TI-89 Titanium

Voyage™ 200

Calculadora Gráfica

Informações importantes

A Texas Instruments não dá qualquer garantia, expressa ou implícita, incluindo, mas não se limitando a quaisquer garantias implícitas de comercialização e de adequação a um propósito específico, com respeito a quaisquer programas ou materiais de livros, e somente disponibiliza esses materiais no estado em que se encontram.

Em hipótese alguma a Texas Instruments poderá ser responsabilizada perante qualquer pessoa por danos especiais, colaterais, incidentais ou conseqüenciais, que tenham qualquer ligação ou que resultem da compra ou utilização desses materiais, e a única e exclusiva responsabilidade da Texas Instruments, independentemente da forma de atuação, não deve exceder qualquer preço de compra aplicável deste artigo ou material. Além disso, a Texas Instruments não poderá estar sujeita a qualquer reivindicação, seja de que espécie for, com respeito ao uso desses materiais por qualquer outra parte.

© 2005 Texas Instruments Incorporated
Windows, Macintosh são marcas comerciais dos respectivos proprietários.

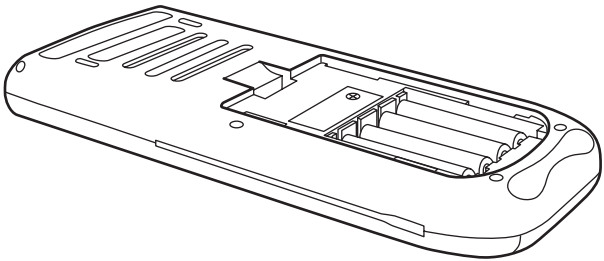
Como começar

Arranque inicial

Instalar as pilhas AAA

A TI-89 Titanium utiliza quatro pilhas alcalinas AAA e uma pilha de segurança de óxido de prata (SR44SW ou 303). A Voyage™ 200 utiliza quatro pilhas alcalinas AAA e uma pilha de segurança de lítio (CR1616 ou CR1620). As pilhas de segurança vêm já instaladas e as pilhas AAA são fornecidas com os produtos.

1. Remova a tampa do compartimento de pilhas da parte posterior da unidade portátil.
2. Desembrulhe as quatro pilhas AAA fornecidas com o produto e insira-as no compartimento de pilhas. Coloque as pilhas de acordo com o diagrama de polaridade (+ e –) do compartimento de pilhas.



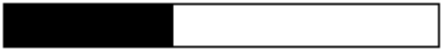
3. Coloque a tampa do compartimento de pilhas na unidade portátil. A tampa deve encaixar-se correctamente.

Ligar a TI-89 Titanium ou a Voyage™ 200 pela primeira vez

Depois de instalar as pilhas incluídas com a unidade portátil, prima [ON]. Aparece o ambiente de trabalho das Apps.

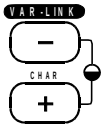
Nota: se a unidade portátil iniciar a Apps pré-instalada, aparece uma barra de progresso com a mensagem "`Installation in progress . . . Do not interrupt!`"em vez do ambiente de trabalho da Apps. Para não perder a Apps, não remova as pilhas durante a inicialização. (Pode reinstalar a Apps a partir do CD-ROM de recursos ou de education.ti.com.)

Barra de progresso



Ajustar o contraste

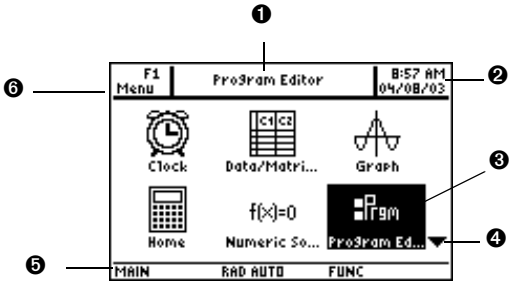
- Para clarear o visor, prima sem soltar [◆] e toque em [−].
- Para escurecer o visor, prima sem soltar [◆] e toque em [+].



Ambiente de trabalho das Apps

O ambiente de trabalho da Apps é o ponto de partida para operar com a TI-89 Titanium ou a Voyage™ 200. As Apps instaladas aparecem no ambiente de trabalho das Apps como ícones organizados em categorias para fácil acesso. No ambiente de trabalho das Apps pode:

- Abrir Apps.
- Seleccionar e editar categorias das Apps.
- Ver todas as Apps instaladas na unidade portátil.
- Ver o nome completo da App realçada.
- Ver e editar a hora e a data.
- Verificar as informações da linha de estado.
- Ver as informações do modo Split Screen.



Ambiente de trabalho das Apps da TI-89 Titanium

❶ Veja o nome completo da App realçada.

❷ Veja a data e a hora.

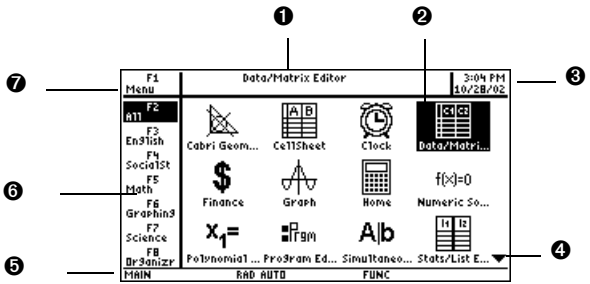
❸ Prima [ENTER] para abrir a App realçada.

❹ Consulte para ver as Apps adicionais.

❺ Verifique as informações da linha de estado.

❻ Edite as categorias.

Nota: como o visor é mais pequeno, o ambiente de trabalho das Apps da TI-89 Titanium parece ligeiramente diferente do ambiente de trabalho das Apps da Voyage™ 200. A lista de categorias de Apps, incluída do lado esquerdo do ambiente de trabalho da Voyage™ 200, é omitida do ambiente de trabalho da TI-89 Titanium, mas as categorias são seleccionadas da mesma forma em ambos os produtos.



Ambiente de trabalho das Apps da Voyage™ 200

❶ Veja o nome completo da App realçada.

❷ Prima [ENTER] para abrir a App realçada.

❸ Veja a data e a hora.

❹ Consulte para ver as Apps adicionais.

❺ Verifique as informações da linha de estado.

❻ Seleccione a categoria das Apps.

❼ Edite as categorias.

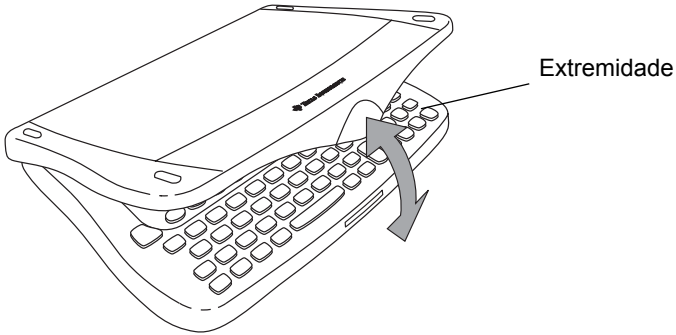
Para voltar ao ambiente de trabalho das Apps a qualquer momento, prima [APPS]. A última categoria seleccionada aparece com a última App aberta realçada.

Remover e instalar a tampa (Voyage™ 200)

Para remover a tampa:

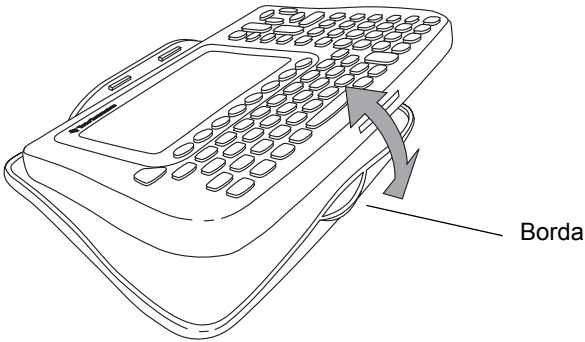
1. Segure a unidade com uma mão.
2. Agarre a tampa pela extremidade com a outra mão.
3. Levante a tampa pela extremidade.

Para instalar a tampa, posicione-a sobre a calculadora com a extremidade virada para a frente e encaixe-a.



Guardar a tampa

Para guardar a tampa, posicione-a virada ao contrário por baixo da unidade portátil com a borda virada para a frente e encaixe-a no lugar.



Desligar a unidade portátil

Prima [2nd] [OFF]. Da próxima vez que ligar a unidade portátil o ambiente de trabalho das Apps aparece com as mesmas definições e conteúdos de memória retidos. (Se desligar o ambiente de trabalho das Apps, aparece o ecrã inicial da calculadora.)

Pode utilizar uma das seguintes teclas para desligar a TI-89 Titanium / Voyage™ 200.

Prima:	Descrição
[2nd] [OFF] (prima [2nd] e, em seguida, [OFF])	As definições e os conteúdos da memória são retidos pela função Constant Memory™. • No entanto, não pode utilizar [2nd] [OFF], se aparecer uma mensagem de erro. • Quando voltar a ligar a TI-89 Titanium / Voyage™ 200, mostra o ecrã Home (Inicial) ou o ambiente de trabalho das Apps (independentemente da última aplicação utilizada).
[◆] [OFF] (prima [◆] e, em seguida, [OFF])	Similar a [2nd] [OFF] excepto: • Pode utilizar [◆] [OFF], se aparecer uma mensagem de erro. • Quando voltar a ligar a TI-89 Titanium / Voyage™ 200, ficará exactamente como estava.

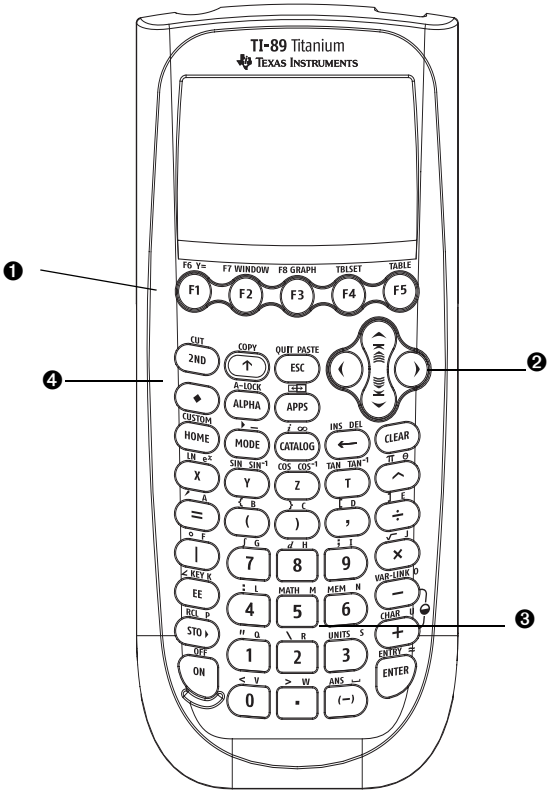
Nota: [OFF] é a segunda função da tecla [ON].

A função Automatic Power Down™ (APD™) da unidade portátil prolonga a duração das pilhas, desligando automaticamente a unidade portátil após vários minutos de inactividade. Quando ligar a unidade portátil após a APD:

- O visor, o cursor e quaisquer condições de erro são exactamente iguais como antes da APD.
- Todos os conteúdos da memória e definições são retidos.

Nota: O APD não ocorre se existir um cálculo ou programa em progresso, excepto se o programa estiver interrompido. Se um programa estiver em execução, mas estiver à espera de um batimento de tecla, o APD ocorrerá após vários minutos de inactividade.

Teclas da TI-89 Titanium e da Voyage™ 200



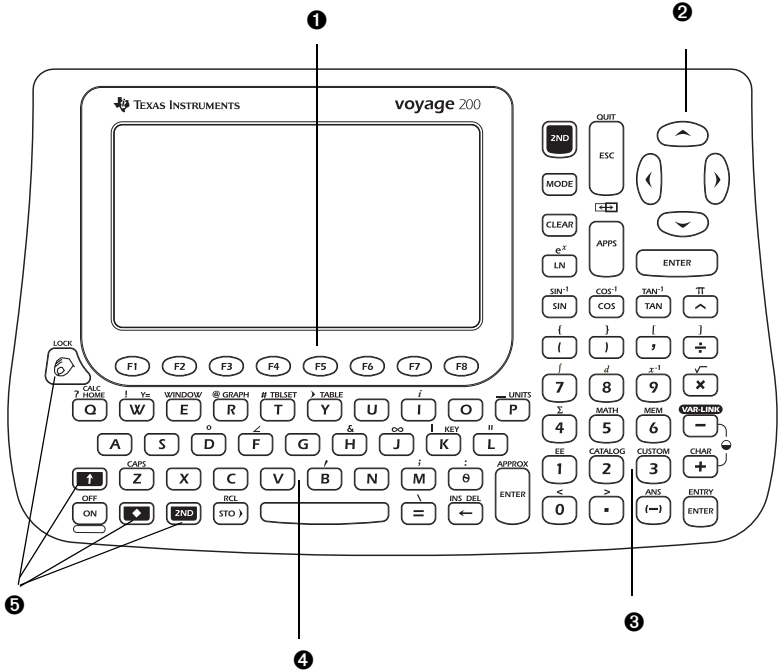
Teclas da TI-89 Titanium

❶ As teclas de função (F1– F8) abrem menus das barras de ferramentas, acedem às Apps e editam categorias das Apps.

❷ As teclas de cursor (◀, ▶, ▲, ▼) movem o cursor.

❸ O teclado numérico opera funções matemáticas e científicas.

❹ As teclas modificadoras (2nd, ◆, ⇧) adicionam funcionalidades, aumentando o número de comandos de teclas.



Teclas da Voyage™ 200

❶ As teclas de função (F1– F8) abrem menus das barras de ferramentas, acedem às Apps e editam categorias das Apps.

❷ As teclas de cursor (◀, ▶, ▲, ▼) movem o cursor.

❸ O teclado numérico opera funções matemáticas e científicas.

❹ As teclas modificadoras (2nd, ◆, ↑ F1 ⊘) adicionam funcionalidades, aumentando o número de comandos de teclas.

Teclado QWERTY (Apenas para a Voyage™ 200)

Se souber escrever num computador, utilizará facilmente o teclado QWERTY da Voyage™ 200, tendo as seguintes diferenças:

- Para escrever um carácter alfabético maiúsculo, prima ↑ e a tecla da letra.
- Para activar o bloqueio das maiúsculas, prima 2nd [CAPS]. Para desactivar o bloqueio das maiúsculas, prima novamente 2nd [CAPS].

Introduzir caracteres especiais

Utilize o menu CHAR e os comandos das teclas para introduzir os caracteres especiais. O menu CHAR permite aceder aos caracteres gregos, matemáticos, internacionais e outros especiais. Um mapa do teclado mostra as localizações dos atalhos utilizados para introduzir os outros caracteres mais utilizados.

Para seleccionar os caracteres do menu CHAR:

1. Prima 2nd [CHAR]. Aparece o menu CHAR.
2. Utilize as teclas de cursor para seleccionar uma categoria. Um submenu lista os caracteres dessa categoria.
3. Utilize as teclas de cursor para seleccionar um carácter e prima ENTER.

Exemplo: introduza o símbolo da seta direita (→) no Text Editor.

Prima	Resultado
[2nd] [CHAR]	CHAR 1:Greek ▸ 2:Math ▸ 3:Punctuation ▸ 4:Special ▸ 5:International ▸
4	CHAR 1:Greek ▸ 2:Math ▸ 3:Punctuation ▸ 4:Special ▸ 5:International ▸ 1: 2: 3: 4: 5: 6↓ Vá para baixo para ver mais caracteres.
9 – ou – Prima várias vezes ⊖ para seleccionar **9:**→ e prima [ENTER]	F1▾ Tools F2▾ Command F3▾ View F4 Execute F5 Find... :x^3-2x^2+x-1→ Símbolo mostrado no local do cursor. MAIN RAD AUTO FUNC 0/30

Para abrir o mapa do teclado, prima [◆] [KEY]. Aparece o mapa do teclado.

Para escrever a maioria dos caracteres, prima [◆] e a tecla correspondente. Prima [ESC] para fechar do mapa.

TI-89 Titanium Exemplo: utilize o mapa do teclado para localizar o símbolo “diferente de” (≠) e introduza o símbolo no Program Editor.

Prima	Resultado
◆ [KEY]	≠ GREEK @ SYSDATA ! = ( ) , ÷ FMT LOG ROOT & 1 7 8 9 X SYMB KBDPRGM 4 - 6 EE 4 5 6 KBDPRGM 1 - 3 → 1 2 3 OFF ≤ ≥ HOMEDATA ON 0 . (-)
◆ =	F1 Tools F2 Control F3 I/O F4 Var F5 Find... F6 Mode :chat() :Prgm :ClrIO :If A≠ — Símbolo mostrado no local do cursor : :EndPrgm MAIN RAD AUTO FUNC

Voyage™ 200 Exemplo: utilize o mapa do teclado para localizar o atalho do símbolo de aspas (") e introduza o símbolo no Program Editor..

Atalho	Caracteres escritos
◆ [KEY]	? ! é @ # ▸ Ü ¡ ô _ Q W E R T Y U I O P à ß ° ∠ GREEK & ∞ | " A S D F G H J K L CAPS @ Ç ≠ ' ~ ; Z X C V B N M

Atalho	Caracteres escritos
2nd L	

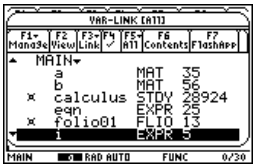
Teclas modificadoras

As teclas modificadoras adicionam funcionalidades, aumentando o número de operações do teclado ao seu dispor. Para aceder a uma função modificadora, prima uma tecla modificadora e, em seguida, a tecla para a operação correspondente.

Teclas	Descrição
2nd (Second)	Acede às Apps, opções dos menus e outras operações. As funções Second estão impressas por cima das teclas correspondentes da mesma cor como a tecla 2nd.
◆ (Diamond)	Acede às Apps, opções dos menus e outras operações. As funções Diamond estão impressas por cima das teclas correspondentes da mesma cor como a tecla ◆.
⇧ (Shift)	Escreve um carácter maiúsculo na próxima tecla de letra que premir. Utilizada também com ◀ e ▶ para realçar caracteres durante a edição.

Teclas	Descrição
alpha (Alpha; apenas para a TI-89 Titanium)	Permite digitar caracteres alfabéticos sem um teclado QWERTY. Os caracteres Alpha são impressos por cima das teclas correspondentes da mesma cor da tecla alpha.
(Hand; apenas para a Voyage™ 200)	Permite utilizar as teclas de cursor para manipular objectos geométricos. Utilizada também para desenhar um gráfico.

Exemplo: acede ao ecrã **VAR-LINK [All] (Tds)**, onde pode gerir variáveis e Apps.

Prima	Resultado
2nd [VAR-LINK]	

Teclas de função

Utilize as teclas de função para efectuar as seguintes operações:

- No ambiente de trabalho das Apps, abra as Apps e seleccione ou edite as categorias das Apps.
- No ecrã inicial da calculadora, abra os menus da barra de ferramentas para seleccionar as operações relacionadas com a matemática.

- Nas Apps, abra os menus das barras de ferramentas para seleccionar as opções da App.

Teclas de cursor

Se premir ◀, ▶, ▲ ou ▼, move o cursor na direcção correspondente. Dependendo da App e da tecla modificadora [2nd] ou [◆] utilizada, as teclas de cursor movem o cursor de várias formas.

- ▲ ou ▼ move o cursor para cima ou para baixo uma linha de cada vez.
- [2nd] ◀ ou [2nd] ▶ move o cursor para o início ou o fim de uma linha.
- [2nd] ▲ ou [2nd] ▼ move o cursor para cima ou para baixo uma página de cada vez.
- [◆] ▲ ou [◆] ▼ move o cursor para o topo ou o fundo de uma página.
- ▲ e ◀, ▲ e ▶, ▼ e ◀ ou ▼ e ▶ move o cursor na diagonal. (Prima cada par de teclas de cursor indicado em simultâneo.)

Teclado numérico

O teclado numérico permite introduzir números positivos e negativos.

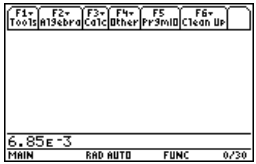
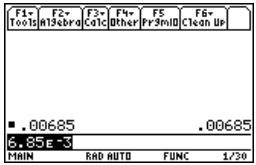
Para introduzir um número negativo, prima antes de escrever o número.

Nota: não confunda o sinal de menos posicional ([(-)]) com o sinal de menos operacional ([−]).

Para introduzir um número em notação científica:

- Escreva os números que antecedem o expoente. (Este valor pode ser uma expressão.)
- Prima [EE] (TI-89 Titanium) or [2nd] [EE] (Voyage™ 200). O símbolo de expoente (E) aparece a seguir aos números introduzidos.
- Escreve o expoente como um número inteiro até três dígitos. (Como o exemplo seguinte mostra, pode utilizar um expoente negativo.)

Exemplo: no ecrã inicial da calculadora, introduza 0.00685 com a notação científica.

Prima	Resultado
6 [.] 8 5 TI-89 Titanium: [EE] Voyage™ 200: [2nd] [EE] **[(-)] 3**	
[ENTER]	

Outras teclas importantes

Comando da tecla	Descrição
◆ [Y=] Apenas para a TI-89 Titanium	Mostra o Y= Editor.
◆ [WINDOW] Apenas para a TI-89 Titanium	Mostra o Window Editor.
◆ [GRAPH] Apenas para a TI-89 Titanium	Mostra o ecrã Graph.
◆ [TBLSET] Apenas para a TI-89 Titanium	Define os parâmetros do ecrã Table.
◆ [TABLE] Apenas para a TI-89 Titanium	Mostra o ecrã Table.
TI-89 Titanium: ◆ [CUT] ◆ [COPY] ◆ [PASTE] Voyage™ 200: ◆ X (cortar) ◆ C (copiar) ◆ V (colar)	Estas teclas permitem editar as informações introduzidas com uma operação de corte, cópia ou colagem.

Comando da tecla	Descrição
[◆] S Apenas para a Voyage™ 200	Mostra a caixa de diálogo SAVE COPY AS, pedindo para seleccionar uma pasta e escrever um nome de variável onde os dados introduzidos no ecrã são guardados.
[◆] N Apenas para a Voyage™ 200	Cria um ficheiro novo.
[◆] O Apenas para a Voyage™ 200	[ON] Abre um ficheiro existente especificado.
[◆] F Apenas para a Voyage™ 200	[ON] Mostra a caixa de diálogo FORMATS ou GRAPH FORMATS onde introduz as informações de formato para a aplicação activa.
[APPS]	Mostra o ambiente de trabalho das Apps. [ON] [APPS]
[◆] [APPS]	Com o ambiente de trabalho das Apps desactivado, mostra o menu FLASH APPLICATIONS.
[2nd] [⇄]	Comuta entre as duas últimas aplicações seleccionadas.
[2nd] [CUSTOM]	Activa e desactiva o menu Custom.
[2nd] [▶]	Converte as unidades de medição.
TI-89 Titanium: [◆] [_] Voyage™ 200: [2nd] [_]	Designa uma unidade de medição.
[←]	Apaga o carácter à esquerda do cursor (backspace).

Comando da tecla	Descrição
[◆] [DEL]	Apaga o carácter à direita do cursor.
[2nd] [INS]	Comuta entre os modos Insert e Overwrite.
[2nd] [MEM]	Mostra o ecrã MEMORY.
TI-89 Titanium: [CATALOG] Voyage™ 200: [2nd] [CATALOG]	Mostra uma lista de comandos.
[2nd] [RCL]	Volta a apresentar os conteúdos de uma variável.
[STO▸]	Guarda um valor numa variável.
[2nd] [CHAR]	Mostra o menu CHAR, que permite seleccionar as letras gregas, caracteres acentuados internacionais e outros caracteres especiais.
[2nd] [QUIT]	• No modo Full Screen, mostra o ambiente de trabalho das Apps. • No modo Split Screen, mostra a vista de ecrã completo da App activa. • Com o ambiente de trabalho das Apps desactivado, mostra o ecrã inicial da calculadora.

Definições dos modos

Os modos controlam como a TI-89 Titanium ou a Voyage™ 200 mostra e interpreta as informações. Todos os números, incluindo os elementos de matrizes e listas, aparecem

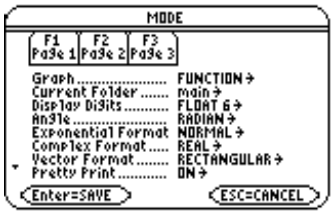
de acordo com as definições actuais dos modos. Quando desligar a TI-89 Titanium ou a Voyage™ 200, a função Constant Memory™ retém todas as definições de modos seleccionadas.

Para ver as definições de modos da TI-89 Titanium ou a Voyage™ 200:

1. Prima [MODE]. Aparece a página 1 da caixa de diálogo MODE.
2. Prima [F2] ou [F3] para ver os modos listados na página 2 ou 3.

Nota: os modos a cinzento só estão disponíveis, se seleccionar as definições de modos necessárias. Por exemplo, o modo Custom Units listado na página 3 só está disponível, se definir o modo Unit System para CUSTOM.

Ver definições de modos

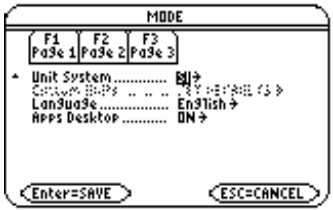
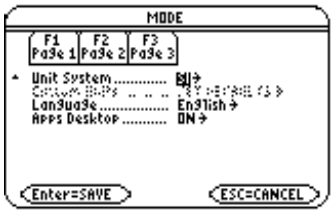
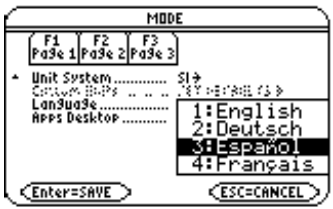
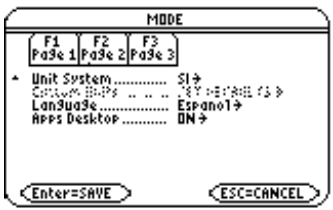
Prima	Resultado
[MODE]	

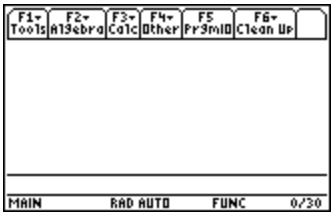
Prima	Resultado
F2	MODE F1 Page 1 F2 Page 2 F3 Page 3 Split Screen ..... FULL→ Split 1 App ..... Home→ Exact/Approx ..... AUTO→ Base ..... DEC→ Enter=SAVE ESC=CANCEL
F3	MODE F1 Page 1 F2 Page 2 F3 Page 3 Unit System ..... SI→ Language ..... English→ Apps Desktop ..... ON→ Enter=SAVE ESC=CANCEL

Alterar as definições de modos

Exemplo: altere a definição do modo Language para Spanish.

Prima	Resultado
MODE	MODE F1 Page 1 F2 Page 2 F3 Page 3 Graph ..... FUNCTION→ Current Folder ..... main→ Display Digits ..... FLOAT 6→ Angle ..... RADIAN→ Exponential Format NORMAL→ Complex Format ..... REAL→ Vector Format ..... RECTANGULAR→ Pretty Print ..... ON→ Enter=SAVE ESC=CANCEL

Prima	Resultado
F3	
Vá para o campo Language. ⊖	
Prima ⊕ e, em seguida, prima ⊖ até **3:Español** estar realçado. **Nota:** a lista de menus pode variar, dependendo dos idiomas instalados.	
ENTER	

Prima	Resultado
ENTER **Nota:** aparece a App aberta anteriormente (neste exemplo, o ecrã inicial da calculadora).	

Para devolver a definição do modo Language para English, repita os passos, seleccionando **1:English** no campo do Language.

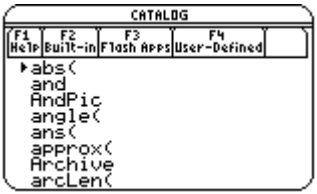
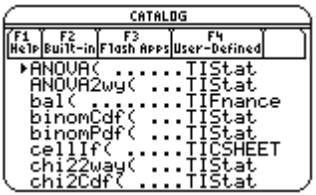
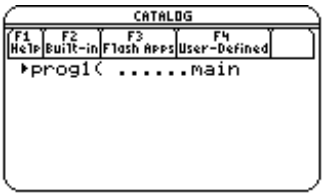
Utilizar o Catalog para aceder aos comandos

Utilize o Catalog para aceder a uma lista de comandos da TI-89 Titanium ou da Voyage™ 200, incluindo funções, instruções e programas definidos pelo utilizador. Os comandos estão listados alfabeticamente. Os comandos sem uma letra no início estão no fim da lista (&, /, +, -, etc.).

A Catalog Help App inclui detalhes sobre cada comando.

As opções inválidas estão a cinzento. Por exemplo, a opção do menu AplFlash (F3) fica a cinzento, se não houver nenhuma aplicação Flash instalada na TI-89 Titanium ou na Voyage™ 200; a opção do menu User-Defined (F4) fica a cinzento, se não criar uma função ou programa.

Nota: a escrita de uma letra leva-o para o primeiro comando começado por essa letra.

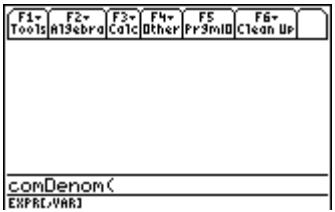
Prima	Resultado
TI-89 Titanium: CATALOG Voyage™ 200: 2nd [CATALOG] (mostra os comandos integrados)	
F3 (mostra os comandos das Flash Apps, se existirem)	
F4 (mostra os comandos definidos pelo utilizador, se existirem)	

Seleccione os comandos do Catalog e insira-os no ecrã Home (Inicial) da calculadora ou cole-os noutras Apps, como o Y= Editor, Text Editor ou a CellSheet™ Apps.

Exemplo: insira o comando **comDenom** na linha de entrada do ecrã inicial da calculadora.

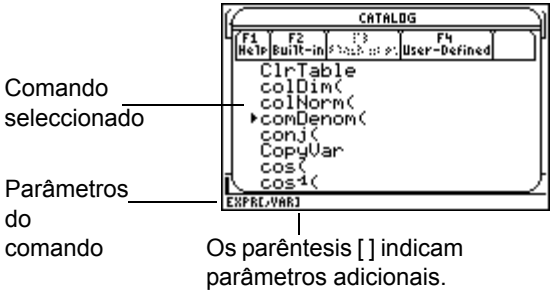
Nota: antes de seleccionar um comando, posicione o cursor onde pretende que o comando apareça.

Se premir [2nd] [▼], avança a lista do Catalog uma página de cada vez.

Prima	Resultado
TI-89 Titanium: [CATALOG] **C** Voyage™ 200: [2nd] [CATALOG] **C** [2nd] [▼] Em seguida, prima [▼] até o ponteiro estar na função **comDenom(**.	
[ENTER]	

A linha de estado mostra qualquer parâmetro necessário e opcional para o comando seleccionado. Os parâmetros opcionais aparecem entre parêntesis rectos.

Nota: se premir [F1] mostra também os parâmetros do comando seleccionado.



Para sair do Catalog sem seleccionar um comando, prima [ESC].

Ecrã inicial da calculadora

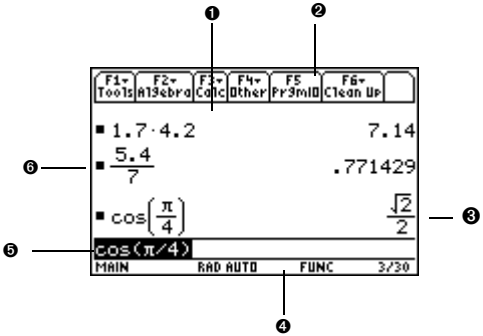
O ecrã inicial da calculadora é o ponto de partida para as operações matemáticas, incluindo a execução de instruções, avaliação de expressões e a visualização de resultados.

Para ver o ecrã inicial da calculadora, prima:

TI-89 Titanium: [HOME]

Voyage™ 200: [◆] [CALC HOME].

Pode também ver o ecrã inicial da calculadora a partir do ambiente de trabalho Apps, realçando o ícone Ecrã inicial e premindo [ENTER].



❶ A área do histórico lista os pares de entrada/resposta introduzidos.

❷ Os separadores mostram os menus para seleccionar as listas de operações. Prima [F1], [F2], etc., para ver os menus.

❸ O resultado da última entrada aparece aqui. (Não se esqueça de que os resultados não aparecem na linha de entrada.)

❹ A linha de estado mostra o estado actual da TI-89 Titanium ou da Voyage™ 200.

❺ A linha de entrada mostra a entrada actual.

❻ A entrada anterior aparece aqui.

Para voltar ao ambiente de trabalho das Apps a partir do ecrã inicial da calculadora, prima [APPS].

Acerca da área do histórico

A área do histórico mostra até oito pares de entradas/respostas, dependendo da complexidade e da altura das expressões. Quando o visor está cheio, as informações voltam ao topo do ecrã. Utilize a área do histórico para:

- Rever entradas e respostas anteriores. Utilizar as teclas de cursor para ver as entradas e respostas que desapareceram do ecrã.
- Voltar a apresentar ou colar automaticamente uma entrada ou resposta anterior na linha de entrada para reutilização ou edição. (Para mais informações, consulte o módulo electrónico *Utilizar a calculadora* da TI-89 Titanium ou da Voyage™ 200.)

Pode mover o cursor, que permanece geralmente na linha de entrada, para a área do histórico. A tabela seguinte mostra como mover o cursor na área do histórico.

Para	Efectuar este procedimento
Ver entradas/respostas desaparecidas do ecrã	Na linha de entrada, prima ⊕ para realçar a última resposta. Continue a utilizar ⊕ para mover o cursor da resposta para a entrada através da área do histórico.
Ir para o par de entrada/resposta mais antigo ou mais recente	Se o cursor estiver na área do histórico, prima [◆] ⊕ ou [◆] ⊖.
Ver uma entrada ou resposta muito longa para uma linha (▶ aparece no fim da linha)	Mova o cursor para a entrada ou resposta. Utilize ◀ ou ▶ para mover para a esquerda ou para a direita e [2nd] ◀ ou [2nd] ▶ para ir para o início ou o fim.

Para	Efectuar este procedimento
Colocar o cursor na linha de entrada	Prima [ESC] ou ⊖ até o cursor voltar à linha de entrada.

Interpretar as informações do histórico na linha de estado

Utilize o indicador de histórico na linha de estado para obter informações sobre os pares de entradas/respostas. Por exemplo:

Se o cursor estiver na linha de entrada:

Número total de pares guardados actualmente ——— 8/30 ——— Número máximo de pares que pode guardar

Se o cursor estiver na área do histórico:

Número do par da entrada/resposta realçada ——— 8/30 ——— Número total de pares guardados actualmente

Modificar a área do histórico

Para alterar o número de pares que pode guardar:

1. No ecrã inicial da calculadora, prima [F1] e seleccione **9:Format (9:Formato)**.
2. Prima ⊕ e utilize ⊕ ou ⊖ para realçar o número novo.
3. Prima [ENTER] [ENTER].

Para limpar a área do histórico e apagar todos os pares guardados:

- No ecrã inicial da calculadora, prima [F1] e seleccione **8:Clear Home**.

  – ou –

- Introduza **ClrHome** na linha de entrada do ecrã inicial do ecrã.

Para apagar um par de entrada/resposta, mova o cursor para a entrada ou resposta e prima [←] ou [CLEAR].

Trabalhar com Apps

A TI-89 Titanium ou a Voyage™ 200 organiza as Apps por categoria no ambiente de trabalho das Apps. Para seleccionar uma categoria, prima uma tecla de função ([F2] - [2nd] [F8] ou [F2] a [F8]). Os ícones da App da categoria seleccionada aparecem no ambiente de trabalho das Apps.

Nota: se o nome de um ícone do ambiente de trabalho das Apps estiver truncado, utilize as teclas de cursor para realçar o ícone. Veja o nome completo no topo do ambiente de trabalho das Apps.

Abrir Apps

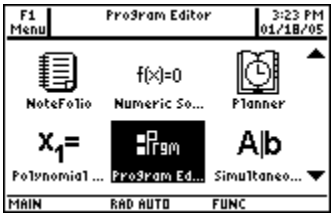
Utilize as teclas de cursor para realçar o ícone Apps no ambiente de trabalho das Apps e prima [ENTER]. A App abre-se directamente ou mostra uma caixa de diálogo. A caixa de diálogo mais usual lista estas opções para a App:

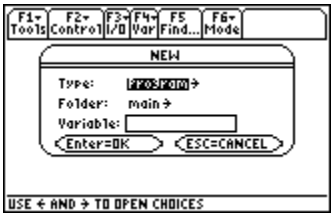
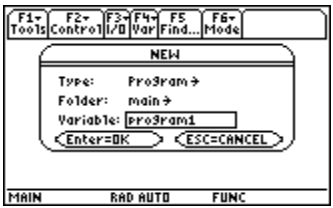
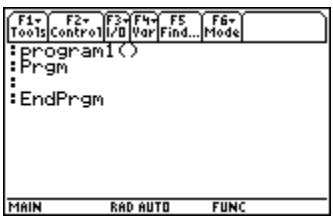
Nota: a TI-89 Titanium ou a Voyage™ 200 utiliza uma *variável* geral para se referir aos ficheiros da App criados.

Opção	Descrição
Current	Devolve o ecrã apresentado da última vez que viu a App. Se não existir nenhuma variável App actual, aparece a caixa de diálogo New.
Open	Permite abrir um ficheiro existente.
New	Cria um ficheiro novo com o nome escrito no campo.

Seleccione uma opção, introduza qualquer informação requerida e prima [ENTER]. Aparece a App.

Exemplo: crie um programa novo com o Program Editor.

Prima	Resultado
Utilize as teclas de cursor para realçar Program Ed...	
[ENTER]	

Prima	Resultado
3	
ENTER	
⊖ ⊖ **p r o g r a m 1**	
ENTER ENTER	

A variável de programa criada recentemente, *program1*, é guardada na pasta principal.

Voltar ao ambiente de trabalho das Apps a partir de uma App

Prima [APPS]. Os ícones da última categoria Apps seleccionada aparecem no ambiente de trabalho das Apps com o ícone da última App aberta realçado.

Pode também voltar ao ambiente de trabalho das Apps, premindo [2nd] [QUIT] no modo Full Screen. No modo Split Screen, prima duas vezes [2nd] [QUIT].

Para voltar à última App aberta a partir do ambiente de trabalho das Apps, prima [2nd] [⇥].

Seleccionar uma categoria Apps

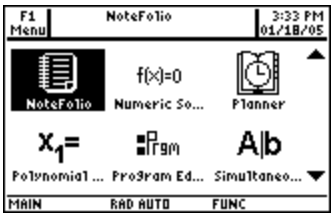
Na TI-89 Titanium, os nomes de categorias das Apps aparecem apenas no menu F1. Para seleccionar uma categoria de Apps, prima [F1] **2:Select Category** e utilize as teclas de cursor para realçar uma categoria de Apps e, em seguida, prima [ENTER] para seleccionar a categoria realçada. Pode também utilizar os atalhos das teclas de função para seleccionar uma categoria do teclado (utilize a tecla [2nd], se necessário). Os ícones da App da categoria seleccionada aparecem no ambiente de trabalho das Apps.

Na Voyage™ 200, os nomes das categorias Apps aparecem no lado esquerdo do ambiente de trabalho das Apps. Para seleccionar uma categoria Apps, prima a tecla de função correspondente (apresentada por cima do nome da categoria no ambiente de trabalho das Apps). Os ícones da App da categoria seleccionada aparecem no ambiente de trabalho das Apps.

Tecla	Descrição
[F2] All	Aparecem os ícones de todas as Apps instaladas. Não personalizável.

Tecla	Descrição
F3 English	Categoria personalizável. English é a predefinição.
F4 SocialSt	Categoria personalizável. SocialSt é a prededinição.
F5 Math	Categoria personalizável. Math é a predefinição.
2nd F6 Graphing ou F6 Graphing	Categoria personalizável. Graphing é a predefinição.
2nd F7 Science ou F7 Science	Categoria personalizável. Science é a predefinição.
2nd F8 Organizr ou F8 Organizr	Categoria personalizável. Organizr é a predefinição.

Exemplo: seleccione a categoria All.

Prima	Resultado
F2	

Se seleccionar uma categoria Apps sem Apps, aparece uma mensagem a confirmar que a categoria está vazia e a apontar para o menu [F1] **1:Edit Categories**, onde pode adicionar atalhos App para a categoria. (Para mais informações sobre como personalizar as categorias do ambiente de trabalho das Apps, consulte *Personalizar as categorias Apps.*)

Prima [ENTER] ou [ESC] para limpar a mensagem e voltar ao ambiente de trabalho das Apps.

Personalizar as categorias Apps

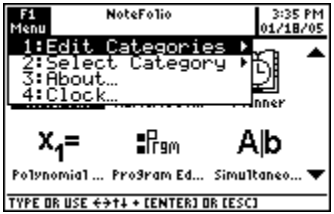
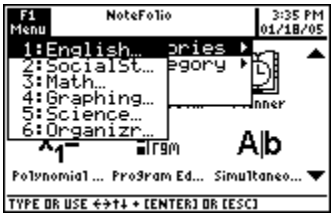
A TI-89 Titanium ou a Voyage™ 200 organiza as Apps em sete categorias, seis das quais pode personalizar para adaptar às suas necessidades individuais. (A categoria All (Tds) contém todas as Apps instaladas e não pode ser editada.)

Para personalizar as categorias Apps de [F3] a [2nd] [F8] (▦) ou [F8] (▬):

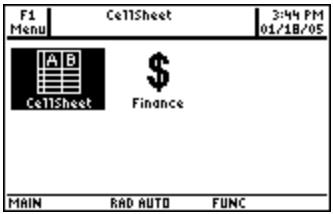
1. Seleccione [F1] **1:Edit Categories**. Aparece um submenu com seis nomes de categorias Apps personalizáveis. (A categoria All não é listada.)
2. Realce uma categoria Apps e prima [ENTER]. A caixa de diálogo Edit Categories aparece com uma lista das Apps instaladas e uma caixa de texto com o nome da categoria realçado.
3. Para alterar o nome da categoria Apps, escreva o nome pretendido.

   Nota: introduza um nome até oito caracteres, incluindo letras minúsculas e maiúsculas, números, pontuação e caracteres acentuados.
4. Para adicionar ou remover um atalho App da categoria, prima ⊖ para realçar a caixa junto à App e, em seguida, prima ⊕ para adicionar ou remover a marca de verificação (✓).
5. Para guardar as alterações e voltar ao ambiente de trabalho das Apps, prima [ENTER].

Exemplo: substitua a categoria Social Studies pela categoria Business e adicione os atalhos das Apps CellSheet™ e Finance.

Prima	Resultado
F1	
⊙	
2 – ou – ⊖ ENTER	

Prima	Resultado
TI-89 Titanium: 2nd [a-lock] ↑ **B u s i n e s s** Voyage™ 200: ↑ **B u s i n e s s**	Edit Categories Category Name: Business Use → to choose APP shortcuts. Calendar ☐ CellSheet ☐ Clock ☐ Contacts ☐ ▼ Data/Matrix Editor ☐ Enter=OK ESC=CANCEL
⊖ ⋮ ⊕	Edit Categories Category Name: Business Use → to choose APP shortcuts. Calendar ☐ CellSheet ☑ → Clock ☐ Contacts ☐ ▼ Data/Matrix Editor ☐ Enter=OK ESC=CANCEL
⊖ ⋮ ⊕	Edit Categories Category Name: Business Use → to choose APP shortcuts. ▲ Clock ☐ Contacts ☐ Data/Matrix Editor ☐ EEPro ☐ ▼ Finance ☑ → Enter=OK ESC=CANCEL
ENTER	F1 Menu — NoteFolio — 3:43 PM 01/18/05 NoteFolio, f(x)=0 Numeric So..., Planner Polynomial ..., Program Ed..., Simultaneo... MAIN RAD AUTO FUNC

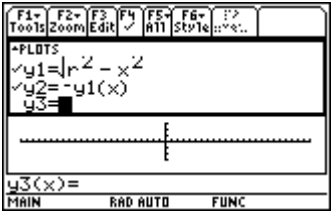
Prima	Resultado
F4	

Abrir o estado das Apps e do modo Split Screen

A TI-89 Titanium ou a Voyage™ 200 permite dividir o ecrã para duas Apps em simultâneo. Por exemplo, veja os ecrãs Y= Editor e Graph para ver a lista de funções e os gráficos.

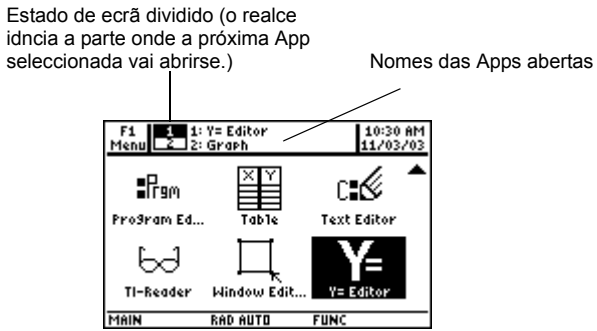
Seleccione o modo Split Screen na página 2 do ecrã MODE. A TI-89 Titanium ou a Voyage™ 200 mostra as Apps seleccionadas na vista de ecrã dividido. Divida o ecrã na horizontal (cima-baixo) ou na vertical (esquerda-direita).

Ecrã dividido de cima para baixo

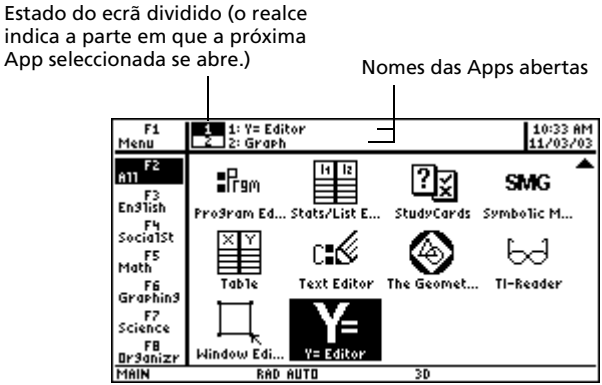


Para voltar ao ambiente de trabalho das Apps, prima O. O estado do ecrã dividido aparece no topo do ambiente de trabalho das Apps com os nomes das Apps abertas e as partes do ecrã em que aparece cada uma delas. O número realçado indica a parte do ecrã dividido em que aparece a próxima App aberta.

Nota: o ambiente de trabalho das Apps aparece sempre na vista de ecrã completo.



Indicadores de ecrã dividido no ambiente de trabalho das Apps da TI-89 Titanium

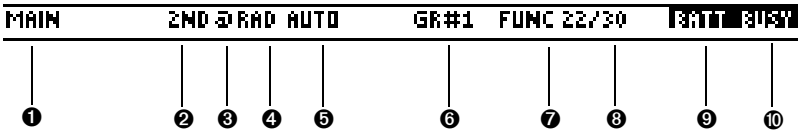


Indicadores de ecrã dividido no ambiente de trabalho das Apps da Voyage™ 200

Existem mais informações disponíveis sobre como utilizar ecrãs divididos. (Para mais informações, consulte o capítulo electrónico ***Ecrãs divididos***.)

Verificar informações de estado

Observe a linha de estado, localizada na parte inferior do ecrã para obter informações sobre o estado actual da TI-89 Titanium ou da Voyage™ 200.



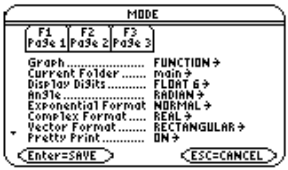
Indicador	Significado
❶ Pasta actual	Nome da pasta seleccionada (Main (Principal) é a pasta predefinida.)
❷ Tecla modificadora	Tecla modificadora seleccionada (2nd, ◆, ↑), se houver.
❸ Tecla Mão (Apenas para a Voyage™ 200).	A tecla modificadora foi seleccionada. (Apenas para a Voyage™ 200).
❹ Modo Angle	Unidades seleccionadas onde os valores dos ângulos são apresentados e interpretados (RAD, DEG, GRAD)
❺ Modo Exact/Approx	Modo onde as respostas são calculadas e apresentadas (AUTO, EXACT, APPROX)
❻ Número do gráfico	Activo de dois gráficos independentes no modo Split Screen (GR#1 (GR#1), GR#2 (GR#2))
❼ Modo Graph	Tipo de gráfico seleccionado que pode ser desenhado (FUNC, PAR, POL, SEQ, 3D, DE)
❽Pares de entradas/respostas	22/30-Número de pares de entradas/respostas (predefinição é 30, máximo é 99) na área do histórico do ecrã inicial da calculadora

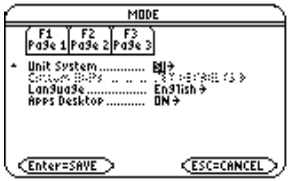
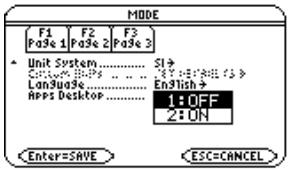
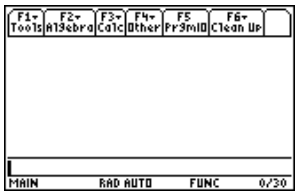
Indicador	Significado
❾ Substituir pilhas	Apresentada quando as pilhas estão com a carga baixa (BATT). Se BATT estiver realçada com um fundo preto, mude as pilhas o mais rapidamente possível (BATT).
❿ Variável bloqueada/arquivada, ocupada/pausa	OCUPADA-Cálculo ou gráfico em progresso PAUSA-Interrompeu um gráfico ou programa 🔒-Variável aberta no editor actual está bloqueada ou arquivada e não pode ser modificada

Desactivar o ambiente de trabalho das Apps

Pode desactivar o ambiente de trabalho das Apps na caixa de diálogo MODE. Se o quiser fizer, abra as Apps no menu APPLICATIONS. Para abrir o menu APPLICATIONS, prima [APPS].

Exemplo: desactive o ambiente de trabalho das Apps.

Prima	Resultado
[MODE]	

Prima	Resultado
F3	
⊖ ⊖ ⊕ ⊕	
ENTER ENTER **Nota:** aparece a App aberta previamente (neste exemplo, o ecrã inicial da calculadora).	

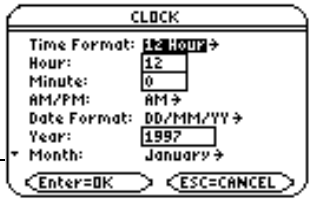
Para activar o ambiente de trabalho das Apps, repita o procedimento, seleccionando ON no campo do modo Apps Desktop. Para voltar ao ambiente de trabalho das Apps a partir do ecrã inicial da calculadora, prima APPS.

Utilizar o relógio

Utilize a caixa de diálogo CLOCK para definir a hora e a data, seleccionar o formato de visualização do relógio, ligar e desligar o relógio.

O relógio está ligado por predefinição. Se desligar o relógio, todas as opções da caixa de diálogo CLOCK, excepto Clock ON/OFF, aparecem a cinzento.

▼ indica a deslocação para as opções de activação/desactivação do relógio e do dia



Ver a caixa de diálogo CLOCK

1. Utilize as teclas de cursor para realçar o ícone Clock no ambiente de trabalho das Apps.
2. Prima [ENTER]. Aparece a caixa de diálogo CLOCK com o campo Time Format realçado.

Nota: como a caixa de diálogo CLOCK mostra as definições actuais com a hora em que abriu a caixa de diálogo, pode necessitar de actualizar a hora antes de sair.

Definir a hora

1. Prima ⊙ para abrir a lista de formatos de hora.
2. Prima ⊙ ou ⊙ para realçar uma opção e, em seguida, prima [ENTER]. O formato seleccionado aparece no campo Time Format.
3. Prima ⊙ para realçar o campo Hour.
4. Escreva a hora e, em seguida, prima ⊙ para realçar o campo Minute.

5. Escreva o(s) minuto(s).
6. Se definir o formato de hora para 24 horas, avance para o passo 9.

   – ou –

   Se definir o formato de hora para 12 horas, prima ⊖ para realçar o campo AM/PM.
7. Prima ⊕ para abrir a lista de opções AM/PM.
8. Prima ⊕ ou ⊖ para realçar uma opção AM/PM e, em seguida, prima [ENTER]. Aparece a opção AM/PM seleccionada.
9. Defina a data (para mais informações sobre os procedimentos, consulte *Definir a data*).

   – ou –

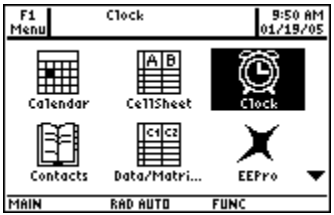
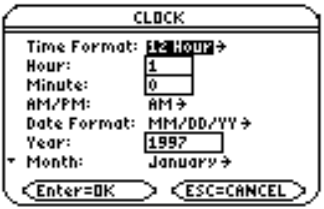
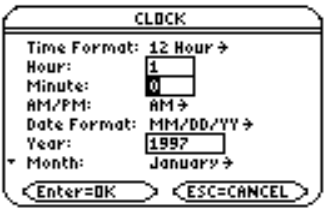
   Para guardar as definições e sair, prima [ENTER]. A hora é actualizada no canto superior direito do ambiente de trabalho das Apps.

Definir a data

1. Prima ⊕ ou ⊖ para realçar o campo Date Format.
2. Prima ⊕ para abrir a lista de formatos de data.
3. Prima ⊕ ou ⊖ para realçar uma opção e, em seguida, prima [ENTER]. O formato seleccionado aparece no campo Date Format.
4. Prima ⊖ para realçar o campo Year.
5. Escreva o ano e prima ⊖ para realçar o campo Month.
6. Prima ⊕ para abrir a lista dos meses.
7. Prima ⊕ ou ⊖ para realçar uma opção e, em seguida, prima [ENTER]. O mês seleccionado aparece no campo Month.

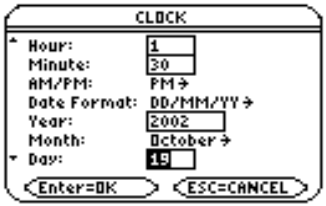
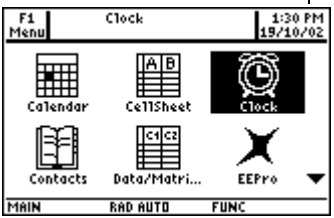
8. Prima ⊖ para realçar o campo Day.
9. Escreva o dia e prima [ENTER] [ENTER] para guardar as definições e sair. A data é actualizada no campo superior direito do ambiente de trabalho das Apps.

Exemplo: Defina a hora e a data para 19/10/02 (19 de Outubro de 2002) à 1:30 p.m.

Prima	Resultado
Utilize as teclas de cursor para realçar	Hora e data 
[ENTER]	
⊖ **1** ⊖	

Prima	Resultado
3 0 ⊖	CLOCK Time Format: 12 Hour → Hour: 1 Minute: 30 AM/PM: AM → Date Format: MM/DD/YY → Year: 1997 Month: January → Enter=OK ESC=CANCEL
⊕ ⊖	CLOCK Time Format: 12 Hour → Hour: 1 Minute: 30 AM/PM: Date Format: 1:AM 2:PM Year: Month: January → Enter=OK ESC=CANCEL
ENTER ⊖	CLOCK Time Format: 12 Hour → Hour: 1 Minute: 30 AM/PM: PM → Date Format: MM/DD/YY → Year: 1997 Month: January → Enter=OK ESC=CANCEL
⊕ ⊖	CLOCK Time Format: 12 Hour → Hour: Minute: AM/PM: Date Format: Year: Month: Enter=OK 1:MM/DD/YY 2:DD/MM/YY 3:MM.DD.YY 4:DD.MM.YY 5:YY.MM.DD 6:MM-DD-YY 7:DD-MM-YY 8:YY-MM-DD

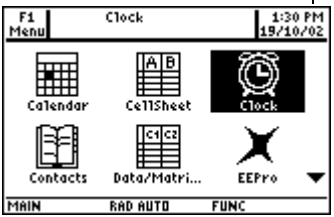
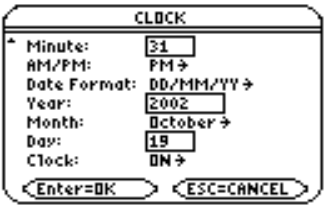
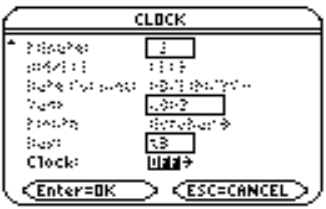
Prima	Resultado
ENTER ⊖	CLOCK Time Format: 12 Hour → Hour: 1 Minute: 30 AM/PM: PM → Date Format: DD/MM/YY → Year: 1997 ▾ Month: January → Enter=OK ESC=CANCEL
2 0 0 2	CLOCK Time Format: 12 Hour → Hour: 1 Minute: 30 AM/PM: PM → Date Format: DD/MM/YY → Year: 2002 ▾ Month: January → Enter=OK ESC=CANCEL
⊖ ⊕	CLOCK Time Format: 12 Hour → Hour: Minute: AM/PM: Date Format: Year: ▾ Month: Enter=OK 1:January 2:February 3:March 4:April 5:May 6:June 7:July 8↓August
Prima ⊕ ou ⊖ até realçar Outubro e prima ENTER	CLOCK Time Format: 12 Hour → Hour: 1 Minute: 30 AM/PM: PM → Date Format: DD/MM/YY → Year: 2002 ▾ Month: October → Enter=OK ESC=CANCEL

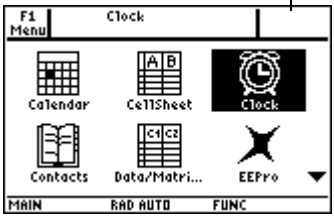
Prima	Resultado
⊖ 1 9	
ENTER ENTER	Data e hora revistas 

Desligar o relógio

No ambiente de trabalho das Apps, abra a caixa de diálogo CLOCK e seleccione OFF no campo Clock.

Exemplo: desligue o relógio.

Prima	Resultado
Utilize as teclas de cursor para realçar	Relógio ligado 
ENTER Vá para o campo Clock.	
▶ ▲ ENTER	

Prima	Resultado
ENTER	Relógio desligado 

Para ligar o relógio, repita o procedimento, seleccionando ON no campo Clock. Não se esqueça de reiniciar a data e a hora.

Utilizar os menus

Para seleccionar a maioria dos menus da TI-89 Titanium ou da Voyage™ 200, prima as teclas de função correspondentes às barras de ferramentas existentes no topo do ecrã inicial da calculadora e à maioria dos ecrãs App. Seleccione outros menus com os comandos de teclas.

Menus das barras de ferramentas

O ponto de partida das operações matemáticas da TI-89 Titanium ou da Voyage™ 200 é o ecrã inicial da calculadora que apresenta os menus das barras de ferramentas que permitem seleccionar as operações relacionadas com a matemática (consulte o exemplo na página seguinte).

Os menus das barras de ferramentas aparecem também no topo da maioria dos ecrãs da App. Estes menus listam as funções gerais da App activa.

Outros menus

Utilize os comandos de teclas para seleccionar os menus seguintes. Estes menus contêm as mesmas opções, independentemente do ecrã apresentado ou a App activa.

Prima	Para ver
2nd [CHAR]	Menu CHAR. Lista os caracteres não disponíveis no teclado; os caracteres são organizados por categoria (gregos, matemáticos, pontuação, especiais e internacionais).
2nd [MATH]	Menu MATH. Lista as operações matemáticas por categoria.
APPS	Menu APPLICATIONS. Lista as Apps instaladas. (O menu só está disponível quando o ambiente de trabalho das Apps está desactivado; as Apps são geralmente acedidas a partir do ambiente de trabalho das Apps.)
◆ APPS	Menu FLASH APPLICATIONS. Lista as Flash Apps. (O menu só está disponível quando o ambiente de trabalho das Apps está desactivado; as Flash Apps são geralmente acedidas a partir do ambiente de trabalho das Apps.)

Seleccionar opções dos menus

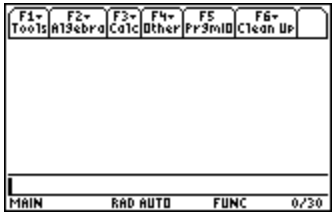
- Prima o número ou a letra que se encontra do lado esquerdo da opção que pretende seleccionar.

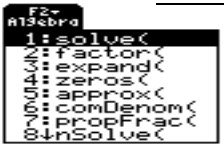
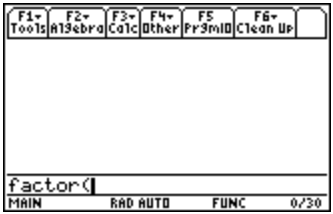
  — ou —

- Prima ⊕ ou ⊖ para seleccionar a opção e prima [ENTER].

Nota: se seleccionar a primeira opção do menu, prima ⊕ para seleccionar a última opção do menu. Se seleccionar a última opção do menu, prima ⊖ para seleccionar a primeira opção do menu.

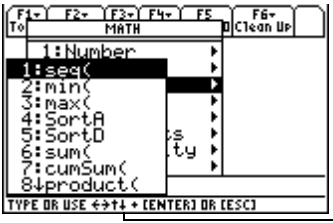
Exemplo: seleccione **factor(** no menu Algebra no ecrã inicial da calculadora.

Prima	Resultado
Prima: TI-89 Titanium: [HOME] Voyage™ 200: [◆] [CALC HOME] – ou – No ambiente de trabalho das Apps, utilize as teclas de cursor para realçar e prima [ENTER]	

Prima	Resultado
F2	 ▾ indica que o menu Algebra abrirá quando premir F2.
2 – ou – ⊖ ENTER	

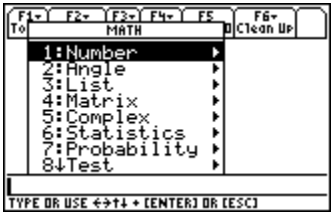
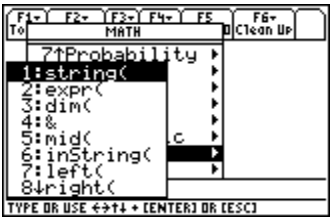
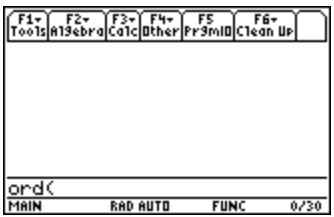
Seleccionar opções dos submenus

Um símbolo de seta pequena (▶) no lado direito de uma opção do menu indica que a selecção da opção abrirá um submenu.



↓ pontos para opções adicionais.

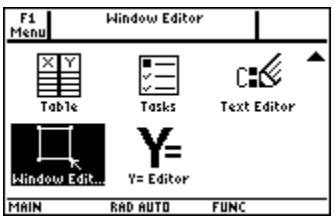
Exemplo: seleccione **ord(** no menu MATH no ecrã inicial da calculadora.

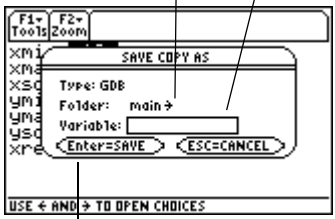
Prima	Resultado
2nd [MATH]	
D – ou – ▲ ▲ ▶	
B – ou – ▲ [ENTER]	

Utilizar as caixas de diálogo

Uma elipse (...) no fim de uma opção do menu indica que a selecção da opção abrirá uma caixa de diálogo. Seleccione a opção e prima [ENTER].

Exemplo: abra a caixa de diálogo SAVE COPY AS no Window Editor.

Prima	Resultado
APPS Utilize as teclas de cursor para realçar e prima ENTER	
F1	

Prima	Resultado
2 – ou – ⊖ [ENTER]	Prima ⊕ para ver a lista de pastas. Escreva o nome da variável.  Prima duas vezes [ENTER] para guardar e fechar a caixa de diálogo.

Nota: se premir a tecla de atalho [◆] S, abre também a caixa de diálogo SAVE COPY AS na maioria das Apps.

Cancelar um menu

Para cancelar um menu sem fazer uma selecção, prima [ESC].

Mover entre os menus das barras de ferramentas

Para mover entre os menus das barras de ferramentas sem seleccionar uma opção de menu:

- Prima a tecla de função ([F1] a [F8]) de um menu de barra de ferramentas.

- Prima uma tecla de função e, em seguida, ⊙ ou ⊙ para mover de um menu de barra de ferramentas para outro. Prima ⊙ no último menu para ir para o primeiro menu. Prima ⊙ para ir do primeiro ao último menu.

Nota: se premir ⊙ quando seleccionar uma opção de menu com um submenu, o submenu aparece em vez do menu da barra de ferramentas seguinte. Prima novamente ⊙ para ir para o menu seguinte.

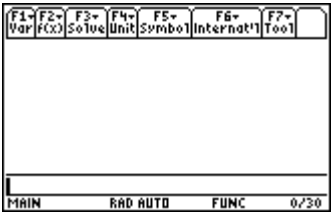
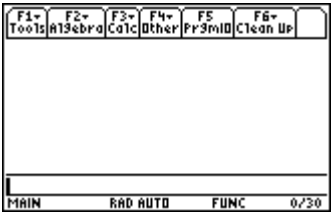
Existem mais informações disponíveis sobre os menus. (Consulte o capítulo electrónico *Operar a calculadora*.)

Menu Custom

O menu Custom proporciona acesso rápido à maioria das funções mais utilizadas. Utilize o menu Custom predefinido ou crie um menu Custom próprio com o Program Editor. Pode incluir qualquer comando ou carácter da TI-89 Titanium ou Voyage™ 200 disponível.

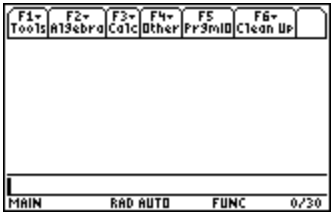
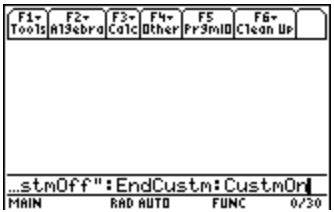
O menu Custom substitui o menu da barra de ferramentas standard do ecrã inicial da calculadora. (Para mais informações sobre como criar um menu Custom, consulte o módulo on-line Programar da TI-89 Titanium ou da Voyage™ 200.) Existem mais informações disponíveis sobre os menus personalizados. (Consulte o capítulo electrónico *Operar a calculadora*.)

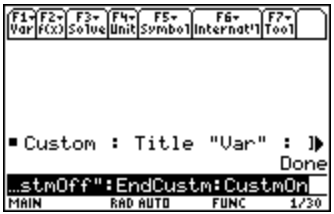
Exemplo: active e desactive o menu Custom no ecrã inicial da calculadora.

Prima	Resultado
2nd [CUSTOM]	Menu Custom predefinido 
2nd [CUSTOM]	Menu da barra de ferramentas normal 

Exemplo: restaure o menu Custom predefinido.

Nota: a restauração do menu Custom predefinido apaga o menu Custom anterior. Se criou o menu Custom anterior com um programa, pode executar novamente o programa para reutilizar o menu.

Prima	Resultado
[2nd] [CUSTOM] (para desactivar o menu Custom e activar o menu da barra de ferramentas standard)	
TI-89 Titanium: [2nd] [F6] Voyage™ 200: [F6]	
3 – ou – ⊖ ⊖ [ENTER]	

Prima	Resultado
ENTER	

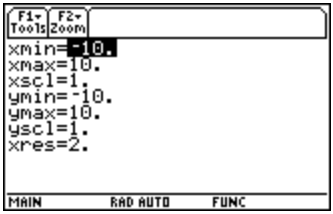
Abrir as Apps com o ambiente de trabalho das Apps desactivado

Se desactivar o ambiente de trabalho das Apps, utilize o menu APPLICATIONS para abrir as Apps. Para abrir o menu APPLICATIONS com o ambiente de trabalho das Apps desactivado, prima APPS.

Nota: se premir APPS com o ambiente de trabalho das Apps activado, o ambiente de trabalho das Apps aparece em vez do menu APPLICATIONS.

Exemplo: Com o ambiente de trabalho das Apps desactivado, abra o Window Editor no menu APPLICATIONS.

Prima	Resultado
APPS	

Prima	Resultado
3 – ou – ⊖ ⊖ [ENTER]	

Para aceder a Apps não listadas no menu APPLICATIONS, seleccione **1:FlashApps**.

Utilizar os ecrãs divididos

A TI-89 Titanium ou a Voyage™ 200 permite dividir o ecrã para mostrar duas Apps em simultâneo. Por exemplo, veja os ecrãs Y= Editor e Graph para comparar a lista de funções e os gráficos.

Definir o modo Split Screen

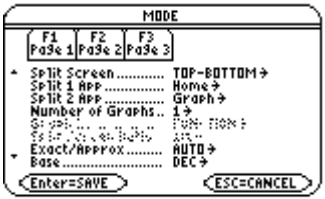
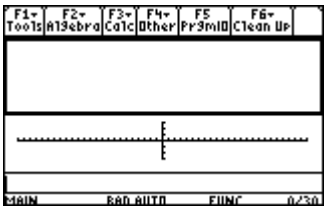
Pode dividir o ecrã de cima para baixo ou da esquerda para a direita na caixa de diálogo MODE. A definição de ecrã dividido permanece activa até a alterar.

1. Prima [MODE] para ver a caixa de diálogo MODE.
2. Prima [F2] para ver a definição do modo Split Screen.
3. Prima ⊕ para abrir o menu do modo Split Screen.
4. Prima ⊖ para realçar TOP-BOTTOM ou LEFT-RIGHT.
5. Prima [ENTER]. A definição do modo Split Screen mostra a opção seleccionada.

6. Prima [ENTER] novamente para guardar esta alteração e ver o ecrã dividido.

Exemplo: defina o modo Split Screen para TOP-BOTTOM.

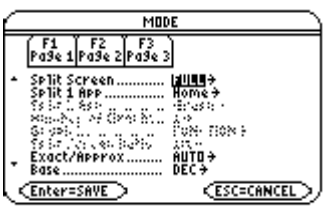
Prima	Resultado
[MODE]	MODE F1 Page 1 F2 Page 2 F3 Page 3 Graph ..... FUNCTION → Current Folder ..... main → Display Digits ..... FLOAT 6 → Angle ..... RADIAN → Exponential Format NORMAL → Complex Format ..... REAL → Vector Format ..... RECTANGULAR → Pretty Print ..... ON → Enter=SAVE ESC=CANCEL
[F2]	MODE F1 Page 1 F2 Page 2 F3 Page 3 Split Screen ..... FULL → Split 1 App ..... Home → Exact/Approx ..... AUTO → Base ..... DEC → Enter=SAVE ESC=CANCEL
[▶] [▼]	MODE F1 Page 1 F2 Page 2 F3 Page 3 Split Screen ..... Split 1 App ..... 1:FULL 2:TOP-BOTTOM 3:LEFT-RIGHT Exact/Approx ..... AUTO → Base ..... DEC → Enter=SAVE ESC=CANCEL

Prima	Resultado
ENTER	
ENTER	

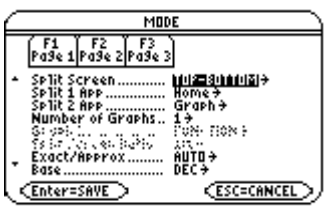
Definir as Apps iniciais para o ecrã dividido

Depois de seleccionar o modo Split Screen TOP-BOTTOM ou LEFT-RIGHT, as definições adicionais do modo ficam disponíveis.

Modo Full Screen



Modo Split Screen



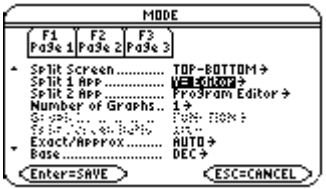
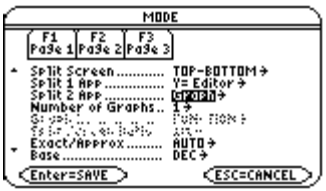
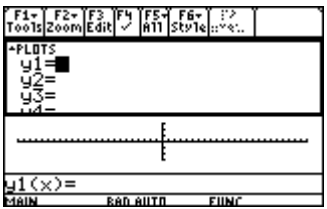
Modo	Descrição
Split 2 App	Permite especificar a App apresentada na parte inferior ou direita do ecrã dividido. Funciona juntamente com a Split 1 App, que permite especificar a App apresentada na parte superior ou esquerda do ecrã dividido.
Number of Graphs	Permite configurar e ver dois gráficos independentes.
Split Screen Ratio	Permite alterar o rácio das vistas do ecrã dividido.

Para definir a App inicial para cada parte do ecrã dividido:

1. Seleccione a definição do modo Split 1 App e prima ⊙ para ver um menu das Apps disponíveis (Para mais informações sobre como seleccionar o modo Split Screen, consulte *Definir o modo Split Screen*, página 64).
2. Prima ⊖ ou ⊕ para realçar a App e prima [ENTER].
3. Repita os passos 1 e 2 para a definição do modo Split 2 App.

Exemplo: veja o Y= Editor no ecrã superior e o Graph App no ecrã inferior.

Prima	Resultado
⊖ ⊙	

Prima	Resultado
2	
⊖ ⊕	
4	
ENTER	

Se definir Split 1 App e Split 2 App para a mesma App não gráfica ou para a mesma App gráfica com Number of Graphs definido para 1, a TI-89 Titanium ou a Voyage™ 200 sai do modo Split Screen e mostra a App no modo Full Screen.

Seleccionar a App activa

No modo Split Screen, só pode estar uma App activa de cada vez.

- Para comutar entre as Apps activas, prima [2nd] [⊞].
- Para seleccionar uma terceira App, prima [APPS] e seleccione a App. Esta App substitui a App activa no modo Split Screen.

Sair do modo Split Screen

Saia do modo Split Screen de uma das seguintes formas:

- Prima [2nd] [QUIT] para fechar a App activa e ver a vista de ecrã completo da outra App aberta.
- Se o ambiente de trabalho das Apps estiver desactivado, premir [2nd] [QUIT] substitui a App activa no ecrã dividido pelo ecrã inicial da calculadora. Se premir novamente [2nd] [QUIT], desactiva o modo Split Screen e mostra o ecrã inicial da calculadora no modo Full Screen.
- Seleccione Split Screen na página 2 da caixa de diálogo MODE, defina o modo Split Screen para FULL e prima [ENTER].
- Prima duas vezes [2nd] [QUIT] para ver o ambiente de trabalho das Apps.

Gerir versões das Apps e do sistema operativo (SO)

As funções de conectividade da TI-89 Titanium ou da Voyage™ 200 permitem transferir as Apps do:

- Site Web TI Educational & Productivity Solutions (E&PS) em: education.ti.com/latest
- CD-ROM fornecido com a unidade portátil gráfica.
- Uma unidade portátil compatível.

A adição de Apps à TI-89 Titanium ou à Voyage™ 200 é como carregar software num computador. Tudo o que necessita é o software TI Connect™ e o cabo USB fornecidos com a TI-89 Titanium ou a Voyage™ 200.

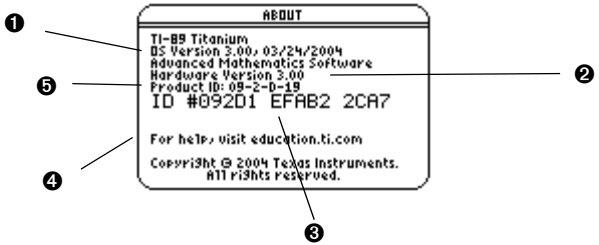
Para obter informações sobre os requisitos do sistema, instruções de ligação de unidades portáteis compatíveis e transferir o software TI Connect software, Apps e as versões do SO, consulte o site Web da TI E&PS.

Antes de transferir as Apps para a TI-89 Titanium ou a Voyage™ 200, leia o acordo de licenciamento no CD-ROM ou no site Web da TI.

Localizar a versão do SO e o número de identificação (ID)

Se comprar software no site Web da TI E&PS ou telefonar para o número de suporte a clientes, ser-lhe-á pedido para fornecer informações sobre a TI-89 Titanium ou a Voyage™ 200. Estas informações estão no ecrã ABOUT.

Para ver o ecrã ABOUT, prima F1 **3:About** no ambiente de trabalho das Apps. O ecrã ABOUT mostra as seguintes informações sobre a TI-89 Titanium ou a Voyage™ 200:



❶ Versão SO

❷ Versão do hardware

❸ ID da unidade (necessária para obter certificados para instalar as Apps adquiridas). Similar a um número de série. Escreva este número e guarde-o num local seguro em caso de perda ou furto da unidade portátil.

❹ Número da revisão do certificado das Apps (Rev. cert.)

❺ Identificador do produto (ID do produto). Similar a um número de modelo.

Não se esqueça de que o seu ecrã será diferente do ecrã apresentado acima.

Apagar uma aplicação

Se apagar uma aplicação, remove-a da TI-89 Titanium ou a Voyage™ 200 e aumenta o espaço para outras aplicações. Antes de apagar uma aplicação, armazene-a num computador para a poder reinstalar mais tarde.

1. Saia da aplicação.

2. Prima 2nd [VAR-LINK] para ver o ecrã VAR-LINK (All).
3. Prima 2nd [F7] (TI-89 Titanium) ou a F7 para ver a lista das aplicações instaladas.
4. Prima F4 para seleccionar a aplicação que pretende apagar. (Prima novamente F4 para anular a selecção.)
5. Prima F1 **1:Delete**. Aparece a caixa de diálogo de confirmação da eliminação de VAR-LINK.
6. Prima ENTER para apagar a aplicação.

Nota: só pode apagar Flash Apps.

Ligar a TI-89 Titanium ou a Voyage™ 200 a outros dispositivos

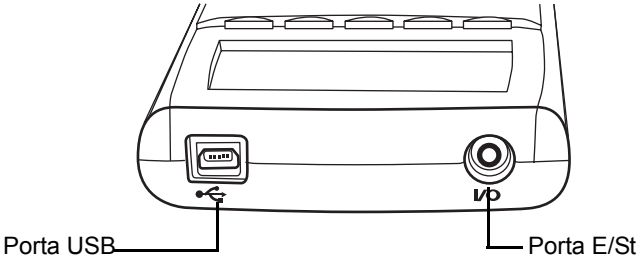
A TI-89 Titanium inclui uma mini porta USB. A TI-89 Titanium e a Voyage™ 200 incluem uma porta E/S standard. Estas portas são utilizadas para ligar duas unidades portáteis gráficas compatíveis, ligar a um computador ou a um dispositivo periférico. Além disso, o modelo do professor das unidades portáteis TI-89 Titanium e Voyage™ 200 incluem uma porta de acessório. Esta porta é utilizada para enviar dados visuais para que uma turma possa ver o visor da unidade portátil num dispositivo de vídeo ou num retroprojector.

Para ligar a unidade portátil a um computador – Ligue a TI-89 Titanium com a porta USB e o cabo USB incluído ou ligue a Voyage™ 200 com a porta E/S e o cabo TI Connectivity USB incluído.

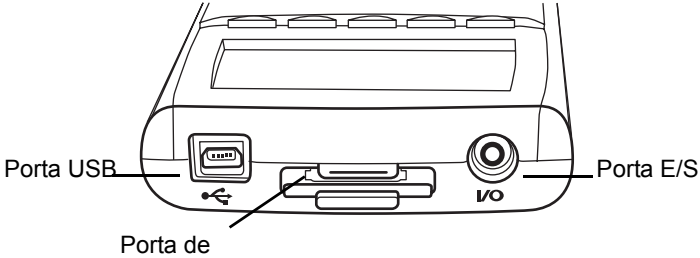
Para ligar a unidade portátil a outra unidade portátil – Utilize o cabo de unidade a unidade USB ou standard para ligar a TI-89 Titanium ou a Voyage™ 200 a uma unidade

portátil gráfica compatível ou a um dispositivo periférico, como uma unidade portátil gráfica TI-89 ou TI-92 Plus, ou os sistemas CBL 2™ e CBR™.

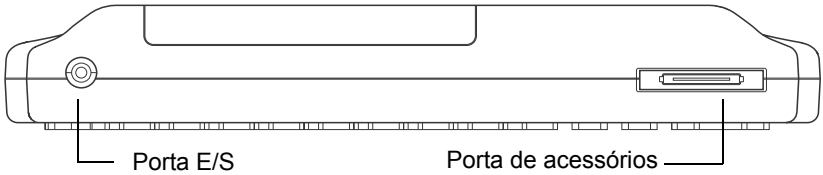
Para mostrar o visor da unidade portátil à turma – •Utilize a porta de acessórios para ligar o adaptador de vídeo TI-Presenter™ à Voyage™ 200. O adaptador de vídeo TI-Presenter proporciona uma interface de vídeo entre a Voyage™ 200 e os dispositivos de gravação e de visualização de vídeo. Utilize a porta de acessórios para ligar o painel retroprojector TI ViewScreen™ à unidade portátil. O painel retroprojector TI ViewScreen amplia e projecta o visor para que uma turma inteira o possa ver. Para mais informações sobre o adaptador de vídeo TI-Presenter e o painel TI ViewScreen, consulte o site Web TI E&PS em education.ti.com.



Portas da TI-89 Titanium



Portas TI-89 Titanium (modelo do



Portas da Voyage™ 200

Pilhas

A TI-89 Titanium utiliza quatro pilhas alcalinas AAA e uma pilha de segurança de óxido de prata (SR44SW ou 303). A Voyage™ 200 utiliza quatro pilhas alcalinas AAA e uma pilha de segurança de lítio (CR1616 ou CR1620). As pilhas de segurança já vêm instaladas e as pilhas AAA são fornecidas com os produtos.

Informações importantes sobre a transferência do SO

Deve instalar pilhas novas antes de iniciar uma transferência do SO.

Quando está no modo de transferência do SO, a função APD™ não funciona. Se deixar a unidade portátil no modo de transferência durante um longo período de tempo antes de iniciar a transferência, as pilhas podem ficar gastas. Necessita de substituir as pilhas gastas por pilhas novas antes de iniciar a transferência.

Pode também transferir o SO para outra TI-89 Titanium ou Voyage™ 200 com um cabo de ligação de unidade a unidade. Se interromper a transferência antes de estar completa, necessita de reinstalar o SO através de um computador. Não se esqueça de instalar pilhas novas antes de iniciar a transferência.

Contacte a Texas Instruments como descrito em Informações sobre a Assistência aos Produtos e a Garantia TI se tiver um problema.

Precauções com as pilhas

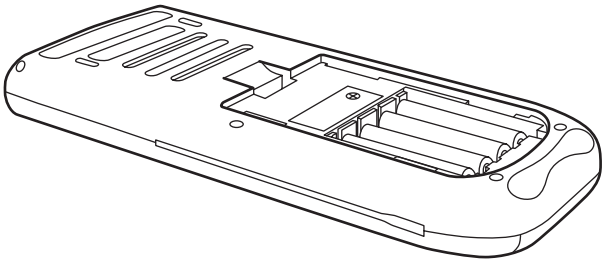
Tome estas precauções quando substituir as pilhas.

- Não deixe as pilhas ao alcance das crianças.
- Não misture pilhas novas e velhas. Não misture marcas (ou tipos nas marcas) das pilhas.
- Não misture pilhas recarregáveis e normais.
- Instale as pilhas de acordo com os esquemas de polaridades (+ e -).
- Não coloque pilhas normais num carregador de pilhas.
- Coloque as pilhas usadas no lixo.

- Não incinere nem desmonte as pilhas.

Instalar as pilhas AAA

1. Remova a tampa do compartimento de pilhas da parte posterior da unidade portátil.
2. Desembrulhe as quatro pilhas AAA fornecidas com o produto e insira-as no compartimento de pilhas. Coloque as pilhas de acordo com o diagrama de polaridade (+ e –) no compartimento de pilhas.



3. Coloque a tampa do compartimento de pilhas na unidade portátil. A tampa deve encaixar-se no lugar.

Substituir as pilhas AAA (alcalinas)

À medida que as pilhas perdem carga, o visor começa a escurecer, especialmente durante os cálculos. Se necessitar de aumentar o contraste frequentemente, substitua as pilhas alcalinas.

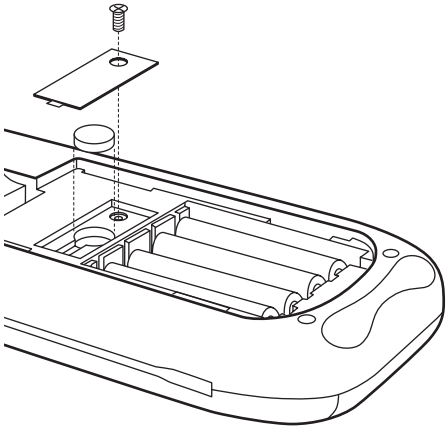
A linha de estado da Voyage™ 200 fornece também informações das pilhas.

Indicador	Significado
BATT	As pilhas estão fracas.
BATT	Substitua as pilhas o mais rapidamente possível.

Antes de substituir as pilhas, desligue a TI-89 Titanium ou a Voyage™ 200, premindo [2nd] [OFF] para evitar a perda de informações armazenadas na memória. Não remova a pilha de lítio de segurança e as pilhas alcalinas AAA em simultâneo.

Substituir a pilha de segurança (óxido de prata)

1. Para substituir a pilha de segurança de óxido de prata, remova a tampa do compartimento de pilhas e desaparafuse o parafuso pequeno que mantém a TAMPA DO COMPARTIMENTO DA PILHA DE SEGURANÇA no lugar.



2. Remova a pilha gasta e instale uma nova pilha SR44SW ou 303 com lado positivo (+) virado para cima. Coloque a tampa e o parafuso.

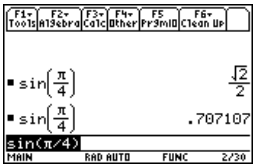
•

Apresentações

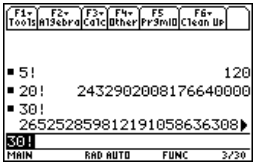
Execução de cálculos

Esta secção fornece vários exemplos para executar no ecrã Calculator Home da unidade portátil que demonstram algumas das funcionalidades computacionais da TI-89 Titanium / Voyage™ 200. A área do histórico de cada ecrã é esvaziada, fazendo clique em [F1] e seleccionando **8:Clear Home**, antes de executar cada exemplo, para ver apenas os resultados das teclas premidas o exemplo actual.

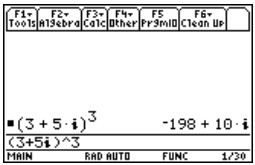
Apresentação dos cálculos

Etapas e teclas	Ecrã
Calcule **sin(π/4)** e veja o resultado nas formas simbólica e numérica. Para limpar os dados da área do histórico, prima [F1] e seleccione **8:Clear Home**. [2nd] [SIN] [2nd] [π] [÷] 4 [)] [ENTER] [◆] [≈] [SIN] [2nd] [π] [÷] 4 [)] [ENTER] [◆] [≈]	

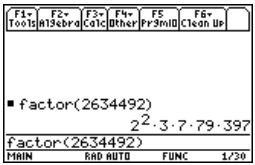
Cálculo do valor factorial de um número

Etapas e teclas	Ecrã
Calcule o valor factorial de vários números para ver como a TI-89 Titanium / Voyage™ 200 trata números inteiros muito grandes. Para obter o operador factorial (!), prima [2nd] [MATH], seleccione **7:Probability** e, em seguida, **1:!**. 5 [2nd] [MATH] 7 1 [ENTER] 2 0 [2nd] [MATH] 7 1 [ENTER] 3 0 [2nd] [MATH] 7 1 [ENTER] 5 [2nd] W [ENTER] 2 0 [2nd] W [ENTER] 3 0 [2nd] W [ENTER]	

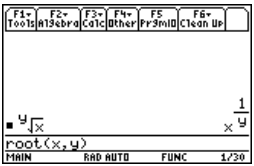
Expansão de números complexos

Etapas e teclas	Ecrã
Calcule $(3+5i)^3$ para ver como a TI-89 Titanium / Voyage™ 200 trata as operações com números complexos. Prima [(] 3 [+] 5 [2nd] [i] [)] [^] 3 [ENTER]	

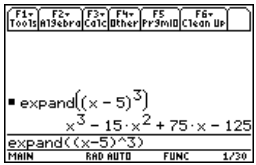
Decomposição em factores primos

Etapas e teclas	Ecrã
Calcule os factores do número inteiro 2634492. É possível introduzir “factor” na linha de entrada, digitando **FACTOR** no teclado ou premindo [F2] e seleccionando **2:factor(**. Prima [F2] 2 2 6 3 4 4 9 2 [)] [ENTER] (*Opcional*) Se quiser, introduza outros números.	

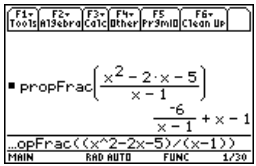
Cálculo de raízes

Etapas e teclas	Ecrã
Calcule a raiz da expressão (x,y). Pode introduzir “root” na linha de entrada, digitando **ROOT** no teclado ou premindo [◆] 9. Este exemplo ilustra a utilização da função de raiz e como a expressão aparece no modo “Pretty print” na área do histórico. Prima [◆] 9 X [,] Y [)] [ENTER]	

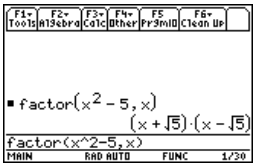
Expansão das expressões

Etapas e teclas	Ecrã
Expanda a expressão $(x-5)^3$. Pode introduzir “expand” na linha de entrada, escrevendo **EXPAND** no teclado ou premindo [F2] e seleccionando **3:expand(**. Prima [F2] 3 [(] X [−] 5 [)] [^] 3 [)] [ENTER] (*Opcional*) Introduza outras expressões na sua.	

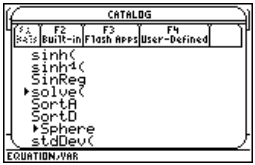
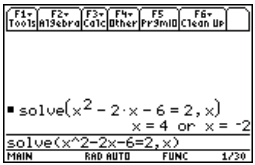
Simplificação de expressões

Etapas e teclas	Ecrã
Reduza a expressão $(x^2-2x-5)/(x-1)$ à sua forma mais simples. É possível introduzir “propFrac” na linha de entrada, digitando **PROPFRAC** no teclado ou premindo [F2] e seleccionando **7:propFrac(**. Prima [F2] 7 [(] X [^] 2 [−] 2 X [−] 5 [)] [÷] [(] X [−] 1 [)] [)] [ENTER]	

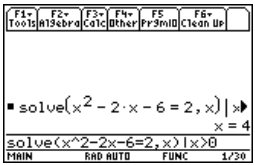
Cálculo de polinómios

Etapas e teclas	Ecrã
Factorize o polinómio (x^2-5) em relação a x. É possível introduzir “factor” na linha de entrada, digitando **FACTOR** no teclado ou premindo [F2] e seleccionando **2:factor(**. Prima [F2] 2 X [^] 2 [−] 5 [,] X [)] [ENTER]	

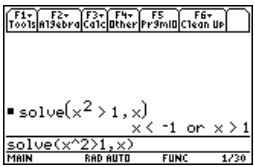
Resolução de equações

Etapas e teclas	Ecrã
Resolva a equação $x^2-2x-6=2$ em relação a x. É possível introduzir “**solve(**” na linha de entrada, seleccionando “**solve(**“no menu Catalog, digitando **SOLVE(** no teclado, ou premindo [F2] e seleccionando **1:solve(**. A linha de estado mostra a sintaxe pedida do item seleccionado no menu **Catalog**. Prima [F2] 1 X [^] 2 [−] 2 X [−] 6 [=] 2 [,] X [)] [ENTER]	 

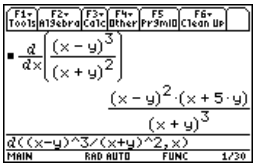
Resolução de equações com restrição de domínio

Etapas e teclas	Ecrã
Resolva a equação $x^2-2x-6=2$ em relação a x, onde x é maior que zero. O operador "*with*" (I) permite restringir o domínio. [F2] 1 X [^] 2 [−] 2 X [−] 6 [=] 2 [,] X [)] [|] X [2nd] [>] 0 [ENTER] [F2] 1 X [^] 2 [−] 2 X [−] 6 [=] 2 [,] X [)] [2nd] [|] X [2nd] [>] 0 [ENTER]	

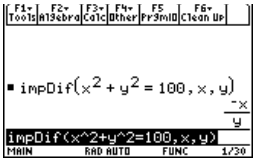
Cálculo de desigualdades

Etapas e teclas	Ecrã
Resolva a desigualdade ($x^2>1$,x) em relação a x. Prima [F2] 1 X [^] 2 [2nd] [>] 1 [)] [ENTER]	

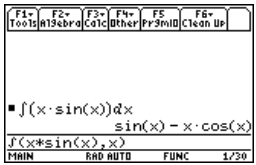
Obtenção da derivada de uma função

Etapas e teclas	Ecrã
Calcule a derivada de $(x-y)^3/(x+y)^2$ em relação a x. Este exemplo demonstra o utilização da função derivada e a forma de apresentação do modo “Pretty print” na área do histórico. Prima [2nd] [d] [(] X [−] Y [)] [^] 3 [÷] [(] X [+] Y [)] [^] 2 [,] X [)] [ENTER]	

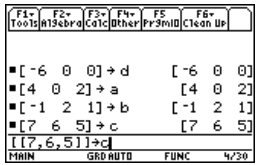
Cálculo de derivadas implícitas

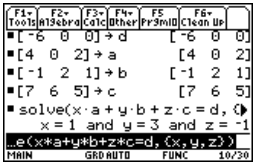
Etapas e teclas	Ecrã
Calcule as derivadas implícitas para equações em duas variáveis em que uma variável é definida implicitamente nos termos da outra. Este exemplo ilustra a utilização da função de cálculo de derivadas implícitas. Prima [F3] D X [^] 2 [+] Y [^] 2 [=] 100 [,] X [,] Y [)] [ENTER]	

Obtenção do integral de uma função

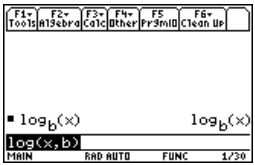
Etapas e teclas	Ecrã
Calcule o integral de **x∗sin(x)** em relação a x. Este exemplo demonstra a utilização da função integral. [2nd] [∫] X [×] [2nd] [SIN] X [)] [,] X [)] [ENTER] [2nd] [∫] X [×] [SIN] X [)] [,] X [)] [ENTER]	

Resolução de problemas com vectores

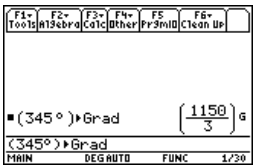
Etapas e teclas	Ecrã
1. Introduza uma linha ou coluna de vectores. [2nd] [[] [(-)] 6 [,] 0 [,] 0 [2nd] []] [STO▸] [alpha] d [ENTER] [2nd] [[] 4 [,] 0 [,] 2 [2nd] []] [STO▸] [alpha] a [ENTER] [2nd] [[] [(-)] 1 [,] 2 [,] 1 [2nd] []] [STO▸] [alpha] b [ENTER] [2nd] [[] 7 [,] 6 [,] 5 [2nd] []] [STO▸] [alpha] c [ENTER] [2nd] [[] [(-)] 6 [,] 0 [,] 0 [2nd] []] [STO▸] d [ENTER] [2nd] [[] 4 [,] 0 [,] 2 [2nd] []] [STO▸] a [ENTER] [2nd] [[] [(-)] 1 [,] 2 [,] 1 [2nd] []] [STO▸] b [ENTER] [2nd] [[] 7 [,] 6 [,] 5 [2nd] []] [STO▸] c [ENTER]	

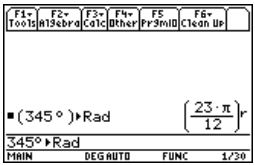
Etapas e teclas	Ecrã
2. Resolve (x* a+y*b+z*c=d {x,y,z}) F2 1 X × alpha a + y × alpha b + z × alpha c = alpha d , 2nd [{] X , Y , Z 2nd [}]) ENTER F2 1 X × a + y × b + z × c = d , 2nd [{] X , Y , Z 2nd [}]) ENTER	

Log para qualquer base

Etapas e teclas	Ecrã
Calcule log (x,b). Pode introduzir “log” na linha de entrada, escrevendo **LOG** no teclado ou premindo ◆ 7. ◆ 7 X , alpha b) ENTER ◆ 7 X , b) ENTER	

Conversão de medidas de ângulos

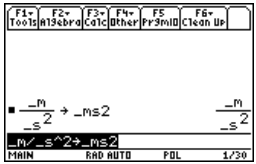
Etapas e teclas	Ecrã
1. Visualize a caixa de diálogo **MODE**. Para o modo **Angle**, seleccione **DEGREE**. Converta 345 graus para a medida do ângulo em gradianos. Pode introduzir "▶**GRAD**" na linha de entrada, seleccionando "▶**GRAD**" no menu Catalog ou Math, premindo [2nd] [MATH] e seleccionando **2:angle, A:▶GRAD**. [MODE] ⊖ ⊖ ⊖ ⊕ 2 [ENTER] 345 [2nd] [°] [2nd] [MATH] 2 [alpha] A [ENTER] [MODE] ⊖ ⊖ ⊖ ⊕ 2 [ENTER] 345 [2nd] [°] [2nd] [MATH] 2 A [ENTER]	

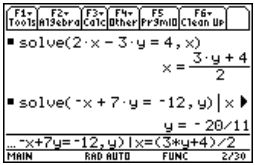
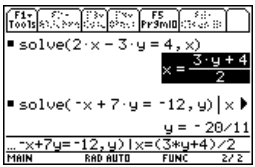
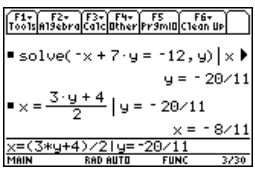
Etapas e teclas	Ecrã
2. Converta 345 graus para a medida do ângulo em radianos. Pode introduzir "▶**GRAD**" na linha de entrada, seleccionando "▶**GRAD**" no menu Catalog ou Math, premindo [2nd] [MATH] e seleccionando **2:angle, A:▶GRAD**. [MODE] ⊖ ⊖ ⊖ ⊕ 2 [ENTER] 345 [2nd] [°] [2nd] [MATH] 2 [alpha] B [ENTER] [MODE] ⊖ ⊖ ⊖ ⊕ 2 [ENTER] 345 [2nd] [°] [2nd] [MATH] 2 B [ENTER] **Nota:** pode também utilizar °, ʳ ou ᴳ para substituir a definição do modo de ângulo temporariamente.	

Manipulação simbólica

Resolva o sistema de equações 2x − 3y = 4 e ⁻x + 7y = ⁻12. Resolva a primeira equação para que x seja expresso em função de y. Substitua a expressão para x na segunda

equação e calcule o valor de y. Em seguida, substitua novamente o valor de y na primeira equação para encontrar o valor de x.

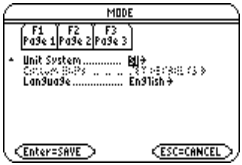
Etapas e teclas	Ecrã
1. Visualize o ecrã inicial e limpe a linha de entrada. Resolva a equação 2x − 3y = 4 para encontrar x. [F2] **1** selecciona **solve(** do menu Algebra. Pode também digitar **solve(** directamente no teclado ou seleccionar em **Catalog**. [HOME] [CLEAR] [CLEAR] [F2] 1 2 X [−] 3 Y [=] 4 [,] X [)] [ENTER] [◆] [CALC HOME] [CLEAR] [CLEAR] [F2] 1 2 X [−] 3 Y [=] 4 [,] X [)] [ENTER]	
2. Comece a resolver a equação ⁻x + 7y = ⁻12 para encontrar y, mas não prima ainda [ENTER]. Prima [F2] 1 [(-)] X [+] 7 Y [=] [(-)] 1 2 [,] Y [)]	

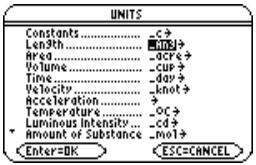
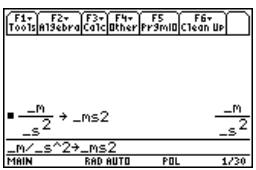
Etapas e teclas	Ecrã
3. Utilize o operador “*with*” para substituir a expressão de x calculada com a primeira equação. Isto fornece o valor de y. O operador “*with*” é mostrado como | no ecrã. Utilize a função de colagem automática para realçar a última resposta da área do histórico e colá-la na linha de entrada. [|] [▲] [ENTER] [ENTER] [2nd] [|] [▲] [ENTER] [ENTER]	
4. Realce a equação em x na área do histórico. Prima [▲] [▲] [▲]	
5. Cole automaticamente a expressão realçada na linha de entrada. Em seguida, substitua o valor de y calculado com a segunda equação. [ENTER] [|] [▲] [ENTER] [ENTER] [ENTER] [2nd] [|] [▲] [ENTER] [ENTER] A solução é: x = ⁻8/11 e y = ⁻20/11	

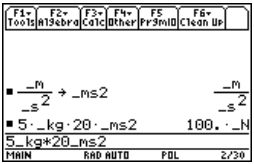
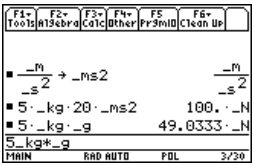
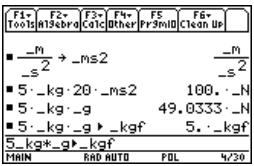
Este exemplo é uma demonstração da manipulação simbólica. Uma função de uma etapa está disponível para resolver sistemas de equações.

Constantes e unidades de medida

Utilize a equação f = m∗a para calcular a força quando m = 5 quilogramas e a = 20 metros/segundo^2. Qual é a força quando a = 9,8 metros/segundo^2? (Esta é a aceleração da gravidade representada pela constante _g). Converta o resultado de newtons para quilogramas-força.

Etapas e teclas	Ecrã
1. Visualize a caixa de diálogo **MODE** na página 3. Para o modo **Unit System**, seleccione **SI** para o sistema métrico de medidas. Os resultados são apresentados de acordo com estas unidades padrão. Prima [MODE] [F3] ▶ 1 [ENTER]	

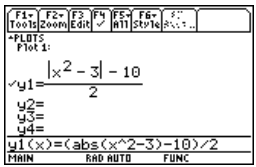
Etapas e teclas	Ecrã
2. Crie uma unidade de aceleração metros/segundo^2 chamada _ms2. A caixa de diálogo **UNITS** permite seleccionar unidades a partir de uma lista de categorias por ordem alfabética. Pode utilizar [2nd] ⊖ e [2nd] ⊕ para percorrer uma páginas de categorias de cada vez. Se utilizar a caixa de diálogo **UNITS** para seleccionar uma unidade, o _ é introduzido automaticamente. Agora, em vez reintroduzir _m/_s^2 cada vez que for necessário, pode utilizar _ms2. Além disso, pode utilizar a caixa de diálogo **UNITS** para seleccionar _ms2 a partir da categoria Acceleration. [2nd] [UNITS] ⊖ ⊕ M [ENTER] [÷] [2nd] [UNITS] ⊖ ⊖ ⊖ ⊖ ⊕ S [ENTER] [^] 2 [STO▸] [◆] [_] [2nd] [a-lock] M S [alpha] 2 [ENTER] [◆] [UNITS] ⊖ ⊕ M [ENTER] [÷] [◆] [UNITS] ⊖ ⊖ ⊖ ⊖ ⊕ S [ENTER] [^] 2 [STO▸] [2nd] [_] M S 2 [ENTER]	 

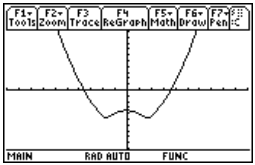
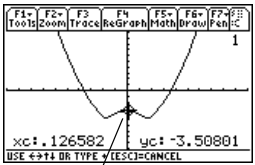
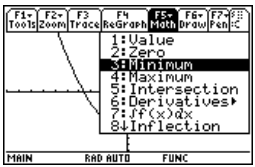
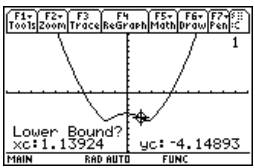
Etapas e teclas	Ecrã
3. Calcule a força quando m = 5 quilogramas (_kg) e a = 20 metros/segundo^2 (_ms2). Se conhecer a abreviatura de uma unidade, pode digitá-la no teclado. [calc] 5 [◆] [_] [2nd] [a-lock] K G [alpha] [×] 2 0 [◆] [_] [2nd] [a-lock] M S [alpha] 2 [ENTER] [calc] 5 [2nd] [_] K G [×] 2 0 [2nd] [_] M S 2 [ENTER]	
4. Utilize o mesmo m para calcular a força gerada pela aceleração da gravidade (a constante _g). Para obter _g, pode utilizar a constante predefinida disponível na caixa de diálogo **UNITS** ou pode digitar _g. [calc] 5 [◆] [_] [2nd] [a-lock] K G [alpha] [×] [2nd] [UNITS] [▶] [alpha] G [ENTER] [ENTER] [calc] 5 [2nd] [_] K G [×] [◆] [UNITS] [▶] G [ENTER] [ENTER]	
5. Converta em quilogramas-força (_kgf). [2nd] [▶] mostra o operador de conversão ▶. [calc] [▶] [2nd] [▶] [◆] [_] [2nd] [a-lock] K G F [alpha] [ENTER] [calc] [▶] [2nd] [▶] [2nd] [_] K G F [ENTER]	

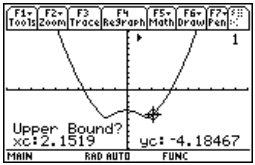
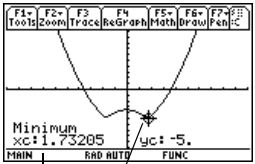
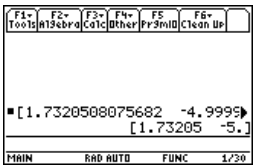
Representação gráfica de funções básicas I

O exemplo desta secção demonstra algumas das capacidades de representação gráfica da TI-89 Titanium / Voyage™ 200 e explica como representar graficamente uma função com o **Y= Editor**. O utilizador aprenderá a introduzir uma função, desenhar o gráfico, traçar uma curva, encontrar um ponto mínimo e transferir as coordenadas mínimas para o ecrã principal.

Explore as capacidades gráficas da TI-89 Titanium / Voyage™ 200 através da representação gráfica da função $y=(|x^2-3|-10)/2$.

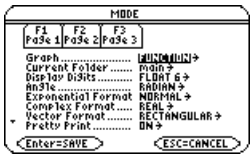
Etapas e teclas	Ecrã
1. Visualize o **Y= Editor**. Prima [◆] [Y=]	
2. Introduza a função **(abs(x²−3)−10)/2**. Apresentação em “Pretty print” da função na linha Y1= [([CATALOG] A [ENTER] X [^] 2 [−] 3 [)] [−] 1 0 [)] [÷] 2 [ENTER] [([2nd] [CATALOG] A [ENTER] X [^] 2 [−] 3 [)] [−] 1 0 [)] [÷] 2 [ENTER]	

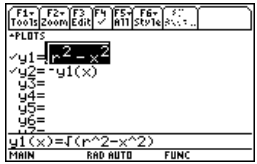
Etapas e teclas	Ecrã
3. Visualize o gráfico da função. Seleccione **6:ZoomStd**, premindo **6** ou deslocando o cursor para **6:ZoomStd** e premindo [ENTER]. Prima [F2] 6	
4. Active **Trace**. Aparece o cursor de traçado e as coordenadas x e y. Prima [F3]	 Cursor de traçado
5. Abra o menu **MATH** e seleccione **3:Minimum**. Prima [F5] ⊖ ⊖ [ENTER]	
6. Defina o limite inferior. Prima ⊕ (cursor direito) para posicionar o cursor de traçado até que o limite inferior da função em relação a x fique situado esquerda do ponto mínimo e volte a premir [ENTER]. Prima ⊕ ... ⊕ [ENTER]	

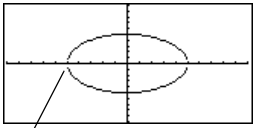
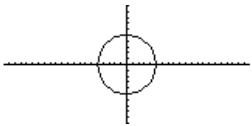
Etapas e teclas	Ecrã
7. Defina o limite superior. Prima ⊙ (cursor direito) para posicionar o cursor de traçado até que o limite superior da função em relação a x fique situado à direita do ponto mínimo. Prima ⊙ ... ⊙	
8. Localize o ponto mínimo no gráfico entre o limite inferior e o superior. Prima [ENTER]	
9. Transfira o resultado para o ecrã principal e visualize-o. [◆] [(-)] [HOME] [◆] H [◆] [CALC HOME]	

Representação gráfica de funções básicas II

Represente uma circunferência com raio igual a 5, centrada na origem do sistema de coordenadas. Visualize a circunferência na janela de visualização padrão (**ZoomStd**) e ajuste a janela com **ZoomSqr**.

Etapas e teclas	Ecrã
1. Visualize a caixa de diálogo **MODE**. Seleccione o modo **Graph FUNCTION**. Prima [MODE] ⊙ 1 [ENTER]	
2. Visualize o ecrã principal e guarde o raio 5 na variável r. [HOME] 5 [STO▸] [alpha] R [ENTER] [◆] [CALC HOME] 5 [STO▸] R [ENTER]	5→r

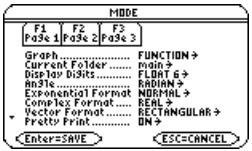
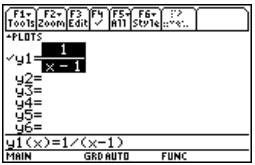
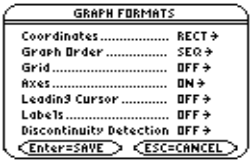
Etapas e teclas	Ecrã
3. Visualize e cancele o **Y= Editor**. Defina y1(x) = $\sqrt{(r^2 - x^2)}$, a parte superior de uma circunferência. Na representação de funções, tem de definir funções separadas para as metades superior e inferior de uma circunferência. [♦] [Y=] [F1] 8 [ENTER] [ENTER] [2nd] [√] [alpha] R [^] 2 [−] X [^] 2 [)] [ENTER] [♦] [Y=] [F1] 8 [ENTER] [ENTER] [2nd] [√] R [^] 2 [−] X [^] 2 [)] [ENTER]	
4. Defina y2(x) = $-\sqrt{r^2 - x^2}$, a função da metade inferior da circunferência. A metade inferior corresponde à função negativa da metade superior para que possa definir y2(x) = ⁻y1(x). Utilize o nome completo da função **y1(x),** não apenas y1. Prima [ENTER] [(-)] Y 1 [(] X [)] [ENTER]	

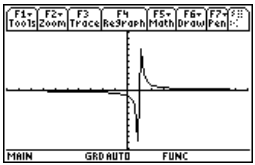
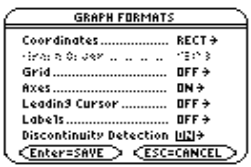
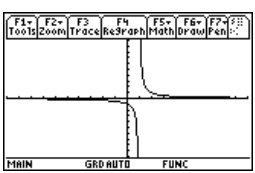
Etapas e teclas	Ecrã
5. Seleccione a janela de visualização **ZoomStd**, que representa as funções automaticamente. Na janela de visualização padrão, os eixos x e y compreendem valores de ⁻10 a 10. No entanto, o comprimento horizontal do ecrã é maior do que o comprimento vertical e, consequentemente, a circunferência aparece como uma elipse. Prima F2 6	 Não se esqueça de que as metades superior e inferior estão fisicamente separadas.
6. Seleccione **ZoomSqr**. **ZoomSqr** incrementa o intervalo do eixo x para que as circunferências e os quadrados apareçam na proporção correcta. Prima F2 5	

Nota: as metades superior e inferior da circunferência estão fisicamente separadas porque cada uma representa uma função independente. Os extremos matemáticos de cada metade são (⁻5,0) e (5,0). Dependendo da janela de visualização, os extremos representados de cada metade podem variar dos extremos matemáticos.

Representação gráfica de funções básicas III

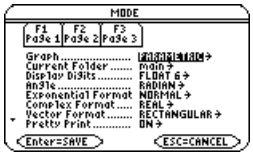
Utilize o formato de gráfico “Detect Discontinuities” para eliminar falsas assimptotas e ligações num descontinuidade de saltos.

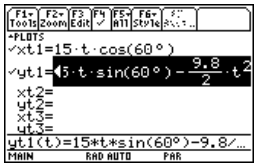
Etapas e teclas	Ecrã
1. Visualize a caixa de diálogo **MODE**. Para o modo **Graph**, seleccione **FUNCTION**. Para o modo **Angle**, seleccione **RADIAN**. Prima [MODE] ▶ 1 ▼ ▼ ▼ ▶ 1 [ENTER]	
2. Abra o Y = Editor e introduza y1(x)=1/(x-1). Prima ◆ [Y=] 1 ÷ (X − 1) [ENTER]	
3. Visualize a caixa de diálogo Graph Formats e desactive “Detect Discontinuities” **Nota:** O segundo item da caixa de diálogo Graph Format não está acinzentado, o que significa que pode ser definido para sequencial “Seq” ou simultâneo “Simul”. ▦ ◆ [|] ▼ ▼ ▼ ▼ ▼ ▼ ▶ 1 [ENTER] ▭ ◆ F ▼ ▼ ▼ ▼ ▼ ▼ ▶ 1 [ENTER]	

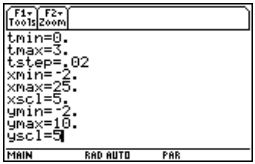
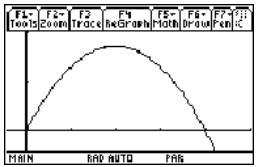
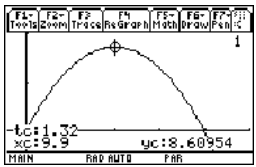
Etapas e teclas	Ecrã
4. Execute o comando **Graph**, que visualiza automaticamente o ecrã Graph. Observe as “falsas” assimptotas contidas no gráfico. Prima ◆ [GRAPH]	
5. Visualize a caixa de diálogo Graph Formats e active “Detect Discontinuities”. **Nota:** O segundo item na caixa de diálogo Graph Format está acinzentado, o que significa que a ordem do gráfico está definida para sequencial “Seq”. ◆ [I] ▼ ▼ ▼ ▼ ▼ ▼ ▶ 2 [ENTER] ◆ F▼ ▼ ▼ ▼ ▼ ▼ ▶ 2 [ENTER]	
6. Execute o comando **Graph**, que visualiza automaticamente o ecrã Graph. Não existem “falsas” assimptotas no gráfico. **Nota:** A velocidade da função gráfica pode diminuir consideravelmente quando activar “Detect Discontinuities”. Prima ◆ [GRAPH]	

Representação gráfica de equações paramétricas

Represente graficamente as equações paramétricas que descrevem a trajectória de uma bola chutada com um ângulo (θ) de 60° e uma velocidade inicial (v_0) de 15 metros por segundo. A constante de gravidade é g = 9.8 metros/seg.2. Ignorando a resistência do ar e outras forças de atrito, qual é a altura máxima da bola e em que momento tocará o solo?

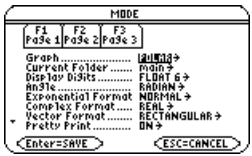
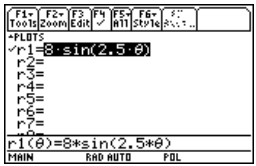
Etapas e teclas	Ecrã
1. Visualize a caixa de diálogo **MODE**. Defina o modo **Graph** em **PARAMETRIC**. Prima [MODE] ▶ 2 [ENTER]	

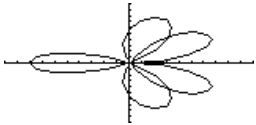
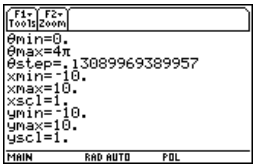
Etapas e teclas	Ecrã
2. Visualize e limpe **Y= Editor**. Defina a componente horizontal xt1(t) = v_0t cos θ. Introduza um valor para v_0 e θ. [◆] [Y=] [F1] 8 [ENTER] [ENTER] 15T [×] [2nd] [COS] 60 [2nd] [°] [)] [ENTER] [◆] [Y=] [F1] 8 [ENTER] [ENTER] 15T [×] [COS] 60 [2nd] [°] [)] [ENTER] **TI-89 Titanium:** Escreva T [×] [2nd] [COS], não T [2nd] [COS]. **Voyage™ 200:** Escreva T [×] [COS], não T [COS]. Introduza o símbolo °, digitando [2nd] [°] ou [2nd] [MATH] 2 1. Desta forma, o número é interpretado em graus independentemente do modo do ângulo.	xt1(t)=15t∗cos(60°)
3. Defina a componente vertical yt1(t) = v_0t sin θ − (g/2)t^2. Introduza um valor para v_0, θ, e g. [ENTER] 1 5 T [×] [2nd] [SIN] 6 0 [2nd] [°] [)] [−] [(] 9 . 8 [÷] 2 [)] T [^] 2 [ENTER] [ENTER] 1 5 T [×] [SIN] 6 0 [2nd] [°] [)] [−] [(] 9 . 8 [÷] 2 [)] T [^] 2 [ENTER]	

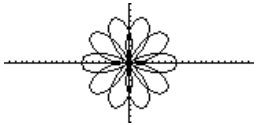
Etapas e teclas	Ecrã
4. Visualize o **Window Editor**. Introduza as variáveis Window adequadas a este exemplo. Para introduzir um valor e passar à próxima variável, prima ⊖ ou [ENTER]. Prima [◆] [WINDOW] 0 ⊖ 3 ⊖ . 0 2 ⊖ [(-)] 2 ⊖ 2 5 ⊖ 5 ⊖ [(-)] 2 ⊖ 1 0 ⊖ 5	
5. Represente graficamente as equações paramétricas para obter o modelo da trajectória da bola. Prima [◆] [GRAPH]	
6. Seleccione **Trace**. Desloque o cursor ao longo da trajectória para encontrar: • o valor de y relativo à altura máxima. • o valor de t relativo ao momento em que a bola bate no solo. Prima [F3] ⊕ ou ⊖ conforme necessário	

Representação gráfica de equações polares

O gráfico da equação polar r1(θ) = A sin Bθ possui a forma de uma rosa. Represente-a graficamente para A=8 e B=2,5. Em seguida, analise a forma da rosa com outros valores para A e B.

Etapas e teclas	Ecrã
1. Visualize a caixa de diálogo **MODE**. No modo **Graph**, seleccione **POLAR**. No modo **Angle**, seleccione **RADIAN**. Prima [MODE] ▶ 3 ▼ ▼ ▼ ▶ 1 [ENTER]	
2. Visualize e limpe o **Y= Editor**. Em seguida, defina a equação polar r1(θ) = A sin Bθ. Introduza 8 e 2.5 para A e B, respectivamente. [◆] [Y=] [F1] 8 [ENTER] [ENTER] 8 [2nd] [SIN] 2 . 5 [◆] [θ] [)] [ENTER] [◆] [Y=] [F1] 8 [ENTER] [ENTER] 8 [SIN] 2 . 5 [θ] [)] [ENTER]	

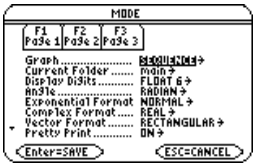
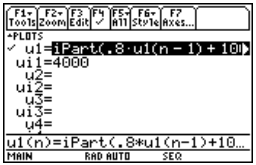
Etapas e teclas	Ecrã
3. Seleccione a janela de visualização **ZoomStd**, que representa a equação graficamente. • O gráfico apresenta apenas cinco pétalas de rosa. - Na janela de visualização padrão, a variável Window θmax é igual a 2π. As pétalas restantes possuem valores θ maiores do que 2π. • A rosa não está simétrica. - Os eixos x e y possuem valores entre ⁻10 e 10. Entretanto, este intervalo é mais espaçado ao longo do eixo x do que no eixo y. Prima [F2] 6	
4. Visualize o **Window Editor** e mude θmax para 4π. O valor de 4π é obtido quando sair do **Window Editor**. Prima [◆] [WINDOW] ⊖ 4 [2nd] [π]	

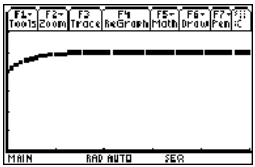
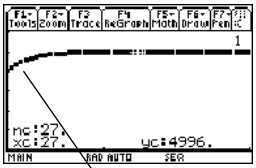
Etapas e teclas	Ecrã
5. Seleccione **ZoomSqr**, que representa graficamente a equação novamente. **ZoomSqr** aumenta o intervalo entre os valores ao longo do eixo x para que o gráfico apareça na proporção correcta. Prima F2 5	
Pode mudar os valores de *A* e *B* como desejado e representar graficamente a equação.	

Representação gráfica de sucessões

Uma pequena floresta contém 4000 árvores. Cada ano, 20% das árvores serão cortadas (permanecendo 80%) e 1000 novas árvores serão plantadas. Calcule o número de árvores na floresta no final de cada ano com uma sucessão. Este número estabiliza-se?

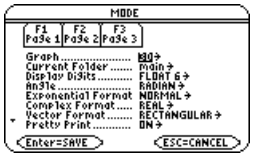
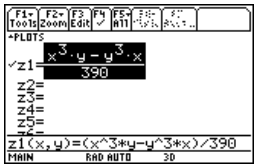
Início	Depois de 1 ano	Depois de 2 anos	Depois de 3 anos	. . .
4000	.8 x 4000 + 1000	.8 x (.8 x 4000 + 1000) + 1000	.8 x (.8 x (.8 x 4000 + 1000) + 1000) + 1000	. . .

Etapas e teclas	Ecrã
1. Visualize a caixa de diálogo **MODE**. Para o modo **Graph**, seleccione **SEQUENCE**. Prima [MODE] ▶ 4 [ENTER]	
2. Visualize e limpe **Y= Editor**. Em seguida, defina a sucessão como u1(n) = iPart(.8∗u1(n−1)+1000). Utilize **iPart** para recuperar a parte inteira do resultado. Não são cortadas fracções de árvores. Para aceder a **iPart(**, pode utilizar [2nd] [MATH], basta digitá-lo ou seleccioná-lo a partir de **CATALOG**. [◆] [Y=] [F1] 8 [ENTER] [ENTER] [2nd] [MATH] 1 4 . 8 [alpha] U 1 [(] [alpha] N [−] 1 [)] [+] 1 0 0 0 [)] [ENTER] [◆] [Y=] [F1] 8 [ENTER] [ENTER] [2nd] [MATH] 1 4 . 8 U 1 [(] N [−] 1 [)] [+] 1 0 0 0 [)] [ENTER]	
3. Defina ui1 como o valor inicial, que será utilizado como o primeiro termo. Prima [ENTER] 4 0 0 0 [ENTER]	

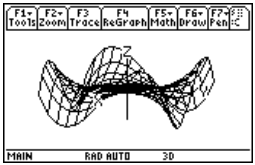
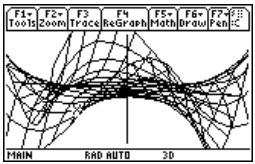
Etapas e teclas	Ecrã
4. Visualize o **Window Editor**. Defina n e desenhe as variáveis de Window. **nmin=0** e **nmax=50** calculam o tamanho da floresta nos próximos 50 anos. Prima ◆ [WINDOW] 0 ⊖ 5 0 ⊖ 1 ⊖ 1 ⊖	nmin=0. nmax=50. plotStrt=1. plotStep=1. xmin=0. xmax=50. xscl=10. ymin=0. ymax=6000. yscl=1000.
5. Defina as variáveis Window x e y com os valores adequados para este exemplo. Prima 0 ⊖ 5 0 ⊖ 1 0 ⊖ 0 ⊖ 6 0 0 0 ⊖ 1 0 0 0	
6. Visualize o ecrã Graph. Prima ◆ [GRAPH]	
7. Seleccione **Trace**. Mova o cursor para representar ano a ano. Quantos anos (nc) são necessários para estabilizar o número de árvores (yc)? A representação começa em nc=0. nc é o número de anos. xc = nc já que n é representado no eixo x. yc = u1(n), o número de árvores no ano n. Prima F3 ⊕ e ⊖ como necessário	 Por predefinição, as sucessões utilizam o estilo de visualização Square.

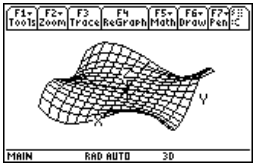
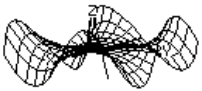
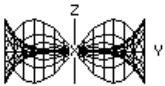
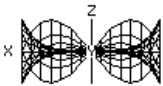
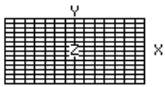
Representação gráfica em 3D

Represente graficamente a equação 3D z(x,y) = ($x^3y - y^3x$) / 390. Controle a animação do gráfico com o cursor para alterar os valores da variável eye de Window de forma interactiva, que controla o ângulo de visualização. Em seguida, visualize o gráfico em estilos de formatos gráficos diferentes.

Etapas e teclas	Ecrã
1. Visualize a caixa de diálogo **MODE**. Seleccione **3D** para o modo **Graph**. Prima [MODE] ▶ 5 [ENTER]	
2. Visualize e limpe **Y= Editor**. Em seguida, defina a equação 3D z1(x,y) = ($x^3y - y^3x$) / 390. Não se esqueça de que a multiplicação implícita é utilizada nas teclas premidas. Prima [◆] [Y=] [F1] 8 [ENTER] [ENTER] [(] X [^] 3 Y [−] Y [^] 3 X [)] [÷] 3 9 0 [ENTER]	

Etapas e teclas	Ecrã
3. Mude o formato gráfico para visualizar e identifique os eixos. Além disso, defina **Style = WIRE FRAME**. É possível animar qualquer estilo de formato gráfico, mas **WIRE FRAME** é mais rápido. [◆] [I] ⊖ ⊕ 2 ⊖ ⊕ 2 ⊖ ⊕ 1 [ENTER] [◆] F ⊖ ⊕ 2 ⊖ ⊕ 2 ⊖ ⊕ 1 [ENTER]	GRAPH FORMATS Coordinates RECT→ Axes..... AXES→ Labels..... ON→ Style..... WIRE FRAME→ Enter=SAVE ESC=CANCEL

Etapas e teclas	Ecrã
4. Seleccione o cubo de visualização **ZoomStd**, que representa graficamente a equação de forma automática. À medida que a equação é calculada (antes de ser representada graficamente), a “percentagem de cálculo” aparece na parte superior esquerda do ecrã. Prima [F2] 6 **Nota:** Se já tiver utilizado a representação gráfica 3D, o gráfico pode aparecer na visualização expandida. Quando o gráfico é animado, o ecrã volta à visualização normal automaticamente. (Excepto para a animação, a visualização normal e expandida oferecem as mesmas possibilidades.) Prima [×] (prima [×] para alternar entre visualização normal e expandida)	 

Etapas e teclas	Ecrã
5. Anime o gráfico através da diminuição do valor da variável eyeϕ de Window. ⊖ ou ⊕ pode afectar eyeθ e eyeψ. No entanto, de forma mais suave do que a eyeϕ. Para animar o gráfico continuamente, prima sem soltar o cursor durante 1 segundo e, em seguida, solte-o. Para interromper, prima [ENTER]. Prima ⊖ ⊖ ⊖ ⊖ ⊖ ⊖ ⊖ ⊖	
6. Coloque o gráfico novamente na orientação inicial. Em seguida, mova o ângulo de visualização ao longo da "órbita de visualização" à volta do gráfico. Prima 0 (zero, não a letra O) ⊖ ⊖ ⊖	
7. Visualize o gráfico ao longo do eixo x , do eixo y e, em seguida, do eixo z. Prima X Este gráfico possui a mesma forma ao longo do eixo y e do eixo x. Prima Y Prima Z	  

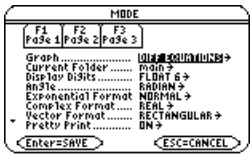
Etapas e teclas	Ecrã
8. Volte à orientação inicial. Prima 0 (zero)	

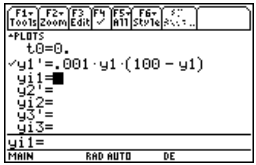
Etapas e teclas	Ecrã
9. Visualize o gráfico em diferentes estilos de formato de gráfico. [I] (prima [I] para mudar de um estilo para o próximo) F (prima F para mudar de um estilo para o próximo)	 SUPERFÍCIE OCULTA NÍVEIS DE CONTORNO (pode ser necessário algum tempo para calcular os contornos) FIO E CONTORNO MOLDURA

Nota: Também é possível visualizar o gráfico como um traçado implícito com a caixa de diálogo **GRAPH FORMATS** (F1 **9** ou TI-89 Titanium: ◆ I Voyage™ 200: ◆ **F**). Se TI-89 Titanium: I Voyage™ 200: **F** for premido para mudar de estilo, o gráfico implícito não aparece.

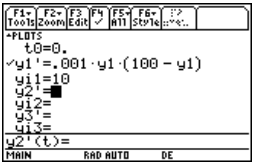
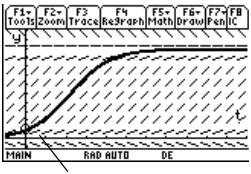
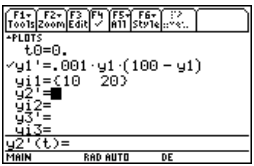
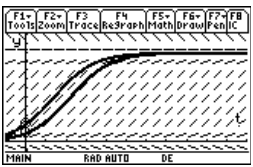
Representação gráfica de equação diferencial

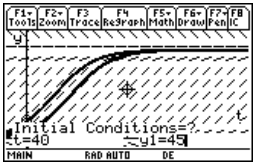
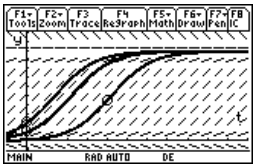
Represente graficamente a solução da equação diferencial de 1ª ordem y' = .001y∗((100−y). Comece por desenhar apenas o campo de inclinação. Em seguida, introduza as condições iniciais em **Y= Editor** e interactivamente a partir do ecrã Graph.

Etapas e teclas	Ecrã
1. Visualize a caixa de diálogo **MODE**. Para o modo **Graph**, seleccione **DIFF EQUATIONS**. Pressione MODE ▷ 6 ENTER	

Etapas e teclas	Ecrã
2. Visualize e limpe **Y= Editor**. Em seguida, defina a equação diferencial de 1ª ordem y1'(t)=.001y1∗(100−y1) Prima [×] para introduzir o ∗ apresentado acima. Não utilize multiplicação implícita entre uma variável e o parêntesis. Caso contrário, o conjunto é tratado como uma chamada de uma função. Deixe a condição inicial **yi1** em branco. **Nota:** Com y1' seleccionado, a TI-89 Titanium / Voyage™ 200 representará graficamente a curva solução de y1, não a derivada y1'. Prima [◆] [Y=] [F1] 8 [ENTER] [ENTER] . 0 0 1 Y 1 [×] [(] 1 0 0 [−] Y 1 [)] [ENTER]	

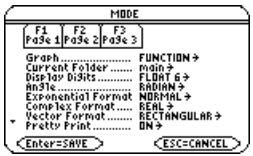
Etapas e teclas	Ecrã
3. Visualize a caixa de diálogo **GRAPH FORMATS**. Em seguida, defina **Axes = ON**, **Labels = ON**, **Solution Method = RK** e **Fields = SLPFLD**. **Nota:** Para representar uma equação diferencial, tem de definir Fields como **SLPFLD** ou **FLDOFF**. Se **Fields=DIRFLD**, ocorre um erro na representação gráfica. [♦] [I] ⊖ ⊖ ⊕ 2 ⊖ ⊖ ⊕ 2 ⊖ ⊕ 1 ⊖ ⊕ 1 [ENTER] [♦] F ⊖ ⊖ ⊕ 2 ⊖ ⊖ ⊕ 2 ⊖ ⊕ 1 ⊖ ⊕ 1 [ENTER]	GRAPH FORMATS Coordinates..... RECT→ Grid..... OFF→ Axes..... ON→ Leading Cursor. OFF→ Labels..... ON→ Solution Method RK→ Fields..... SLPFLD→ Enter=SAVE ESC=CANCEL
4. Visualize o **Window Editor** e defina as variáveis de Window como apresentado do lado direito. Pressione [♦] [WINDOW] 0 ⊖ 1 0 ⊖ .1 ⊖ 0 ⊖ [(-)] 1 0 ⊖ 1 1 0 ⊖ 1 0 ⊖ [(-)] 1 0 ⊖ 1 2 0 ⊖ 1 0 ⊖ 0 ⊖ . 0 0 1 ⊖ 2 0	t0=0. tmax=10. tstep=.1 tplot=0. xmin=-10. xmax=110. xscl=10. ymin=-10. ymax=120. yscl=10. ncurves=0. diftol=.001 fldres=20.
5. Visualize o ecrã Graph. Como não foi especificada uma condição inicial, só é desenhado o campo de inclinação (como especificado por **Fields=SLPFLD** na caixa de diálogo **GRAPH FORMATS**). Prima [♦] [GRAPH]	F1▾ Tools F2▾ Zoom F3 Trace F4 ReGraph F5▾ Math F6▾ Draw F7▾ Pen F8 IC y t MAIN RAD AUTO DE

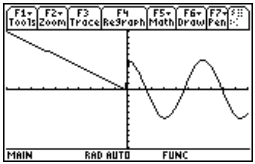
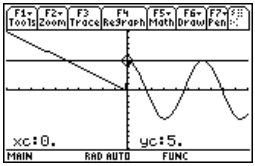
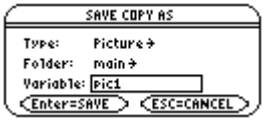
Etapas e teclas	Ecrã
6. Volte ao **Y= Editor** e introduza a condição inicial: **yi1=10** Prima ◆ [Y=] [ENTER] 1 0 [ENTER]	
7. Volte ao ecrã Graph. As condições iniciais introduzidas em **Y= Editor** ocorrem sempre em t0. O gráfico começa na condição inicial e é traçado para a direita. Em seguida, é traçado para a esquerda. Prima ◆ [GRAPH]	 A condição inicial é marcada com um círculo.
8. Volte ao **Y= Editor** e mude **yi1** para introduzir duas condições iniciais em forma de uma lista: **yi1={10,20}** Prima ◆ [Y=] ⊙ [ENTER] [2nd] [{] 1 0 [,] 2 0 [2nd] [}] [ENTER]	
9. Volte ao ecrã Graph. Prima ◆ [GRAPH]	

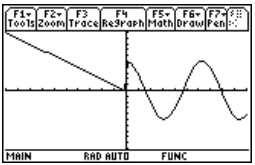
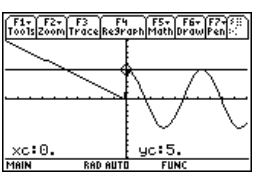
Etapas e teclas	Ecrã
10. Para seleccionar uma condição inicial interactivamente, prima: [2nd] [F8] [F8] No aviso da linha de comandos, digite t=40 e y1=45.	
Se seleccionar uma condição inicial interactivamente, um valor diferente do valor t_0 introduzido em **Y= Editor** ou **Window Editor** é especificado para t. Em vez de introduzir **t** e **y1** depois de premir [2nd] [F8] [F8] pode mover o cursor para um ponto sobre o ecrã e, em seguida, premir [ENTER]. Pode utilizar [F3] para traçar curvas para as condições iniciais especificadas no **Y= Editor**. Entretanto, não pode traçar a curva para uma condição inicial seleccionada interactivamente. [2nd] [F8] 4 0 [ENTER] 4 5 [ENTER] [F8] 4 0 [ENTER] 4 5 [ENTER]	

Tópicos complementares de representação gráfica

Represente graficamente a função definida por partes: y = ⁻x para x < 0 e y = 5 cos(x) para x ≥ 0 no ecrã principal. Trace uma recta horizontal que contenha o máximo da curva do co-seno. Em seguida, guarde a imagem do gráfico apresentado.

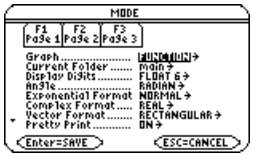
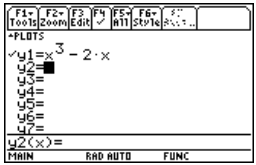
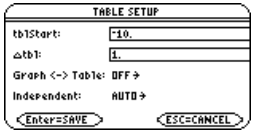
Etapas e teclas	Ecrã
1. Visualize a caixa de diálogo **MODE**. Para o modo **Graph**, seleccione **FUNCTION**. Para o modo **Angle**, seleccione **RADIAN**. Prima [MODE] ▶ 1 ▼ ▼ ▼ ▶ 1 [ENTER]	
2. Visualize o ecrã principal. Utilize o comando **Graph** e a função **when** para especificar a função definida por partes. Pode seleccionar **Graph** do menu **Other** da barra de ferramentas com [F4] **2** e adicionar um espaço automaticamente. [HOME] [F4] 2 [2nd] [a-lock] W H E N [alpha] [(] X [2nd] [<] 0 [,] [(-)] X [,] 5 [×] [2nd] [COS] X [)] [)] [◆] [CALC HOME] [F4] 2 W H E N [(] X [2nd] [<] 0 [,] [(-)] X [,] 5 [×] [COS] X [)] [)]	Gráfico quando(x<0,⁻x,

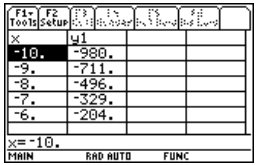
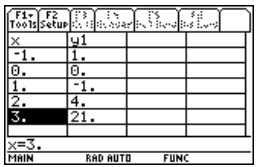
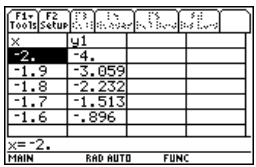
Etapas e teclas	Ecrã
3. Execute o comando **Graph**, que apresenta automaticamente o ecrã Graph. O gráfico utiliza as variáveis Window actuais, que são assumidas como sendo os valores padrão ([F2] **6**) para este exemplo. Prima [ENTER]	
4. Trace uma recta horizontal através da parte superior da curva do co-seno. A calculadora permanece no modo “horizontal” até seleccionar uma operação diferente ou premir [ESC]. [2nd] [F7] 5 ⊕ (até que a recta seja posicionada) [ENTER] [F7] 5 ⊕ (até que a recta seja posicionada) [ENTER]	
5. Guarde uma imagem do gráfico. Utilize **PIC1** como o nome da imagem. Não se esqueça de definir **Type = Picture**. Por predefinição, a definição é **GDB**. [F1] 2 ⊕ 2 ⊖ ⊖ P I C [alpha] 1 [ENTER] [ENTER] [F1] 2 ⊕ 2 ⊖ ⊖ P I C 1 [ENTER] [ENTER]	

Etapas e teclas	Ecrã
6. Apague a recta horizontal traçada. Pode premir também [F4] para traçar novamente o gráfico. [2nd] [F6] 1 [F6] 1	
7. Abra a imagem guardada para visualizar novamente o gráfico com a recta. Não se esqueça de definir **Type = Picture**. Por predefinição, a definição é **GDB**. Prima [F1] 1 ▶ 2 (se não aparecer, defina também Variable = pic1) [ENTER]	 

Tabelas

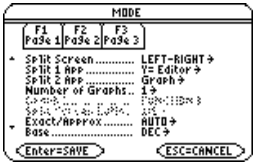
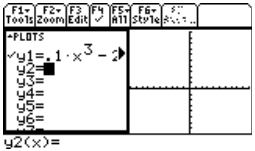
Determine os valores assumidos pela função $y=x^3-2x$ para cada número inteiro compreendido entre ⁻10 e 10. Quantas são as trocas de sinais e onde ocorrem?

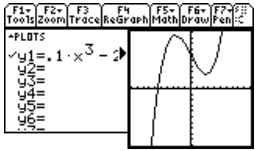
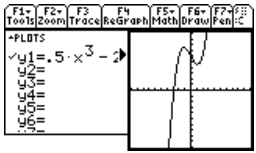
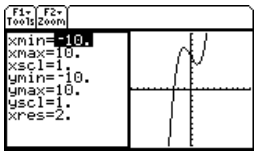
Etapas e teclas	Ecrã
1. Visualize a caixa de diálogo **MODE**. No modo **Graph**, seleccione **FUNCTION**. Prima [MODE] ▶ 1 [ENTER]	
2. Visualize e cancele **Y= Editor** e, em seguida, defina $y1(x) = x^3 - 2x$. Prima [◆] [Y=] [F1] 8 [ENTER] [ENTER] X [^] 3 [−] 2 X [ENTER]	
3. Defina os parâmetros da tabela em: **tblStart = ⁻10** **Δtbl = 1** **Graph < - > Table = OFF** **Independent = AUTO** Prima [◆] [TBLSET] [(-)] 1 0 ▼ 1 ▼ ▶ 1 ▼ ▶ 1 [ENTER]	

Etapas e teclas	Ecrã
4. Visualize o ecrã Prima [◆] [TABLE]	
5. Percorra a tabela. Não se esqueça de que **y1** muda de sinal em x = ⁻1, 1, e 2 Para percorrer as páginas, utilize [2nd] ⊖ e [2nd] ⊕. Prima ⊖ e ⊕ conforme necessário	
6. É possível aumentar a área da troca de sinal entre x = ⁻2 e x = ⁻1, modificando os parâmetros da tabela da seguinte maneira: **tblStart = ⁻2** **Δtbl = .1** Prima [F2] [(-)] 2 ⊖ .1 [ENTER] [ENTER]	

Ecrãs divididos

Divida o ecrã para visualizar o **Y= Editor** e o ecrã Graph. Em seguida, explore o comportamento de um polinómio conforme a variação dos coeficientes.

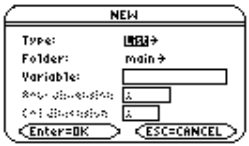
Etapas e teclas	Ecrã
1. Visualize a caixa de diálogo **MODE**. Na opção Graph, seleccione **FUNCTION**. Na opção **Split Screen**, seleccione **LEFT-RIGHT**. Na opção **Split 1 App**, seleccione **Y= Editor**. Na opção **Split 2 App**, seleccione **Graph**. Prima [MODE] ▶ 1 [F2] ▶ 3 ▼ ▶ 2 ▼ ▶ 4 [ENTER]	
2. Cancele **Y= Editor** e desactive todas as representações gráficas de dados estatísticos. Em seguida, defina $y1(x) = .1x^3 - 2x+6$. Um limite grosso à volta de **Y= Editor** indica que este ecrã está activo. Quando estiver activo, a linha de entrada atravessa o visor. Prima [F1] 8 [ENTER] [F5] 5 [ENTER] .1 X [^] 3 [−] 2 X [+] 6 [ENTER]	

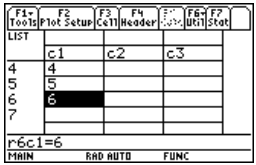
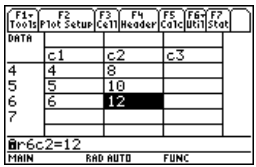
Etapas e teclas	Ecrã
3. Seleccione a janela de visualização **ZoomStd**, que alterna para o ecrã Graph e representa graficamente a função. O limite grosso aparece à volta do ecrã Graph. Prima [F2] 6	
4. Alterne para **Y= Editor**. Em seguida, edite **y1(x)** para substituir $.1x^3$ to $.5x^3$. [2nd] [⊞] é a segunda função de [APPS]. O limite grosso aparece à volta de **Y= Editor**. Prima [2nd] [⊞] ⊕ [ENTER] ⊖ ⊕ ⊕ ← 5 [ENTER]	
5. Alterne para o ecrã Graph, que representa graficamente a função editada novamente. O limite grosso aparece à volta do ecrã Graph. Prima [2nd] [⊞]	
6. Alterne para o **Y= Editor**. Em seguida, abra o **Window Editor** no seu lugar. Prima [2nd] [⊞] [◆] [WINDOW]	

Etapas e teclas	Ecrã
7. Abra o ecrã principal. Em seguida, saia para um ecrã principal de tamanho máximo. Prima: [2nd] [QUIT] [HOME] [2nd] [QUIT]	

Editor de Dados/Matrizes

Utilize o **Editor de Dados/Matrizes** para criar uma variável de lista de uma única coluna e, em seguida, adicione uma segunda coluna de informações. Não se esqueça de que a variável de lista (que pode ter uma única coluna) é convertida automaticamente numa variável de dados (que pode ter várias colunas).

Etapas e teclas	Ecrã
1. Inicie o **Editor de Dados/Matrizes** e crie uma nova variável de lista denominada **TEMP**. Prima 3 ▶ 3 ▼ ▼ T E M P [ENTER] [ENTER]	

Etapas e teclas	Ecrã
2. Introduza uma coluna de números e desloque o cursor para a célula superior (para verificar se o valor da célula destacada está na linha de entrada). **LIST** aparece no canto superior esquerdo para indicar uma variável de lista. Para introduzir uma informação numa célula, é possível utilizar ⊖ em vez de [ENTER]. Prima 1 [ENTER] 2 [ENTER] 3 [ENTER] 4 [ENTER] 5 [ENTER] 6 [ENTER] ⊕	
3. Desloque o cursor para a coluna 2 e defina o cabeçalho da coluna de forma que seja o dobro da coluna 1. **DATA** aparece no canto superior esquerdo para indicar que a variável de lista foi convertida numa variável de dados. [calc] ⊕ [F4] 2 [×] [alpha] C 1 [ENTER] [calc] ⊕ [F4] 2 [×] C 1 [ENTER]	 🔒 significa que a célula está numa coluna definida.

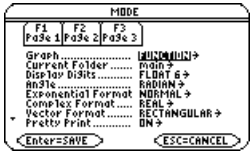
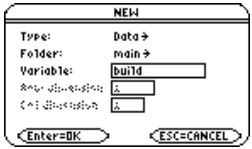
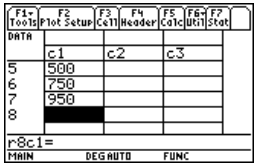
Etapas e teclas	Ecrã
4. Desloque o cursor para a célula de cabeçalho da coluna 2 para visualizar a definição na linha de entrada. Quando o cursor estiver posicionado na célula de cabeçalho, não tem de premir [F4] para a definir; basta começar a digitar a expressão. Prima [2nd] ⊕ ⊕	F1▾ Tools F2 Plot Setup F3 Cell F4 Header F5 Calc F6▾ Util F7 Stat DATA c1 c2 c3 1 1 2 2 2 4 3 3 6 4 4 8 c2=2*c1 MAIN RAD AUTO FUNC
5. Cancele o conteúdo da variável. A variável de dados não se converte numa variável de lista devido ao cancelamento de dados. Prima [F1] 8 [ENTER]	F1▾ Tools F2 Plot Setup F3 Cell F4 Header F5 Calc F6▾ Util F7 Stat DATA c1 c2 c3 1 2 3 4 r1c1= MAIN RAD AUTO FUNC

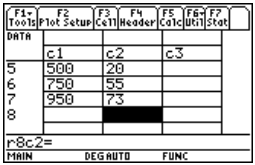
Nota: se não quiser guardar a variável actual, pode utilizá-la como memória intermédia. Da próxima vez que precisar utilizar uma variável provisoriamente, cancele a actual e volte a utilizá-la. Permite introduzir dados temporários sem desperdiçar memória através da criação de uma nova variável de cada vez.

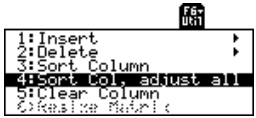
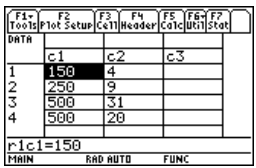
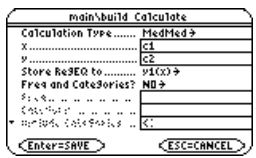
Gráficos estatísticos e de dados

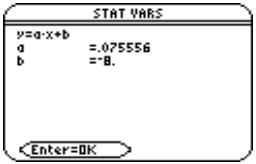
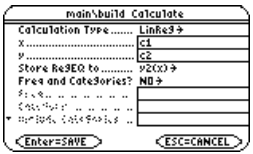
Com base numa amostra de 7 cidades, introduza dados que relacionem a população com o número de edifícios com mais de 12 andares. Utilize o comando Median-Median e os cálculos de regressão linear para localizar e traçar as equações de regressão correspondentes aos dados. Em cada equação de regressão, faça uma estimativa do

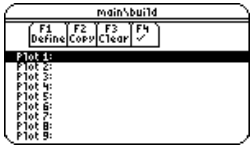
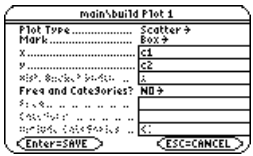
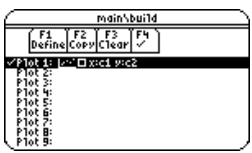
número de edifícios com mais de 12 andares que pode ter uma cidade com 300.000 habitantes.

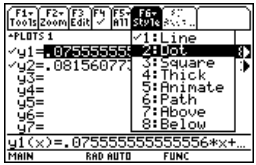
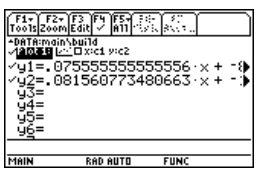
Etapas e teclas	Ecrã
1. Visualize a janela de diálogo **MODE**. Para o modo **Graph**, seleccione **FUNCTION**. Prima [MODE] ⊙ 1 [ENTER]	
2. Visualize o **Editor de Dados/Matrizes** e crie uma nova variável de dados com o nome **BUILD**. Prima 3 ⊙ ⊙ B U I L D [ENTER] [ENTER]	
3. Utilize os dados da amostra para introduzir a população na coluna 1. Pop. (em 1000) Edifícios > 12 andares 150 4 500 31 800 42 250 9 500 20 750 55 950 73 Prima 1 5 0 [ENTER] 5 0 0 [ENTER] 8 0 0 [ENTER] 2 5 0 [ENTER] 5 0 0 [ENTER] 7 5 0 [ENTER] 9 5 0 [ENTER]	

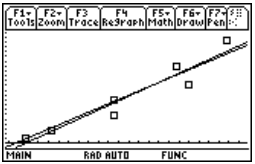
Etapas e teclas	Ecrã
4. Posicione o cursor na linha 1 da coluna 2 (r1c2). Em seguida, introduza o número correspondente de edifícios. [◆] ⊕ move o cursor para a parte superior da página. Depois de digitar os dados de uma célula, é possível premir [ENTER] ou ⊖ para introduzir os dados e deslocar o cursor uma célula para baixo. Quando premir ⊕, introduz os dados e desloca o cursor uma célula para cima. ⊕ [◆] ⊕ 4 [ENTER] 3 1 [ENTER] 4 2 [ENTER] 9 [ENTER] 2 0 [ENTER] 5 5 [ENTER] 7 3 [ENTER] ⊕ [2nd] ⊕ 4 [ENTER] 3 1 [ENTER] 4 2 [ENTER] 9 [ENTER] 2 0 [ENTER] 5 5 [ENTER] 7 3 [ENTER]	

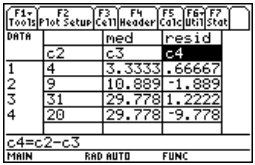
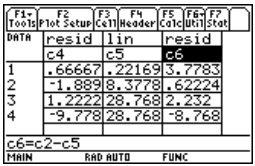
Etapas e teclas	Ecrã
5. Desloque o cursor para a linha 1 da coluna 1 (r1c1). Classifique os dados da população em ordem crescente. Esta operação classifica a coluna 1 e adapta as outras colunas para manter a ordem estabelecida na coluna 1. Esta classificação é essencial para manter a relação entre as colunas de dados. Para classificar a coluna 1, é possível posicionar o cursor em qualquer ponto da coluna. Neste exemplo, foi premido [◆] [▲] [2nd] [▲] para ver as quatro primeiras linhas. [◀] [◆] [▲] [2nd] [F6] 4 [◀] [2nd] [▲] [F6] 4	 
6. Visualize a caixa de diálogo **Calculate**. Defina: **Calculation Type = MedMed** **x = C1** **y = C2** **Store RegEQ to = y1(x)** [F5] [▶] 7 [▼] C [alpha] 1 [▼] [alpha] C 2 [▼] [▶] [▼] [ENTER] [F5] [▶] 7 [▼] C 1 [▼] C 2 [▼] [▶] [▼] [ENTER]	

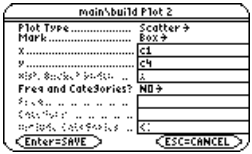
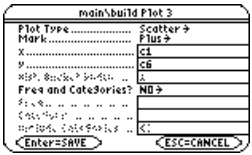
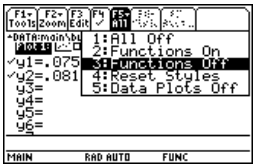
Etapas e teclas	Ecrã
7. Execute o cálculo para visualizar a equação de regressão MedMed. Como especificado na caixa de diálogo **Calculate**, esta equação é guardada em **y1(x)**. Prima [ENTER]	
8. Feche o ecrã **STAT VARS**. Aparece o **Editor de Dados/Matrizes**. Prima [ENTER]	
9. Visualize a caixa de diálogo **Calculate**. Defina: **Calculation Type = LinReg** **x = C1** **y = C2** **Store RegEQ to = y2(x)** Prima [F5] ▶ 5 ▼ ▼ ▼ ▶ ▼ [ENTER]	
10. Execute o cálculo para visualizar a equação de regressão LinReg. Esta equação é guardada em **y2(x)**. Prima [ENTER]	
11. Feche o ecrã **STAT VARS**. Aparece o **Editor de Dados/Matrizes**. Prima [ENTER]	

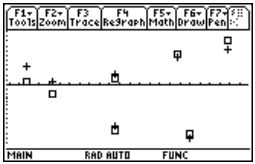
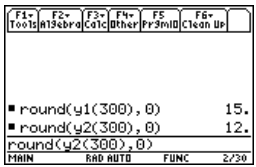
Etapas e teclas	Ecrã
12. Visualize o ecrã Plot Setup. **Plot 1** é realçado por predefinição. [F3] permite apagar a definição Plot realçada. Prima [F2]	
13. Defina **Plot 1** como: **Plot Type = Scatter** **Mark = Box** **x = C1** **y = C2** Não se esqueça de que a similaridade com a caixa de diálogo **Calculate**. [F1] ▶ 1 ▼ ▶ 1 ▼ C [alpha] 1 ▼ [alpha] C 2 [F1] ▶ 1 ▼ ▶ 1 ▼ C 1 ▼ C 2	
14. Guarde a definição do gráfico e volte ao ecrã Plot Setup. Não se esqueça da notação resumida da definição **Plot 1**. Prima [ENTER] [ENTER]	

Etapas e teclas	Ecrã
15. Visualize o **Y= Editor**. Para **y1(x)**, que é a equação de regressão MedMed, defina o estilo de visualização em **Dot**. **Nota:** dependendo do conteúdo anterior de **Y= Editor**, tem de deslocar o cursor para **y1**. **PLOTS 1** está na parte superior do ecrã e indica a selecção de **Plot 1**. Não se esqueça de que **y1(x)** e **y2(x)** foram seleccionadas quando as equações de regressão foram guardadas. [◆] [Y=] [2nd] [F6] 2 [◆] [Y=] [F6] 2	
16. Percorra para realçar **Plot 1**. A definição resumida apresentada é igual à definição do ecrã Plot Setup. Prima [⊙]	

Etapas e teclas	Ecrã
17. Utilize **ZoomData** para representar graficamente **Plot 1** e as equações de regressão **y1(x)** e **y2(x)**. **ZoomData** examina os dados dos gráficos estatísticos seleccionados e regula o tamanho da janela de visualização para que possa incluir todos os dados. Prima [F2] 9	
18. Volte à sessão actual do **Editor de Dados/Matrizes.** Prima [APPS] ⊖ [ENTER] [ENTER]	
19. Introduza um título para a coluna 3. Defina o cabeçalho da coluna 3 com os valores previstos pela reta MedMed. Para introduzir um título, tem de realçar a célula de cabeçalho localizada na parte superior da coluna. [F4] permite definir o cabeçalho a partir de qualquer parte da coluna. Se o cursor estiver na célula de cabeçalho, não tem de premir [F4]. ▯ ⊕ ⊕ ⊖ ⊖ [2nd] [a-lock] M E D [alpha] [ENTER] [F4] Y 1 [(] [alpha] C 1 [)] [ENTER] ▯ ⊕ ⊕ ⊖ ⊖ M E D [ENTER] [F4] Y 1 [(] C 1 [)] [ENTER]	

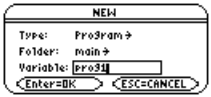
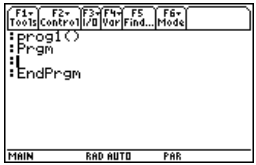
Etapas e teclas	Ecrã
20. Introduza um título para coluna 4. Defina como cabeçalho da coluna 4 os resíduos (diferença entre os valores observados e previstos) de MedMed. [calc] ▶ ▲ [2nd] [a-lock] R E S I D [alpha] [ENTER] [alpha] C 2 [−] [alpha] C 3 [ENTER] [classwiz] ▶ ▲ R E S I D [ENTER] [F4] C 2 [−] C 3 [ENTER]	
21. Introduza um título para coluna 5. Defina o cabeçalho da coluna 5 como os valores previstos pela recta LinReg. [calc] ▶ ▲ ▲ [2nd] [a-lock] L I N [alpha] [ENTER] [F4] Y 2 [(] [alpha] C 1 [)] [ENTER] [classwiz] ▶ ▲ L I N [ENTER] [F4] Y 2 [(] C 1 [)] [ENTER]	
22. Introduza um título para a coluna 6. Defina o cabeçalho da coluna 6 com os resíduos de LinReg. [calc] ▶ ▲ [2nd] [a-lock] R E S I D [alpha] [ENTER] [F4] [alpha] C 2 [−] [alpha] C 5 [ENTER] [classwiz] ▶ ▲ R E S I D [ENTER] [F4] C 2 [−] C 5 [ENTER]	
23. Visualize o ecrã Plot Setup e anule **Plot 1**. Prima [F2] [F4]	

Etapas e teclas	Ecrã
24. Realce **Plot 2** e defina-o da seguinte maneira: **Plot Type = Scatter** **Mark = Box** **x = C1** **y = C4** (resíduos de MedMed) ▼ F1 ▼ ▼ C alpha 1 ▼ alpha C 4 ENTER ENTER ▼ F1 ▼ ▼ C 1 ▼ C 4 ENTER ENTER	
25. Realce **Plot 3** e defina-o da seguinte maneira: **Plot Type = Scatter** **Mark = Plus** **x = C1** **y = C6** (resíduos de LinReg) ▼ F1 ▼ ▶ 3 ▼ C alpha 1 ▼ alpha C 6 ENTER ENTER ▼ F1 ▼ ▶ 3 ▼ C 1 ▼ C 6 ENTER ENTER	
26. Visualize o **Y= Editor** e desactive todas as funções **y(x)**. F5, seleccione **3:Functions Off** e não **1:All Off**. **Plots 2** e **3** continuam seleccionados. Prima ◆ [Y=] F5 3	

Etapas e teclas	Ecrã
27. Utilize **ZoomData** para representar graficamente os resíduos. □ identifica os resíduos de MedMed; + identifica os resíduos de LinReg. Pressione [F2] 9	
28. Visualize o ecrã principal. [HOME] [◆] [CALC HOME]	
29. Utilize as equações de regressão MedMed (**y1(x)**) e LinReg (**y2(x)**) para calcular os valores com x = 300 (300.000 habitantes). A função **round** ([2nd] [MATH] 1 3) garante que o resultado visualizado seja um número inteiro de edifícios. Depois de calcular o primeiro resultado, substitua na linha de entrada **y1** por **y2**. Prima [2nd] [MATH] 1 3 Y 1 [(] 3 0 0 [)] [,] 0 [)] [ENTER] ▶ ◀ ◀ ◀ ◀ ◀ ◀ ◀ ◀ [←] 2 [ENTER]	

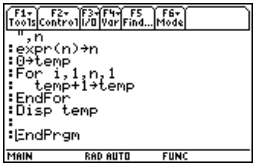
Programação

Escreva um programa que peça ao utilizador para introduzir um número inteiro, calcule a soma de todos os números inteiros a partir de 1 até o número introduzido e visualize o resultado.

Etapas e teclas	Ecrã
1. Inicie um novo programa no **Editor de Programa**. Prima 3	
2. Digite **PROG1** (sem espaços) como o nome da nova variável de programa. ▼ ▼ P R O G alpha 1 ▼ ▼ P R O G 1	
3. Visualize o “modelo” de um novo programa. O nome do programa, **Prgm**, e **EndPrgm** aparecem automaticamente. Depois de introduzir dados numa caixa de entrada, como Variable, tem de premir ENTER duas vezes. Prima ENTER ENTER	

Etapas e teclas	Ecrã

4. Digite as linhas de programa seguintes.



`Request "Enter an integer",n`

Visualize uma caixa de diálogo que pede “Introduza um número inteiro”, aguarda que o utilizador introduza um valor e guarda (como uma cadeia de caracteres) numa variável n.

`expr(n)→n`

Converte a cadeia de caracteres numa expressão numérica.

`0→temp`

Cria uma variável denominada temp e inicia-a como 0.

`For i,1,n,1`

Inicia o loop da instrução For com base na variável i. Na primeira passagem pelo loop, i = 1. Depois de executar o loop, i é incrementado de 1. O loop continua até que i > n.

`temp+i→temp`

Adiciona o valor actual de i a temp.

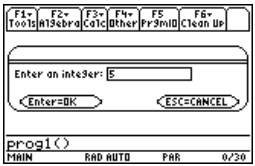
`EndFor`

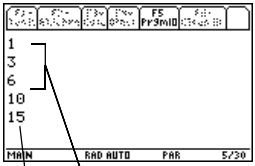
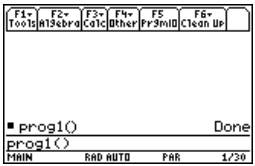
Marca o fim do loop For.

`Disp temp`

Exibe o valor final de temp.

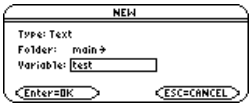
Digite as linhas do programa como apresentado. Prima [ENTER] no fim de cada linha.

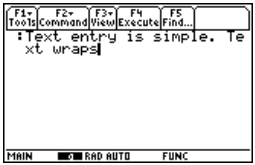
Etapas e teclas	Ecrã
5. Vá para o ecrã principal. Introduza o nome do programa, seguido por dois parêntesis. Tem de incluir ( ) mesmo quando o programa não possuir argumentos. O programa apresenta uma caixa de diálogo com a solicitação especificada no programa. [HOME] [2nd] [a-lock] P R O G [alpha] 1 [(] [)] [ENTER] [◆] [CALC HOME] P R O G 1 [(] [)] [ENTER]	prog1()
6. Digite 5 na caixa de diálogo apresentada. Prima 5	

Etapas e teclas	Ecrã
7. Continue a execução do programa. O comando **Disp** apresenta o resultado no ecrã Program I/O. O resultado é a soma dos inteiros de 1 a 5. Apesar do ecrã Program I/O aparecer com o ecrã principal, é utilizado apenas para entrada e saída de dados do programa. Não é possível fazer cálculos no ecrã Program I/O. Prima [ENTER] [ENTER]	 A saída de outros programas ainda pode estar no ecrã Resultado baseado na introdução do nº inteiro 5
8. Saia do ecrã Program I/O e volte para o ecrã principal. É também possível premir [ESC], [2nd] [QUIT] ou [HOME] [◆] [CALC HOME] para voltar ao ecrã principal.. Prima [F5]	

Operações com texto

Inicie uma nova secção no **Editor de Texto**. Em seguida, digite um texto qualquer no **Editor de Texto**. À medida que digita o texto, exercite a movimentação do cursor de texto e a correcção de possíveis erros.

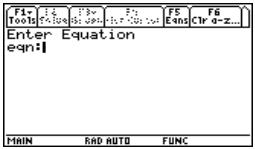
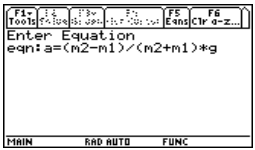
Etapas e teclas	Ecrã
1. Inicie uma nova secção no **Editor de Texto**. Prima 3	
2. Crie uma variável de texto denominada **TEST**, que guarda o texto introduzido automaticamente na nova secção. Utilize a pasta **MAIN** apresentada na caixa de diálogo **NEW** por predefinição. Depois de introduzir dados numa caixa de entrada, como **Variable**, tem de premir [ENTER] duas vezes. Prima ⊖ T E S T [ENTER] [ENTER]	

Etapas e teclas	Ecrã
3. Digite algum texto. • Para digitar apenas uma letra maiúscula, prima [↑] e, em seguida, a letra. Apenas para a TI-89 Titanium: - Para digitar um espaço, prima [alpha] [␣] (função alfabética da tecla [(-)]). - Para digitar um ponto, prima [alpha] para desactivar o alpha-lock; prima [.] e, em seguida, [2nd] [a-lock] para activar o alpha-lock novamente. Pratique a edição do texto com: • As teclas de cursor para movimentar o cursor de texto. • [←] ou [◆] [DEL] para excluir o carácter à esquerda ou à direita do cursor, respectivamente. [2nd] [a-lock] Digite um texto qualquer Digite um texto qualquer	

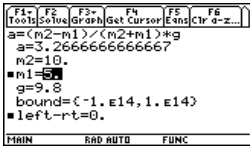
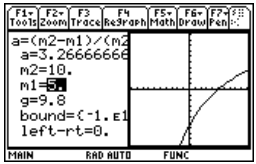
Etapas e teclas	Ecrã
4. Saia do **Editor de Texto** e visualize o ecrã principal. A secção de texto é guardada automaticamente à medida que digita o texto. No entanto, não tem de guardar a secção manualmente quando sair do **Editor de Texto**. [HOME] [◆] [CALC HOME]	
5. Volte à secção actual do **Editor de Texto**. Não se esqueça de que a secção apresentada permanece exactamente como a deixou. Prima [2nd] [⇥]	

Solucionador numérico

Considere a equação a=(m2−m1)/(m2+m1)∗g, onde os valores conhecidos são m2=10 e g=9.8. Assumindo que a=1/3 g, encontre o valor de m1.

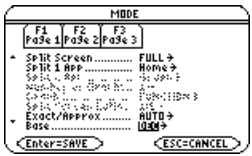
Etapas e teclas	Ecrã
1. Visualize o solucionador numérico. Prima [APPS]	
2. Introduza a equação. Quando prime [ENTER] ou ⊖, o ecrã lista as variáveis utilizadas na equação. [alpha] A [=] [(] [alpha] M 2 [−] [alpha] M 1 [)] [÷] [(] [alpha] M 2 [+] [alpha] M 1 [)] [×] [alpha] G [ENTER] A [=] [(] M 2 [−] M 1 [)] [÷] [(] M 2 [+] M 1 [)] [×] G [ENTER]	

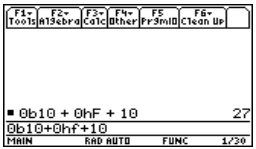
Etapas e teclas	Ecrã
3. Introduza valores para cada variável, excepto para a incógnita m1. Defina m2 e g em primeiro lugar. Em seguida, defina a. (Deve definir g antes de definir a em termos de g.) Aceite o valor predefinido para limites (bound). Se já tiver definido uma variável previamente, o valor aparece como predefinição. ▼ 1 0 ▼ ▼ 9 . 8 ▲ ▲ ▲ alpha G ÷ 3 ▼ 1 0 ▼ ▼ 9 . 8 ▲ ▲ ▲ G ÷ 3	F1▾ Tools F2 Solve F3▾ Graph F4 Get Cursor F5 Eqns F6 Clr a-z... a=(m2-m1)/(m2+m1)*g a=g/3 m2=10. m1= g=9.8 bound={-1.E14,1.E14} MAIN RAD AUTO FUNC
4. Mova o cursor para a incógnita m1. Opcionalmente, é possível introduzir um valor suposto inicial para m1. Mesmo que introduza um valor para todas as variáveis, o solucionador numérico encontra o valor da variável realçada pelo cursor. Prima ▼ ▼	F1▾ Tools F2 Solve F3▾ Graph F4 Get Cursor F5 Eqns F6 Clr a-z... a=(m2-m1)/(m2+m1)*g a=3.2666666666667 m2=10. m1= g=9.8 bound={-1.E14,1.E14} MAIN RAD AUTO FUNC g/3 é calculado quando move o cursor para fora da linha.

Etapas e teclas	Ecrã
5. Encontre o valor da incógnita. Para verificar a exactidão da solução, os lados esquerdo e direito da equação são calculados separadamente. A diferença aparece como left-rt. Se a solução for exacta, left-rt=0. Prima [F2]	 ■ marca os valores calculados.
6. Represente graficamente a solução com a janela de visualização **ZoomStd**. O gráfico aparece num ecrã dividido. É possível explorar o gráfico através de traçado, ampliação, etc. A variável realçada pelo cursor (incógnita m1) está no eixo x, e left-rt está no eixo y. Prima [F3] 3	
7. Volte ao solucionador numérico e saia do ecrã dividido. É possível premir [ENTER] ou ⊖ para visualizar novamente a lista de variáveis. Prima [2nd] [⊞] [F3] 2	

Bases numéricas

Calcule 10 binário (base 2) + F hexadecimal (base 16) + 10 decimal (base 10). Em seguida, utilize o operador ▶ para converter o resultado inteiro de uma base para outra. Por fim, observe como a mudança do modo da base afecta o resultado apresentado.

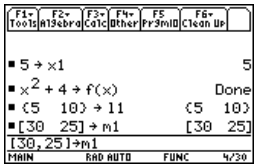
Etapas e teclas	Ecrã
1. Visualize a caixa de diálogo **MODE**, página 2. Para o modo **Base**, seleccione **DEC** como a base numérica predefinida. Os resultados inteiros são apresentados de acordo com o modo **Base**. Os resultados em fracções e de ponto flutuante são sempre apresentados na forma decimal. Prima [MODE] [F2] (utilize ⊖ para mover para o modo **Base**) ⊕ 1 [ENTER]	

Etapas e teclas	Ecrã
2. Calcule 0b10+0hF+10. Para introduzir um número binário ou hexadecimal, é preciso utilizar o prefixo 0b ou 0h (zero e a letra B ou H). Caso contrário, o valor introduzido é tratado como número decimal. **Nota:** O prefixo 0b ou 0h é um zero, não a letra O, seguido por B ou H. 0 [alpha] B 1 0 [+] 0 [2nd] [a-lock] H F [alpha] [+] 1 0 [ENTER] 0 B 1 0 [+] 0 H F [+] 1 0 [ENTER]	
3. Adicione 1 ao resultado e converta-o em binário. [2nd] [▶] apresenta o operador de conversão ▶. [+] 1 [2nd] [▶] [2nd] [a-lock] B I N [alpha] [ENTER] [+] 1 [2nd] [▶] B I N [ENTER]	
4. Adicione 1 ao resultado e converta-o em hexadecimal. [+] 1 [2nd] [▶] [2nd] [a-lock] H E X [alpha] [ENTER] [+] 1 [2nd] [▶] H E X [ENTER]	

Etapas e teclas	Ecrã
5. Adicione 1 ao resultado e deixe-o na base decimal predefinida. Os resultados utilizam o prefixo 0b ou 0h para identificar a base. Prima [+] 1 [ENTER]	F1▾ Tools F2▾ Algebra F3▾ Calc F4▾ Other F5 PrgmIO F6▾ Clean Up ■ 0b10 + 0hF + 10 27 ■ (27 + 1)▸Bin 0b11100 ■ (0b11100 + 1)▸Hex 0h1D ■ 0h1D + 1 30 ans(1)+1 MAIN RAD AUTO FUNC 4/30
6. Mude o modo **Base** para **HEX**. Quando **Base = HEX** ou **BIN**, a ordem de grandeza de um resultado fica restrita a determinadas limitações de tamanho. Prima [MODE] [F2] (utilize ⊖ para mover para o modo **Base**) ⊕ 2 [ENTER]	
7. Calcule 0b10+0hF+10 0 [alpha] B 1 0 [+] 0 [2nd] [a-lock] H F [alpha] [+] 1 0 [ENTER] 0 B 1 0 [+] 0 H F [+] 1 0 [ENTER]	F1▾ Tools F2▾ Algebra F3▾ Calc F4▾ Other F5 PrgmIO F6▾ Clean Up ■ 0b10 + 0hF + 10 27 ■ (27 + 1)▸Bin 0b11100 ■ (0b11100 + 1)▸Hex 0h1D ■ 0h1D + 1 30 ■ 0b10 + 0hF + 10 0h1B 0b10+0hf+10 MAIN RAD AUTO FUNC 5/30
8. Mude o modo **Base** para **BIN**. Prima [MODE] [F2] (utilize ⊖ para mover para o modo **Base**) ⊕ 3 [ENTER]	
9. Introduza novamente 0b10+0hF+10. Prima [ENTER]	F1▾ Tools F2▾ Algebra F3▾ Calc F4▾ Other F5 PrgmIO F6▾ Clean Up ■ (27 + 1)▸Bin 0b11100 ■ (0b11100 + 1)▸Hex 0h1D ■ 0h1D + 1 30 ■ 0b10 + 0hF + 10 0h1B ■ 0b10 + 0hF + 10 0b11011 0b10+0hf+10 MAIN RAD AUTO FUNC 6/30

Gestão da memória e das variáveis

Atribua valores a diversos tipos de dados de variáveis. Utilize o ecrã **VAR-LINK** para visualizar uma lista das variáveis definidas. Em seguida, mova uma variável para a memória de arquivo de dados do utilizador e analise as formas pelas quais pode ou não aceder a uma variável arquivada. (As variáveis arquivadas são automaticamente bloqueadas). Por fim, recupere a variável e exclua as variáveis não utilizadas de forma a que não ocupem espaço em memória.

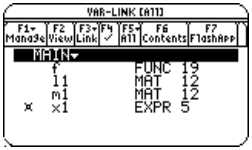
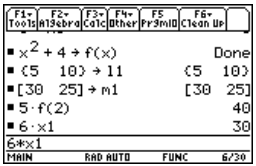
Etapas e teclas	Ecrã
1. No ecrã principal, atribua variáveis dos seguintes tipos. Expressão: 5 →x1 Função: x +4 →f(x) Lista: {5,10} →L1 Matriz: [30,25] →m1 [HOME] [CLEAR] 5 [STO▸] X 1 [ENTER] X [^] 2 [+] 4 [STO▸] [alpha] F [(] X [)] [ENTER] [2nd] [{] 5 [,] 1 0 [2nd] [}] [STO▸] [alpha] L 1 [ENTER] [2nd] [[] 3 0 [,] 2 5 [2nd] []] [STO▸] [alpha] M1 [ENTER] [◆] [CALC HOME] [CLEAR] 5 [STO▸] X 1 [ENTER] X [^] 2 [+] 4 [STO▸] F [(] X [)] [ENTER] [2nd] [{] 5 [,] 1 0 [2nd] [}] [STO▸] L 1 [ENTER] [2nd] [[] 3 0 [,] 2 5 [2nd] []] [STO▸] M 1 [ENTER]	

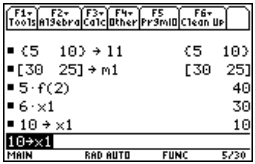
Etapas e teclas	Ecrã
2. Suponha que é necessário iniciar uma operação com uma variável de função, cujo nome não é possível lembrar. Prima 5 [×]	5∗
3. Visualize o ecrã **VAR-LINK**. Este exemplo pressupõe que as únicas variáveis definidas são as definidas acima. Prima [2nd] [VAR-LINK]	VAR-LINK [ALL] F1 Manage F2 View F3 Link F4 ✓ F5 All F6 Contents F7 FlashApp MAIN f FUNC 19 l1 MAT 12 m1 MAT 12 x1 EXPR 5
4. Modifique a visualização do ecrã para mostrar apenas as variáveis de função. Mesmo que esta operação pareça inútil num exemplo com quatro variáveis, pense nas vantagens na utilização de muitas variáveis de diversos tipos. Prima [F2] ⊖ ⊖ ⊕ 5 [ENTER]	VAR-LINK VIEW View..... Variables→ Folder... All→ Var Type Function→ Enter=OK ESC=CANCEL VAR-LINK [ALL] F1 Manage F2 View F3 Link F4 ✓ F5 All F6 Contents F7 FlashApp MAIN f FUNC 19
5. Realce a variável de função **f** e visualize o conteúdo. Observe que a função foi atribuída com **f(x)**, mas é listada no ecrã como **f**. ⊖ [2nd] [F6] ⊖ [F6]	x^2+4

Etapas e teclas	Ecrã
6. Feche a janela Contents. Prima [ESC]	
7. Com a variável **f** ainda realçada, feche **VAR-LINK** para colcar os conteúdos da variável na linha de entrada. Observe que " (" é colado. Prima [ENTER]	5∗f(
8. Complete a operação. Prima 2 [)] [ENTER]	5∗f(2)

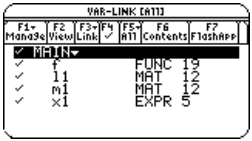
Gestão de uma variável

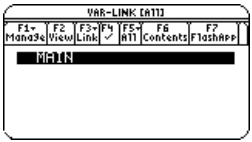
Etapas e teclas	Ecrã
1. Visualize **VAR-LINK** novamente e realce a variável que deseja arquivar. A mudança anterior na visualização já não está activada. O ecrã lista todas as variáveis definidas. Prima [2nd] [VAR-LINK] (utilizar ⊖ para realçar **x1**)	VAR-LINK [ATT] F1▾ Manage F2 View F3▾ Link F4 ✓ F5▾ ATT F6 Contents F7 FlashAPP MAIN▾ f FUNC 19 l1 LIST 10 m1 MAT 12 t1 FIG 26 x1 EXPR 5

Etapas e teclas	Ecrã
2. Utilize o menu [F1] **Manage** da barra de ferramentas para arquivar a variável. ⌘ indica que a variável está arquivada. Prima [F1] 8	 
3. Volte ao ecrã principal e utilize a variável arquivada nos cálculos. [HOME] 6 [×] X 1 [ENTER] [◆] [CALC HOME] 6 [×] X 1 [ENTER]	
4. Tente armazenar um valor diferente na variável arquivada. Prima 1 0 [STO▸] X 1 [ENTER]	
5. Cancele a mensagem de erro. Prima [ESC]	
6. Utilize **VAR-LINK** para recuperar uma variável. Prima [2nd] [VAR-LINK] (use ⊖ para realçar **x1**) [F1] 9	

Etapas e teclas	Ecrã
7. Volte ao ecrã principal e armazene um valor diferente na variável recuperada. [HOME] [ENTER] [◆] [CALC HOME] [ENTER]	

Exclusão de variáveis

Etapas e teclas	Ecrã
1. Visualize **VAR-LINK** e utilize o menu da barra de ferramentas [F5] **All** para seleccionar todas as variáveis. Uma marca ✓ indica os itens que estão seleccionados. Não se esqueça de que também seleccionou a pasta **MAIN**. **Nota:** Em vez de utilizar [F5] (se não desejar excluir todas as suas variáveis), pode seleccionar as variáveis individualmente. Realce cada variável a excluir e prima [F4]. Prima [2nd] [VAR-LINK] [F5] 1	 
2. Utilize [F1] para excluir. **Nota:** Pode premir [←] (em vez de [F1] **1**) para excluir as variáveis marcadas. Prima [F1] 1	

Etapas e teclas	Ecrã
3. Confirme a exclusão. Prima [ENTER]	
4. Como [F5] **1** também selecciona a pasta **MAIN**, uma mensagem de erro afirma que não é possível excluir a pasta **MAIN**. Confirme a mensagem. Quando **VAR-LINK** aparecer novamente, as variáveis excluídas não serão listadas. Prima [ENTER]	
5. Feche **VAR-LINK** e volte à aplicação actual (o ecrã principal neste exemplo). Quando utiliza [ESC] (em vez de [ENTER]) para fechar **VAR-LINK** , o nome realçado não é colado na linha de entrada. Prima [ESC]	

Utilização da unidade portátil

Activação e desactivação da TI-89 Titanium / Voyage™ 200

Para ligar e desligar manualmente a TI-89 Titanium / Voyage™ 200 Graphing Calculator, utilize as teclas [ON] e [2nd] [OFF] (ou [◆] [OFF]). Para aumentar a duração das pilhas, a TI-89 Titanium / Voyage™ 200 contém a função APD™ (Automatic Power Down™) que a desliga automaticamente.

Activação da TI-89 Titanium / Voyage™ 200

Digite [ON].

- Se desligar com [2nd] [OFF], a TI-89 Titanium / Voyage™ 200 volta ao ecrã principal ou ao ambiente de trabalho das Apps.
- Se desligar com [◆] [OFF] ou se a calculadora se desligar automaticamente com a função APD, a TI-89 Titanium / Voyage™ 200 volta à última aplicação utilizada.

Desactivação da TI-89 Titanium / Voyage™ 200

Para desligar a TI-89 Titanium / Voyage™ 200, é possível utilizar uma das seguintes teclas.

Prima:	Descrição
[2nd] [OFF] (prima [2nd] e, em seguida, [OFF])	A função Constant Memory™ permite conservar todas as definições e o conteúdo da memória; no entanto: • Não é possível utilizar [2nd] [OFF], se houver uma mensagem de erro. • Quando voltar a ligar a TI-89 Titanium / Voyage™ 200, aparece o ecrã principal ou a ambiente de trabalho das Apps (independentemente da última aplicação utilizada).
[◆] [OFF] (prima [◆] e então [OFF])	Similar a [2nd] [OFF] excepto: • É possível utilizar [◆] [OFF] mesmo em caso de uma mensagem de erro. • Se voltar a ligar, a TI-89 Titanium / Voyage™ 200 mostra exactamente o conteúdo exibido antes de a desligar.

Nota: [OFF] é a segunda função da tecla [ON].

APD (desligar automático)

Depois de vários minutos sem actividade, a TI-89 Titanium / Voyage™ 200 desliga-se automaticamente. Esta função chama-se APD.

Se premir [ON], a TI-89 Titanium / Voyage™ 200 mostra exactamente o conteúdo exibido antes de a desligar.

- O ecrã, o cursor e qualquer condição de erro estarão exactamente como antes de a desligar.
- Todas as definições e os dados da memória permanecem intactos.

A função APD não é activada durante uma operação de cálculo ou a execução de um programa, excepto se estiver no modo de espera. Se o programa estiver em execução, mas à espera de um batimento de tecla, o APD verificar-se-á decorridos alguns minutos de inactividade.

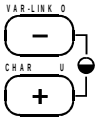
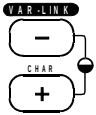
Ajuste do contraste do ecrã

O brilho e o contraste do ecrã dependem da iluminação do ambiente, da carga das pilhas, do ângulo de visão e do ajuste do contraste. Este ajuste fica memorizado quando desligar a TI-89 Titanium / Voyage™ 200 Graphing Calculator.

Ajuste do contraste

É possível variar o contraste conforme o ângulo de visão e as condições de iluminação.

Para:	Prima sem soltar
Diminuir o contraste (mais claro)	[◆] e [−]
Aumentar o contraste (mais escuro)	[◆] e [+]
Teclas de contraste	Teclas de contraste

Para:	Prima sem soltar
	

Se premir sem soltar as teclas [◆] [−] ou [◆] [+] durante muito tempo, é possível que o ecrã apareça excessivamente claro ou escuro. Para efectuar um ajuste mais preciso, prima sem soltar [◆] e prima [−] ou [+].

Quando trocar as pilhas

À medida que as pilhas se descarregam, o ecrã escurece (especialmente nas operações de cálculo). Por consequência, aumente o contraste. Se aumentar o contraste frequentemente, substitua as quatro pilhas alcalinas.

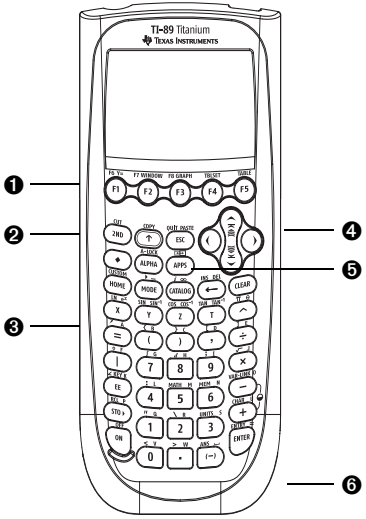
Nota: depois de trocar as pilhas, o ecrã pode estar muito escuro; utilize [◆] [−] para o clarear.

A linha de estado na parte interior do ecrã fornece também informações sobre a carga das pilhas.

Indicador da linha de estado	Descrição
BATT	Pilhas com pouca carga.
BATT	Pilhas descarregadas. Troque-as assim que for possível.

Teclado da TI-89 Titanium

A maioria das teclas pode executar uma ou mais funções, se premir primeiro uma tecla modificadora.

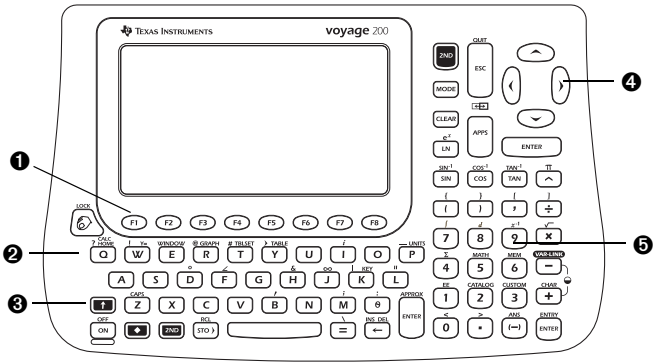


❶ [F1] – [2nd] [F8] abrem os menus da barra de ferramentas. Selecciona aplicações (quando utilizadas com [◆]).
❷ [2nd], [◆], [↑] e [alpha] ampliam a funcionalidade, aumentando o número de comandos de tecla disponíveis.
❸ X, Y e Z normalmente são utilizadas em cálculos simbólicos.

❹ ◀, ▶, ▲ e ▼ movem o cursor.
❺ APPS permite seleccionar uma aplicação.
❻ ENTER calcula uma expressão, executa uma instrucção, selecciona um item de menu, etc.

Teclado da Voyage™ 200

Devido ao desenho ergonómico e a distribuição das teclas, é possível aceder a qualquer parte do teclado, mesmo quando estiver a segurar a unidade com as duas mãos. O teclado está dividido em várias áreas de teclas funcionais.



❶ F1 – F8 abrem os menus da barra de ferramentas.
❷ O teclado QWERTY funciona da mesma maneira que um teclado de computador.
❸ ◆, 2nd, ↑ e ⊡ ampliam a funcionalidade, aumentando o número de comandos de tecla disponíveis.

❹ ◀, ▶, ▲ e ▼ movem o cursor.
❺ O teclado numérico executa funções matemáticas e científicas.

Mover o cursor

Para mover o cursor para uma direção específica, prima a tecla de cursor adequada (◀, ▶, ▲ ou ▼).

Algumas aplicações da TI-89 Titanium permitem também premir:

- [2nd] ◀ ou [2nd] ▶ para o mover para o início ou fim de uma linha.
- [2nd] ▲ ou [2nd] ▼ para o mover um ecrã de cada vez para cima ou para baixo.
- [◆] ▲ ou [◆] ▼ para o mover para o topo ou base de uma página.
- ▲ e ◀, ▲ e ▶, ▼ e ◀, ou ▼ e ▶ para movimentos diagonais. (Prima as teclas de cursor indicadas em simultâneo.)

Teclas modificadoras

Teclas modificadoras

Modificador	Descrição
[2nd] (segunda)	Acede à segunda função da tecla seguinte premida. No teclado, têm a mesma cor que a tecla [2nd].
[◆] (losango)	Activa as teclas que selecionnam determinadas aplicações, itens de menu e outras operações do teclado. No teclado, têm a mesma cor que a tecla [◆].
[⇧] (shift)	Digita o carácter maiúsculo da próxima tecla premida. [⇧] é também utilizada com ▶ e ◀ para realçar os caracteres da linha de entrada para fins de edição.

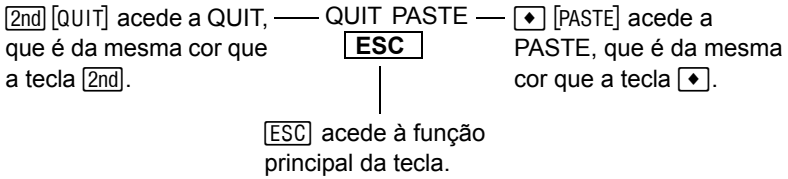
Modificador	Descrição
alpha (apenas)	Utilizada para digitar letras, inclusive o espaço em branco. No teclado, têm a mesma cor que a tecla alpha.
(mão) (apenas)	Utilizada com a tecla do cursor para manipular objectos geométricos. é também utilizada para desenhar num gráfico.

Nota: Para mais informações sobre a utilização de alpha e ↑, consulte “Introduzir caracteres alfabéticos”.

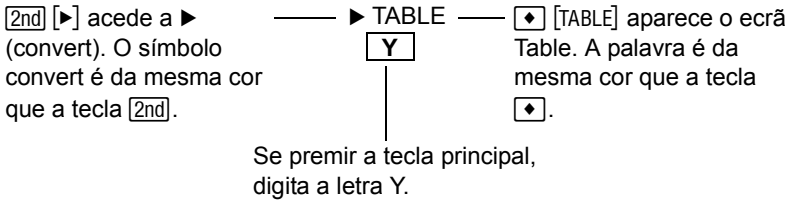
Exemplos dos modificadores [2nd] e [losango]

Por exemplo, a tecla ESC pode realizar três operações, dependendo de premir primeiro 2nd ou ◆.

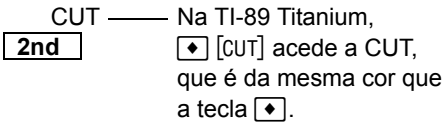
O exemplo da TI-89 Titanium seguinte mostra a utilização da tecla modificadora 2nd ou ◆ com a tecla ESC.



O exemplo da Voyage™ 200 seguinte mostra a utilização da tecla modificadora 2nd ou ◆ com a tecla alfabética Y.



Algumas teclas realizam apenas uma operação adicional, que pode exigir que prima 2nd ou ◆, dependendo da cor da operação e da posição acima da tecla.



Se premir um modificador como 2nd ou ◆, aparece um indicador 2ND ou ◆ na linha de estado na parte inferior do ecrã. Se premir um modificador acidentalmente, prima-o novamente (ou prima ESC) para cancelar o efeito.

Outras teclas importantes que precisa de conhecer

Tecla	Descrição
◆ [Y=]	Mostra o Y= Editor.
◆ [WINDOW]	Mostra o Editor de Janelas.
◆ [GRAPH]	Mostra o ecrã Graph.

Tecla	Descrição
[◆] [TBLSET]	Define parâmetros do ecrã Table.
[◆] [TABLE]	Mostra o ecrã Table.
: [◆] [CUT] [◆] [COPY] [◆] [PASTE] : [◆] **X** (cut) [◆] **C** (copy) [◆] **V** (paste)	Permite editar as informações introduzidas com uma operação de cortar, copiar e colar.
[2nd] [⇔]	Alterna entre as duas últimas Apps escolhidas ou entre as partes do ecrã dividido.
[2nd] [CUSTOM]	Liga e desliga o menu Custom.
[2nd] [▶]	Converte unidades de medida.
[◆] [_] [2nd] [_]	Designa uma unidade de medida.
[←]	Exclui o carácter à esquerda do cursor (retrocesso).
[2nd] [INS]	Alterna entre os modos de inserção e sobreposição para a entrada de informações.
[◆] [DEL]	Exclui o carácter à direita do cursor.
[|] [2nd] [|]	Introduz o operador “*with*” utilizado nos cálculos simbólicos.
[2nd] [∫], [2nd] [d]	Calcula integrais e derivadas.

Tecla	Descrição
2nd [∠]	Designa um ângulo em coordenadas polares, cilíndricas e esféricas.
2nd [MATH]	Mostra o menu MATH.
2nd [MEM]	Mostra o ecrã MEMORY.
2nd [VAR-LINK]	Mostra o ecrã VAR-LINK para a gestão de variáveis e aplicações Flash.
2nd [RCL]	Chama o conteúdo de uma variável.
2nd [UNITS] [◆] [UNITS]	Mostra a caixa de diálogo UNITS.
2nd [CHAR]	Mostra o menu CHAR, que permite seleccionar letras gregas, caracteres internacionais acentuados, etc.
2nd [ENTRY], 2nd [ANS]	Chama a entrada anterior e a última resposta.

Introdução de caracteres alfabéticos

Os caracteres alfabéticos são utilizados em expressões do tipo $\mathbf{x^2+y^2}$, para introduzir nomes de variáveis (Armazenamento e recuperação dos valores de variáveis) e também no módulo *Text Editor*.

Digitação de uma letra na TI-89 Titanium

As letras x, y, z e t são geralmente utilizadas em expressões algébricas. Para as digitar rapidamente, essas letras são as teclas principais do teclado TI-89 Titanium.

[X] [Y] [Z] [T]

Outras letras estão também disponíveis, como a função [alpha] de uma outra tecla, semelhante aos modificadores [2nd] e [◆] descritos na secção anterior. Por exemplo:

[2nd] [′] digita ′, que é da mesma cor que a tecla [2nd]. ——— ′ A ——— [alpha] [A] digita um A, que é da mesma cor que a tecla [alpha].
[=]

Digitação de caracteres alfabéticos na TI-89 Titanium / Voyage™ 200

Para:	Na [calculadora], prima:	Na [teclado], prima:
Digitar uma única letra minúscula	[alpha] e a tecla da letra (a linha de estado mostra ◧)	a tecla da letra
Digitar uma única letra maiúscula.	[↑] e a tecla da letra (a linha de estado mostra ⬆)	[↑] e, em seguida, a tecla da letra (a linha de estado mostra ⬆)
Digitar um espaço.	[alpha] [␣] (função alfabética da tecla [(-)])	barra de espaço
Activar o alpha-lock de minúsculas.	[2nd] [a-lock] (a linha de estado mostra ◧)	(nenhuma ação é necessária)
Activar o ALPHA-lock de maiúsculas.	[↑] [a-lock] (a linha de estado mostra ◧)	[2nd] [CAPS]

Para:	Na , prima:	Na , prima:
Desactivar alpha-lock.	[alpha] (desactiva o travamento de maiúsculas e minúsculas)	[2nd] [CAPS] (desactiva o travamento de maiúsculas)

Nota:

- Na TI-89 Titanium, não precisa de [alpha] ou de alpha-lock para digitar x, y, z ou t. Mas precisa de [↑] ou do ALPHA-lock de maiúsculas para X, Y, Z ou T.
- Na TI-89 Titanium, Alpha-lock fica sempre desactivado quando muda de aplicação. Por exemplo, quando muda do Text Editor para o ecrã principal.

Enquanto houver um dos tipos de alpha-lock activado na TI-89 Titanium:

- Para digitar um ponto, vírgula ou outro carácter que seja a função principal de uma tecla, é necessário desactivar o alpha-lock.
- Para digitar um carácter de segunda função como [2nd] [{], não é necessário desactivar o alpha-lock. Depois de digitar o carácter, o alpha-lock permanece activado.

Alpha-Lock automático nas caixas de diálogo da TI-89 Titanium

Há determinados momentos em que não precisa premir [alpha] ou [2nd] [a-lock] para digitar caracteres alfabéticos na TI-89 Titanium. O alpha-lock automático é activado sempre

que aparece uma caixa de diálogo. A função de alpha-lock automático aplica-se nas seguintes caixas de diálogo:

Caixa de diálogo	Alpha-lock
Catalog	Todos os comandos são listados por ordem alfabética. Prima uma letra para ir para o primeiro comando que inicia com aquela letra. Consulte “Utilizar a caixa de diálogo Catalog na página 30
Units	Em cada categoria de unidade, digite a primeira letra de uma unidade ou constante. Consulte no módulo *Constantes e unidades de medida*.
Caixas de diálogo com campos de introdução	Incluem, mas não estão limitadas a: Create New Folder, Rename e Save Copy As.

Nota: para digitar um número, prima [alpha] para desactivar o alpha-lock. Prima [alpha] ou [2nd] [a-lock] para voltar a digitar letras.

O alpha-lock não é activado nas caixas de diálogo que requeiram apenas entradas numéricas. As caixas de diálogo que só aceitam entradas numéricas são: Resize Matrix, Zoom Factors e Table Setup.

Para caracteres especiais

Utilize o menu [2nd] [CHAR] para seleccionar um dos vários caracteres especiais. Para mais informações, consulte “Digitação de caracteres especiais” no módulo *Text Editor*.

Introdução de números

O teclado permite digitar números positivos e negativos. Também é possível digitar números em notação científica.

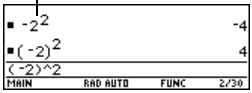
Introdução de um número negativo

1. Prima a tecla de negação [(-)]. (Não utilize a tecla de subtracção [−].)
2. Digite o número.

Para ver como a TI-89 Titanium / Voyage™ 200 Graphing Calculator interpreta uma negação em relação às outras funções, consulte a hierarquia do Sistema Operativo de Equações (EOS™) no módulo *Referência técnica*. É importante saber que determinadas funções, como por exemplo x^2 são calculadas antes da negação.

Utilize [(] e [)] para pôr parênteses em caso de dúvidas sobre como será interpretada uma negação.

Calculado como $-(2^2)$



A utilização de [−] em vez de [(-)] (ou vice-versa), pode gerar uma mensagem de erro ou um resultado imprevisto. Por exemplo:

- 9 [×] [(-)] 7 = -63
  – mas –
  9 [×] [−] 7 mostra uma mensagem de erro.

- 6 [−] 2 = 4
  – mas –
  6 [(-)] 2 = ⁻12 sendo interpretado como 6(⁻2), multiplicação implícita.
- [(-)] 2 [+] 4 = 2
  – mas –
  [−] 2 [+] 4 subtrai 2 do resultado precedente e adiciona 4.

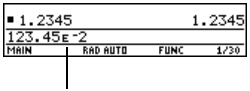
Importante: Utilize [−] para a subtracção e [(-)] para indicar o número negativo.

Introdução de um número em notação científica

1. Digite a parte do número que precede o expoente. Este valor pode ser uma expressão.
2. Prima:
   [EE]
   [2nd] [EE]
   O ecrã mostra E.
3. Digite o expoente como um número inteiro com até três algarismos. É possível utilizar um expoente negativo.

A introdução de um número em notação científica não pressupõe a apresentação dos resultados em notação científica ou técnica.

O formato de apresentação é determinado pela definição dos modos e pelo valor absoluto do número.



Representa 123.45×10^{-2}

Introdução de expressões e instruções

É possível operar com expressões. Uma acção é iniciada mediante uma instrução adequada. As expressões são calculadas e os resultados apresentados de acordo com a definição dos modos.

Definições

Expressão	É composta por números, variáveis, operadores, funções e respectivos argumentos cujo resultado é uma resposta única. Por exemplo: πr^2+3. • Insira uma expressão pela mesma ordem como se escreve normalmente. • Uma expressão pode ser utilizada na maior parte dos casos onde é solicitada a introdução de um valor.
Operador	Executa operações como +, –, ∗, ^. • Operadores requerem um argumento antes e depois. Por exemplo: 4+5 e 5^2.
Função	Devolve um valor. • Funções necessitam de um ou mais argumentos (entre parênteses) depois. Por exemplo: $\sqrt{}$(5) e **min(5,8)**.

Instrução	Inicia uma acção. • Instruções não podem ser utilizadas em expressões. • Algumas instruções não requerem nenhum argumento. Por exemplo: ClrHome. • Outras requerem um ou mais argumentos. Por exemplo: **Circle 0,0,5**. Para as instruções, não ponha os argumentos entre parêntesis.

Nota:

- O módulo *Referência técnica* descreve todas as funções e instruções integradas na TI-89 Titanium / Voyage™ 200 Graphing Calculator.
- Este manual utiliza a expressão comando como uma referência genérica para as funções e as instruções.

Multiplicação implícita

A TI-89 Titanium / Voyage™ 200 reconhece uma multiplicação implícita, desde que não esteja em conflito com uma notação reservada.

	Se digitar:	**A / interpreta como:**
Válido	2π 4 sin(46) 5(1+2) or (1+2)5 [1,2]a 2(a)	2∗π 4∗sin(46) 5∗(1+2) or (1+2)∗5 [a 2a] 2∗a

	Se digitar:	A / interpreta como:
Inválido	xy	Variável simples definida xy
	a(2)	Chamada de função
	a[1,2]	Índice matricial para o elemento a[1,2]

Parêntesis

Expressões são calculadas de acordo com a hierarquia do Sistema Operativo de Equações (EOS™), descrito no módulo *Referência técnica*. Para modificar a ordem de cálculo ou garantir que as operações sejam feitas segundo uma ordem desejada, utilize os parêntesis.

Os cálculos entre parêntesis são efectuados em primeiro lugar. Por exemplo: em 4(1+2), o EOS calcula em primeiro lugar (1+2) e, em seguida, multiplica o resultado por 4.

Introdução de uma expressão

Digite a expressão e, em seguida, prima [ENTER] para a calcular. Para introduzir o nome de uma função ou de uma instrução na linha de entrada, efectue o procedimento apresentado em seguida:

- Se disponível, prima a tecla correspondente. Por exemplo: prima:
  [2nd] [SIN]
  [SIN]

  – ou –

- Se disponível, seleccione-o num menu. Por exemplo: seleccione **2:abs** no submenu Number do menu MATH.

  – ou –

- Digite o nome, letra a letra, no teclado. (Na TI-89 Titanium, utilize [alpha] e [2nd] [a-lock] para digitar as letras). Pode utilizar qualquer mistura de letras maiúsculas ou minúsculas. Por exemplo, digite **sin(** ou **Sin(**.

Exemplo

Digite o nome da função.

Calcule 3.76 ÷ (-7.9 + √5) + 2 log 45.

Na	Na	Ecrã
3.76 [÷] [(] [(-)] 7.9 [+] [2nd] [√]	3.76 [÷] [(] [(-)] 7.9 [+] [2nd] [√]	3.76/(-7.9+√([2nd] [√] insere "√(" porque o argumento deve estar entre parêntesis.
5 [)] [)]	5 [)] [)]	3.76/(-7.9+√(5)) Utilize [)] uma vez para fechar √(5) e outra vez para fechar (-7.9 + √5).
[+] 2 [2nd] [a-lock] LOG [alpha] [(] 45 [)]	[+] 2 LOG [(] 45 [)]	3.76/(-7.9+√(5))+2log(45) log requer ( ) para delimitar o argumento.

Na	Na	Ecrã
ENTER	ENTER	$\frac{3.76}{-7.9+\sqrt{5}}+2\cdot\log(45)$ 2.64258 3.76/(-7.9+√(5))+2log(45) MAIN RAD AUTO FUNC 1/30

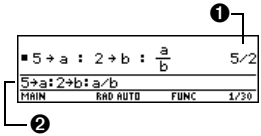
Nota: pode também seleccionar **log** com

CATALOG

2nd [CATALOG]

Introdução de várias expressões numa linha

Para introduzir mais de uma expressão ou instrução simultaneamente, separe-as com dois pontos, premindo 2nd [:].



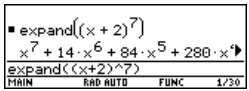
❶ Aparece apenas o último resultado.

❷ → aparece quando pressionar STO▸ para memorizar um valor numa variável.

Se uma entrada ou resposta for maior que o comprimento da linha

Se não for possível mostrar a entrada e o resultado numa única linha na área do histórico, o resultado aparece na linha seguinte.

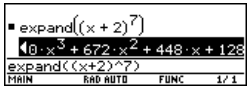
Se uma entrada ou resposta superar o comprimento de uma linha, no fim da linha aparece ▶.



Para ver a entrada ou a resposta por inteiro:

1. Prima ⊕ para deslocar o cursor da linha de entrada para a área do histórico; este procedimento realça a última resposta.
2. Prima ⊕ e ⊖ para realçar a entrada ou resposta desejada. Por exemplo: ⊕ permite mover do resultado para a entrada na área do histórico.
3. Utilize ⊕ e ⊖ ou [2nd] ⊕ e [2nd] ⊖ para ir para a direita ou para a esquerda.

   Nota: se for para a direita, ◄ aparece no início da linha.



4. Para voltar à linha de entrada, prima [ESC].

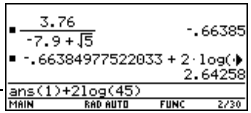
Continuação de uma operação

Se premir [ENTER] para calcular uma expressão, a TI-89 Titanium / Voyage™ 200 mantém a expressão na linha de entrada e realça-a. É possível continuar com a última resposta ou introduzir outra expressão.

Se premir:	A / :
[+], [−], [×], [÷], [^], ou [STO▸]	Substitui a linha de entrada com a variável **ans(1)**, que permite utilizar a última resposta como início de uma outra expressão.
Qualquer outra tecla	Cancela a linha de entrada e inicia uma outra.

Exemplo

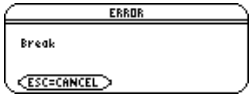
Calcule 3.76 ÷ (-7.9 + √5). Em seguida, adicione 2 log 45 ao resultado.

Na	Na	Ecrã
3.76 [÷] [(] [(-)] 7.9 [+] [2nd] [√] 5 [)] [)] [ENTER] [+] 2 [2nd] [a-lock] LOG [alpha] [(] 45 [)] [ENTER]	3.76 [÷] [(] [(-)] 7.9 [+] [2nd] [√] 5 [)] [)] [ENTER] [+] 2 LOG [(] 45 [)] [ENTER]	 Se premir [+], a linha de entrada é substituída pela variável **ans(1)**, que contém a última resposta.

Interrupção de um cálculo

O indicador BUSY aparece à direita da linha de estado durante a realização de um cálculo. Para o interromper, prima [ON].

É possível que passem alguns momentos antes de aparecer a mensagem “Break”.



Prima [ESC] para voltar à aplicação actual.

Formatos de apresentação dos resultados

Um resultado pode ser calculado e apresentado em formatos diferentes. Esta secção explica os modos da TI-89 Titanium / Voyage™ 200 Graphing Calculator e as características que determinam os formatos de apresentação.

Modo Pretty Print

Por predefinição, a definição deste modo é **Pretty Print = ON**. Expoentes, raízes, fracções, etc., são apresentadas da mesma forma em que foram escritas. É possível utilizar [MODE] para activar ou desactivar Pretty print.

Pretty Print	
ON	**OFF**
π^2, $\frac{\pi}{2}$, $\sqrt{\frac{x-3}{2}}$	π^2, π/2, √((x–3)/2)

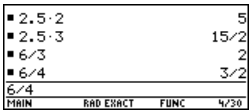
As expressões não aparecem em Pretty print na linha de entrada. Se este modo estiver activo e premir [ENTER][F1] a área do histórico apresenta a entrada e o resultado no modo Pretty print.

Modo Exact/Approx

A predefiniçao é **Exact/Approx = AUTO**. É possível utilizar MODE para seleccionar um dos três modos.

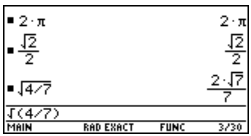
Visto que AUTO é uma combinação das duas outras opções, o utilizador deve conhecer os efeitos das três definições.

EXACT — Todos os resultados que não sejam números inteiros aparecem em forma simbólica ou fracionária (1/2, π, $\sqrt{2}$, etc.).



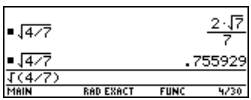
Mostra os resultados constituídos por um número inteiro.

Mostra os resultados fracionários simplificados.



Mostra o símbolo π.

Mostra a forma simbólica das raízes que não podem ser calculadas como números inteiros.

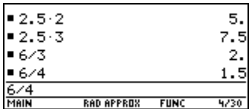


Prima [◆] [ENTER] para cancelar temporariamente a definição EXACT e apresentar um resultado com ponto flutuante.

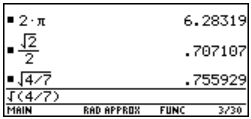
Nota: o modo EXACT permite conservar a forma fracionária e simbólica, reduzindo o risco de erros de arredondamento que podem ocorrer após uma série de cálculos intermediários numa série de cálculos concatenados.

APPROXIMATE — Todos os resultados numéricos, sempre que possível, aparecem na forma decimal (com ponto flutuante).

Nota: os resultados são arredondados com a precisão de cálculo da TI-89 Titanium / Voyage™ 200 e apresentados de acordo com o estado de definição actual.



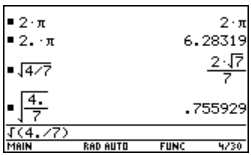
As fracções são tratadas como números decimais.



Os formatos simbólicos são calculados numericamente (sempre que possível).

As variáveis não definidas não podem ser calculadas e são tratadas de forma algébrica. Por exemplo: se a variável **r** não está definida, $\pi\ r^2 = 3.14159 \cdot r^2$.

AUTO — Este modo utiliza o formato EXACT, quando possível, ou o formato APPROXIMATE quando a entrada efectuada incluir um ponto decimal. Além disso, algumas funções podem apresentar resultados no formato APPROXIMATE mesmo que a entrada não inclua um ponto decimal.



A entrada de um número decimal determina um resultado em decimal.

Nota: para conservar um formato EXACT, utilize fracções em vez de decimais. Por exemplo, utilize 3/2 em vez de 1.5.

A tabela seguinte compara as três definições.

Entrada	Resultado exacto	Resultado aproximado	Resultado automático
8/4	2	2.	2
8/6	4/3	1.33333	4/3
8.5∗3	51/2	25.5	25.5
√(2)/2	$\frac{\sqrt{2}}{2}$	.707107	$\frac{\sqrt{2}}{2}$
π∗2	2·π	6.28319	2·π
π∗2.	2·π	6.28319	6.28319

— A entrada de um decimal determina o resultado em decimal no formato AUTO.

Nota: para calcular uma entrada no formato APPROXIMATE, independentemente da definição actual, prima [♦] [ENTER].

Modo Display Digits

A predefinição é **Display Digits = FLOAT 6**; isto significa que os resultados são arredondados com seis dígitos no máximo. Para seleccionar diferentes opções, utilize [MODE]. As opções são válidas para todos os formatos exponenciais.

A TI-89 Titanium / Voyage™ 200 calcula e conserva internamente todos os resultados decimais até 14 dígitos significativos (embora apenas apareçam 12 no máximo).

Definição	Exemplo		Descrição
FIX (0–12)	123. 123.5 123.46 123.457	(FIX 0) (FIX 1) (FIX 2) (FIX 3)	Os resultados são arredondados de acordo com a quantidade de casas decimais seleccionadas.
FLOAT	123.456789012		O número de casas decimais varia conforme o resultado.
FLOAT (1–12)	1.E 2 1.2E 2 123. 123.5 123.46 123.457	(FLOAT 1) (FLOAT 2) (FLOAT 3) (FLOAT 4) (FLOAT 5) (FLOAT 6)	Os resultados são arredondados conforme o número total de casas decimais seleccionadas.

Nota:

- Independentemente da definição do **Display Digits**, a calculadora utiliza nos cálculos internos de decimais o valor exacto para garantir a máxima precisão.
- Se não for possível apresentar o resultado com o número de dígitos seleccionado, a calculadora passa automaticamente ao sistema de notação científica.

Modo Exponential Format

Por predefinição, **Exponential Format = NORMAL**. Para seleccionar uma das três opções, utilize [MODE].

```
1:NORMAL
2:SCIENTIFIC
3:ENGINEERING
```

Definição	Exemplo	Descrição
NORMAL	12345.6	Se não for possível apresentar um resultado com o número de casas decimais especificadas no modo Display Digits, a TI-89 Titanium / Voyage™ 200 passa de NORMAL para SCIENTIFIC apenas para este resultado.
SCIENTIFIC	1.23456E 4 (❶ ❷)	1.23456×10^4
ENGINEERING	1.23456E 3 (❸ ❹)	12.3456×10^3

❶ Sempre 1 casa decimal à esquerda do ponto decimal.
❷ Expoente (potência de 10).

❸ À esquerda do ponto decimal, pode-se ter 1, 2, ou 3 casas decimais.
❹ O expoente é um múltiplo de 3.

Nota: na área do histórico, um número de uma entrada aparece no modo SCIENTIFIC, se o valor absoluto é menor que .001.

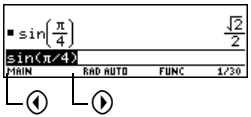
Edição de uma expressão na linha de entrada

O conhecimento do procedimento de edição de uma entrada economiza muito tempo. É mais fácil corrigir o erro de uma expressão do que a digitar de novo.

Remoção do realce de uma entrada precedente

Depois de premir [ENTER][F1] a expressão calculada permanece realçada na linha de entrada. Para a modificar, é preciso que não esteja realçada, para não a cancelar sem querer juntamente com uma nova entrada.

Para remover o realce, desloque o cursor para o lado da expressão que deseja modificar.



◀ desloca o cursor para o início.
▶ desloca o cursor para o final da expressão.

Deslocação do cursor

Depois de efectuar esta operação, desloque o cursor para a posição desejada da expressão.

Para deslocar o cursor:	Prima:	
Para a esquerda ou para a direita de uma expressão.	◀ ou ▶	Prima sem soltar a tecla para repetir o movimento.
Para o início da expressão.	[2nd] ◀	

Para deslocar o cursor:	Prima:
Para o fim da expressão.	2nd ⊙

Nota: se premir sem querer ⊙ em vez de ⊙ ou ⊙, o cursor vai para a área do histórico. Prima ESC ou ⊙ até que o cursor volte à linha de entrada.

Para cancelar um carácter

Para cancelar:	Prima:	
O carácter à esquerda do cursor.	←	Prima sem soltar ← para cancelar vários caracteres.
O carácter à direita do cursor.	◆ ←	
Todos os caracteres à direita do cursor.	CLEAR (apenas uma vez)	Se não houver caracteres à direita do cursor, CLEAR cancela toda a linha de entrada.

Para cancelar a linha de entrada

Para cancelar a linha de entrada, prima:

- CLEAR se o cursor estiver no início ou no fim da linha de entrada.

  – ou –

- CLEAR CLEAR se o cursor não estiver no início nem no fim da linha de entrada. O primeiro toque da tecla cancela todos os caracteres à direita do cursor e o segundo cancela toda a linha de entrada.

Inserção ou sobreposição de um carácter

A TI-89 Titanium / Voyage™ 200 contém um modo de inserção e um modo de sobreposição. Por predefinição, a TI-89 Titanium / Voyage™ 200 está definida para o modo de inserção; para alternar, prima [2nd] [INS].

Se a / estiver em:	O próximo carácter digitado:
`Insert mode` Cursor interposto entre dois caracteres	Será inserido no cursor.
`Overtype mode` O cursor realça um carácter	Substituirá o carácter realçado.

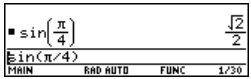
Nota: observe o cursor para verificar o modo actual.

Substituição ou cancelamento vários caracteres

Realce os caracteres desejados e, em seguida, substitua-os ou cancele-os.

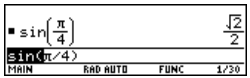
Para realçar vários caracteres:

1. Desloque o cursor para o início ou o fim dos caracteres que deseja realçar.



Para substituir **sin(** por **cos(**, posicione o cursor no início de **sin**.

2. Prima sem soltar [↑], prima ◀ ou ▶ para realçar os caracteres à esquerda ou à direita do cursor.

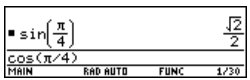


Prima sem soltar [↑] e prima ▶ ▶ ▶ ▶.

Para substituir ou cancelar os caracteres realçados:

1. Digite os novos caracteres.
2. Prima [←].

   Nota: se realçar os caracteres que deseja substituir, não se esqueça de que algumas teclas de função abrem automaticamente parêntesis.



Menus

Para reduzir a utilização do teclado, a TI-89 Titanium / Voyage™ 200

Graphing Calculator permite aceder a muitas aplicações através de menus. Esta secção explica como seleccionar um item de qualquer menu. Cada menu é descrito no módulo correspondente deste manual.

Visualização de um menu

Prima:	Para ver:
F1, F2, etc.	Um menu da barra de ferramentas — pendente na barra de ferramentas na parte superior da maioria dos ecrãs da aplicação. Permite seleccionar operações relativas à aplicação em questão.
APPS	O ambiente de trabalho das Apps ou o menu **APPLICATIONS** — permite aceder a uma lista das aplicações.
2nd [CHAR]	O menu **CHAR** — permite aceder a vários caracteres especiais (gregos, matemáticos, etc.).
2nd [MATH]	O menu **MATH** — permite aceder às categorias das operações matemáticas.
CATALOG 2nd [CATALOG]	O menu **CATALOG** — permite aceder ao índice alfabético completo das funções e instruções integradas na TI-89 Titanium / Voyage™ 200. Permite também seleccionar funções definidas pelo utilziador ou funções de aplicação Flash (se tiver definido ou carregado alguma).

Prima:	Para ver:
[2nd] [CUSTOM]	O menu **CUSTOM** — permite aceder a um menu que pode ser personalizado para listar qualquer função, instrução ou carácter disponível. A TI-89 Titanium / Voyage™ 200 inclui um menu personalizado predefinido, que pode ser modificado ou redefinido. Consulte o módulo Ecrã principal da calculadora e/ou o módulo Programação para obter mais informações sobre o menu personalizado.

Selecção de um item do menu

Para seleccionar um item do menu apresentado:

- Prima o número ou a letra apresentado à esquerda deste item. Para obter uma letra, prima [alpha] na TI-89 Titanium e, em seguida, a tecla correspondente à letra.

  – ou –

- Utilize a tecla do cursor ⊖ e ⊕ para realçar o elemento e, em seguida, prima [ENTER]. (Se premir ⊕ no primeiro item, o realce vai para o último item.)

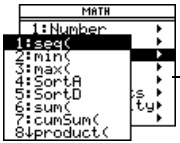
▾ indica que se premir [F2][F1] aparece um menu pendente da barra de ferramentas.

Para seleccionar **factor**, prima 2 ou ⊖ [ENTER]. Esta operação fecha o menu e insere a função no lugar do cursor.

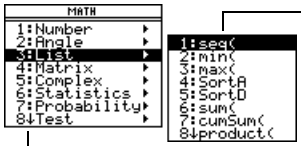
factor(

Itens com o símbolo ▶ (submenus)

Se seleccionar um item do menu com o símbolo ▶, aparece um submenu, que permite efetuar uma selecção.



Devido ao tamanho limitado do ecrã, a TI-89



↓ indica que com a tecla do cursor é possível ver

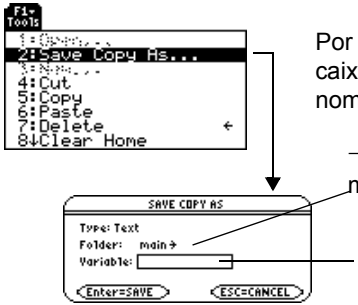
Por exemplo, List apresenta um submenu, que permite seleccionar funções específicas de List.

Para os itens que permitem aceder a um submenu, é possível utilizar a tecla do cursor, como explicado em seguida.

- Para mostrar o submenu do item realçado, prima ⊙.
  (É o mesmo procedimento utilizado para seleccionar o item.)
- Para cancelar o submenu sem efectuar uma selecção, prima ⊙. (O mesmo efeito que premir [ESC].)
- Para ir directamente do primeiro item do menu para o último item de menu, prima ⊙. Para ir directamente do último item do menu para o primeiro item de menu, prima ⊙.

Itens com reticências “. . .” (caixas de diálogo)

Se seleccionar um item do menu com “. . .” (reticências), aparece uma caixa de diálogo a pedir mais informações.



Por exemplo: **Save Copy As ...** aparece uma caixa de diálogo que pede a introdução do nome da pasta e de uma variável.

→ indica que é possível premir ⊙ para mostrar e efectuar uma selecção num menu.

Uma caixa de entrada indica que é preciso digitar um valor. (Alpha-lock é automaticamente activado para a TI-89

Depois de digitar uma variável numa caixa de introdução, é preciso premir [ENTER] duas vezes para memorizar a informação e fechar a caixa de diálogo.

Cancelamento de um menu

Para cancelar o menu actual sem efectuar uma selecção, prima [ESC]. Se aparecerem submenus, prima várias vezes [ESC] para cancelar todos os submenus.

Deslocação de um menu da barra de ferramentas para outro

Para ir de um menu da barra de ferramentas para outro sem efectuar nenhuma selecção:

- Prima a tecla ([F1], [F2], etc.) correspondente ao outro menu.

  – ou –

- Utilize a tecla do cursor para ir para o menu da barra de ferramentas posterior (prima ⊙) ou ao anterior (prima ⊙). Se premir ⊙ no último menu seleccionado, vai para o primeiro e vice-versa.

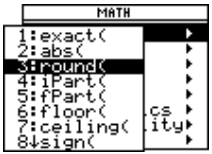
Quando utilizar ⊙, verifique se existe um item com um submenu realçado; se estiver, ⊙ mostra o submenu em vez de ir para o próximo menu da barra de ferramentas.

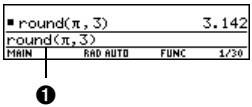
Exemplo: selecção de um item do menu

Arredonde o valor de π com três casas decimais. Suponha que a linha de entrada do ecrã principal está vazia:

1. Prima [2nd] [MATH] para mostrar o menu **MATH**.

2. Prima **1** para mostrar o submenu **Number**. (Ou prima [ENTER] porque o primeiro item é realçado automaticamente.)
3. Prima **3** para seleccionar **round**. (Ou prima ⊙ ⊙ e [ENTER].)
4. Prima [2nd] [π] [,] **3** [)] e, em seguida, [ENTER] para obter o resultado.





❶ Se seleccionar a função no 3º passo, escreve automaticamente **round(** na linha de entrada.

Selecção de uma aplicação

A TI-89 Titanium / Voyage™ 200 Graphing Calculator tem diversas aplicações que permitem resolver e analisar vários tipos de problemas. É possível seleccionar uma

aplicação de um menu, o ambiente de trabalho das Apps ou aceder às aplicações mais utilizadas a partir do teclado.

No menu APPLICATIONS

1. Se o ambiente de trabalho das Apps estiver desligado, prima [APPS] para ver o menu que contém a lista das aplicações.

   Nota: para cancelar o menu sem efectuar uma selecção, prima [ESC].

2. Para seleccionar uma aplicação:

   - Utilize ⊖ ou ⊕ para realçar a aplicação e, em seguida, prima [ENTER].

     – ou –

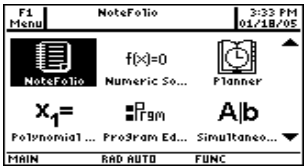
   - Prima o número correspondente à aplicação.

Aplicação:	Permite:
FlashApps	Mostrar uma lista das aplicações Flash, se houver.
Y= Editor	Definir, editar e seleccionar funções ou equações para a representação gráfica.
Window Editor	Definir o tamanho da janela de visualização dos gráficos.
Graph	Mostrar os gráficos.
Table	Mostrar uma tabela dos valores das variáveis correspondentes a uma função inserida.

Aplicação:	Permite:
Data/Matrix Editor	Introduzir e editar listas, dados e matrizes. É possível efectuar cálculos estatísticos e representar gráficos estatísticos.
Program Editor	Introduzir e editar programas e funções.
Text Editor	Introduzir e editar um texto.
Numeric Solver	Introduzir uma expressão ou equação, definir os valores de todas as variáveis, excepto uma e, em seguida, encontrar a solução para a variável desconhecida.
Home	Inserir expressões, instruções e efectuar cálculos.

No ambiente de trabalho das Apps

Prima a primeira letra do nome da aplicação ou utilize as teclas de cursor para realçar um ícone da aplicação no ambiente de trabalho das Apps e prima [ENTER]. (Se premir a primeira letra da aplicação e existir mais do que uma aplicação que comece por essa letra, a primeira por ordem alfabética é realçada). A aplicação abre-se directamente ou mostra uma caixa de diálogo. (O ambiente de trabalho das Apps pode variar do ambiente apresentado abaixo.)



A caixa de diálogo mais comum lista estas opções para a aplicação:

Opção	Descrição
Current	Volta ao ecrã apresentado na última visualização da App. (Se não houver nenhum arquivo/variável para a App seleccionada, a definição desta opção será New quando premir [ENTER].)
Open	Permite seleccionar um arquivo existente.
New	Cria um novo arquivo com o nome digitado no campo.

Seleccione uma opção e prima [ENTER]. Aparece a aplicação.

Nota: o termo geral *variável* é utilizado para os arquivos de dados de aplicação criados.

Utilize qualquer um destes métodos para voltar ao ambiente de trabalho das Apps quando estiver numa aplicação:

- Prima [APPS].
- No modo de ecrã cheio, prima [2nd] [QUIT].
- No modo de ecrã dividido, prima [2nd] [QUIT] para abrir a apresentação do ecrã cheio da aplicação activa; em seguida, prima [2nd] [QUIT] novamente.

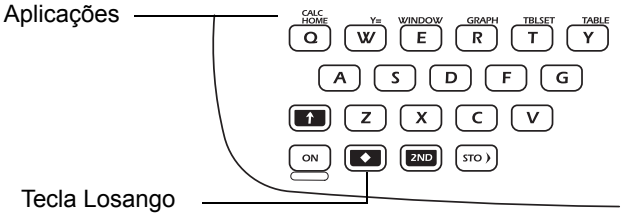
Para voltar à última aplicação aberta a partir do ambiente de trabalho das Apps, prima [2nd] [⇥].

No teclado

É possível aceder às aplicações mais utilizadas a partir do teclado. Na TI-89 Titanium, por exemplo, [◆] [Y=] é o mesmo que premir [◆] e, em seguida, [F1]. Este manual utiliza a notação [◆] [Y=], semelhante à notação utilizadas nas segundas funções.

Aplicação:	Premir:
Home	[HOME] [CALC HOME]
Y= Editor	[◆] [Y=]
Window Editor	[◆] [WINDOW]
Graph	[◆] [GRAPH]
Table Setup	[◆] [TBLSET]
Table Screen	[◆] [TABLE]

Algumas das aplicações mais utilizadas estão indicadas em cima das teclas QWERTY na Voyage™ 200.



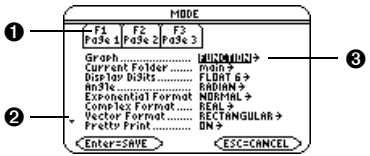
Nota: 2nd funções de tecla não aparecem no elemento gráfico acima.

Definição de modos

Os modos determinam a apresentação e interpretação de números e gráficos. Estas definições são guardadas automaticamente pela função Constant Memory™ quando desligar a calculadora. Todos os números, incluindo os elementos de matrizes e listas, aparecem de acordo com a definição escolhida.

Verificação das definições dos modos

Prima MODE para ver a caixa de diálogo MODE, que contém uma lista dos modos e as respectivas definições actuais.



❶ As definições de modos estão em três páginas. Prima F1, F2 ou F3 para ver directamente uma determinada página.
❷ Indica que é possível ir para baixo para ver mais modos.
❸ → indica que se premir ▷ ou ◁ é possível ver um menu e efectuar uma selecção.

Nota: os modos inválidos actuais estão inactivos. Por exemplo: na segunda página 2, **Split 2 App** não é válido, se **Split Screen = FULL**. Quando percorrer a lista, o cursor ignora as definições inactivas.

Modificação das definições dos modos

Na caixa de diálogo MODE:

1. Realce a definição do modo que deseja modificar. Utilize ⊖ ou ⊕ (com [F1], [F2], ou [F3]) para percorrer a lista.
2. Prima ⊕ ou ⊖ para ver um menu com as definições válidas. A definição actual está realçada.
3. Seleccione a definição desejada:
   - Realce a definição com ⊖ ou ⊕ e prima [ENTER].

     – ou –
   - Prima o número ou a letra correspondente à definição desejada.

   Nota: para sair de um menu e voltar à caixa de diálogo **MODE** sem efectuar uma selecção, prima [ESC].
4. Modifique, se necessário, as outras definições dos modos.
5. Depois de efectuar todas as modificações, prima [ENTER] para as memorizar e sair da caixa de diálogo.

Importante: se premir [ESC] em vez [ENTER] para sair da caixa de diálogo **MODE**, todas as modificações efectuadas serão canceladas.

Descrição geral dos modos

Nota: para mais informações sobre um modo específico, consulte a respectiva secção.

Modo	Descrição
Graph	Tipos de representação gráfica: FUNCTION, PARAMETRIC, POLAR, SEQUENCE, 3D, ou DE.
Current Folder	Pasta utilizada para armazenar e recuperar variáveis. Se não criar outra pasta, apenas a pasta MAIN está disponível. Consulte “Utilização de pastas para armazenar conjuntos independentes de variáveis” – no módulo *Ecrã principal da calculadora*.
Display Digits	Número máximo de dígitos (FLOAT) ou número fixo de casas decimais (FIX) apresentados num resultado decimal. Independentemente da definição escolhida, o número total de dígitos apresentados num resultado decimal não pode superar 12. (Consulte “Mostrar modo de dígitos na página 18
Angle	Unidades em que os valores dos ângulos são interpretados e visualizados: RADIANOS, GRAUS ou GRADOS.
Exponential Format	Notação utilizada para apresentar resultados: NORMAL, SCIENTIFIC ou ENGINEERING.
Complex Format	Formato utilizado para apresentar resultados complexos - se houver: REAL (os resultados complexos aparecem apenas se utilizar uma entrada complexa), RECTANGULAR, ou POLAR.
Vector Format	Formato utilizado para apresentar vectores de 2 e 3 elementos: RECTANGULAR, CYLINDRICAL ou SPHERICAL.

Modo	Descrição
Pretty Print	Activa e desactiva a função Pretty print.
Split Screen	Divide o ecrã em duas partes, especificando a sua disposição: FULL (ecrã não dividido), TOP-BOTTOM, ou LEFT-RIGHT. Consulte o módulo *Ecrãs divididos*.
Split 1 App	Aplicação apresentada na parte superior ou à esquerda de um ecrã dividido. Se não dividir o ecrã, indica a aplicação actual.
Split 2 App	Aplicação apresentada na parte inferior ou à direita de um ecrã dividido. Activa apenas o ecrã dividido.
Number of Graphs	O ecrã dividido permite definir as partes do ecrã para apresentar conjuntos de gráficos independentes.
Graph 2	Se **Number of Graphs = 2**, selecciona o tipo de gráfico da segunda parte do ecrã dividido. Consulte o módulo *Ecrã principal da calculadora*.
Split Screen Ratio	Proporção das duas partes do ecrã dividido: 1:1, 1:2, ou 2:1. ([◀][×][÷][8][⇥][e^x] [▭])
Exact/Approx	Calcula as expressões e apresenta os resultados em forma numérica ou racional/simbólica: AUTO, EXACT ou APPROXIMATE.
Base	Permite realizar cálculos através da digitação dos números nas formas decimal (DEC), hexadecimal (HEX) ou binária (BIN).
Unit System	Permite seleccionar um dos três sistemas de medidas para especificar as unidades padrão para os resultados apresentados: **SI** (métrico ou MKS); **Eng/US** (pés, libras, etc.); ou **Custom**.

Modo	Descrição
Custom Units	Permite seleccionar padrões personalizados. O modo fica inactivo até que Unit System, **3:CUSTOM** seja seleccionado.
Language	Permite localizar a TI-89 Titanium / Voyage™ 200 para um dos idiomas possíveis, dependendo dos idiomas das aplicações Flash que estiverem instalados.
Apps Desktop	Liga ou desliga o ambiente de trabalho das Apps.

Utilização do menu Clean Up para iniciar um novo problema

O menu da barra de ferramentas **Clean Up** permite iniciar um novo cálculo a partir de um estado inicia, sem reinicializar a memória da TI-89 Titanium / Voyage™ 200 Graphing Calculator.

Menu da barra de ferramentas Clean Up

Veja o menu **Clean Up** no ecrã principal, premindo:

[2nd] [F6]

[F6]

Item do menu	Descrição
Clear a–z	Limpa (exclui) todos os nomes de variável de um único carácter na pasta actual, excepto se as variáveis estiverem bloqueadas ou arquivadas. Aparece uma mensagem a pedir que prima [ENTER] para confirmar a acção.
	Os nomes de variáveis com um único carácter são frequentemente utilziados em cálculos simbólicos, como: **solve(aøx^2+bøx+c=0,x)** Se uma das variáveis já tiver um valor atribuído, o seu cálculo pode produzir resultados incorrectos. Para impedir isto, seleccione **1:Clear a–z** antes de começar o cálculo.
NewProb	Coloca **NewProb** na linha de entrada. Em seguida, é necessário premir [ENTER] para executar o comando. **NewProb** realiza várias operações que permitem começar um novo problema a partir de um estado inicial sem reconfigurar a memória: • Limpa todos os nomes de variáveis de um único carácter na pasta actual (o mesmo que **1:Clear a–z**), excepto se as variáveis estiverem bloqueadas ou arquivadas. • Desactiva todas as funções e gráficos estatísticos (**FnOff** e **PlotsOff**) no modo gráfico actual. • Executa **ClrDraw**, **ClrErr**, **ClrGraph**, **ClrHome**, **ClrIO**, e **ClrTable**.
Restore custom default	Se um menu personalizado diferente da predefinição estiver activo, é possível restaurar a predefinição. Consulte o módulo *Ecrã principal da calculadora* para obter informações sobre o menu personalizado.

Nota:

- Quando definir uma variável que deseja manter, utilize mais do que um carácter no seu nome. Este procedimento impede que seja cancelada inadvertidamente por **1:Clear a–z**.
- Para mais informações sobre a verificação e reconfiguração da memória e outras predefinições do sistema, consulte o módulo *Gestão da memória e das variáveis*.

Utilização da caixa de diálogo Catalog

A caixa de diálogo CATALOG permite aceder a qualquer comando embutido da TI-89 Titanium / Voyage™ 200 Graphing Calculator (funções e instruções) a partir de uma lista conveniente. Além disso, a caixa de diálogo CATALOG permite seleccionar funções utilizadas nas aplicações Flash ou nas funções definidas pelo utilizador (se tiver carregado ou definido alguma).

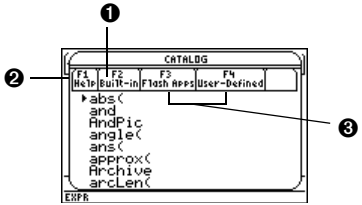
Visualização da caixa de diálogo CATALOG

Para ver a caixa de diálogo CATALOG, prima:

[CATALOG]

[2nd] [CATALOG]

A predefinição da caixa de diálogo CATALOG é [F2] **Built-in**, que mostra uma lista alfabética de todos os comandos pré-instalados da TI-89 Titanium / Voyage™ 200 (funções e instruções).



❶ Mostra [F2] Built-in.
❷ [F1] Help mostra os parâmetros de um comando numa caixa de diálogo.
❸ [F3] e [F4] permitem aceder às funções da aplicação Flash e às funções e programas definidos pelo utilizador.

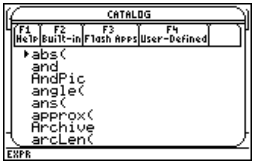
Nota: as opções inválidas no momento ficam inactivas. Por exemplo, [F3] **Flash Apps** fica inactiva, se não houver uma aplicação Flash instalada. [F4] **User-Defined** fica inactiva, se não tiver criado uma função ou programa.

Selecção do menu CATALOG

Quando seleccionar um comando, o nome é transcrito na linha de entrada na posição do cursor. É necessário posicionar correctamente o cursor antes de seleccionar um comando.

1. Prima:
   [CATALOG]
   [2nd] [CATALOG]

2. Prima [F2] **Built-in**.



- Os comandos aparecem por ordem alfabética. Os comandos que não começam por uma letra (+, %, √, Σ, etc.), estão no fim da lista.
- Para sair de **CATALOG** sem seleccionar um comando, prima [ESC].

Nota: quando a lista **Built-in** aparecer pela primeira vez, começa no topo da lista. Da próxima vez que a lista aparecer, começa no mesmo lugar onde ficou.

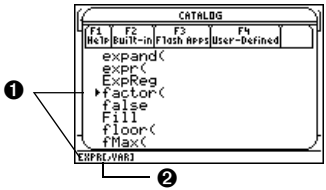
3. Posicione o indicador ▶ no comando e prima [ENTER].

Para deslocar o indicador ▶:	**Prima ou digite:**
Um comando de cada vez	⊖ ou ⊕
Uma página de cada vez	[2nd] ⊖ ou [2nd] ⊕
Para o primeiro comando que inicia com uma determinada letra	A tecla de letra. (Na TI-89 Titanium, não prima [alpha] primeiro. Se o fizer, precisa de premir [alpha] ou [2nd] [a-lock] novamente antes de poder digitar uma letra.)

Nota: prima ⊕ no início da lista para ir para o fim; prima ⊖ para ir do fim para o princípio.

Informações sobre os parâmetros

Para os comandos assinalados com ▶, a linha de estado indica os parâmetros pedidos e opcionais (se disponíveis), e os respectivos tipos.



❶ Comando indicado e respectivos parâmetros.
❷ Os parêntesis rectos [ ] indicam os parâmetros opcionais.

No exemplo acima, a sintaxe do **factor** é:

factor(*expressão***)**	necessário
– ou –	
factor(*expressão*, *variável***)**	opcional

Nota: para mais informações sobre os parâmetros, consulte a descrição correspondente no módulo *Referência técnica*.

Visualização da ajuda do CATALOG

Prima F1 **Help** para apresentar os parâmetros de um comando numa caixa de diálogo. Os parâmetros são iguais aos parâmetros apresentados na linha de estado.

Comando indicado e respectivos parâmetros.

Alguns comandos, como o **ClrDraw**, não necessitam de parâmetros. Se seleccionar um desses comandos, os parâmetros não aparecem na linha de estado; além disso, Unavailable aparece, se premir [F1] **Help.**

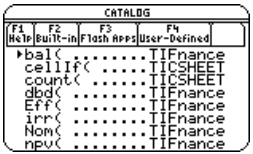
Prima [ESC] para sair da caixa de diálogo CATALOG Help.

Selecção de uma função de aplicação Flash

Uma aplicação Flash pode conter uma ou mais funções. Se seleccionar uma função, o seu nome é inserido na linha de entrada na posição do cursor. Portanto, posicione o cursor conforme necessário antes de seleccionar a função.

1. Prima:
   - [CATALOG]
   - [2nd] [CATALOG]

2. Prima [F3] **Flash Apps**. (Esta opção fica inactiva, se não tiver instalado nenhuma aplicação Flash na TI-89 Titanium / Voyage™ 200.)



- A lista é classificada pelo nome da função. A coluna da esquerda lista as funções. A coluna da direita lista a aplicação Flash que contém a função
- As informações sobre uma função aparecem na linha de estado.
- Para sair sem seleccionar uma função, prima [ESC].

3. Mova o indicador ▶ para a função e prima [ENTER].

Para mover o indicador ▶:	**Prima ou digite:**
Uma função por vez	⊖ ou ⊕
Uma página por vez	[2nd] ⊖ ou [2nd] ⊕
Para a primeira função que comece com uma letra especificada	A tecla da letra. (Na TI-89 Titanium, não prima [alpha] primeiro. Se o fizer, precisa de premir [alpha] ou [2nd] [a-lock] novamente antes de digitar uma letra.)

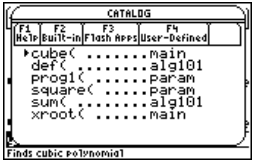
Selecção de uma função ou programa definido pelo utilizador

Pode criar os seus próprios programas ou funções e, em seguida, utilizar [F4] **User-Defined** para lhes aceder. Para mais informações sobre como criar funções, consulte “Criação e cálculo de funções definidas pelo utilizador” no módulo *Ecrã*

principal da calculadora e “Visão geral da digitação de uma função” no módulo *Programação*. Consulte o módulo *Programação* para obter mais informações sobre como criar e executar um programa.

Se seleccionar uma função ou programa, o seu nome é inserido na linha de entrada na posição do cursor. Portanto, posicione o cursor conforme necessário antes de seleccionar a função ou programa.

1. Prima:
   [CATALOG]
   [2nd] [CATALOG]
2. Prima [F4] **User-Defined**. (Esta opção fica inactiva, se não tiver definido nenhuma função ou criado nenhum programa.)



- A lista é classificada pelo nome da função/programa. A coluna da esquerda lista as funções e programas. A coluna da direita lista a pasta que contém a função ou programa.
- Se a primeira linha da função ou programa for um comentário, o texto do comentário aparece na linha de estado.
- Para sair sem seleccionar uma função ou programa, prima [ESC].

Nota: utilize o ecrã **VAR-LINK** para gerir variáveis, pastas e aplicações Flash. Consulte o módulo *Gestão da memória e das variáveis*.

3. Mova o indicador ▶ para a função ou programa e prima [ENTER].

Para mover o indicador ▶:	Prima ou digite:
Uma função ou programa por vez	⊖ ou ⊕
Uma página por vez	[2nd] ⊖ ou [2nd] ⊕
Para a primeira função ou programa que comece com uma letra especificada	A tecla da letra. (Na TI-89 Titanium, não prima [alpha] primeiro. Se o fizer, precisa de premir [alpha] ou [2nd] [a-lock] novamente antes de digitar uma letra.)

Armazenamento e recuperação dos valores de variáveis

Os valores são armazenados com nomes de variáveis. É possível utilizar o nome em vez do valor nas expressões. Quando a TI-89 Titanium / Voyage™ 200 Graphing Calculator encontrar o nome numa expressão, substitui o valor armazenado da variável.

Regras para nomes das variáveis

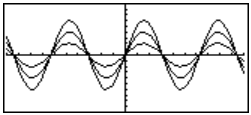
O nome de uma variável:

- Pode conter de 1 a 8 caracteres composto por letras e dígitos, inclusive letras do alfabeto grego (excepto π), letras com acento e internacionais.
  - Não introduza espaços.
  - O primeiro carácter não pode ser um dígito.
- É possível utilizar letras maiúsculas e minúsculas. Os nomes **AB22**, **Ab22**, **aB22**, e **ab22** referem-se à mesma variável.
- Não é possível utilizar um nome predefinido pela TI-89 Titanium / Voyage™ 200. Os nomes predefinidos incluem:
  - funções (como **abs**) e instruções (tais como **LineVert**). Consulte o módulo *Referência técnica*.
  - variáveis de sistema (como **xmin** e **xmax**, utilizadas para armazenar valores associados a gráficos). Consulte o módulo *Referência técnica*.

Exemplos

Variável	Descrição
myvar	OK
a	OK
Log	Não é válido; nome já dado à função **log**.
Log1	OK
3rdTotal	Não é válido, pois começa com um dígito.
circumfer	Não é válido, pois supera 8 caracteres.

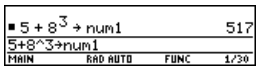
Tipos de dados

Tipos de dados	Exemplos
Expressões	2.54, 1.25E6, 2π, xmin/10, 2+3i, $(x-2)^2$, $\sqrt{2}/2$
Listas	{2 4 6 8}, {1 1 2}
Matrizes	$\begin{bmatrix}1 & 0 & 0\end{bmatrix}$, $\begin{bmatrix}1 & 0 & 0\\3 & 4 & 6\end{bmatrix}$
Cadeia de caracteres	“Hello”, “The answer is:”, “xmin/10”
Imagens	
Funções	myfunc(arg), ellipse(x,y,r1,r2)

Armazenar um valor numa variável

1. Introduza o valor que deseja armazenar, que pode ser uma expressão.
2. Prima [STO▸]. Aparece o símbolo de armazenamento (→).
3. Digite o nome da variável.

   Nota: Os utilizadores da TI-89 Titanium devem utilizar [alpha] conforme necessário, se digitarem nomes de variáveis.

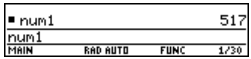


4. Prima [ENTER].

Para armazenar temporariamente uma variável, é possível utilizar o operador “*with*”. Consulte “Substituição de valores e configuração das restrições” – módulo *Manipulação Simbólica*.

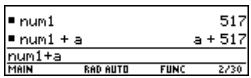
Visualização de uma variável

1. Digite o nome da variável.
2. Prima [ENTER].



Se a variável não estiver definida, aparece o seu nome como resultado.

Neste exemplo, a variável a não está definida; portanto, é utilizada como variável simbólica.

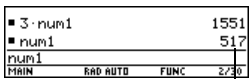


Nota: consulte o módulo *Manipulação Simbólica* para informações anteriores sobre a manipulação simbólica.

Utilização de uma variável numa expressão

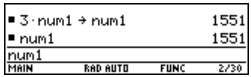
1. Digite o nome da variável na expressão.
2. Prima [ENTER] para calcular a expressão.

   Nota: para ver uma lista dos nomes das variáveis existentes, utilize [2nd] [VAR-LINK] como descrito no módulo *Gestão da memória e das variáveis*.



O valor da variável permaneceu o mesmo.

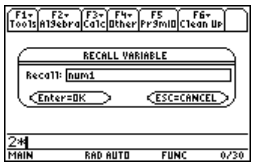
Se quiser que o resultado substitua o valor anterior da variável, tem de armazenar o resultado.



Recuperação do valor de uma variável

Em alguns casos, pode ser útil utilizar o valor efectivo de uma variável em vez do nome.

1. Prima [2nd] [RCL] para visualizar uma caixa de diálogo.
2. Digite o nome da variável.
3. Prima duas vezes [ENTER].



Neste exemplo, o valor armazenado em **num1** será transcrito para a posição do cursor na linha de entrada.

Indicadores da linha de estado no ecrã

A linha de estado aparece na parte inferior do ecrã de todas as aplicações. Contém informações sobre o estado actual da TI-89 Titanium / Voyage™ 200 Graphing Calculator, inclusive importantes configurações dos modos.

Indicadores da linha de estado

❶ Pasta actual
❷ Tecla modificadora
❸ Modo Angle
❹ Modo Exact/Approx
❺ Número gráfico
❻ Modo Graph
❼ Trocar pilhas
❽ Pares do histórico, Indicador de ocupado, pausa, variáveis bloqueadas

Indicador	Significado
Pasta actual	Mostra nome da pasta activa. Consulte “Utilização das pastas para a armazenamento de conjuntos independentes de variáveis” no módulo *Ecrã principal da calculadora*. MAIN é a pasta predefinida que aparece automaticamente se utilizar a TI-89 Titanium / Voyage™ 200.
Tecla modificadora	Mostra a tecla modificadora activa como descrito abaixo.
2nd	[2nd] — utilizará a segunda função da próxima tecla premida.
◆	[◆] — utilizará o recurso do losango da próxima tecla premida.

Indicador	Significado
⬆	[↑] — digitará a letra maiúscula da próxima tecla premida. Na TI-89 Titanium, é possível utilizar [↑] para digitar uma letra sem ter que utilizar [alpha].
(🖩)	[alpha] — digitará a letra minúscula da próxima tecla premida.
(🖩)	[2nd] [a-lock] — o alpha-lock minúsculo está activo. Até que o desative, será digitada a letra minúscula de cada tecla premida. Para cancelar o alpha-lock, prima [alpha].
(🖩)	[↑] [alpha] — o ALPHA-lock maiúsculo está activo. Até que o desative, será digitada a letra maiúscula de cada tecla premida. Para cancelar o alpha-lock, prima [alpha].
(⌨)	Quando utilizado juntamente com uma tecla de cursor, a Voyage™ 200 utilizará qualquer recurso "arrastar" disponível nos gráficos e na geometria.
Modo Angle	Indica a unidade com a qual os ângulos são interpretados e apresentados. Para modificar o modo Angle, utilize a tecla [MODE].
RAD	Radianos
DEG	Graus
GRD	Grados
Modo Exact/Approx	Indica como as respostas são calculadas e apresentadas. (Consulte “Modo Exact/Approc” na página 16.) Para modificar o modo Exact/Approx, utilize a tecla [MODE].
AUTO	Automático

Indicador	Significado
EXACT	Exacto
APPROX	Aproximado
Número gráfico	Se o ecrã estiver dividido para mostrar dois gráficos, indica o activo — **G1** ou **G2**. (Mostra **GR#1** ou **GR#2** na Voyage™ 200.)
Modo gráfico	Indica os tipos de gráfico que podem ser representados. (Para modificar o modo gráfico, prima a tecla [MODE].)
FUNC	funções **y(x)**
PAR	equações paramétricas **x(t)** e **y(t)**
POL	equações polares **r(θ)**
SEQ	sucessões **u(n)**
3D	Gráfico tridimensional de equações **z(x,y)**
DE	equações diferenciais **y'(t)**
Indicador das pilhas	Aparece apenas quando as pilhas estão a acabar. Se aparecer BATT num fundo preto, troque as pilhas assim que for possível.
Pares do histórico, Indicador de ocupado, pausa, variáveis bloqueadas	As informações apresentadas nesta parte da linha de estado dependem da aplicação utilizada.

Indicador	Significado
23/30	Apresentado no ecrã principal para mostrar o número de pares de entrada/resposta na área do histórico. Consulte as Informações do histórico na linha de estado no módulo *Ecrã principal da calculadora*.
BUSY	Um cálculo ou gráfico está em execução.
PAUSE	Foi efectuada uma pausa num gráfico ou programa.
🔒	A variável aberta no editor actual (Editor de Dados/Matrizes, Editor de Programas ou Editor de Texto) está bloqueada ou arquivada e não pode ser modificada.

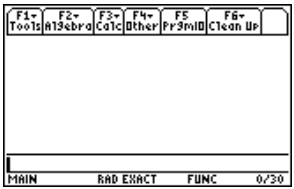
Nota:

- Para cancelar [2nd], [◆], [alpha] ou [↑], prima a mesma tecla novamente ou prima uma tecla modificadora diferente.
- Se a próxima tecla premida não tiver o recurso do losango ou uma letra associada, a tecla executa a função normal.

Ecrã principal da calculadora

Ecrã principal da calculadora

O ecrã principal da calculadora é o ponto de início das operações matemáticas, incluindo executar instruções, avaliar expressões e ver resultados.



Um ecrã principal da calculadora em branco

Este módulo descreve as partes do ecrã principal da calculadora, como percorrer o ecrã principal ou modificar a área do histórico; como cortar, copiar, colar, etc.

Nota: o termo “ecrã principal da calculadora” é utilziado neste módulo. Outros módulos utilizam o termo “Ecrã principal.” Ambos os termos são permutáveis e referem-se ao mesmo ecrã.

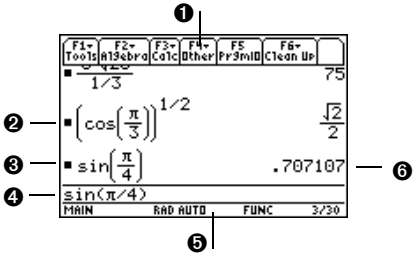
Visualização do ecrã principal da calculadora

Quando ligar a TI-89 Titanium ou a Voyage™ 200 Graphing Calculator pela primeira vez, aparece o ambiente de trabalho das Apps. Para ver o ecrã principal da calculadora, realce o ícone Home e prima [ENTER]. Pode também ver o ecrã principal da calculadora,

premindo [HOME] (TI-89 Titanium) or [◆] [CALC HOME] (Voyage™ 200). Se desactivar o modo Apps Desktop, o ecrã principal da calculadora aparece automaticamente.

Partes do ecrã principal da calculadora

O exemplo seguinte contém os dados introduzidos previamente e descreve as partes principais do ecrã principal da calculadora. Os pares de entradas/respostas na área do histórico aparecem em “pretty print.” Pretty print mostra as expressões da forma que estão escritas no quadro ou nos livros de exercícios.



❶ Toolbar
Permite ver os menus de selecção de operações aplicáveis no ecrã principal da calculadora. Para ver um menu da barra de ferramentas, prima [F1], [F2], etc.

❷ Pretty Print Display
Mostra os expoentes, raízes quadradas, fracções, etc., da forma tradicional.

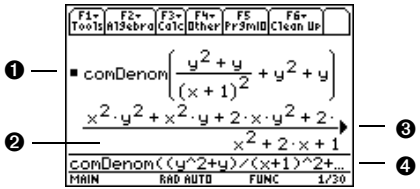
❸ Last Entry
Última entrada.

❹ Entry Line
Onde introduz expressões ou instruções.

❺ Status Line
Mostra o estado actual da calculadora, incluindo várias definições de modos importantes.

❻ Last Answer
Resultado da última entrada. Não se esqueça de que os resultados não aparecem na linha de entrada. Nota: [♦] [ENTER] (Approx) foi utilizado neste exemplo.

O exemplo seguinte mostra uma resposta que não está na mesma linha da expressão. Não se esqueça de que a resposta é maior do que a largura do ecrã. Uma seta (▶) indica que a resposta continua. A linha de entrada contém a elipse (…). A elipse indica que a entrada é maior do que a largura do ecrã.



❶ Last Entry
"Pretty print" está activada. Os exponentes, raízes quadradas, fracções, etc., aparecem da mesma forma que são escritas tradicionalmente.

À History Area
Lista os pares de entradas/respostas introduzidos. Os pares percorrem o ecrã à medida que efectua novas entradas.

Ã Answer Continues
Realce a resposta e prima B para ir para a direita para ver o resto da resposta. Não se esqueça de que a resposta não está na mesma linha da expressão.

Õ Expression C**ontinues (…)**
Prima ⊙ para ir para a direita para ver o resto da entrada. Prima 2nd ⊙ ou 2nd ⊙ para ir para o início ou o fim da linha de entrada.

Área do histórico

A área do histórico mostra até oito pares de entradas/respostas anteriores (dependendo da complexidade e da altura das expressões apresentadas). Quando o visor estiver cheio, as informações ultrapassam o topo do ecrã. Pode utilizar a área do histórico para:

- Rever as entradas e as respostas anteriores. Pode utilizar o cursor para ver as entradas e as respostas no ecrã.
- Voltar a chamar ou colar automaticamente uma entrada ou resposta anterior na linha de entrada para que a possa reutilizar ou editar.

Visualização da área do histórico

Geralmente, o cursor está na linha de entrada. No entanto, pode mover o cursor para a área do histórico.

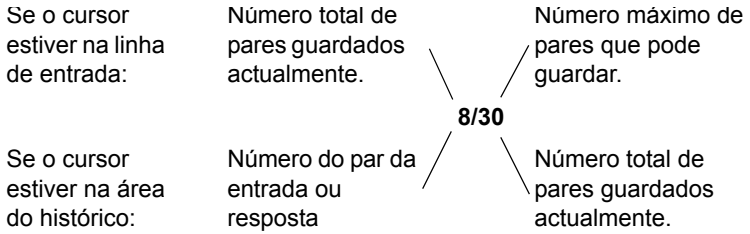
Para:	Efectue o procedimento apresentado em seguida:
Ver as entradas ou as respostas que estão fora do ecrã	• Prima ⊕ para realçar a última resposta na linha de entrada. • Continue a utilizar ⊕ para mover o cursor da resposta para a entrada até à área do histórico.
Ir para o par do histórico mais antigo ou mais recente	Se o cursor estiver na área do histórico, prima [◆] ⊕ ou [◆] ⊖, respectivamente.
Ver uma entrada ou resposta demasiado longa para uma linha (▶ está no fim da linha)	Mova o cursor para a entrada ou a resposta. Utilize ◀ e ▶ para ir para a esquerda e a direita (ou [2nd] ◀ e [2nd] ▶ para ir para o início ou o início), respectivamente.
Colocar o cursor na linha de entrada	Prima [ESC] ou ⊖ até o cursor ir para a linha de entrada.

Nota: está disponível um exemplo de visualização de uma resposta longa.

Informações do histórico na linha de estado

Utilize o indicador do histórico na linha de estado para obter informações sobre os pares de entradas/respostas. Por exemplo:

Se o cursor estiver na linha de entrada:	Número total de pares guardados actualmente.	**8/30**	Número máximo de pares que pode guardar.
Se o cursor estiver na área do histórico:	Número do par da entrada ou resposta		Número total de pares guardados actualmente.



Por predefinição, são guardados os últimos 30 pares de entradas/respostas. Se a área do histórico estiver cheia quando efectuar uma nova entrada (indicada por 30/30), o novo par de entrada/resposta é guardado e o par antigo é eliminado. O indicador do histórico não se altera.

Modificação da área do histórico

Para:	**Efectue o procedimento apresentado em seguida:**
Alterar o número de pares que pode guardar	Prima [F1] e seleccione **9:Format** ou prima [◆] [I] [◆] **F** Em seguida, prima ▶, utilize ▲ ou ▼ para realçar o novo número e prima duas vezes [ENTER].
Limpar a área do histórico e eliminar todos os pares guardados	Prima [F1] e seleccione **8:Clear Home** ou introduza **ClrHome** na linha de entrada.
Eliminar um par de entrada/resposta específico	Mova o cursor para a entrada ou a resposta. Prima [←] ou [CLEAR].

Armazenamento de entradas do ecrã principal da calculadora como um script para o editor de texto

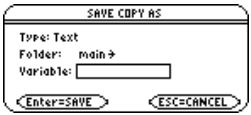
Para guardar todas as entradas da área do histórico, é possível guardar o ecrã principal numa variável de texto. Quando quiser tornar a executar estas entradas, utilize o Text Editor para abrir a variável como um script de comando.

Armazenamento das entradas na área do histórico

1. No ecrã principal da calculadora:
2. Prima F1 e seleccione **2:Save Copy As**.

3. Especifique a pasta e a variável de texto que deseja utilizar para guardar as entradas.

   Nota: o armazenamento refere-se às entradas e não as respostas.



Item	Descrição
Type	É configurada automaticamente como Text e não é possível modificá-la.
Folder	Mostra a pasta onde será guardada a variável de texto. Se quiser utilizar uma pasta diferente, prima ⊙ para ver o menu das pastas existentes e seleccione a pasta desejada.
Variable	Digite um nome válido de variável - um nome não utilizado.

Nota: para mais informações sobre as pastas, consulte o módulo *Gestão da memória e das variáveis*.

4. Prima [ENTER] (após uma entrada numa caixa de entrada, tal como Variable, prima duas vezes [ENTER]).

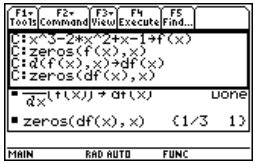
Recuperação de entradas memorizadas

Visto que as entradas são guardadas com o formato de script, não é possível recuperá-las no ecrã principal da calculadora. (No menu da barra de ferramentas [F1] do ecrã principal da calculadora, a opção **1:Open** não está disponível.) Efectue o procedimento apresentado em seguida:

1. Utilize o Text Editor para abrir a variável que contém as entradas guardadas do ecrã principal da calculadora.

   Estas entradas aparecem como uma série de linhas de comando que podem ser executadas individualmente na ordem desejada.

2. Com o cursor posicionado na primeira linha do script, prima várias vezes [F4] para executar os comandos linha por linha.
3. Visualize o ecrã principal da calculadora.



   Este ecrã dividido mostra o Text Editor (com o script da linha de comando) e o ecrã principal da calculadora restabelecido.

Nota: para mais informações sobre a utilização do Text Editor e como executar um script de comando, consulte o módulo *Editor de Texto*.

Corte, cópia e colagem de informações

O procedimento para cortar, copiar e colar pode ser utilizado na mesma aplicação ou em aplicações diferentes. Estas operações utilizam uma área de transferência reservada da TI-89 Titanium / Voyage™ 200 Graphing Calculator para armazenamento temporário de dados.

Comparação entre colar automaticamente e cortar/copiar/colar

O procedimento de colagem automática é a forma mais rápida para copiar uma entrada ou resposta da área do histórico e colá-la na linha de entrada.

1. Utilize ⊕ e ⊖ para realçar um item na área do histórico.
2. Prima [ENTER] para colar automaticamente o item na linha de entrada.

As operações para cortar (cut), copiar (copy) ou colar (paste) permitem transferir informações para a linha de entrada; uma informação da área do histórico pode ser copiada, mas não cortada ou colada.

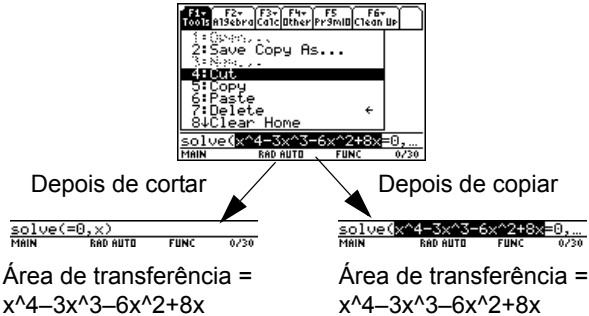
Operação para cortar ou copiar informações na área de transferência

Uma informação cortada ou copiada é colocada na área de transferência. A operação de corte remove a informação da posição (utilizada para deslocação), enquanto a operação de cópia deixa a informação na posição original.

1. Realce os caracteres que deseja cortar ou copiar.

   Na linha de entrada, posicione o cursor à direita ou à esquerda dos caracteres. Prima sem soltar [↑] e, em seguida, prima ◀ ou ▶ para realçar respectivamente os caracteres à direita e à esquerda do cursor.
2. Prima [F1] e seleccione **4:Cut** ou **5:Copy**.

Áreas de transferência = (área vazia ou com o conteúdo anterior)



Nota: pode cortar, copiar e colar sem utilizar o menu da barra de ferramentas de [F1]. Prima:

[◆] [CUT], [◆] [COPY], ou [◆] [PASTE]

[◆] **X**, [◆] **C**, ou [◆] **V**

Cortar não é o mesmo que cancelar, pois a informação cancelada não fica guardada na área de transferência e portanto não pode ser recuperada.

Nota: uma informação cortada ou copiada substitui o eventual conteúdo da área de transferência.

Operação para colar uma informação da área de transferência

A operação para colar insere o conteúdo da área de transferência na posição do cursor na linha de entrada. O conteúdo da área de transferência permanece igual.

1. Posicione o cursor no ponto onde deseja colar a informação.

2. Prima [F1] e seleccione **6:Paste** (ou utilize as teclas de atalho):
   [◆] [PASTE]
   [◆] **V**

Exemplo: copiar e colar

Para reutilizar uma expressão sem digitá-la de novo por inteiro.

1. Copie a informação desejada.

   a) Utilize [↑] [▶] ou [↑] [◀] para realçar a expressão.

   b) Prima:
      [◆] [COPY]
      [◆] **C**

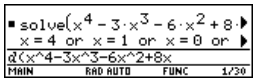
   c) c.Neste exemplo, prima [ENTER] para calcular a expressão.

2. Cole a informação na nova entrada.

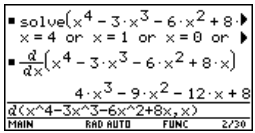
   a) Inicie uma nova entrada e posicione o cursor no local onde deseja colar a informação copiada.

b) Prima [F3] **1** para seleccionar a função *d* differentiate.

c) Prima
[◆] [PASTE]
[◆] **V**
para colar a expressão copiada.



d) Complete a nova entrada e prima [ENTER].



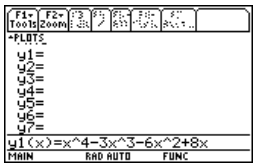
Nota: também é possível reutilizar uma expressão através de uma função definida pelo utilizador.

3. Cole a informação copiada noutra aplicação.

a) Prima [◆] [Y=] para ver Y= Editor.

b) Prima [ENTER] para definir **y1(x)**.

c) Prima
[◆] [PASTE]
[◆] **V**
para colar.



d) Prima [ENTER] para guardar a nova definição.

Nota: a operação de copiar e colar permite transferir facilmente informações de uma aplicação para outra.

Reutilização de uma entrada precedente ou da última resposta

É possível reutilizar uma entrada precedente, executando-a de novo “tal e qual” ou modificando-a antes de a tornar a executar. Também é possível reutilizar a última resposta, inserindo-a numa nova expressão.

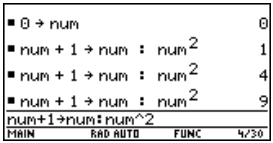
Reutilização da expressão na linha de entrada

Se premir [ENTER], a expressão calculada permanece realçada na linha de entrada. Conforme a situação, é possível sobrepô-la ou reutilizá-la.

Por exemplo: utilize uma variável para calcular o quadrado de 1, 2, 3, etc.

Como mostrado em seguida, ajuste o valor da variável inicial e digite a expressão variável. Em seguida, volte a digitar para incrementar a variável e calcular o quadrado.

		Ecrã
0 [STO▸] [2nd] [a-lock] NUM [ENTER]	0 [STO▸] NUM [ENTER]	■ 0 → num 0 0→num MAIN RAD AUTO FUNC 1/30
NUM [alpha] [+] 1 [STO▸] [2nd] [a-lock] NUM [2nd] [:] NUM [^] 2 [ENTER]	NUM [+] 1 [STO▸] NUM [2nd] [:] NUM [^] 2 [ENTER]	■ 0 → num 0 ■ num + 1 → num : num^2 1 num+1→num:num^2 MAIN RAD AUTO FUNC 2/30

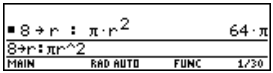
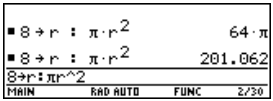
		Ecrã
[ENTER] [ENTER]	[ENTER] [ENTER]	

Nota: tornar a executar a entrada “tal e qual”, é muito útil nos cálculos interactivos que incluem variáveis.

Com a equação **A=π r**^{2}, utilize teste e erro para calcular o raio de um círculo que cobre 200 centímetros quadrados

Nota: editar uma entrada permite realizar pequenas correcções sem a necessidade de a reescrever.

O exemplo seguinte utiliza 8 como primeiro palpite e, em seguida, mostra a resposta na forma aproximada de ponto flutuante. É possível editar e voltar a executar com `7.95` e continuar até que a resposta esteja tão precisa quanto desejado.

		Ecrã
8 [STO▸] [alpha] R [2nd] [:] [2nd] [π] [alpha] R [^] 2 [ENTER]	8 [STO▸] R [2nd] [:] [2nd] [π] R [^] 2 [ENTER]	
[◆] [ENTER]	[◆] [ENTER]	

		Ecrã
◀ ◆ [DEL] 7.95 [ENTER]	◀ ◆ [DEL] 7.95 [ENTER]	■8→r : π·r² 64·π ■8→r : π·r² 201.062 ■7.95→r : π·r² 198.557 7.95→r:πr^2 MAIN RAD AUTO FUNC 3/30

Nota: quando a entrada contiver um ponto decimal, o resultado aparece automaticamente com o ponto flutuante.

Recuperação de uma entrada precedente

É possível recuperar as entradas precedentes guardadas na área do histórico, mesmo que não apareçam no ecrã. A entrada recuperada substitui o conteúdo da linha de entrada. É possível tornar a executar ou editar a entrada recuperada.

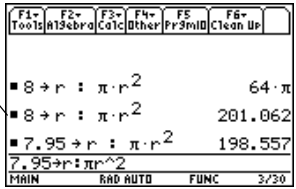
Para recuperar:	Prima:	Efeito:
A última entrada (modificou a linha de entrada)	[2nd] [ENTRY] uma vez	Se a última entrada ainda estiver na linha de entrada, recupera a entrada precedente.
Entradas precedentes	[2nd] [ENTRY] várias vezes	Cada toque recupera a entrada anterior àquela apresentada na linha de entrada.

Nota: também é possível utilizar a função entry para recuperar uma entrada precedente. Consulte **entry( )** no módulo *Referência técnica*.

Por exemplo:

Se a linha de entrada contiver a última entrada, 2nd [ENTRY] recupera-a.

Se a linha de entrada for modificada ou cancelada, 2nd [ENTRY] recupera-a.



Recuperação da última resposta

Cada vez que calcula uma expressão, a TI-89 Titanium / Voyage™ 200 Graphing Calculator guarda o resultado na variável **ans(1)**. Para inserir esta variável na linha de entrada, prima 2nd [ANS].

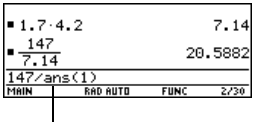
Por exemplo: calcule a área de um jardim rectangular de 1,7 por 4,2 metros. Em seguida, calcule a produção por metro quadrado, sendo a produção total de 147 tomates.

1. Calcule a área.

   1.7 × 4.2 ENTER

2. Calcule a colheita por metro quadrado.

   147 ÷ 2nd [ANS] ENTER



A variável **ans(1)** é inserida e o valor utilizado no cálculo.

De modo analógo à **ans(1)**, **ans(2)**, **ans(3)**, etc., também contêm as respostas precedentes. Por exemplo: **ans(2)** contém a penúltima resposta.

Nota: consulte **ans( )** no módulo *Referência técnica*.

Colagem automática de uma entrada ou resposta da área do histórico

É possível seleccionar qualquer entrada ou resposta da área do histórico e “colar automaticamente” uma duplicação na linha de entrada. Esta função permite inserir uma entrada ou resposta precedente sem digitá-la de novo.

Vantagens da utilização da função de colagem automática

O efeito da função de colagem automática é parecido com a utilização de 2nd [ENTRY] e 2nd [ANS] descritos na secção anterior, embora haja diferenças.

Para as entradas:

Colagem automática permite:	2nd [ENTRY] **permite:**
Inserir qualquer entrada anterior na linha de entrada.	Substituir o conteúdo da linha de entrada por qualquer entrada anterior.

Para as respostas:

Colagem automática permite:	2nd [ANS] **permite:**
Inserir o valor apresentado de qualquer resposta anterior na linha de entrada.	Inserir a variável **ans(1)**, que contém apenas a última resposta. Cada vez que introduzir um cálculo, **ans(1)** é actualizada com a última resposta.

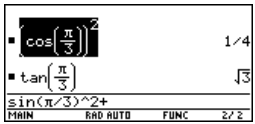
Nota: uma informação pode também ser colada através do menu F1 da barra de ferramentas.

Colar automaticamente uma entrada ou resposta

1. Posicione o cursor no ponto escolhido para inserir a entrada ou a resposta na linha de entrada.
2. Prima ⊕ para deslocar o cursor para cima na área do histórico. Este procedimento realça a última resposta.
3. Utilize ⊕ e ⊖ para realçar a entrada ou a resposta que deseja colar automaticamente.

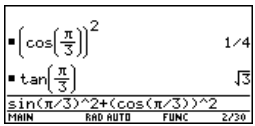
   - ⊕ permite ir da resposta para entrada na área do histórico.
   - Para realçar elementos não contidos no ecrã, utilize ⊕.

   Nota: para anular a operação e voltar à linha de entrada, prima [ESC]. Para ver uma entrada ou resposta cujo comprimento é superior à linha (indicado com ▶ no final da linha), utilize ⊕ e ⊖ ou [2nd] ⊕ e [2nd] ⊖.



4. Prima [ENTER].

   O elemento realçado é inserido na linha de entrada.



Desta forma, cola a entrada ou a resposta completa. Se necessitar apenas uma parte de uma ou de outra, edite a linha de entrada para cancelar as partes não desejadas.

Criação e cálculo de funções definidas pelo utilizador

As funções definidas pelo utilizador permitem economizar muito tempo quando é preciso repetir a mesma expressão várias vezes (com valores diferentes). Estas funções expandem as funções incorporadas da TI-89 Titanium / Voyage™ 200 Graphing Calculator.

Formato de uma função

Os exemplos seguintes contêm funções de um ou dois argumentos definidas pelo utilizador. É possível utilizar um número qualquer de argumentos conforme as necessidades. Nestes exemplos, a definição é composta por uma única expressão (ou enunciado).

cube(x) = x^3
❶ ❷ ❸

xroot(x,y) = $y^{1/x}$
❶ ❷ ❸

❶ Nome da função
❷ Lista de argumentos
❸ Definição

Para definir funções e programas, atribua nomes exclusivos aos argumentos para não os utilizar quando chamar estas funções ou programas posteriormente.

Nota: os nomes das funções seguem as mesmas regras que os nomes das variáveis. Consulte “Guardar e recuperar os valores de variáveis” no módulo *Funcionamento da calculadora*.

Utilize os mesmos argumentos da definição na lista de argumentos. Por exemplo: **cube(n) = x^3** dá resultados imprevistos quando calcular essa função.

Os argumentos (x e y nestes exemplos), são marcadores que contêm os valores transferidos para a função. Não representam as variáveis x e y, excepto se transferir x e y como argumentos durante o cálculo da função.

Criação de uma função definida pelo utilizador

Efectue um dos procedimentos apresentados em seguida.

Método	Descrição
STO▸	Guardar uma expressão num nome de função (incluindo a lista de argumentos).
Comando **Define**	Definir um nome de função (incluindo a lista de argumentos) como uma expressão.
Program Editor	Para mais informações sobre a criação de uma função definida pelo utilizador, consulte o módulo *Programação*.

Criação de uma função de múltiplas expressões

É possível criar também uma função definida por várias expressões. A definição pode incluir muitas das estruturas de decisões e de controlo (**If**, **ElseIf**, **Return**, etc.) utilizadas na programação.

Nota: para mais informações sobre as analogias e diferenças entre as funções e os programas, consulte o módulo *Programação*.

Por exemplo: supondo que deseja criar uma função que some uma série de valores recíprocos baseados num único número inteiro (**n**) introduzido:

$$\frac{1}{n}+\frac{1}{n-1}+\ldots+\frac{1}{1}$$

Durante a criação da definição da função de múltiplas expressões, convém mostrá-la primeiro na forma de bloco.

```
❶ Func
❷ Local temp,i
  If fPart(nn)≠0 or nn≤0
❸   Return "bad argument"
  0→temp
❹ For i,nn,1,-1
    approx(temp+1/i)→temp
  EndFor
❺ Return temp
❶ EndFunc
```

❶ **Func** e **EndFunc** devem estar no início e no fim da função.
❷ As variáveis não incluídas na lista de argumentos devem ser declaradas como

“local”.

❸ Devolve uma mensagem se nn não é um inteiro ou se nn≤0.

❹ Soma os recíprocos.

❺ Devolve a soma.

Se introduzir uma função de múltiplas expressões no ecrã principal da calculadora, tem de a introduzir numa única linha. Utilize o comando **Define** da mesma forma que com uma função de uma única expressão.

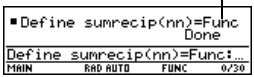
Utilize dois pontos para separar cada expressão.

```
Define sumrecip(nn)=Func:Local temp,i: ... :EndFunc
```

Utilize nomes de argumentos que não serão utilizados quando chamar a função ou o

No ecrã principal da calculadora:

As funções de múltiplas expressões aparecem como **Func**.

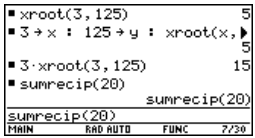


Introduza uma função de múltipla expressão na mesma linha - não esqueça de incluir o sinal de dois pontos.

Nota: é mais fácil criar uma função complexa de múltiplas expressões no Program Editor do que no ecrã principal da calculadora. Consulte o módulo *Programação*.

Calcular uma função

É possível utilizar uma função definida pelo utilizador da mesma forma que qualquer outra função; calcule-as individualmente ou dentro de uma outra expressão.



Apresentar e editar uma função definida

Para:	Operação:
Ver uma lista de todas as funções definidas pelo utilizador	Prima [2nd] [VAR-LINK] para ver o ecrã VAR-LINK. Pode ser necessário utilizar o menu [F2] **View** da barra de ferramentas para especificar a variável **Function**. (Consulte “Gestão da memória e das variáveis”.) – ou – Prima: [CATALOG] [F4] [2nd] [CATALOG] [F4]
Ver uma lista de funções das aplicações Flash	Prima: [CATALOG] [F3] [2nd] [CATALOG] [F3]

Para:	Operação:
Ver a definição de uma função definida pelo utilizador	No ecrã VAR-LINK, realce a função do menu e veja o menu Contents. [2nd] [F6] [F6] – ou – Prima [2nd] [RCL]. no ecrã principal da calculadora. Digite o nome da função, mas não a lista de argumentos (como por exemplo **xroot**), e prima duas vezes [ENTER]. – ou – Em Program Editor, abra a função. (Consulte o módulo *Programação*.)
Editar a definição	No ecrã principal da calculadora, utilize [2nd] [RCL] para ver a definição e efectue as modificações desejadas. Utilize [STO▸] ou **Define** para guardar a nova definição. – ou – Em Program Editor, abra a função, edite-a e guarde as modificações.

Nota: pode ver uma função definida pelo utilizador na caixa de diálogo CATALOG, mas não pode utilizar CATALOG para ver ou editar a própria definição.

Se uma entrada ou resposta for “muito comprida”

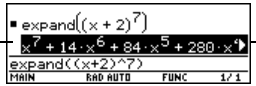
Em alguns casos uma entrada ou resposta pode ser “muito comprida” e/ou “alta” para aparecer completamente na área do histórico. Noutros casos, é possível que a TI-89 Titanium / Voyage™ 200 Graphing Calculator não possa apresentar a resposta por falta de memória disponível.

Se uma entrada ou resposta for maior que uma linha

Posicione o cursor na área do histórico e realce a entrada ou a resposta e utilize a tecla do cursor para a deslocar; por exemplo:

- O exemplo seguinte contém uma resposta demasiado grande para uma linha.

Prima ◀ ou 2nd ◀ para a deslocar para a



Prima ▶ ou 2nd ▶ para a

- O exemplo seguinte contém uma resposta demasiado grande nos sentidos vertical e horizontal para aparecer.

  Nota: este exemplo utiliza a função **randMat** para gerar uma matriz 25 x 25.

❶ Prima ▲ ou ⇧ ▲ para a deslocar para cima

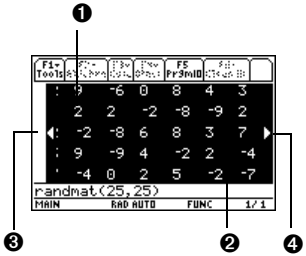
Prima ▲ ou ⎋ ▲ para a deslocar para cima

❷ Prima ⇧ ▼ para a deslocar para baixo

Prima ⎋ ▼ para a deslocar para baixo

❸ Prima ◀ ou 2nd ◀ para a deslocar para a esquerda

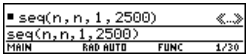
❹ Prima ▶ ou 2nd ▶ para a deslocar para a direita



Memória insuficiente

Quando a memória não for suficiente para apresentar o resultado, aparece o símbolo << ...>> .

Por exemplo:



Nota: este exemplo utiliza a função seq para gerar uma lista sequencial de números inteiros de 1 a 2500.

Se aparecer o símbolo << ...>>, não é possível ver a resposta, mesmo que a realce ou desloque o cursor.

Nestes casos, tente:

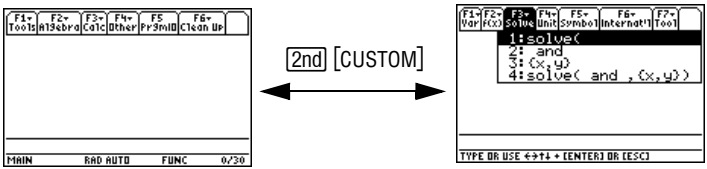
- Libertar alguma memória adicional, excluindo variáveis desnecessárias e/ou aplicações Flash. Utilize [2nd] [VAR-LINK] conforme descrito no módulo *Gestão da memória e das variáveis*.
- Se possível, divida o problema em partes menores que possam ser calculadas e apresentadas com menos memória.

Utilização do menu personalizado

A TI-89 Titanium / Voyage™ 200 Graphing Calculator tem um menu personalizado que pode ser activado e desactivado a qualquer momento. Pode utilizar o menu personalizado predefinido ou criar o seu próprio menu, como descrito no módulo *Programação*.

Activação e desactivação do menu personalizado

Quando activar o menu personalizado, substitui o menu normal da barra de ferramentas. Quando o desactivar, volta ao menu normal. Por exemplo, no menu normal da barra de ferramentas do ecrã principal da calculadora, prima [2nd] [CATALOG] para activar e desactivar o menu personalizado.



Menu da barra de ferramentas normal do ecrã principal da calculadora

Menu personalizado

Nota: também é possível activar ou desactivar o menu personalizado, introduzindo **CustmOn** ou **CustmOff** na linha de entrada e premindo [ENTER].

A menos que o menu tenha sido modificado, aparece o menu personalizado predefinido.

Menu	Função
[F1] **Var**	Nomes das variáveis comuns.
[F2] **f(x)**	Nomes de funções tais como f(x), g(x) e f(x,y).
[F3] **Solve**	Itens relacionados à solução de equações.
[F4] **Unit**	Unidades comuns tais como _m, _ft e _l.

Menu	Função
F5 **Symbol**	Símbolos tais como #, ? e ~.
International 2nd F6 F6	Caracteres habitualmente acentuados tais como é e ê.
Tool 2nd F7 F7	**ClrHome**, **NewProb**, e **CustmOff**.

Nota: um menu personalizado permite aceder aos itens mais utilizados. O módulo *Programação* mostra como criar menus personalizados para os itens utilizados com mais frequência.

Recuperação do menu personalizado predefinido

Se aparecer um menu personalizado diferente da predefinição e se for necessário restaurar a predefinição:

1. No ecrã principal da calculadora, utilize [2nd] [CATALOG] para desactivar o menu personalizado e ver o menu da barra de ferramentas normal do ecrã principal da calculadora.

2. Veja o menu da barra de ferramentas **Clean Up** e seleccione **3:Restore custom default**.

[2nd] [F6]

[F6]

Este procedimento cola os comandos utilizados para criar o menu predefinido na linha de entrada.

Nota: o menu personalizado anterior é apagado. Se criar esse menu num programa, pode recriá-lo mais tarde quando executar o programa novamente.

3. Prima [ENTER] para executar os comandos e recuperar a predefinição.

Localização da versão do software e o número de identificação

Em algumas situações, pode ser necessário obter informações sobre a sua TI-89 Titanium / Voyage™ 200 Graphing Calculator, em especial, a versão do software e o número ID da unidade.

Visualização do ecrã “About”

1. No ecrã principal da calculadora ou no ambiente de trabalho das Apps, prima [F1] e, em seguida, seleccione **A:About**.

   O ecrã ficará diferente do ecrã apresentado à direita.

2. Prima [ENTER] ou [ESC] para fechar o ecrã.

Quando é necessária esta informação?

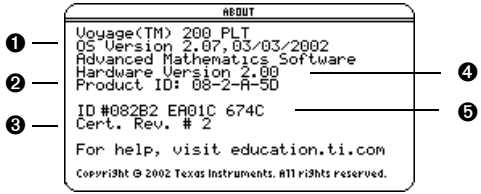
As informações sobre o ecrã About destinam-se a situações como:

- Para obter um novo software ou um software actualizado ou aplicações Flash para TI-89 Titanium / Voyage™ 200, é necessário fornecer a versão de software actual e/ou o número ID da unidade.
- Se tiver dificuldades com a TI-89 Titanium / Voyage™ 200 e for necessário entrar em contacto com suporte técnico, conhecer a versão do software pode tornar a resolução do problema mais fácil.

O ecrã About mostra as seguintes informações sobre a Voyage™ 200:

- Versão do hardware
- Versão do SO (Advanced Mathematics Software)
- Identificador do produto (Product ID)

- ID da unidade
- Número de revisão do certificado de Apps (Cert. Rev.)



❶ Versão do SO
❷ Identificador do produto
❸ Número de revisão do certificado das Apps
❹ Versão do hardware
❺ ID da unidade (necessário para obter certificados para a instalação de Apps adquiridas)

O ecrã apresentado será diferente do ecrã acima.

Manipulação simbólica

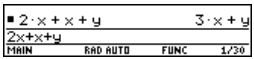
Utilização de variáveis definidas e indefinidas

Se realizar operações algébricas ou de cálculo, é importante entender o efeito da utilização de variáveis definidas e indefinidas. Caso contrário, pode ser obtido um número como resultado em vez da expressão algébrica prevista.

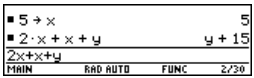
Como as variáveis definidas e indefinidas são tratadas

Se introduzir uma expressão com uma variável, a TI-89 Titanium / Voyage™ 200 Graphing Calculator trata a variável de uma das duas formas seguintes.

- Se a variável for indefinida, é tratada como um símbolo algébrico.

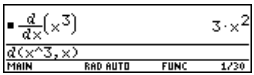


- Se a variável for definida (mesmo se definida como 0), o valor substitui a variável.



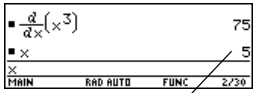
Para entender a sua importância, suponha que quer calcular a primeira derivada de x^3 em relação a x.

- Se x estiver indefinido, o resultado aparece da forma prevista.



- Se x estiver definido, o resultado pode aparecer de uma forma diferente da prevista.

  Nota: se configurar uma variável, recomendamos que utilize mais do que um carácter no nome. Deixe os nomes de um carácter indefinidos para cálculos simbólicos.



A menos que soubesse que 5 havia sido guardado previamente em x, a resposta 75 poderia conduzir ao erro.

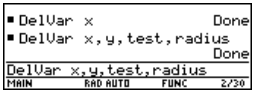
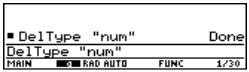
Como determinar se uma variável é indefinida

Método:	Exemplo:
Introduza o nome da variável.	Se definido, aparece o valor da variável. ■ x 5 ■ y y y MAIN RAD AUTO FUNC 2/30 Se não definido, aparece o nome da variável.
Utilize a função **isVar()**.	Se definido, aparece “true”. ■ isVar(x) true ■ isVar(y) false isVar(y) MAIN RAD AUTO FUNC 2/30 Se não definido, aparece “false”.
Utilize a função **getType**.	Se definido, aparece o tipo da variável. ■ getType(x) "NUM" ■ getType(y) "NONE" getType(y) MAIN RAD AUTO FUNC 2/30 Se não definido, aparece “NONE”.

Nota: utilize [2nd] [VAR-LINK] para ver a lista de variáveis definidas, como descrito em *Gestão da memória e das variáveis*.

Exclusão de uma variável definida

Pode-se “indefinir” uma variável definida através da sua exclusão.

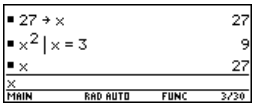
Para excluir:	Operação:
Uma ou mais variáveis especificadas	Utilize a função **DelVar**.  Pode-se também excluir variáveis com o ecrã VAR-LINK (2nd [VAR-LINK]) como descrito em *Gestão da memória e das variáveis.*
Todas as variáveis de um tipo específico **Nota:** a função Deltype elimina todas as variáveis do tipo especificado em todas as pastas.	Utilize a função **Deltype**. 

Para excluir:	Operação:
Todas as variáveis com um carácter (a – z) na pasta actual **Nota:** para mais informações sobre pastas, consulte o módulo *Ecrã principal da calculadora*.	No menu Clean Up do ecrã principal, seleccione **1:Clear a-z**. Aparece um pedido de confirmação. Prima [ENTER] para confirmar a exclusão. Clear a-z Clear 1-character variables a-z in current folder? Enter=YES ESC=CANCEL

Reutilização temporária de uma variável existente

Se digitar o operador "*with*" (|), é possível:

- Omitir temporariamente um valor definido de variável.



- Definir temporariamente um valor para uma variável indefinida.

Nota: para mais informações sobre o operador |, consulte Introdução do operador "*With*".

Para digitar o operador "*with*" (|), prima:

[|]

[2nd] [|]

Utilização dos modos Exact, Approximate e Auto

As definições dos modos Exact/Approx, descritos brevemente no módulo *Funcionamento da calculadora*, influem directamente na precisão e na exactidão com que a TI-89 Titanium / Voyage™ 200 Graphing Calculator calcula um resultado. Esta secção descreve estas configurações, que estão relacionadas com a manipulação simbólica.

Modo EXACT

Quando Exact/Approx = EXACT, a TI-89 Titanium / Voyage™ 200 utiliza a aritmética racional exacta até 614 dígitos no numerador e 614 dígitos no denominador. O modo EXACT:

- Simplifica números irracionais, tanto quanto possível, sem aproximá-los. Por exemplo, $\sqrt{12}$ é transformado em $2\sqrt{3}$ e **ln(1000)** é transformado em **3 ln(10)**.
- Converte números de decimais em números racionais. Por exemplo, 0.25 é transformado em 1/4.

As funções **solve**, **cSolve**, **zeros**, **cZeros**, **factor**, ∫, **fMin** e **fMax** utilizam apenas algoritmos simbólicos exactos. Estas funções não calculam soluções aproximadas no modo EXACT.

- Algumas equações, como $\mathbf{2^{-x} = x}$, possuem soluções que não podem ser representadas finitamente em termos de funções e operadores na TI-89 Titanium / Voyage™ 200.

- Com este tipo de equação, EXACT não calculará soluções aproximadas. Por exemplo, $\mathbf{2^{-x} = x}$ tem a solução aproximada $\mathbf{x \approx 0.641186}$, mas esta não aparece no modo EXACT.

Vantagens	Desvantagens
Os resultados são exactos.	À medida que utiliza números racionais complexos e constantes irracionais, os cálculos podem: • Utilizar mais memória e consumi-la antes de obter a solução. • Levar mais tempo a calcular. • Produzir resultados grandes que são mais difíceis de ser compreendidos do que um número decimal.

Modo APPROXIMATE

Quando Exact/Approx = APPROXIMATE, a TI-89 Titanium / Voyage™ 200 converte os números racionais e constantes irracionais em decimais. Entretanto, há excepções:

- Certas funções incorporadas nas quais um dos seus argumentos tem de ser um número inteiro converterão este número num número inteiro, se possível. Por exemplo: d**(y(x), x, 2.0)** é transformado em d**(y(x), x, 2).**
- Os expoentes com números decimais inteiros são convertidos em números inteiros. Por exemplo: $x^{2.0}$ é transformado em x^2 mesmo no modo APPROXIMATE.

As funções como **solve** e ∫ (integral) podem utilizar o método simbólico exacto e o método numérico aproximado. No modo APPROXIMATE, estas funções ignoram todos ou alguns métodos simbólicos exactos.

Vantagens	Desvantagens
Se não forem necessários resultados exactos, este modo poderá economizar tempo e/ou utilizar menos memória do que o modo EXACT. Resultados aproximados são, às vezes, mais compactos e mais compreensíveis do que resultados exactos.	Os resultados com funções ou variáveis indefinidas frequentemente mostram um cancelamento incompleto. Por exemplo, um coeficiente que deveria ser **0** poderia ser apresentado com uma ordem de grandeza pequena tal como **1.23457E-11**.
Se não quiser utilizar cálculos simbólicos, os resultados aproximados são parecidos com os resultados das calculadoras numéricas tradicionais.	As operações simbólicas tais como limites e integrais têm menos probabilidade de fornecer resultados satisfatórios no modo APPROXIMATE. Algumas vezes, os resultados aproximados são menos compactos e menos compreensíveis do que resultados exactos. Por exemplo, pode ser preferível ver **1/7** em vez de **.142857**.

Modo AUTO

Quando Exact/Approx = AUTO, a TI-89 Titanium / Voyage™ 200 utiliza a aritmética racional exacta sempre que todos os operandos forem números racionais. Caso

contrário, utiliza a aritmética decimal depois de converter todos os operandos racionais em decimais. Por outras palavras, o decimal é “preponderante”. Por exemplo:

1/2 − **1/3** é transformado em **1/6**
mas
0.5 − **1/3** é transformado em **.16666666666667**

Esta preponderância do decimal não ocorre em variáveis indefinidas ou elementos de listas ou matrizes. Por exemplo:

(1/2 − **1/3) x + (0.5** − **1/3)** y é transformado em **x/6 + .16666666666667 y**
e
{1/2 − **1/3, 0.5** − **1/3}** é transformado em **{1/6, .16666666666667}**

No modo AUTO, as funções como **solve** encontram tantas soluções quanto possível com exactidão e, em seguida, utilizam métodos de aproximação numérica para encontrar soluções adicionais, se necessário. Da mesma forma, a ∫ (integral) utiliza métodos numéricos aproximados, se apropriado, onde os métodos de exactidão simbólica falham.

Vantagens	Desvantagens
Pode obter resultados exactos quando possível e resultados numéricos aproximados quando os resultados exactos forem impraticáveis.	Se estiver interessado apenas em resultados exactos, pode perder algum tempo a procurar resultados aproximados.
Pode controlar frequentemente o formato de um resultado, introduzindo alguns coeficientes como números racionais ou como números de decimais.	Se estiver interessado apenas em resultados aproximados, pode perder algum tempo a procurar resultados exactos. Além disso, pode consumir a memória com a procura desses resultados exactos.

Simplificação automática

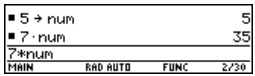
Quando digitar uma expressão na linha de entrada e premir [ENTER], a TI-89 Titanium / Voyage™ 200 Graphing Calculator simplifica automaticamente a expressão de acordo com as regras de simplificação predefinidas.

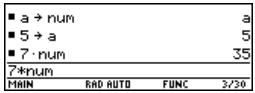
Regras de simplificação predefinidas

Todas as regras seguintes são aplicadas automaticamente. Os resultados intermédios não são apresentados.

- Se uma variável tiver um valor definido, este valor substitui a variável.

  Se a variável estiver definida em função de outra variável, a variável é substituída pelo valor de “nível mais baixo” (denominado busca infinita).





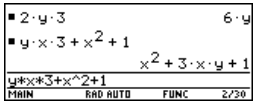
  A simplificação predefinida não modifica as variáveis que utilizam nomes de caminho para indicar uma pasta. Por exemplo: *x+class\x* não é simplificado para *2x*.

Nota: para mais informações sobre pastas, consulte o módulo *Ecrã principal da calculadora*.

- Para funções:
  - Os argumentos são simplificados. (Algumas funções incorporadas atrasam a simplificação de alguns dos argumentos.)

- Se a função for integrada ou definida pelo utilizador, a definição da função é aplicada aos argumentos simplificados. Em seguida, a forma funcional é substituída por este resultado. .

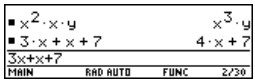
- As subexpressões numéricas são combinadas.
- Os produtos e adições são ordenadas.



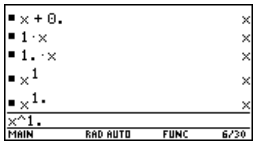
Os produtos e adições que incluem variáveis indefinidas são ordenadas alfabeticamente de acordo com a primeira letra do nome da variável.

- As variáveis indefinidas de *r* a *z* são consideradas variáveis verdadeiras e são colocadas por ordem alfabética no início de uma adição.
- As variáveis indefinidas de *a* a *q* são consideradas constantes e são colocadas por ordem alfabética no final de uma adição (antes dos números).

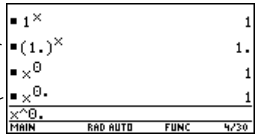
- Os factores e termos semelhantes são agrupados.



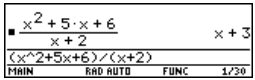
- Procura as igualdades com zeros e uns.



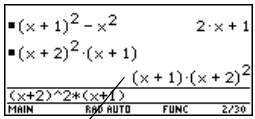
Este número decimal faz com que os resultados numéricos apareçam como decimais.
Se introduzir como expoente um número inteiro na forma decimal, será tratado como um inteiro (e não produzirá um resultado como decimal).



- Os máximos divisores comuns do polinômio são simplificados.

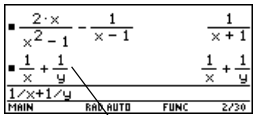


- Os polinómios são calculados, a menos que não seja possível uma simplificação.



Sem cancelamento

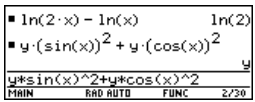
- São formados denominadores comuns, a menos que não seja possível uma simplificação.



Sem cancelamento

- São procuradas as igualdades de funções. Por exemplo:

  ln(2x) = ln(2) + ln(x)
  e
  sin(x)^2 + cos(x)^2 = 1



Quanto demora o processo de simplificação?

Dependendo da complexidade de uma entrada, resultado ou expressão intermediária, pode levar bastante tempo para fatorar uma expressão e cancelar os divisores comuns como é necessário para a simplificação.

Para interromper um processo de simplificação demorado, prima [ON]. Em seguida, pode experimentar simplificar apenas uma parte da expressão (cole automaticamente a expressão inteira na linha de entrada e, em seguida, remova as partes não desejadas.)

Simplificação diferida para algumas funções incorporadas

Normalmente, as variáveis são simplificadas automaticamente no nível mais baixo possível antes de serem passadas para uma função. Em certas funções, entretanto, a simplificação completa é diferida até que a função seja efetuada.

Funções que utilizam a simplificação diferida

As funções que utilizam a simplificação diferida têm um argumento obrigatório var que calcula a função em relação a uma variável. Estas funções possuem pelo menos dois argumentos com a seguinte forma geral:

function(*expressão*, *var* [, …]**)**

Nota: nem todas as funções que utilizam um argumento var utilizam a simplificação diferida.

Por exemplo: **solve(x^2−x−2=0,x)**
d(x^2−x−2,x)
∫(x^2−x−2,x)
limit(x^2−x−2,x,5)

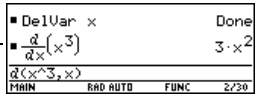
Para uma função que utiliza a simplificação diferida:

1. A variável *var* é simplificada até alcançar o nível mais baixo no qual permanece uma variável (mesmo que possa ser implificada para um valor invariável).
2. A função é calculada com a variável.
3. Se for possível simplificar mais *var*, o valor é substituído no resultado.

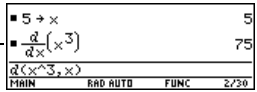
Nota: pode querer ou não definir um valor numérico para *var*, dependendo da situação.

Por exemplo:

x não pode ser simplificado.

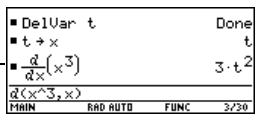


x não é simplificado. A função utiliza x^3 e, em seguida, substitui x por 5.

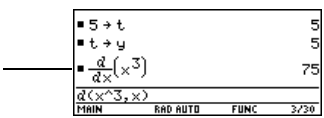


Nota: o exemplo à direita encontra a derivada de x^3 em x=5. Se x^3 fosse inicialmente simplificado em 75, seria calculada a derivada de 75, resultado diferente daquele desejado.

x é simplificado em relação a t. A função utiliza t^3.



x é simplificado em relação a t. A função utiliza t^3 e, em seguida, substitui t por 5.



Substituição de valores e definição de restrições

O operador "*with*" (|) permite substituir valores temporariamente numa expressão ou especificar restrições de domínio.

Introdução do operador "*With*"

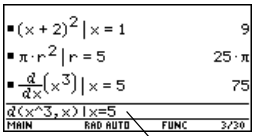
Para digitar o operador "*with*" (|), prima:

[|]

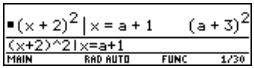
[2nd] [|]

Substituição de uma variável

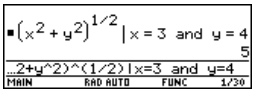
Todas as ocorrências de uma variável especificada podem ser substituídas por um valor numérico ou uma expressão.



Primeira derivada de x^{-3} para x = 5

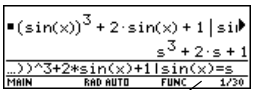


Para substituir diversas variáveis ao mesmo tempo, utilize o operador booleano *and*.



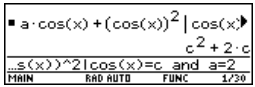
Substituição de uma expressão simples

Todas as ocorrências de uma expressão simples podem ser substituídas por uma variável, um valor numérico ou outra expressão.



A substituição de sin(x) por s mostra que a expressão é um polinómio em termos de sin(x).

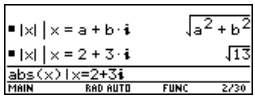
Se substituir um termo utilizado frequentemente (ou um termo grande), os resultados podem aparecer de uma forma mais compacta.



Nota: acos(x) é diferente de **a∗cos(x)**.

Substituição de valores complexos

Os valores complexos podem ser substituídos exactamente como seria feito para outros valores.



Todas as variáveis indefinidas são tratadas como números reais em cálculos simbólicos. Para realizar análises simbólicas complexas, é preciso configurar uma variável complexa. Por exemplo:

x+y*i*→z

Pode utilizar *z* como uma variável complexa. É também possível utilizar *z_*. Para mais informações, consulte o tópico _ (sublinhado) no módulo *Referência técnica*.

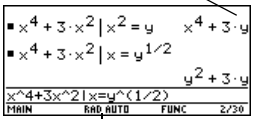
Nota:

- para obter uma descrição sobre números complexos, consulte o módulo *Referência técnica*.
- para obter o número complexo i, prima [2nd] [i]. Não digite simplesmente a letra **i** no teclado.

Limitações das substituições

- A substituição só ocorre quando há uma correspondência exacta para a substituição.

Apenas x^2 foi substituído, não x^4.



Defina a substituição em termos mais simples para obter uma substituição

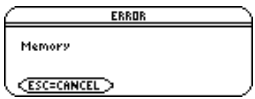
- Pode ocorrer uma recursividade infinita quando definir uma variável de substituição em função dela mesma.

Substitui sin(x+1), sin(x+1+1), sin(x+1+1+1), etc.

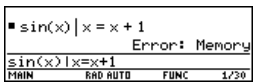
sin(x)|x=x+1

Quando introduzir uma substituição que causa recursividade infinita:

- Aparece uma mensagem de erro.

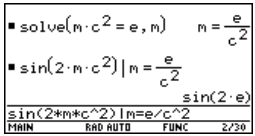


- Se premir ESC, aparece uma mensagem de erro na área do histórico.



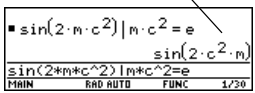
- Internamente, uma expressão é ordenada de acordo com as regras de simplificação automática. Consequentemente, os produtos e as adições podem não estar na mesma ordem em que foram introduzidos.

  - Em regra geral, a substituição deve ser para uma única variável.



  - A substituição em expressões mais genéricas **(m • c^2=e** ou **c^2 • m=e)** pode não funcionar como esperado.

Sem correspondência para substituição



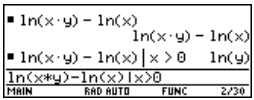
Nota: utilize a função **solve** para ajudar a determinar a substituição de uma única variável.

Especificar as restrições de domínio

Muitas igualdades e transformações são válidas apenas num domínio particular. Por exemplo:

ln(x y) = ln(x) + ln(y)	apenas se **x** e **y** forem não negativos
sin^{-1}(sin(θ)) = θ	apenas se θ ≥ ⁻π**/2** e θ ≤ π**/2** radianos

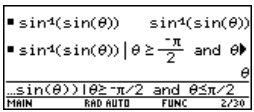
Utilize o operador “*with*” para especificar a restrição do domínio.



Como ln(x∗y) = ln(x) + ln(y) nem sempre é válido, os logaritmos não são combinados.

Com a restrição, a igualdade é válida e a expressão é simplificada.

Nota: introduza **ln(x∗y)** em vez de **ln(xy)**; caso contrário, xy será interpretada como uma única variável chamada xy.



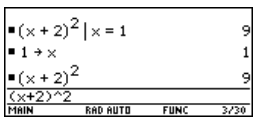
Visto que $\sin^{-1}(\sin(\theta)) = \theta$ não sempre é válido, a expressão não é simplificada.

Com a restrição, a expressão pode ser simplificada.

Nota: para ≥ ou ≤, digite [◆] [>] ou [◆] [<]. Pode utilizar também [2nd] [MATH] **8** ou [2nd] [UNITS] **2** para seleccionar num menu.

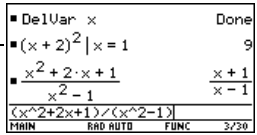
Substituições em confronto com definições de variáveis

Em muitos casos, a definição da variável produz o mesmo efeito de uma substituição.

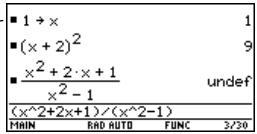


Entretanto, a substituição é preferível na maioria dos casos porque impede que a variável, definida a princípio apenas para o cálculo actual, afecte acidentalmente cálculos posteriores.

A substituição de x=1 não afecta o próximo cálculo.



O armazenamento de 1→x afecta os cálculos subsequentes.



Atenção: depois de definir a variável x, todos os cálculos que incluam esta variável são afectados (até que seja excluída).

Descrição do menu Algebra

Pode utilizar o menu F2 **Algebra** da barra de ferramentas para seleccionar as funções algébricas mais utilizadas.

Menu Algebra

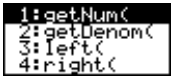
No ecrã principal, prima [F2] para ver:

Este menu está disponível também a partir do menu MATH. Prima [2nd] [MATH] e, em seguida, seleccione **9:Algebra**.

Nota: para obter uma descrição completa de cada função e da sua sintaxe, consulte o módulo *Referência técnica*.

Opção de menu	Descrição
solve	Calcula uma expressão para uma variável especificada. Esta opção devolve apenas soluções reais, independente da configuração do modo **Complex Format**. Mostra respostas com soluções de ligação "*and*" e "*or*".(Para soluções complexas, seleccione **A:Complex** a partir do menu Algebra.)
factor	Decomponha em factores uma expressão em relação a todas as suas variáveis ou apenas a uma variável especificada.
expand	Desenvolve uma expressão em relação a todas as suas variáveis ou apenas a uma variável especificada.
zeros	Determina os valores de uma variável específica que igualam uma expressão a zero. Aparece numa lista.

Opção de menu	Descrição
approx	Calcula uma expressão aritmética com decimal, sempre que possível. Equivale a utilizar [MODE] para configurar Exact/Approx = APPROXIMATE (ou empregar [◆] [ENTER] para calcular uma expressão).
comDenom	Calcula um denominador comum para todos os termos numa expressão e transforma a expressão numa fracção reduzida de um numerador e denominador.
propFrac	Calcula a fracção própria de uma expressão.
nSolve	Calcula uma única solução decimal para uma equação (diferente de **solve**, que pode apresentar algumas soluções na forma simbólica ou racional).
Trig	Exibe o submenu: 1:tExpand(2:tCollect(
	tExpand — Desenvolve expressões trigonométricas com somas angulares e múltiplos ângulos.
	tCollect — Agrupa os produtos de potências inteiras de funções trigonométricas em somas angulares e ângulos múltiplos. **tCollect** é o oposto de **tExpand**.
Complex	Exibe o submenu: 1:cSolve(2:cFactor(3:cZeros(São idênticos a **solve**, **factor** e **zeros**; mas calculam também resultados complexos.

Opção de menu	Descrição
Extract	Mostra o submenu: 
	getNum — Aplica **comDenom** e devolve o numerador resultante.
	getDenom — Aplica **comDenom** e devolve o denominador resultante.
	left — Devolve o lado esquerdo de uma equação ou de uma inequação.
	right — Devolve o lado direito de uma equação ou de uma inequação.

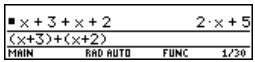
Nota: as funções **left** e **right** são utilizadas também para obter um número específico de elementos ou caracteres do lado esquerdo ou direito de uma lista ou cadeia de caracteres.

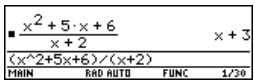
Operações algébricas comuns

Esta secção dá exemplos de algumas das funções disponíveis no menu F2 **Algebra** da barra de ferramentas. Para obter informações completas sobre quaisquer funções, consulte o módulo *Referência técnica*. Algumas operações algébricas não requerem uma função especial.

Adição ou divisão de polinómios

Pode adicionar ou dividir polinómios sem utilizar uma função especial.





Decomposição em factores e desenvolvimento de polinómios

Utilize as funções **factor** (F2 **2**) e **expand** (F2 **3**).

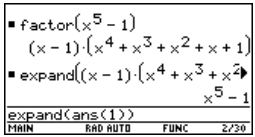
factor(*expressão* [,*var*]**)**

└para decompor em factores em relação a uma

expand(*expressão* [,*var*]**)**

└para desenvolvimento parcial em relação a uma variável

Decomponha em factores $\mathbf{x}^5 - \mathbf{1}$. Em seguida, desenvolva o resultado.

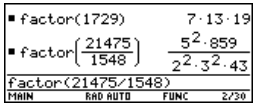


Não se esqueça de que **factor** e **expand** realizam operações opostas.

Obtenção de factores primos de um número

As possibilidades oferecidas pela função **factor** (F2 **2**) não se limitam simplesmente a decompor em factores um polinómio algébrico.

Pode calcular os factores primos de um número racional (um número inteiro ou uma fracção de números inteiros).

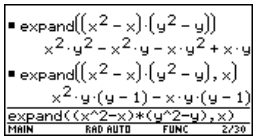


Obtenção de expansões parciais

Com o valor opcional var da função **expand** (F2 **3**), é possível obter um desenvolvimento parcial que agrupa as potências similares de uma variável.

Obtenha um desenvolvimento completo de **(x^2−x) (y^2−y)** em relação a todas as variáveis.

Em seguida, obtenha um desenvolvimento parcial em relação a x.

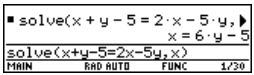


Resolução de uma equação

Utilize a função **solve** ([F2] **1**) para resolver uma equação em relação a uma variável especificada.

solve(*equação*, *var***)**

Calcule **x + y − 5 = 2x − 5y** para x.



Não se esqueça de que solve mostra apenas o resultado final.

Para ver os resultados intermédios, a equação deve ser resolvida manualmente, passo a passo.

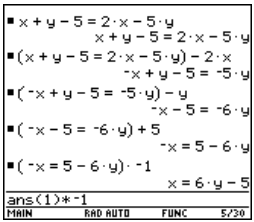
x [+] **y** [−] **5** [=] **2x** [−] **5y**

[−] **2 x**

[−] **y**

[+] **4**

[×] [(-)] **1**



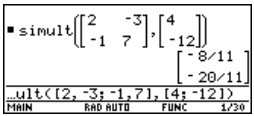
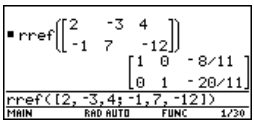
Nota: uma operação tal como [−] **2** [×] subtrai 2x de ambos os lados.

Resolução de um sistema de equações lineares

Considere o sistema de duas equações com duas variáveis desconhecidas:

2x − 3y = 4
⁻x + 7y = ⁻12

Para resolver este sistema de equações, utilize um dos métodos seguintes.

Método	**Exemplo**
Utilize a função **solve** para a solução a um passo.	**solve(**2x−3y=4 and ⁻x+7y=⁻12,{x,y}**)**
Utilize a função **solve** com substituição (|) para a manipulação passo a passo.	Consulte ”Manipulação Simbólica” no módulo *Início rápido da matemática*, que foi calculado para **x = ⁻8/11** e **y = ⁻20/11**.
Utilize a função **simult** com uma matriz.	Introduza os coeficientes como uma matriz e os resultados como uma matriz de coluna constante. 
Utilize a função **rref** com uma matriz.	Introduza os coeficientes como uma matriz aumentada. 

Nota: as funções de matrizes **simult** e **rref** não estão disponíveis no menu F2 **Algebra**. Utilize 2nd [MATH] **4** ou **Catalog**.

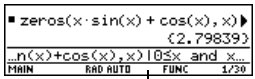
Obtenção dos zeros de uma expressão

Utilize a função **zeros** ([F2] **4**).

zeros(*expressão*, *var***)**

Utilize a expressão **x∗sin(x) + cos(x)**.

Encontre as raízes da equação no intervalo **0** ≤ **x** e **x** ≤ **3**.



Utilize o operador "with" para designar o intervalo.

Nota: para ≥ ou ≤, digite [◆] [>] ou [◆] [<]. Pode também utilizar [2nd] [MATH] **8** ou [2nd] [UNITS] **2** para seleccionar a partir de um menu.

Obtenção de fracções próprias e denominadores comuns

Utilize as funções **propFrac** ([F2] **7**) e **comDenom** ([F2] **6**).

propFrac(*expressão racional* [,*var*]**)**

para fracções próprias com relação a uma variável

comDenom(*expressão* [,*var*]**)**

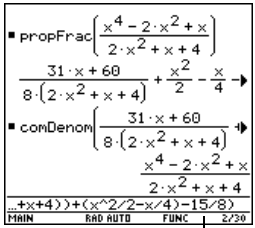
para denominadores comuns que agrupam potências similares desta variável

Encontre uma fracção própria para a expressão **(x^4-2x^2+x) / ($2x^2+x+4$)**.

Em seguida, transforme a resposta numa fracção com numerador e denominador totalmente desenvolvidos.

Não se esqueça de que **propFrac** e **comDenom** realizam operações opostas.

Nota: pode utilizar **comDenom** com uma expressão, lista ou matriz.



Se fizer este exemplo na sua TI-89 Titanium / Voyage™ 200, a função propFrac ultrapassa o topo do ecrã.

Neste exemplo:

- $\frac{31x+60}{8}$ é o resto de x^4-2x^2+x dividido por $2x^2+x+4$.
- $\frac{x^2}{2}-\frac{x}{4}-15/8$ é o quociente.

Descrição do menu Calc

Pode utilizar o menu F3 **Calc** da barra de ferramentas para seleccionar as funções de cálculo mais utilizadas.

Menu Calc

No ecrã principal, prima F3 para ver:

Este menu está disponível também no menu MATH. Prima 2nd [MATH] e, em seguida, seleccione **A:Calculus**.

Nota: para obter uma descrição completa de cada função e sua sintaxe, consulte o módulo *Referência técnica*.

Opção de menu	Descrição
d **differentiate**	Deriva uma expressão em relação a uma variável especificada.
∫ **integrate**	Integra uma expressão em relação a uma variável especificada.
Limit	Calcula o limite de uma expressão em relação a uma variável especificada.
Σ **sum**	Avalia os valores discretos das variáveis de uma expressão dentro de um intervalo e calcula a soma.

Opção de menu	Descrição
Π **product**	Avalia os valores discretos das variáveis de uma expressão dentro de um intervalo e calcula o produto.
fMin	Encontra os possíveis valores de uma variável especificada que minimizam uma expressão.
fMax	Encontra os possíveis valores de uma variável especificada que maximizam uma expressão.
arcLen	Devolve o comprimento do arco de uma expressão em relação a uma variável especificada.
taylor	Calcula uma aproximação de polinómio de Taylor para uma expressão em relação a uma variável especificada.
nDeriv	Calcula a derivada numérica de uma expressão em relação a uma variável especificada.
nInt	Calcula um integral como número decimal com quadratura (uma aproximação que utiliza somas ponderadas dos integrandos).
deSolve	Resolve simbolicamente muitas equações diferenciais de 1ª e 2ª ordem, com ou sem as condições iniciais.
impDif	Calcula derivadas implícitas para equações em duas variáveis em que uma variável é definida implicitamente nos termos da outra.

Nota: o símbolo d (de diferencial) é um símbolo especial. Não é o mesmo que digitar **D** no teclado. Utilize [F3] **1** ou [2nd] [d].

Operações comuns de cálculo

Esta secção fornece exemplos de algumas das funções disponíveis do menu F3 **Calc** da barra de ferramentas. Para obter informações completas sobre uma função de cálculo, consulte no módulo *Referência técnica*.

Integração e derivação

Utilize as funções ∫ **integrate** (F3 **2**) e *d* **differentiate** (F3 **1**).

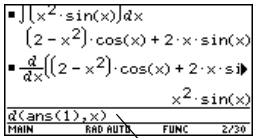
∫ (*expressão*, *var* [,*inferior*] [,*superior*])

permite especificar limites ou uma constante de integração

d (*expressão*, *var* [,*ordem*])

Integre **x²∗sin(x)** em relação a x.

Derive o resultado em relação a x.



Para obter *d*, utilize F3 **1** ou 2nd [*d*]. Não digite simplesmente **D**

Nota: apenas expressões podem ser integradas; porém, pode derivar expressões, listas ou matrizes.

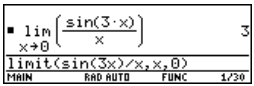
Cálculo de limite

Utilize a função **limit** (F3 **3**).

limit(*expressão*, *var*, *ponto* [,*direção*]**)**

negativo = pela esquerda
positivo = pela direita
omitido ou 0 = ambos

Calcule o limite de **sin(3x) / x** quando x tende a 0.



Nota: pode calcular o limite de uma expressão, lista, ou matriz.

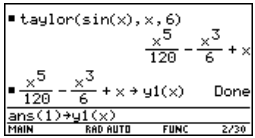
Cálculo de polinómio de Taylor

Utilize a função **taylor** (F3 **9**).

taylor(*expressão*, *var*, *ordem* [,*ponto*]**)**

└─ se omitido, o ponto de expansão é 0

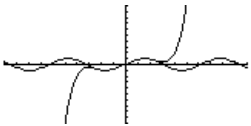
Calcule um polinómio de Taylor de 6ª ordem para **sin(x)** com relação a x.



Guarde o resultado como uma função definida pelo utilizador denominada **y1(x)**.

Em seguida, represente graficamente a função **sin(x)** e o polinómio de Taylor.

Graph sin(x):Graph y1(x)



Importante: utilizar um formato diferente (radiano ou grau) pode afectar a apresentação do cálculo.

Funções definidas pelo utilizador e manipulação simbólica

Pode utilizar uma função definida pelo utilizador como argumento das funções algébricas e de cálculo incorporadas na TI-89 Titanium / Voyage™ 200 Graphing Calculator.

Para mais informações sobre a criação de uma função definida pelo utilizador

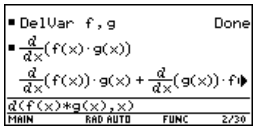
Consulte:

- “Criação e cálculo das funções definidas pelo utilizador” no módulo *Ecrã principal da calculadora*.
- “Representação gráfica de uma função definida no ecrã principal” e “Representação gráfica de uma função definida por intervalos” no módulo *Ecrã principal da calculadora*.
- “Descrição da introdução de uma função” no módulo *Programação*.

Funções não definidas

É possível utilizar funções como **f(x)**, **g(t)**, **r(θ)**, etc., que não tenham sido definidas. Estas funções “não definidas” produzem resultados simbólicos. Por exemplo:

Utilize **DelVar** para garantir que **f(x)** e **g(x)** não estão definidas.



Em seguida, encontre a derivada de **f(x)∗g(x)** em relação a x.

Nota: para seleccionar **d** do menu da barra de ferramentas Calc, prima F3 **1** (ou prima 2nd [d] no teclado).

Funções simples

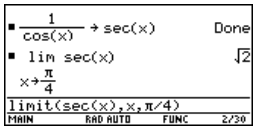
Pode utilizar funções definidas pelo utilizador compostas por uma única expressão. Por exemplo:

- Utilize [STO▸] para criar uma função secante definida pelo utilizador, onde:

$$\sec(x) = \frac{1}{\cos(x)}$$

Em seguida, calcule o limite de **sec(x)** quando x tende a π/4.

Nota: para seleccionar **limit** a partir do menu da barra de ferramentas Calc, prima [F3] **3**.



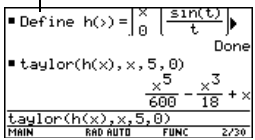
- Utilize **Define** para criar uma função **h(x)** definida pelo utilizador, onde:

$$h(x) = \int_0^x \frac{\sin(t)}{t}$$

Em seguida, calcule um polinómio de Taylor de 5ª ordem para **h(x)** em relação a x.

Nota: para seleccionar ∫ do menu da barra de ferramentas Calc, prima [F3] **2** (ou prima [2nd] [∫] no teclado). Para seleccionar taylor, prima [F3] **9**.

Defina
h(x)= ∫(sin(t)/t,t,0,x).

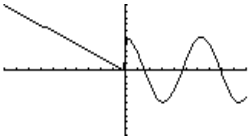


Confronto entre funções de várias expressões e funções simples

As funções com várias expressões definidas pelo utilizador só devem ser utilizadas como um argumento para funções numéricas (tais como **nDeriv** e **nInt**).

Em alguns casos, pode ser possível criar uma função simples equivalente. Por exemplo, considere uma função definida por intervalos composta de duas partes.

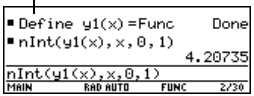
Quando:	Utilize a expressão:
$x < 0$	$-x$
$x \geq 0$	$5\cos(x)$



- Crie uma função definida pelo utilizador com várias expressões com a forma:

```
Func
  If x<0 Then
  Return -x
Else
  Return 5cos(x)
EndIf
EndFunc
```

Define y1(x)=Func:If x<0 Then: ...



Em seguida, integre numericamente **y1(x)** em relação a x.

Nota: para seleccionar **nInt** a partir do menu Calc da barra de ferramentas, prima F3 **B:nInt**.

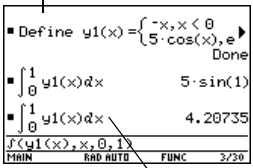
- Crie uma função simples equivalente definida pelo utilizador.

  Utilize a função incorporada **when** da TI-89 Titanium / Voyage™ 200.

  Em seguida, integre **y1(x)** em relação a x.

  Nota: para seleccionar ∫ do menu da barra de ferramentas Calc, prima [F3] **2** (ou prima [2nd] [∫] no teclado).

Define y1(x)=When (x<0,⁻x, 5cos(x))



Prima [◆] [ENTER] para obter um

Se receber uma mensagem de erro de falta de memória

A TI-89 Titanium / Voyage™ 200 Graphing Calculator guarda os resultados intermédios na memória e remove-os posteriormente quando o cálculo termina. Dependendo da complexidade do cálculo, a memória pode acabar antes da apresentação do resultado.

Libertação de memória

- Exclua as variáveis desnecessárias e/ou aplicações Flash, principalmente as maiores.
  - Utilize [2nd] [VAR-LINK] conforme descrito no módulo *Gestão da memória e das variáveis* para ver e excluir variáveis e/ou aplicações Flash.
- No ecrã principal:

- Limpe a área do histórico (F1 **8**) ou exclua os pares desnecessários da área do histórico.
- Pode também utilizar F1 **9** para reduzir o número de pares da área do histórico a guardar.

- Utilize MODE para configurar Exact/Approx = APPROXIMATE. (Nos resultados com um número elevado de dígitos, é utilizada menos memória do que com AUTO ou EXACT. Nos resultados com apenas alguns dígitos, é utilizada mais memória.)

Simplificação de problemas

- Divida o problema em partes.
  - Divida **solve(**a∗b=0,*var***)** em **solve(**a=0,*var***)** e **solve(**b=0,*var***)**. Resolva cada parte e combine os resultados.
- Se algumas variáveis indefinidas ocorrerem apenas numa determinada combinação, substitua a combinação por uma única variável.
  - Se a única ocorrência de m e c é em **m∗c**^{2}, substitua e por **m∗c**^{2}.
  - Na expressão $\frac{(a+b)^2+\sqrt{(a+b)^2}}{1-(a+b)^2}$, substitua c por (a+b) e utilize $\frac{c^2+\sqrt{c^2}}{1-c^2}$. Na solução, substitua novamente c por (a+b).
- Em expressões combinadas com um denominador comum, substitua as somas nos denominadores por novas variáveis indefinidas únicas.
  - Na expressão $\frac{x}{\sqrt{a^2+b^2}+c}+\frac{y}{\sqrt{a^2+b^2}+c}$, substitua d por $\sqrt{a^2+b^2}+c$ e use $\frac{x}{d}+\frac{y}{d}$.

    Na solução, substitua novamente d por $\sqrt{a^2+b^2}+c$.

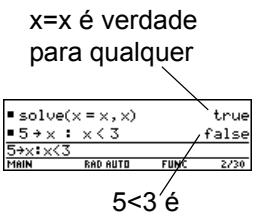
- Substitua as variáveis indefinidas por valores numéricos conhecidos no início do cálculo, principalmente se forem números inteiros simples ou fracções.
- Reformule um problema para evitar potências fraccionárias.
- Omita termos relativamente pequenos quando calcular uma aproximação.

Constantes especiais utilizadas na manipulação simbólica

O resultado de um cálculo pode incluir uma das constantes especiais descritas nesta secção. Em alguns casos, pode ser preciso inserir também uma constante como parte da sua entrada.

true, false

Indicam o resultado de uma igualdade ou de uma expressão booleana.



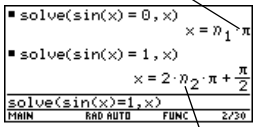
@n1 ... @n255

Esta notação indica um número “inteiro arbitrário” que representa um número inteiro qualquer.

Se um número inteiro arbitrário ocorrer várias vezes na mesma sessão, cada ocorrência é numerada de forma progressiva. Quando a numeração atingir 255, reinicia em @n0. Utilize Clean Up **2:NewProb** para reinicializar em @n1.

Nota: para @, prima:

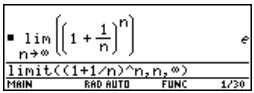
Cada múltiplo inteiro de π, tem uma solução.



Tanto @n1 como @n2 representam qualquer inteiro arbitrário, mas esta notação identifica inteiros

∞, e

∞ representa infinito, e e representa a constante 2.71828... (base dos logaritmos naturais).



Estas constantes são utilizadas frequentemente nas introduções de cálculos e nos resultados.

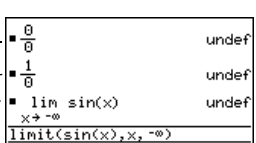
Nota:
para ∞, prima:
[◆] [∞]
2nd [∞]

para *e*, prima:
[◆] [e^x]
2nd [e^x]

undef

Indica que o resultado está indefinido.

Matematicamente não definido
±∞ (sinal indeterminado)
Limite não único



Constantes e unidades de medição

Introdução de constantes ou unidades

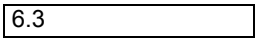
Pode utilizar um menu para seleccionar a partir de uma lista de unidades e constantes disponíveis, ou pode digitá-las directamente no teclado.

Num menu

Em seguida, é mostrado como seleccionar uma unidade, mas pode utiizar o mesmo procedimento geral para seleccionar uma constante.

No ecrã principal:

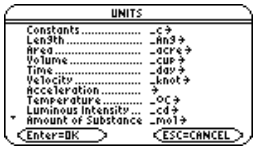
1. Digite o valor ou expressão.



2. Veja a caixa de diálogo **UNITS**. Prima:

   [2nd] [UNITS]

   [◆] [UNITS]

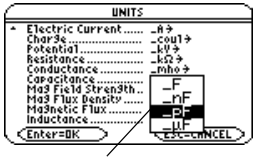


3. Utilize ⊖ e ⊕ para mover o cursor para a categoria aplicável.

   Nota: utilize [2nd] ⊖ e [2nd] ⊕ para percorrer uma página de categorias de cada vez.

4. Para seleccionar a unidade realçada (padrão), prima [ENTER].
   – ou –
   Para seleccionar uma unidade diferente da categoria, prima ⊙. Em seguida, realce a unidade desejada e prima [ENTER].

   Nota: se criar uma unidade definida pelo utilizador para uma categoria existente, ela é listada no menu.



Também é possível mover o cursor digitando a primeira letra de uma

A unidade seleccionada é colocada na linha de entrada. Os nomes da constante e da unidade começam sempre por um sinal de sublinhado (_).

6.3_pF

No teclado

Se conhecer a abreviatura que a TI-89 Titanium / Voyage™ 200 utiliza para uma determinada constante ou unidade, pode digitá-la directamente no teclado. Por exemplo:

256_m

- O primeiro carácter deve ser um sublinhado (_). Para obter o _, prima:
  [◆] [_]
  [2nd] [_]
- Um espaço ou um símbolo de multiplicação (∗) antes do símbolo de sublinhado é opcional. Por exemplo, **256_m**, **256 _m**, and **256∗_m** são equivalentes.

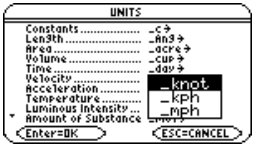
- Entretanto, se estiver a adicionar unidades a uma variável, é preciso colocar um espaço ou ∗ antes do símbolo de sublinhado. Por exemplo, *x_m* é tratado como uma variável, não como x com uma unidade.

Nota: os caracteres das unidades podem ser maiúsculos ou minúsculos.

Combinação de várias unidades

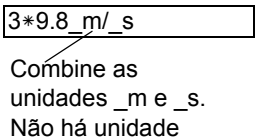
Pode ser preciso combinar duas ou mais unidades a partir de categorias diferentes.

Por exemplo, suponha que deseja introduzir a velocidade em metros por segundo. Entretanto, na caixa de diálogo UNITS, a categoria **Velocity** não contém esta unidade.



Pode introduzir metros por segundo

combinando **_m** e **_s** das categorias **Length** e **Time**, respectivamente.



Nota: crie uma unidade definida pelo utilizador para as combinações mais utilizadas.

Utilização de parêntesis com unidades num cálculo

Num cálculo, pode ser necessário utilizar parêntesis ( ) para agrupar um valor e as suas unidades, de forma que sejam calculadas adequadamente. Isto é particularmente verdade para problemas de divisão. Por exemplo:

Para calcular:	Introduza:
$\frac{100\_m}{2\_s}$	100_m/(2_s) 50 • $\frac{\_m}{\_s}$ É preciso utilizar parêntesis para (2_s). Isto é importante para a divisão. Se os parêntesis forem omitidos, resultados inesperados serão obtidos. Por exemplo: 100_m/2_s 50. ∗_m ∗_s

Nota: se tiver alguma dúvida sobre como um valor e as suas unidades serão calculadas, agrupe-as dentro de parêntesis ( ).

Esta é a explicação porque obtém resultados inesperados, se não utilizar parêntesis não forem utilizados. Num cálculo, uma unidade é tratada de forma semelhante a uma variável. Por exemplo: **100_m** é tratada como **100∗_m** e **2_s** é tratada como **2∗_s**. Sem parêntesis, a entrada é calculada como: **100∗_m / 2∗_s =** $\frac{100*\_m}{2}$ **∗_s = 50. ∗_m ∗_s.**

Conversão de uma unidade para outra

Pode converter de uma unidade para outra na mesma categoria, incluindo quaisquer unidades definidas pelo utilizador.

Para todas as unidades excepto a temperatura

Se utilizar uma unidade num cálculo, é convertida e apresentada automaticamente na unidade actual predefinida para a categoria, a menos que o operador de conversão ▸ seja utilizado, conforme descrito posteriormente. Os exemplos seguintes assumem que as unidades predefinidas são definidas no sistema de unidades métricas Si.

Nota:

- Para obter uma lista de unidades predefinidas.
- Seleccione unidades disponíveis a partir de um menu na caixa de diálogo **UNITS**.

Para multiplicar 20 vezes 6 quilómetros.

20∗6_km

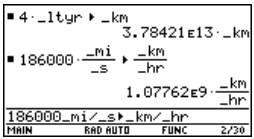
Mostrado na unidade padrão de comprimento (_m no

Se quiser converter uma unidade diferente da predefinição, utilize o operador de conversão ▸.

expressão_unidade1 ▸ *_unidade2*

└ Para ▸, prima 2nd [▸]

Para converter 4 anos-luz em quilómetros:



4_ltyr ▸ _km

Para converter 186000 milhas/segundo em quilómetros/hora:

186000_mi/_s ▸ _km/_hr

Se uma expressão utilizar uma combinação de unidades, pode especificar uma conversão apenas para algumas das unidades. Qualquer unidade sem conversão especificada, aparece de acordo com a predefinição.

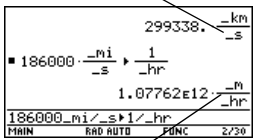
Para converter as milhas de 186000 milhas/segundo em quilómetros:

186000_mi/_s ▸ _km

Para converter os segundos de 186000 milhas/segundo em horas:

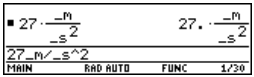
186000_mi/_s ▸ 1/_hr

Como uma conversão de tempo não é especificada, aparece a unidade



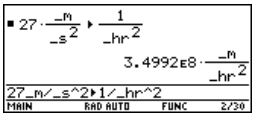
Como a conversão de distância não é especificada, aparece a unidade

Para digitar metros por segundo ao quadrado:



27_m/_s^2

Para converter de metros por segundo ao quadrado em horas:



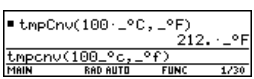
27_m/_s^2 ▸ 1/_hr^2

Para valores de temperatura

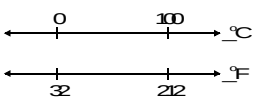
Para converter um valor de temperatura, é preciso utilizar **tmpCnv( )** em vez do operador ▸.

tmpCnv(*expressão_°tempUnid1*, *_°tempUnid2***)**
└ Para °, pressione [2nd] [°]

Por exemplo, para converter 100_°C em _°F:



tmpCnv(*100_°c, _°f***)**

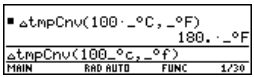


Para intervalos de temperatura

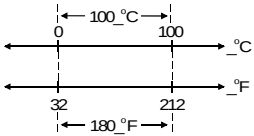
Para converter um intervalo de temperatura (a diferença entre dois valores de temperatura), utilize **ΔtmpCnv( )**.

ΔtmpCnv(*expressão_°tempUnid1*, *_°tempUnid2***)**

Por exemplo, para converter um intervalo de 100_°C no intervalo equivalente em _°F:



ΔtmpCnv(*100_°c, _°f***)**



Nota: para Δ, prima:

[◆] [(] [⇧] [D]

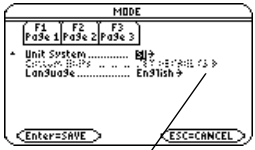
[2nd] **G** [⇧] **D**

Definição de unidades predefinidas para resultados apresentados

Todos os resultados que envolvam unidades aparecem na unidade predefinida para a categoria. Por exemplo, se a unidade predefinida para distância for *_m*, qualquer resultado de distância aparece em metros, mesmo que tenha sido introduzido em *_km* (quilómetros) ou *_ft* (pés) no cálculo.

Se utilizar o sistema SI ou ENG/US

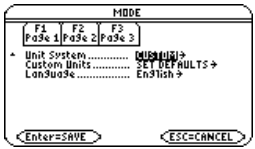
Os sistemas de medição SI e ENG/US (definido a partir da **página 3** do ecrã MODE) utilizam unidades predefinidas integradas, que não podem ser alteradas.



Se Unit System=SI ou ENG/US, o item Custom Units desaparece. Não é possível definir uma

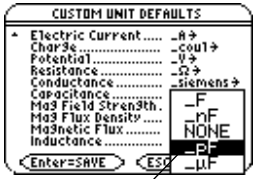
Definição de predefinições personalizadas

Para definir predefinições personalizadas:



1. Prima MODE F3 ▶ **3** para configurar **Unit System = CUSTOM**.
2. Prima ▼ para realçar **SET DEFAULTS**.
3. Prima ▶ para ver a caixa de diálogo **CUSTOM UNIT DEFAULTS**.

4. Para cada categoria, é possível realçar a sua predefinição, prima ⊙ e seleccione uma unidade na lista.
5. Prima duas vezes [ENTER] para guardar as alterações e sair do ecrã **MODE**.



Pode também mover o cursor digitando a primeira letra de uma

Nota:

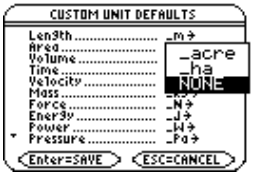
- Pode também utilizar **setUnits( )** ou **getUnits( )** para definir ou voltar às informações sobre as unidades predefinição. Consulte o módulo *Referência técnica*.
- Quando a caixa de diálogo **CUSTOM UNIT DEFAULTS** aparecer pela primeira vez, mostrará as unidades predefinidas actuais.

O que é a predefinição NONE?

Várias categorias permitem seleccionar NONE como unidade predefinida.

Isto significa que os resultados na categoria aparecem exibidos nas unidades predefinidas dos seus componentes.

Por exemplo, **Area = Length**^{2}, portanto **Length** é o componente de **Area**.



- Se as predefinições forem **Area = _acre** (acres) e **Length = _m** (metros), os resultados de área aparecem nas unidades **_acre**.
- Se definir **Area = NONE**, os resultados de área aparecem nas unidades **_m**^{2}.

Nota: NONE não está disponível para categorias de base, como Distância e Massa, que não têm componentes.

Criação de unidades - definidas pelo utilizador

Em qualquer categoria, é possível expandir a lista de unidades disponíveis através da definição de uma nova unidade em termos de uma ou mais unidades predefinidas. Também é possível utilizar unidades “independentes”.

Por que utilizar unidades próprias?

Exemplos de algumas razões para criar uma unidade são:

- Deseja introduzir valores de distância em decâmetros. Defina *10_m* como uma nova unidade chamada *_dm*.
- Em vez de introduzir *_m/_s*^{2} como unidade de aceleração, é possível definir a combinação de unidades como uma única unidade chamada *_ms2*.
- Deseja calcular quantas vezes algo pisca. Pode utilizar *_pisca* como unidade válida sem defini-la. Esta unidade “independente” é tratada de forma similar a uma variável não definida. Por exemplo, *3_pisca* é tratada como *3a*.

Nota: se criar uma unidade definida pelo utilizador para uma categoria existente, pode seleccioná-la no menu da caixa de diálogo UNITS. Mas não é possível utilizar MODE para seleccionar a unidade como uma predefinição para os resultados apresentados.

Regras para nomes de unidades definidas por utilizador

As regras de denominação de unidades são parecidas com as de variáveis.

- Pode ter até 8 caracteres.
- O primeiro carácter precisa ser um sublinhado. Para _, prima:
  [◆] [_]
  [2nd] [_]
- O segundo carácter pode ser qualquer carácter de nome de variável válido excepto _ ou dígito. Por exemplo, _*9f* não é válido.
- Os caracteres restantes (até 6) podem ser qualquer nome de variável excepto um sublinhado.

Definição de unidade

Defina uma unidade da mesma forma que se armazena uma variável.

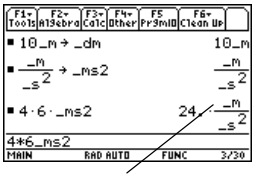
definição → *_novaUnid*
└── Para →, prima [STO▸]

Por exemplo, para definir a unidade decâmetro:

10_m →_dm

Para definir a unidade de aceleração:

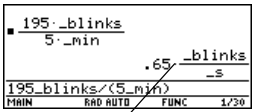
_m/_s^2 →_ms2



Assumindo que as predefinições de unidade para Lenght e Time estejam configuradas como

Para calcular 195 piscadas em 5 minutos como _blinks/_min:

195_blinks/(5_min)



Assumindo que a predefinição de unidade para Time esteja configurada

Notas:

- As unidades definidas pelo utilizador aparecem caracteres minúsculos, independentemente do tipo de letra (maiúscula ou não) utilizado na definição.
- As unidades definidas pelo utilizador como as *_dm* são guardadas como variáveis. Pode cancelá-las da mesma forma como se faria com uma variável.

Lista de unidades e constantes predefinidas

Esta secção lista as unidades e constantes predefinidas por categoria. Pode seleccionar qualquer uma delas a partir da caixa de diálogo UNITS. Se utilizar MODE para configurar unidades predefinidas, não se esqueça de que as categorias apenas com uma unidade definida não são listadas.

Predefinições para SI e ENG/US

Os sistemas de medição SI e ENG/US utilizam unidades predefinidas internas. Nesta secção, as predefinições internas são indicadas por (SI) e (ENG/US). Em algumas categorias, ambos os sistemas utilizam a mesma predefinição.

Não se esqueça de que algumas categorias não possuem unidades predefinições.Constantes.

	Descrição	Valor
_c	velocidade da luz	2.99792458E8_m/_s
_Cc	constante de Coulomb	8.9875517873682E9_N • $\_m^2/\_coul^2$
_g	aceleração da gravidade	$9.80665\_m/\_s^2$
_Gc	constante gravitacional	$6.6742E^{-}11\_m^3/\_kg/\_s^2$
_h	constante de Planck	6.6260693E⁻34_J • _s
_k	constante de Boltzmann	1.3806505E⁻23_J/_°K
_Me	massa do electrão	9.1093826E⁻31_kg
_Mn	massa do neutrão	1.67492728E⁻27_kg

	Descrição	Valor
_Mp	massa do prótão	1.67262171E-27_kg
_Na	número de Avogadro	6.0221415E23 /_mol
_q	carga de electrões	1.60217653E-19_coul
_Rb	raio de Bohr	5.291772108E-11_m
_Rc	constante de gás molar	8.314472_J/_mol/_°K
_Rdb	constante de Rydberg	10973731.568525 /_m
_Vm	volume molar	2.2413996E-2_m^3/_mol
_ε0	permissibilidade do vácuo	8.8541878176204E-12_F/_m
_σ	constante Stefan-Boltzmann	5.670400E-8_W/_m^2/_°K^4
_ϕ0	quantum de fluxo magnético	2.06783372E-15_Wb
_μ0	permeabilidade do vácuo	1.2566370614359E-6_N/_A^2
_μb	magneto de Bohr	9.2740154E 24_J • _m^2/_Wb

Nota:

- A TI-89 Titanium / Voyage™ 200 simplifica as expressões de unidade e mostra os resultados de acordo com as suas unidades predefinidas. Portanto, os valores de constante apresentados no ecrã podem parecer diferentes dos valores desta tabela.

- Para mais informações sobre caracteres gregos, consulte o módulo *Tabela de referência rápida*.

- Estes valores representam as constantes mais actualizadas disponíveis no momento da impressão dos valores recomendados internacionalmente CODATA das constantes físicas fundamentais disponíveis no site Web do National Institute of Standards and Technology (NIST). (http://physics.nist.gov/cuu/Constants/index.html).

Distância

_Ang	angstrom	_mi	milha
_au	unidade astronómica	_mil	1/1000 polegada
_cm	centímetro	_mm	milímetro
_fath	braça	_Nmi	milha náutica
_fm	fermi	_pc	parsec
_ft	pé (ENG/US)	_rod	vara
_in	polegada	_yd	jarda
_km	quilómetro	_µ	mícron
_ltyr	ano-luz	_Å	angstrom
_m	metro (SI)		

Área

_acre	acre	NONE (SI) (ENG/US)	
_ha	hectare		

Volume

_cup	xícara	_ml	mililitro
_floz	onça líquida	_pt	pinta
_flozUK	onça líquida britânica	_qt	quarto
_gal	galão	_tbsp	colher de mesa
_galUK	galão inglês	_tsp	colher de chá
_l	litro	NONE (SI) (ENG/US)	

Tempo

_day	dia	_s	segundo (SI) (ENG/US)
_hr	hora	_week	semana
_min	minuto	_yr	ano
_ms	milisegundo	_µs	microsegundo
_ns	nanosegundo		

Velocidade

_knot	nó	_mph	milhas por hora
_kph	quilómetros por hora	NONE (SI) (ENG/US)	

Aceleração

unidades não predefinidas		

Temperatura

_°C	°Celsius (Para °, prima 2nd [°].)	_°K	°Kelvin
_°F	°Fahrenheit	_°R	°Rankine (não prededinição)

Intensidade de luz

_cd	candela (não predefinição)		

Quantidade de substância

_mol	mol (não predefinição)		

Massa

_amu	unidade de massa atómica	_oz	onça
_gm	grama	_slug	slug
_kg	quilograma (SI)	_ton	tonelada
_lb	libra (ENG/US)	_tonne	tonelada métrica

_mg	miligrama	_tonUK	tonelada longa
_mton	tonelada métrica		

Força

_dyne	dina	_N	newton (SI)
_kgf	quilograma-força	_tonf	tonelada força
_lbf	libra-força (ENG/US)		

Energia

_Btu	unidade térmica britânica (BTU) (ENG/US)	_J	joule (SI)
_cal	caloria	_kcal	quilocaloria
_erg	erg	_kWh	quilowatt-hora
_eV	electron volt	_latm	litro-atmofera
_ftlb	pé-libra		

Potência

_hp	cavalo-vapor (ENG/US)	_W	watt (SI)
_kW	quilowatt		

Pressão

_atm	atmosfera	_mmHg	milímetros de mercúrio
_bar	bar	_Pa	pascal (SI)
_inH2O	polegadas de água	_psi	libras por polegada quadrada (ENG/US)
_inHg	polegadas de mercúrio	_torr	milímetros de mercúrio
_mmH2O	milímetros de água		

Viscosidade cinemática

_St	stokes		

Viscosidade dinâmica

_P	poise		

Frequência

_GHz	gigahertz	_kHz	kilohertz
Hz	hertz (SI) (ENG/US)	_MHz	megahertz

Corrente eléctrica

_A	ampere (SI) (ENG/US)	_mA	miliampere
_kA	quiloampere	_µA	microampere

Carga

_coul	coulomb (SI) (ENG/US)		

Potêncial

_kV	quilovolt	_V	volt (SI) (ENG/US)
_mV	milivolt	_volt	volt

Resistência

_kΩ	quilo ohm	_ohm	ohm
_MΩ	megaohm	_Ω	ohm (SI) (ENG/US)

Condutância

_mho	mho (ENG/US)	_siemens	siemens (SI)
_mmho	millimho	_µmho	micromho

Capacitância

_F	farad (SI) (ENG/US)	_pF	picofarad
_nF	nanofarad	_µF	microfarad

Força de campo magnético

_Oe	oersted		NONE (SI) (ENG/US)

Densidade de fluxo magnético

_Gs	gauss	_T	tesla (SI) (ENG/US)

Fluxo magnético

_Wb	weber (SI) (ENG/US)		

Indutância

_henry	henry (SI) (ENG/US)	_nH	nanohenry
_mH	milihenry	_µH	microhenry

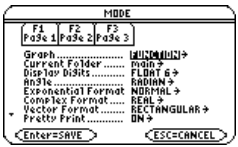
Representação gráfica de funções básicas

Visão geral dos passos para a representação gráfica de funções

Para representar uma ou mais funções **y(x)**, efectue o procedimento geral apresentado em seguida. Para uma explicação detalhada de cada passo, consulte as próximas páginas. A representação gráfica de uma função não requer necessariamente a execução do procedimento inteiro.

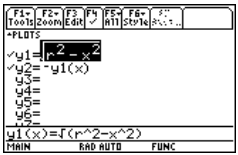
Representação gráfica de funções

1. Defina o modo **Graph** (MODE) em **FUNCTION**. Se necessário, defina também o modo **Angle**.



2. Defina funções em Y= Editor (◆ [Y=]).
3. Seleccione com (F4) as funções que deseja representar graficamente.

   Nota: para desactivar gráficos estatísticos, prima F5 **5** ou utilize F4.



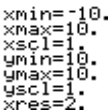
4. Defina o estilo de visualização de uma função.

   [2nd] [F6]

   [F6]

5. Defina a janela de apresentação ([◆] [WINDOW]).

   [F2] **Zoom** também modifica a janela de visualização.



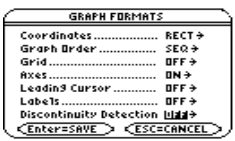
6. Se necessário, modifique o formato gráfico.

   [F1] **9**

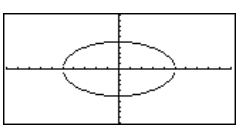
   – ou –

   [◆] [I]

   [◆] **F**



7. Represente as funções seleccionadas ([◆] [GRAPH]).



Exploração do gráfico

No ecrã Graph é possível:

- Mostrar as coordenadas de qualquer pixel com o cursor de movimento livre ou as coordenadas de qualquer ponto representado mediante a função de traçado.
- Utilizar o menu [F2] **Zoom** da barra de ferramentas para ampliar ou reduzir uma parte do gráfico.

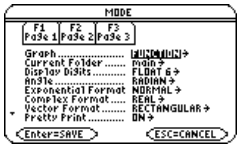
- Utilizar o menu F5 **Math** da barra de ferramentas para encontrar pontos específicos como o zero, o ponto mínimo, o máximo, etc.

Definição do modo Graph

Antes de representar graficamente as funções **y(x)**, é preciso seleccionar o modo de representação gráfica FUNCTION. Pode ser necessário também definir o modo Angle para determinar a modalidade de desenho dos gráficos de funções trigonométricas.

Modo Graph

1. Prima MODE para ver a caixa de diálogo **MODE**, que contém as definições actuais



2. Defina o modo Graph em **FUNCTION**. Consulte “Configuração dos modos” — *Funcionamento da calculadora*.

   Para gráficos que não utilizam números complexos, defina **Complex Format = REAL** para não afectar os gráficos que incluam potências como $\mathbf{x}^{1/3}$.

Este módulo explica exclusivamente os gráficos das funções **y(x)**; todavia, a TI-89 Titanium / Voyage™ 200 Graphing Calculator permite seleccionar 6 modos Graph.

Definição do modo Graph	Descrição
FUNCTION	funções **y(x)**

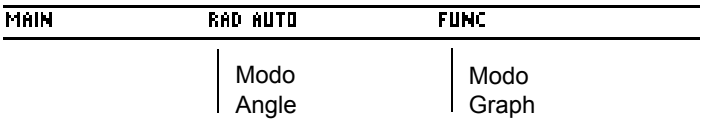
Definição do modo Graph	Descrição
PARAMETRIC	Equações paramétricas **x(t)** e **y(t)**
POLAR	Equações polares **r(θ)**
SEQUENCE	Sucessões **u(n)**
3D	Equações tridimensionais **z(x,y)**
DIFFERENTIAL EQUATION	Equações diferenciais **y'(t)**

Modo Angle

Quando utilizar funções trigonométricas, defina o modo Angle nas unidades (RADIANOS, GRADOS ou GRAUS) onde deseja introduzir e mostrar os valores angulares.

Verificação da linha de estado

Para ver as defiições actuais dos modos **Graph** e **Angle**, verifique a linha de estado apresentada na parte inferior do ecrã.

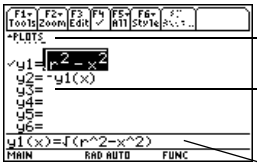


Definição de funções para a representação gráfica

No modo de representação gráfica FUNCTION, é possível representar funções definidas de **y1(x)** a **y99(x)**. Para definir e editar as funções, utilize Y= Editor, que indica os nomes das funções do modo de representação gráfica actual. Por exemplo: no modo POLAR os nomes das funções são **r1(θ), r2(θ)**, etc.

Definição de uma nova função

1. Prima [◆] [Y=] para ver Y= Editor.



Gráficos — desloque-se acima de y1= para ver uma lista de gráficos estatísticos.

Lista das funções — desloque-se através da lista de funções e definições.

Linha de entrada — área na qual se define ou edita a função realçada na lista.

Nota: a lista de funções mostra nomes abreviados como **y1**, embora a linha de entrada apresente o nome completo **y1(x)**.

2. Prima ⊖ e ⊕ para deslocar o cursor para qualquer função não definida. (Utilize [2nd] ⊖ e [2nd] ⊕ para ver uma página de cada vez.)
3. Prima [ENTER] ou [F3] para posicionar o cursor na linha de entrada.
4. Digite a expressão que define a função.
   - A variável independente na representação gráfica de funções é x.

- A expressão pode referir-se a outras variáveis, inclusive matrizes, listas e outras funções. Apenas os “floats” e as listas de “floats” produzirão um gráfico.

Nota: numa função não definida, não é preciso premir [ENTER] ou [F3]. Se começar a escrever, o cursor desloca-se à linha de entrada.

5. Quando terminar a expressão, prima [ENTER].

A lista de funções mostra e selecciona automaticamente a nova função para a representação gráfica.

Nota: se deslocar o cursor sem querer na linha de entrada, prima [ESC] para voltar à lista de funções.

Edição de uma função

De Y= Editor:

1. Prima ⊖ e ⊕ para realçar a função.
2. Prima [ENTER] ou [F3] para posicionar o cursor na linha de entrada.
3. Proceda de uma das seguintes maneiras.
   - Utilize ⊕ e ⊖ para posicionar o cursor na expressão e modifique-a. Consulte “Edição de uma expressão na linha de entrada” — *Funcionamento da calculadora*.

     – ou –
   - Prima uma ou duas vezes [CLEAR] para cancelar a expressão anterior e digite a nova expressão.

4. Prima [ENTER].

   A lista de funções mostra a função modificada e selecciona-a automaticamente para a representação gráfica.

   Nota: para cancelar quaisquer modificações, prima [ESC] em vez de [ENTER].

Cancelamento de uma função

No Y= Editor:

Para cancelar:	Operação:
Uma função da lista de funções	Realce a função e prima [←] ou [CLEAR].
Uma função da linha de entrada	Prima uma ou duas vezes [CLEAR] (segundo a posição do cursor) e prima [ENTER].
Todas as funções	Prima [F1] e seleccione select **8:Clear Functions**. Prima [ENTER] para confirmar.

Nota: [F1] **8** não cancela gráficos estatísticos (*Gráficos estatísticos e de dados*).

Não é preciso cancelar uma função, se não a desejar representar graficamente. É possível seleccionar apenas as funções que deseja representar.

Atalhos para movimentação do cursor

No Y= Editor:

Prima:	Para:
◆ ⊕ ou ◆ ⊖	Ir para a função 1 ou para a última função definida. Se o cursor estiver na última função definida ou houver registo da função, ◆ ⊖ vai para a função 99.

No ecrã principal ou num programa

É possível definir e calcular uma função no ecrã principal ou num programa.

- Utilize os comandos **Define** e **Graph**. Consulte:
  - “Representação gráfica de uma função definida no ecrã principal” e “Representação gráfica de uma função definida por intervalos” — *Tópicos complementares de representação gráfica*.
  - “Descrição da introdução de uma função” — *Programação*.
- Guarde uma expressão directamente numa variável de função. Consulte:
  - “Armazenamento e recuperação de valores de variáveis” no módulo *Funcionamento da calculadora*.
  - “Criação e cálculo de funções definidas pelo utilizador” no módulo *Ecrã principal da calculadora*.

Nota: as funções definidas pelo utilizador podem ter praticamente todos os nomes. Todavia, utilize os nomes **y1(x)**, **y2(x)** para mostrar as funções no Y= Editor, etc.

Selecção de funções para a representação gráfica

Independentemente do número de funções definidas em Y= Editor, é possível seleccionar aquelas que deseja representar graficamente.

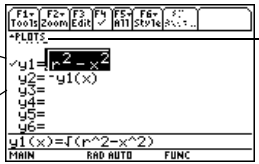
Selecção e não selecção de funções

Prima [◆] [Y=] para ver Y= Editor.

O símbolo “ ✓” indica as funções que serão representadas graficamente na próxima apresentação do ecrã Graph.

Seleccionad

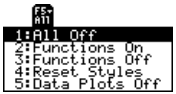
Não seleccionada



Se aparecerem os números dos gráficos (PLOT), seleccione os gráficos estatísticos correspondentes.

Neste exemplo, estão seleccionados os gráficos estatísticos 1 e 2. Para ver os gráficos, desloque-se acima de y1=.

Para seleccionar ou não seleccionar:	Operação:
Uma função específica	• Posicione o cursor para realçar a função. • Prima [F4]. Este procedimento permite seleccionar ou cancelar a selecção de uma função.

Para seleccionar ou não seleccionar:	Operação:
Todas as funções	• Prima F5 para ver o menu **All** da barra de ferramentas. • Seleccione o item correspondente. 

Se introduzir ou modificar uma função, não é preciso seleccionar a função, porque é seleccionada automaticamente. Para desactivar um gráfico estatístico, prima F5 **5** ou utilize F4 para anular o gráfico.

No ecrã principal ou de um programa

Também é possível seleccionar ou cancelar a selecção de funções do ecrã principal ou de um programa.

- Utilize os comandos **FnOn** e **FnOff** (disponíveis no menu F4 **Other** na barra de ferramentas) para as funções. Consulte o módulo *Referência técnica*.
- Utilize os comandos **PlotsOn** e **PlotsOff** para os gráficos estatísticos. Consulte o módulo *Referência técnica*.

Definição do estilo de apresentação de uma função

Para cada função definida, é possível definir um estilo de apresentação do gráfico. Estes parâmetros são muito úteis para a representação gráfica de várias funções. Por

exemplo: é possível definir um parâmetro para as linhas contínuas, um outro para as tracejadas, etc.

Apresentação ou modificação do estilo de uma função

No Y= Editor:

1. Realce a função desejada com o cursor.

   Seleccione o menu **Style**. Prima:

   [2nd] [F6]

   [F6]

- Mesmo que o item Line esteja realçado inicialmente, o estilo actual da função está assinalado com o sinal ✓.
- Para sair do menu sem efectuar modificações, prima [ESC].

2. Para modificar, seleccione o estilo desejado.

Estilo	Descrição
Line	Une os pontos representados com uma linha. É a predefinição.
Dot	Mostra um ponto para cada ponto representado.
Square	Mostra um quadro cheio em cada ponto representado.
Thick	Une os pontos representados com uma linha espessa.

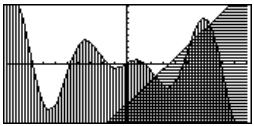
Estilo	Descrição
Animate	Um cursor circular desloca-se ao longo do gráfico sem desenhá-lo.
Path	Um cursor circular desloca-se ao longo do gráfico desenhando-o.
Above	Sombreia a área acima do gráfico.
Below	Sombreia a área abaixo do gráfico.

Para definir Line como estilo padrão de todas as funções, prima F5 e seleccione **4:Reset Styles**.

Utilização do sombreado superior ou inferior

A TI-89 Titanium / Voyage™ 200 Graphing Calculator tem quatro tipos de sombreados utilizados rotativamente. Se definir uma função como sombreada, utiliza o primeiro tipo; a próxima utiliza o segundo e assim por diante. A quinta função sombreada torna a utilizar o primeiro.

Se houver intersecção das áreas sombreadas, sobrepõem-se os respectivos tipos.



No ecrã principal ou num programa

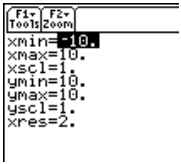
O estilo de uma função também pode ser definido no ecrã principal ou num programa. Consulte o comando **Style** no módulo *Referência técnica*.

Definição da janela de visualização

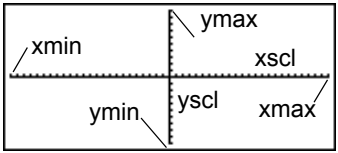
A janela de visualização representa uma parte do plano de coordenadas apresentada no ecrã Graph. Com as variáveis Window, é possível definir os limites e outras características da janela de visualização. Os gráficos de funções, de equações paramétricas, etc., têm o seu próprio conjunto independente de variáveis Window.

Apresentação das variáveis Window no Window Editor

Prima [◆] [WINDOW] para ver o Window Editor.



Variáveis Window (apresentadas no Window Editor)



Janela de visualização correspondente (apresentada no ecrã Graph)

Variável	Descrição
xmin, xmax, ymin, ymax	Limites da janela de visualização.
xscl, yscl	Distância entre as marcas assinalados nos eixos x e y.

Variável	Descrição
xres	Define a resolução pixel (1 a 10) dos gráficos de funções. A predefinição é 2. • Em 1, as funções são calculadas e representadas graficamente em cada pixel ao longo do eixo x. • Em 10, as funções são calculadas e representadas graficamente em cada 10 pixéis ao longo do eixo x.

Para desactivar os pontos assinalados, defina **xscl=0** e/ou **yscl=0**. Um valor baixo de **xres** melhora a resolução gráfica, embora possa diminuir a velocidade de execução.

Modificação dos valores

1. No Window Editor:
2. Desloque o cursor para destacar o valor que deseja modificar.
3. Utilize um dos seguintes modos:
   - Digite um valor ou expressão. O valor precedente é cancelado no início da digitação.

     – ou –
   - Prima [CLEAR] para cancelar o valor precedente e, em seguida, digite o novo valor.

     – ou –
   - Prima ◀ ou ▶ para remover o realce e, em seguida, modifique o valor.

Os valores são guardados à medida que são digitados; portanto, não é necessário pressionar [ENTER]. [ENTER] desloca simplesmente o cursor para a próxima variável

Window. Se digitar uma expressão, será calculada com o deslocamento do cursor para outra variável Window ou ao sair do Window Editor.

No ecrã principal ou num programa

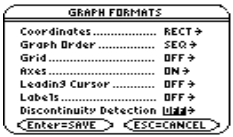
Os valores podem também ser guardados directamente nas variáveis Window do ecrã principal ou de um programa. Consulte “Armazenamento e recuperação dos valores de variáveis” — no módulo *Funcionamento da calculadora*.

Modificação do formato gráfico

É possível definir o formato gráfico para mostrar ou ocultar elementos de referência como os eixos, a grelha e as coordenadas do cursor. Os gráficos de funções, de equações paramétricas, etc., têm o seu próprio conjunto independente de formatos gráficos.

Apresentação das configurações de formato gráfico

No Y= Editor, no Window Editor ou no ecrã Graph, prima [F1] e seleccione **9:Format**.



- A caixa de diálogo GRAPH FORMATS mostra a definição actual.
- Para sair sem modificar, prima [ESC].

Pode também ver a caixa de diálogo GRAPH FORMATS no Y= Editor, no Window Editor ou no ecrã Graph. Prima:

Formato	Descrição
Coordinates	Mostra as coordenadas do cursor em formato rectangular (RECT) ou polar (POLAR), ou oculta (OFF) as coordenadas.
Graph Order	Representa graficamente as funções de uma forma consecutiva (SEQ) ou simultânea (SIMUL). Indisponível quando Discontinuity Detection estiver ligada.
Grid	Mostra (ON) ou oculta (OFF) os pontos da grelha correspondentes às marcas assinalados nos eixos.
Axes	Mostra (ON) ou oculta (OFF) os eixos x e y.
Leading Cursor	Mostra (ON) ou oculta (OFF) um cursor de referência que traça as funções durante a representação gráfica.
Labels	Mostra (ON) ou oculta (OFF) as etiquetas dos eixos x e y.
Discontinuity Detection	Elimina (ON) ou permite (OFF) assimptotas falsas e ligações numa descontinuidade de salto.

Para desactivar as marcas, defina a janela de visualização para que **xscl** e/ou **yscl = 0**.

Modificação das configurações

Na caixa de diálogo GRAPH FORMATS:

1. Realce a definição do formato com o cursor.
2. Prima ⊙ para ver o menu das configurações válidas para este formato.

3. Seleccione uma configuração:
   - Realce a opção com o cursor e, em seguida, prima [ENTER].

     – ou –
   - Prima o número correspondente à opção desejada.
4. Depois de modificar as configurações de formato desejadas, prima [ENTER] para guardar as modificações e fechar a caixa de diálogo **GRAPH FORMATS**.

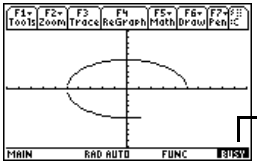
   Nota: para cancelar um menu ou sair da caixa de diálogo sem guardar as modificações, prima [ESC] em vez de [ENTER].

Representação gráfica das funções seleccionadas

Quando terminar as operações relativas à representação gráfica das funções seleccionadas, veja o ecrã Graph, que utilizará o estilo e a janela de visualização anteriormente definidas.

Visualização do ecrã Graph

Prima [◆] [GRAPH]. A TI-89 Titanium / Voyage™ 200 Graphing Calculator representa automaticamente as funções seleccionadas.



Durante a execução do gráfico, aparece o indicador **BUSY**.

Se seleccionar uma opção F2 **Zoom** de Y= Editor ou de Window Editor, a TI-89 Titanium / Voyage™ 200 mostra automaticamente o ecrã Graph.

Interrupção da representação gráfica

Durante a execução de um gráfico:

- Para interromper momentaneamente a representação, prima ENTER. (O indicador PAUSE substitui BUSY.) Para continuar, prima outra vez ENTER.
- Para anular a representação, prima ON. Para recomeçar, prima F4 (**ReGraph**).

Se quiser modificar a janela de visualização

Dependendo das definições, é possível que a representação gráfica de uma função possa ser demasiado grande ou pequena, ou que esteja deslocada num dos lados do ecrã. Para corrigir:

- Torne a definir a janela de visualização com limites diferentes.
- Utilize a opção **Zoom**.

Smart Graph

No ecrã Graph, a função Smart Graph mostra imediatamente o conteúdo da janela precedente, desde que não tenha sido modificada no que respeita à actualização gráfica.

Smart Graph actualiza a janela e o gráfico apenas se:

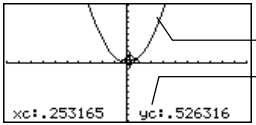
- Modificar uma configuração de modo a que influa na representação gráfica, um atributo de uma função ou formato gráfico.
- Seleccionar ou anular uma função ou um gráfico estatístico. (Se seleccionar apenas uma nova função, Smart Graph adiciona-a o ecrã Graph.)
- Modificar a definição de uma função seleccionada ou o valor de uma variável numa função seleccionada.
- Cancelar um objecto desenhado.
- Modificar a definição de um gráfico estatístico.

Apresentação de coordenadas com o cursor de movimento livre

Para ver as coordenadas em qualquer ponto do ecrã Graph, utilize o cursor de movimento livre. Pode deslocar este cursor em qualquer pixel do ecrã, porque não está limitado ao gráfico da função.

Cursor de movimento livre

Quando o gráfico Graph aparecer pela primeira vez, nenhum cursor está visível. Para ver o cursor, prima uma seta de tecla do cursor. Este desloca-se a partir do centro do ecrã e mostra as coordenadas.



$y1(x)=x^2$

A letra “c” indica que são as coordenadas do cursor. Os valores são guardados nas variáveis xc e yc. As coordenadas rectangulares utilizam xc e yc; as polares rc e qc.

Se o ecrã não mostrar as coordenadas, defina o formato gráfico de forma a que **Coordinates = RECT** ou **POLAR**. Prima:

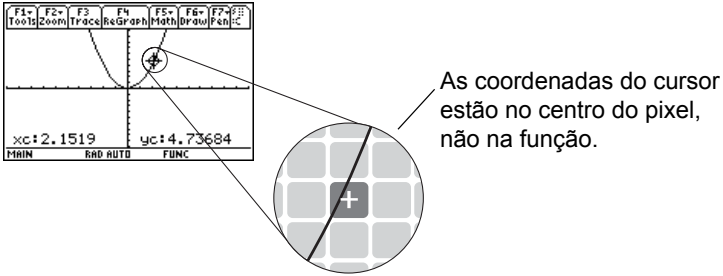
[◆] [|]

[◆] **F**

Para deslocar o cursor de movimento livre:	**Prima:**
Para um pixel adjacente	Uma seta de tecla do cursor para qualquer direção.
Com incrementos de 10 pixéis	[2nd] e, em seguida, a tecla do cursor.

Nota: para ocultar o cursor e as suas coordenadas temporariamente, prima [CLEAR], [ESC], ou [ENTER]. Quando o mover da próxima vez, desloca-se a partir da última posição.

Quando deslocar o cursor para um pixel que parece estar “em cima” da função, não se esqueça de que pode estar junto à função, não necessariamente sobre a função.



Para aumentar a precisão:

- Utilize a ferramenta **Trace** descrita na próxima página para ver as coordenadas da função.
- Utilize uma operação **Zoom** para aumentar ou diminuir uma parte do gráfico.

Traçado de uma função

Para ver as coordenadas exactas de um ponto traçado num gráfico, utilize a ferramenta F3 **Trace**. Diferentemente do cursor de movimento livre, o cursor de traços desloca-se apenas ao longo dos pontos traçados de uma função.

Início do traço

No ecrã Graph, prima F3.

Mostra o cursor de traços na função no ponto do valor **x** médio do ecrã. As coordenadas do cursor aparecem na parte inferior do ecrã.

Se representar várias funções, o cursor de traços aparece na função com o número mais baixo seleccionada no Y= Editor. O número da função está indicado na margem superior direita do ecrã.

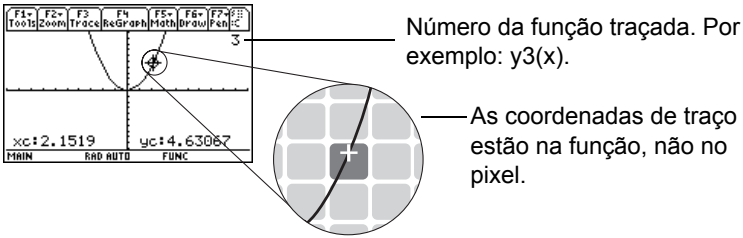
Se representar gráficos estatísticos, o cursor de traços aparece com o número mais baixo.

Movimentação do cursor ao longo de uma função

Para deslocar o cursor de traços:	Operação:
Para o ponto precedente ou sucessivo	Prima ◀ ou ▶.
Aproximadamente 5 pontos (podem ser mais ou menos que 5, dependendo da variável **xres** Window)	Prima 2nd ◀ ou 2nd ▶.
Para um valor específico x na função	Digite o valor x e pressione ENTER.

Nota: se introduzir um valor de x, o mesmo deve estar compreendido entre **xmin** e **xmax**.

O cursor de traços desloca-se apenas a partir de um ponto traçado ao seguinte ao longo da função — não de pixel para outro.



Se as coordenadas não aparecerem no ecrã, defina o formato gráfico de forma a que **Coordinates = RECT** ou **POLAR**. Prima:

◆ |

◆ **F**

Cada valor representado de y, é calculado segundo o valor de x, a saber, **y=yn(x)**. Se a função não estiver definida num valor x, o valor y está em branco.

É possível continuar a traçar uma função além da borda superior ou inferior da janela de visualização. Não vê o cursor quando este ultrapassa o limite do ecrã, mas os valores apresentados das suas coordenadas estão correctos.

Nota: a opção QuickCenter permite traçar uma função que ultrapassa os limites da janela (superior e inferior).

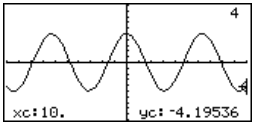
Passagem de uma função para outra

Prima ⊕ ou ⊖ para passar respectivamente para a função anterior ou sucessiva seleccionada com o mesmo valor x. O número da nova função aparece no ecrã.

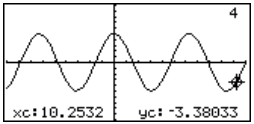
A função “anterior” ou “sucessiva” baseia-se na ordem das funções seleccionadas no Y= Editor e não na ordem pela qual aparecem no ecrã.

Enquadramento automático

Se traçar uma função que ultrapassa a borda direita ou esquerda do ecrã, a janela de visualização desloca-se automaticamente para a direita ou para a esquerda. Durante o desenho da nova parte, há uma breve pausa.



Antes do enquadramento automático



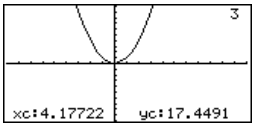
Depois do enquadramento automático

Depois do enquadramento automático, o cursor continua o traçado.

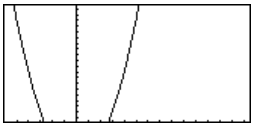
Nota: o enquadramento automático não funciona durante a apresentação de gráficos estatísticos ou se uma função utilizar um estilo de apresentação sombreado.

Utilização do QuickCenter

Se uma função traçada superar a borda superior ou inferior do ecrã de apresentação, prima [ENTER] para centrar a janela de visualização na posição do cursor.



Antes de utilizar QuickCenter



Depois de utilizar QuickCenter

Depois de activar o QuickCenter, o cursor pára o traçado. Se deseja continuar, prima [F3].

A opção QuickCenter pode ser utilizada em qualquer momento durante a operação de traçado, mesmo se o cursor ainda estiver no ecrã.

Cancelamento do traçado

Para cancelar o traçado em qualquer momento, prima [ESC].

O traçado também é cancelado quando vir outras aplicações, como Y= Editor. Quando voltar ao ecrã Graph, prima [F3] para iniciar o traçado:

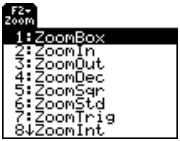
- Se Smart Graph actualizar o ecrã, o cursor aparece no valor médio de x.
- Se Smart Graph não actualizar o ecrã, o cursor aparece na posição anterior da visualização da outra aplicação.

Utilização do zoom para explorar um gráfico

O menu F2 **Zoom** da barra de ferramentas tem várias utilitários que permitem ajustar a janela de visualização. É possível guardar uma janela de visualização para a utilizar uma outra vez.

Visão geral do menu Zoom

Prima F2 no Y= Editor, no Window Editor ou no ecrã Graph.



Os procedimentos para utilizar **ZoomBox**, **ZoomIn**, **ZoomOut**, **ZoomStd**, **Memory**, e **SetFactors** estão descritos nas próximas páginas desta secção.

Para mais informações sobre os outros itens, consulte o módulo *Referência técnica*.

Nota: se seleccionar uma ferramenta **Zoom** do Y=Editor ou Window Editor, a TI-89 Titanium / Voyage™ 200 Graphing Calculator mostra automaticamente o ecrã Graph.

Ferramenta Zoom	Descrição
ZoomBox	Permite desenhar um quadro e ampliá-lo.
ZoomIn, ZoomOut	Permite seleccionar um ponto e ampliá-lo ou reduzi-lo conforme o número definido em **SetFactors**.
ZoomDec	Define Δx e Δy em .1 e centra a origem.

Ferramenta Zoom	Descrição
ZoomSqr	Define as variáveis Window de forma a que um quadrado ou circunferência sejam apresentados com as proporções correctas ao (em vez de um rectângulo ou uma elipse).
ZoomStd	Define as variáveis Window com os respectivos valores predefinidos. xmin = ⁻10 ymin = ⁻10 xres = 2 xmax = 10 ymax = 10 xscl = 1 yscl = 1
ZoomTrig	Define as variáveis Window com valores predefinidos frequentemente indicados para representar funções trigonométricas. Centra a origem e define: $\Delta x = \pi/24$ (.130899... radianos ou 7.5 graus) ymin = ⁻4 ymax = 4 xscl = $\pi/2$ (1.570796... radianos ou 90 graus) yscl = 0.5
ZoomInt	Permite seleccionar um novo ponto central e, em seguida definir Δx e Δy em 1 e **xscl** e **yscl** em 10.
ZoomData	Ajusta as variáveis Window para mostrar todos os gráficos estatísticos seleccionados.
ZoomFit	Ajusta a janela de visualização para mostrar toda a série dos valores da variável dependente respeitantes às funções seleccionadas. Na representação gráfica de funções, conserva os valores **xmin** e **xmax** e define **ymin** e **ymax**.
Memory	Permite guardar e recuperar as variáveis Window para tornar a criar uma janela de visualização personalizada.

Ferramenta Zoom	Descrição
SetFactors	Permite definir os factores **Zoom** para **ZoomIn** e **ZoomOut**.

Δx e Δy representam a distância do centro de dois pixéis adjacentes.

Ampliação com o Zoom Box

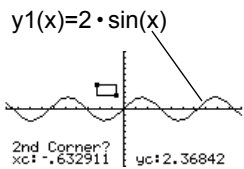
1. No menu [F2] **Zoom**, seleccione **1:ZoomBox**.

   O ecrã mostra **1st Corner?**

2. Posicione o cursor numa borda do quadro que deseja definir e prima [ENTER].

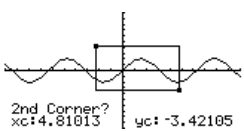
   O cursor torna-se num pequeno rentângulo e no ecrã aparece a pergunta qual é o **2nd Corner?**

   Nota: para deslocar o cursor por incrementos, utilize [2nd] ⊙, [2nd] ⊙, etc.



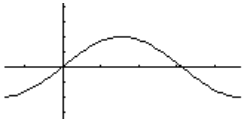
3. Desloque o cursor para o canto oposto do quadro.

   O quadro é desenhado à medida que desloca o cursor.



4. Depois de delimitar a área que deseja ampliar, prima [ENTER].

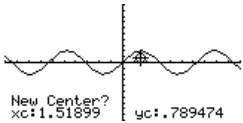
   O ecrã Graph mostra a área ampliada. Para anular **ZoomBox**, prima [ESC] antes de [ENTER].



Ampliação e redução de um ponto

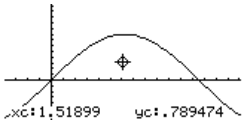
1. No menu [F2] **Zoom**, seleccione **2:ZoomIn** ou **3:ZoomOut**.

   Aparece um cursor e no ecrã aparece a pergunta **New Center?**



2. Posicione o cursor no ponto que deseja ampliar ou reduzir e prima [ENTER].

   A TI-89 Titanium / Voyage™ 200 define as variáveis Window segundo os factores de **Zoom** definidos em **SetFactors**.



- Numa operação **ZoomIn**, as variáveis x são divididas por **xFact** e as variáveis y por **yFact**.

  new **xmin = xmin/xFact** , etc.

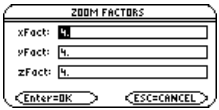
- Numa operação **ZoomOut**, as variáveis x são multiplicadas por **xFact** e as y por **yFact**.

  new **xmin = xmin** ∗ **xFact** , etc.

Modificação dos factores Zoom

Os factores Zoom definem o aumento ou a redução utilizados por **ZoomIn** e **ZoomOut**.

1. No menu [F2] **Zoom**, seleccione **C:SetFactors** para ver a caixa de diálogo **ZOOM FACTORS**.



Os factores Zoom devem ser ≥ 1, embora não precisem de ser números inteiros. O valor predefinido é 4.

Nota: para sair sem guardar as modificações, prima [ESC].

2. Utilize ⊖ e ⊕ para realçar o valor que deseja modificar. Em seguida:
   - Digite o novo valor; o valor precedente é cancelado automaticamente.

     – ou –
   - Prima ⊲ ou ⊳ para remover o realce e, em seguida, modifique o valor precedente.
3. Prima [ENTER] (depois de introduzir numa caixa de entrada, prima duas vezes [ENTER]) para guardar as modificações e sair da caixa de diálogo.

Armazenamento ou visualização de uma janela de visualização

Depois de utilizar as várias ferramentas **Zoom**, é possível voltar à janela de visualização anterior ou guardar a janela actual.

1. No menu F2 **Zoom**, seleccione **B:Memory** para ver um submenu.

2. Seleccione o item desejado.

Seleccione	Para:
1:ZoomPrev	Voltar à janela de visualização precedente durante a utilização do zoom.
2:ZoomSto	Memorizar a janela de visualização actual (os valores das variáveis Window actual são guardados nas variáveis de sistema **zxmin**, **zxmax**, etc.)
3:ZoomRcl	Recuperar a última janela de visualização guardada com **ZoomSto**.

Nota: é possível guardar apenas uma série de valores de variáveis Window de cada vez. Se memorizar uma nova, esta sobrepõe a série precedente.

Restabelecimento da janela de visualização predefinida

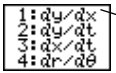
É possível restabelecer a qualquer momento os valores predefinidos das variáveis Window. No menu F2 **Zoom**, seleccione **6:ZoomStd**.

Utilização de ferramentas do menu Math para analisar funções

No ecrã Graph, é possível aceder ao menu da barra de ferramentas F5 **Math** e às suas diversas ferramentas que ajudam a analisar funções representadas graficamente.

Descrição do menu Math

Prima F5 no ecrã Graph.



No submenu Derivatives, apenas dy/dx está disponível para a representação gráfica de funções. As outras derivadas estão disponíveis para os outros modos de representação (paramétricas, polares, etc.).

Ferramenta Math	Descrição
Value	Calcula uma função **y(x)** seleccionada com um valor especificado de x.
Zero, **Minimum**, **Maximum**	Calcula o zero (intersecção com o eixo x), o ponto mínimo e máximo num intervalo de x.
Intersection	Calcula o ponto de intersecção de duas funções.
Derivatives	Calcula a derivada (inclinação) num ponto.
]f(x)dx	Calcula o integral numérico aproximado num intervalo.

Ferramenta Math	Descrição
Inflection	Calcula o ponto de inflexão de uma curva onde a segunda derivada troca de sinal (ponto onde muda a concavidade da curva).
Distance	Desenha e mede uma linha entre dois pontos da mesma função ou de duas funções diferentes.
Tangent	Desenha a tangente num ponto e mostra a respectiva equação.
Arc	Calcula o comprimento de arco entre dois pontos de uma curva.
Shade	Depende do número de funções representadas. • Se estiver representada apenas uma função, sombreia a área acima ou abaixo do eixo x. • Se estiver representada mais que uma função, sombreia a área entre as duas funções num intervalo.

Nota: as coordenadas do cursor são guardadas nas variáveis de sistema xc e yc (rc e θc se utilizar coordenadas polares) no menu Math. As derivadas, os integrais, as distâncias, etc., são guardadas na variável de sistema *sysMath*.

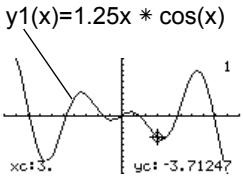
Cálculo de y(x) num ponto específico

1. No ecrã **Graph**, prima F5 e seleccione **1:Value**.

2. Digite o valor de x compreendido entre um valor real de **xmin** e **xmax**. O valor pode ser uma expressão.

3. Prima [ENTER].

   O cursor desloca-se para o valor de x da primeira função seleccionada em Y= Editor, mostrando as coordenadas.



4. Prima ⊖ ou ⊕ para deslocar o cursor entre as funções com o valor introduzido de x. Aparece o valor correspondente de y .

   Se premir ◀ ou ▶, aparece o cursor de movimento livre; nem sempre é possível deslocá-lo para o valor introduzido de x.

Também é possível ver as coordenadas de uma função, traçando a função ([F3]), digitando um valor x e premindo [ENTER].

Cálculo do zero, do ponto mínimo e do máximo num intervalo

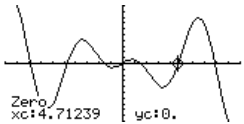
1. No ecrã **Graph**, prima [F5] e seleccione **2:Zero**, **3:Minimum**, ou **4:Maximum**.
2. Utilize ⊖ e ⊕ para seleccionar a função desejada.

   Nota: digitar os valores de x é um modo rápido para estabelecer os limites.
3. Defina o limite inferior de x. Utilize ◀ e ▶ para posicionar o cursor até este limite ou digite o seu valor x.

4. Prima [ENTER]. O símbolo ▸ na parte superior do ecrã assinala o limite inferior.

5. Defina o limite superior e prima [ENTER].

   O cursor desloca-se para a solução e mostra as coordenadas.

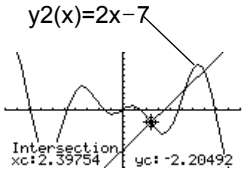


Cálculo do ponto de intersecção de duas funções num intervalo

1. No ecrã **Graph**, prima [F5] e seleccione **5:Intersection**.
2. Seleccione a primeira função com ⊖ ou ⊕ e prima [ENTER]. O cursor desloca-se para a próxima função representada graficamente.
3. Seleccione a segunda função e prima [ENTER].
4. Defina o limite inferior de x. Posicione o cursor no limite inferior com ◀ e ▶ ou digite o respectivo valor x.
5. Prima [ENTER]. O símbolo ▸ na parte superior do ecrã assinala o limite inferior.

6. Defina o limite superior e prima [ENTER].

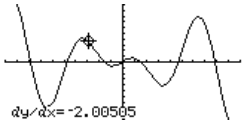
   O cursor desloca-se para o ponto de intersecção e mostra as coordenadas.



Cálculo da derivada (inclinação) num ponto

1. No ecrã **Graph**, prima F5 e seleccione **6:Derivatives**. Em seguida, seleccione **1:dy/dx** no submenu.
2. Utilize ⊖ e ⊕ para seleccionar a função desejada.
3. Introduza o ponto da derivada. Posicione o cursor no ponto ou digite o respectivo valor de x.
4. Prima ENTER.

   Aparece a derivada neste ponto.



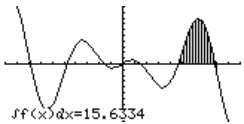
Cálculo do integral numérico num intervalo

1. No ecrã **Graph**, prima F5 e seleccione **7:∫f(x)dx**.
2. Utilize ⊖ e ⊕ para seleccionar a função desejada.

   Nota: digitar os valores de x é um modo rápido para estabelecer os limites.
3. Introduza o limite inferior de x. Posicione o cursor no limite inferior com ⊖ e ⊕ ou digite o respectivo valor de **x**.
4. Prima ENTER. O símbolo ▸ na parte superior do ecrã assinala o limite inferior.

   Nota: para cancelar a área sombreada, prima F4 (**ReGraph**).
5. Defina o limite superior e prima ENTER.

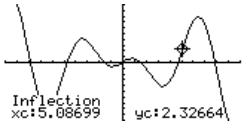
   O intervalo é sombreado e aparece o respectivo integral numérico aproximado.



Obtenção de um ponto de inflexão num intervalo

1. No ecrã **Graph**, prima [F5] e seleccione **8:Inflection**.
2. Utilize ⊖ e ⊕ para seleccionar a função desejada.
3. Defina o limite inferior de x. Posicione o cursor no limite inferior com ⊖ e ⊕ ou digite o respectivo valor de x.
4. Prima [ENTER]. O símbolo ▸ na parte superior do ecrã assinala o limite inferior.
5. Defina o limite superior e prima [ENTER].

   O cursor desloca-se para o ponto de inflexão (se disponível) no intervalo e aparecem as coordenadas.



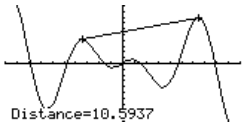
Cálculo da distância entre dois pontos

1. No ecrã **Graph**, prima [F5] e seleccione **9:Distance**.
2. Utilize ⊖ e ⊕ para seleccionar o primeiro ponto da função.
3. Defina o primeiro ponto. Posicione o cursor no ponto com ⊖ e ⊕ ou digite o respectivo valor de x.
4. Pressione [ENTER]. Um sinal **+** indica o ponto.
5. Se o segundo ponto estiver situado noutra função, utilize ⊖ e ⊕ para seleccionar a função.

6. Defina o segundo ponto. (Se utilizar o cursor para o fazer, desenha uma linha à medida que desloca o cursor.)

7. Prima [ENTER].

   A distância entre os dois pontos aparece junto à linha de ligação.



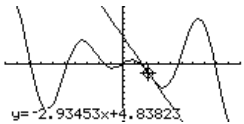
Desenho da recta tangente

1. No ecrã **Graph**, prima [F5] e seleccione **A:Tangent**.
2. Utilize ⊖ e ⊕ para seleccionar a função desejada.

   Nota: para cancelar a recta tangente desenhada, prima [F4] (**ReGraph**).

3. Defina o ponto da tangente. Posicione o cursor no ponto ou digite o respectivo valor de x.
4. Prima [ENTER].

   A tangente é desenhada e aparece a respectiva equação.

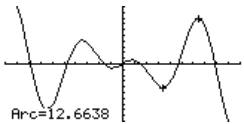


Cálculo do comprimento de um arco

1. No ecrã **Graph**, prima [F5] e seleccione **B:Arc**.
2. Utilize ⊖ e ⊕ para seleccionar a função desejada.

3. Defina o primeiro ponto do arco. Utilize ◀ ou ▶ para deslocar o cursor ou digite o respectivo valor de x.
4. Prima [ENTER]. Um sinal **+** indica o primeiro ponto.
5. Defina o segundo ponto e prima [ENTER].

   Um sinal **+** indica o segundo ponto e aparece o comprimento do arco.



Sombreado da área entre uma função e o eixo X

Apenas uma única função pode estar representada graficamente; caso contrário a ferramenta Shade sombreará a área compreendida entre as duas funções.

1. No ecrã **Graph**, prima [F5] e seleccione **C:Shade**. O ecrã pergunta **Above X axis?**
2. Seleccione uma das seguintes opções. Para sombrear a área da função:
   - Acima do eixo x, prima [ENTER].
   - Abaixo do eixo x, prima:

     [alpha] **N**

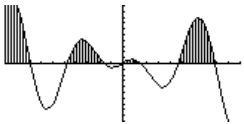
     N
3. Defina o limite inferior de x. Utilize ◀ ou ▶ para deslocar o cursor no limite inferior ou digite o respectivo valor de x.

   Nota: se não utilizar as teclas ◀ ou ▶, ou não digitar um valor de x quando estabelecer o limite superior e o inferior, **xmin** e **xmax** serão utilizados como o limite inferior e superior.

4. Prima [ENTER]. O símbolo ▸ na parte superior do ecrã assinala o limite inferior.

5. Defina o limite superior e prima [ENTER].

   A área delimitada é sombreada. Para cancelar a área sombreada, prima [F4] (**ReGraph**).



Sombreado da área entre duas funções num intervalo

É preciso haver pelo menos duas funções representadas; caso contrário (uma única função representada), a ferramenta Shade sombreia a área entre a função e o eixo de x.

1. No ecrã **Graph**, prima [F5] e seleccione **C:Shade**. O ecrã pergunta **Above?**
2. Utilize ⊖ e ⊕ para seleccionar uma função. (O sombreado está em cima da função.)
3. Prima [ENTER]. O cursor desloca-se para o gráfico da próxima função e o ecrã pergunta **Below?**
4. Utilize ⊖ e ⊕ para seleccionar uma outra função (O sombreado está em baixo da função.)
5. Prima [ENTER].
6. Defina o limite inferior de x. Utilize ◀ ou ▶ para deslocar o cursor para o limite inferior ou digite o valor de x.

   Nota: se não utilizar as teclas ◀ ou ▶, ou não digitar um valor de x quando estabelecer o limite superior e o inferior, **xmin** e **xmax** serão utilizados como o limite inferior e superior.

7. Prima [ENTER]. O símbolo ▸ na parte superior do ecrã assinala o limite inferior.

8. Defina o limite superior e prima [ENTER].

   A área delimitada é sombreada. Para cancelar a área sombreada, prima [F4] (**ReGraph**).

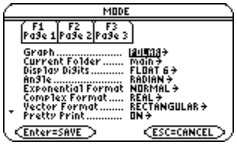
Representação gráfica de equações polares

Descrição das etapas de representação gráfica das equações polares

Para representar equações polares, efectue o mesmo procedimento para funções **y(x)** como descrito no módulo *Representação gráfica de funções básicas*. As diferenças válidas para equações polares são descritas em seguida.

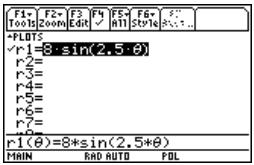
Representação gráfica de equações polares

1. Defina o modo **Graph** (MODE) para **POLAR**. Defina também o modo **Angle**, se necessário.



2. Defina as equações polares no Y= Editor (◆ [Y=]).
3. Seleccione (F4) que equações definidas serão representadas graficamente.

   Nota: Para desactivar qualquer gráfico estatístico, prima F5 **5** ou utilize F4.



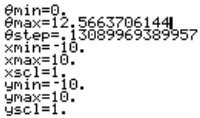
4. Defina o estilo de apresentação para uma equação.

   [2nd] [F6]
   [F6]

   Esta etapa é opcional. No caso de várias equações, ajuda a distinguir visualmente uma de outra.

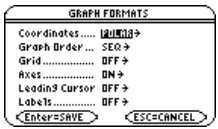
5. Defina a janela de visualização ([◆] [WINDOW]).

   [F2] **Zoom** também muda a janela de visualização.

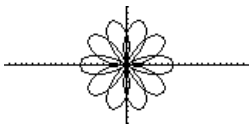


6. Mude o formato gráfico se necessário.

   [F1] **9**
   – or –
   [◆] [I]
   [◆] **F**



7. Represente as equações seleccionadas graficamente ([◆] [GRAPH]).



Exploração do gráfico

No ecrã Graph, é possível:

- Mostrar as coordenadas de qualquer pixel com o cursor de movimento livre ou as coordenadas de um ponto traçado através do traçado da equação polar.

- Utilizar o menu F2 **Zoom** da barra de ferramentas para ampliar ou reduzir uma parte do gráfico.
- Utilizar o menu F5] **Math** da barra de ferramentas para encontrar derivadas, tangentes etc. Algumas opções de menu não estão disponíveis para gráficos representados por coordenadas polares.

Diferenças entre a representação gráfica de equações polares e de funções

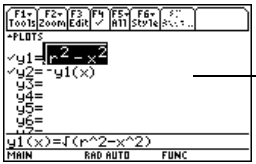
Este módulo assume que o leitor já está familiarizado com a representação gráfica de funções **y(x)** como descrito no módulo *Representação gráfica de funções básicas*. Esta secção descreve as diferenças que podem ser aplicadas às equações polares.

Definição do modo Graph

Utilize MODE para configurar **Graph = POLAR** antes de definir equações ou variáveis Window. O Y= Editor e o Window Editor permitem introduzir informações exclusivas para o modo **Graph** actual.

O modo **Angle** deve também ter as unidades (RADIAN ou DEGREE) definidas de acordo com as que deseja utilizar para θ.

Definição de equações polares no Y= Editor



É possível definir equações polares de r1(θ) a r99(θ).

É possível utilizar o comando **Define** no ecrã principal (consulte o módulo *Referência técnica*) para definir as funções e as equações em qualquer modo de representação gráfica independente do modo actual.

O Y= Editor mantém uma lista de funções independentes para cada modo de definição **Graph**. Suponha o exemplo seguinte:

- Defina um conjunto de funções **y(x)** no modo de representação gráfica FUNCTION. Em seguida, mude para o modo de representação POLAR e defina um conjunto de equações **r(θ)**.
- Quando voltar para o modo de representação FUNCTION, as suas funções **y(x)** ainda estão definidas para o Y= Editor. Quando voltar ao modo de representação POLAR, as suas equações **r(θ)** ainda estão definidas.

Selecção do estilo de apresentação

Os estilos **Above** e **Below** não estão disponíveis para equações polares e não estão no menu **Style** da barra de ferramentas do Y= Editor.

Variáveis Window

O Window Editor mantém um conjunto independente de variáveis Window para cada definição do modo **Graph** (como o Y= Editor mantém listas de função independentes). As representações gráficas polares utilizam as seguintes variáveis Window.

Variável	Descrição
θmin, θmax	Valores mínimo e máximo de θ que serão calculados.
θstep	Incremento para os valores de θ. As equações polares são calculadas em: r(θmin) r(θmin+θstep) r(θmin+2(θstep)) ..... que não ultrapasse ….. r(θmax)
xmin, **xmax**, **ymin**, **ymax**	Extremos da janela de visualização.
xscl, **yscl**	Distância entre as marcas nos eixos x e y.

Nota: é possível utilizar um incremento θ**step** negativo. Neste caso, θ**min** precisa ser maior que θ**max**.

Os valores predefinidos (definidos quando seleccionar **6:ZoomStd** no menu F2 **Zoom** da barra de ferramentas) são:

θmin = 0.		xmin = ⁻10.	ymin = ⁻10.
θmax = 2π	(6.2831853... radianos or 360 graus)	xmax = 10.	ymax = 10.

θstep = π/24	(.1308996... radianos or 7.5 graus)	xscl = 1.	yscl = 1.

Pode ser necessário alterar os valores predefinidos para as variáveis θ (θ**min**, θ**max**, θ**step**) para garantir a representação de um número suficiente de pontos.

Definição do formato gráfico

Para ver coordenadas como os valores **r** e θ, utilize

F1 **9**
– or –
◆ |
◆ **F**

para configurar **Coordinates = POLAR**. Se **Coordinates = RECT**, as equações polares serão representadas de forma adequada, mas as coordenadas serão representadas como x e y.

Quando traçar uma equação polar, a coordenada θ aparece mesmo se **Coordinates = RECT**.

Exploração de um gráfico

Como na representação de funções, os gráficos podem ser trabalhados através das ferramentas seguintes. As coordenadas apresentadas adoptam a forma polar ou rectangular conforme configurado no formato gráfico.

Ferramenta	Para gráficos em coordenadas polares:
Cursor de movimento livre	Funciona da mesma forma que nos gráficos de funções.
F2 **Zoom**	Funciona da mesma forma que nos gráficos de funções. • Apenas as variáveis **x** (**xmin**, **xmax**, **xscl**) e **y** (**ymin**, **ymax**, **yscl**) em Window são afectadas. • As variáveis Window θ (**θmin**, **θmax**, **θstep**) não são afectadas a menos que **6:ZoomStd** (que define **θmin = 0**, **θmax = 2π** e **θstep = π/24**) seja seleccionado.
F3 **Trace**	Permite mover o cursor ao longo do gráfico um **θstep** por vez. • Quando iniciar um traçado, o cursor posiciona-se sobre a primeira equação seleccionada em **θmin**. • QuickCenter aplica-se a todas as direcções. Se o cursor for movido para fora do ecrã (parte superior, inferior, esquerda ou direita), prima ENTER para centralizar a janela de visualização na posição do cursor. • O acompanhamento automático de visualização não está disponível. Se o cursor for movido para fora do ecrã pelo lado esquerdo ou direito, a TI-89 /Voyage™ 200 não acompanhará automaticamente a janela de visualização. Entretanto, pode utilizar o QuickCenter.

Ferramenta	Para gráficos em coordenadas polares:
F5 **Math**	Apenas **1:Value**, **6:Derivatives**, **9:Distance**, **A:Tangent**, e **B:Arc** estão disponíveis para os gráficos de equações polares. Estas ferramentas baseiam-se em valores de θ. Por exemplo: • **1:Value** mostra um valor de **r** (ou de x e y, dependendo do formato gráfico) para um valor θ especificado. • **6:Derivatives** encontra **dy/dx** ou **dr/d**θ num ponto definido para um valor de θ especificado.

É possível calcular também **r(**θ**)** durante um traçado, digitando o valor de θ e premindo ENTER.

Nota: é possível utilizar o QuickCenter a qualquer momento durante um traçado, mesmo se o cursor ainda estiver no ecrã.

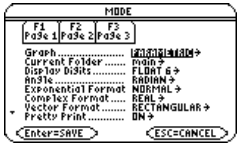
Representação gráfica de equações paramétricas

Descrição das etapas de representação gráfica de equações paramétricas

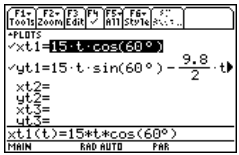
Para representar equações paramétricas graficamente, o procedimento é o mesmo utilizado para as funções **y(x)**, descrito no módulo *Representação gráfica de funções básicas*. As diferenças referentes às equações paramétricas são descritas em seguida.

Representação gráfica de equações paramétricas

1. Defina o modo **Graph** ([MODE]) para **PARAMETRIC**. Se necessário, defina também o modo **Angle**.



2. Defina as componentes x e y em Y= Editor ([◆] [Y=]).
3. Seleccione com ([F4]) as equações definidas que deseja representar graficamente. Seleccione o componente x, y ou ambos.



Nota: para desactivar um gráfico de dados estatísticos, prima [F5] **5** ou [F4].

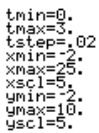
4. Defina o estilo de apresentação das equações. É possível definir o componente x ou y.

[2nd] [F6]

[F6]

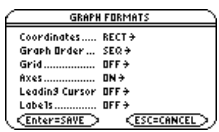
Esta etapa é opcional. No caso de várias equações, ajuda a distinguir visualmente uma da outra.

5. Defina a janela de visualização ([◆] [WINDOW]).



[F2] **Zoom** também modifica a janela de visualização.

6. Modifique, se necessário, o formato gráfico.



[F1] **9**

– ou –

[◆] [|]

[◆] **F**

7. Represente graficamente as equações seleccionadas ([◆] [GRAPH]).

Exploração do gráfico

No ecrã Graph, é possível:

- Mostrar as coordenadas dos pixéis com o cursor de movimento livre ou as coordenadas de um ponto representado através do traçado da equação paramétrica.
- Utilizar o menu F2 **Zoom** da barra de ferramentas para ampliar ou reduzir uma parte do gráfico.
- Utilizar o menu F5 **Math** da barra de ferramentas para calcular derivadas, tangentes, etc. Para os gráficos de equações paramétricas, alguns itens do menu não estão disponíveis.

Diferenças entre a representação de equações paramétricas e de funções

Este módulo pressupõe o conhecimento do procedimento de representação gráfica das funções **y(x)** descrito no módulo *Representação gráfica de funções básicas*. Esta secção descreve as diferenças relativas às equações paramétricas.

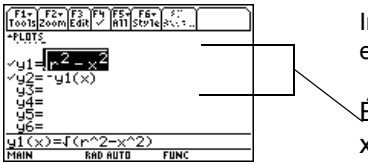
Definição do modo Graph

Utilize MODE para estabelecer a definição de **Graph = PARAMETRIC**, antes de definir as equações ou as variáveis Window. É possível introduzir informações relativas exclusivamente ao modo **Graph** actual no Y= Editor e Window Editor.

Definição de equações paramétricas no Y= Editor

Para representar uma equação paramétrica graficamente, é preciso definir ambas as componentes x e y; caso contrário, a equação não pode ser representada. (Todavia, é

possível utilizar uma única componente para gerar uma tabela automática como descrito no módulo *Tabelas*.)



Introduza as componentes x e y em linhas separadas.

É possível definir de xt1(t) a xt99(t) e de yt1(t) a yt99(t).

Tenha atenção à uma multiplicação implícita com **t**. Por exemplo:

Introduza:	Em vez de:	Motivo:
t∗cos(60)	tcos(60)	**tcos** é interpretado como a função definida pelo utilizador denominada **tcos** e não como uma multiplicação implícita. Na maioria dos casos, isto tem a ver com uma função que não existe; por isso, a TI-89 Titanium / Voyage™ 200 Graphing Calculator mostra simplesmente o nome da função e não um número.

Nota: utilize **t** para verificar se a multiplicação implícita é válida para a situação. É possível utilizar o comando **Define** (consulte o módulo *Referência técnica*) no ecrã principal para definir funções e equações em qualquer modo de representação gráfica, independentemente do modo actual.

O Y= Editor mantém uma lista de funções independente para cada estado do modo **Graph**. Observe exemplo seguinte:

- Defina um conjunto de funções **y(x)** no modo de representação gráfica FUNCTION. Em seguida, mude para o modo PARAMETRIC e defina um conjunto de componentes x e y.
- Quando voltar ao modo FUNCTION, as funções **y(x)** continuam definidas no **Y= Editor**. Quando voltar para o modo PARAMETRIC, a definição das componentes x e y continuam definidas.

Selecção de equações paramétricas

Para representar uma equação paramétrica graficamente, seleccione o componente x ou y, ou ambos. Quando introduzir ou editar um componente, será automaticamente seleccionado.

A selecção separada de x e y pode ser útil para as tabelas como descrito no módulo *Tabelas*. Várias equações paramétricas permitem seleccionar e comparar todas as componentes x e y.

Selecção do estilo de apresentação

É possível definir o estilo de apresentação para o componente x ou y.
Por exemplo: se definir o componente x em **Dot**, a TI-89 Titanium / Voyage™ 200 define automaticamente o componente y em **Dot**.

Nota: utilize os efeitos relativos à trajectória de um projéctil: **Animate** e **Path**.

Os estilos **Above** e **Below** não estão disponíveis para as equações paramétricas e estão inactivos no menu **Style** da barra de ferramentas do Y= Editor.

Variáveis Window

O Window Editor mantém um conjunto independente de variáveis Window para cada definição do modo **Graph** (da mesma forma que o Y= Editor mantém listas de funções independentes). Os gráficos paramétricos utilizam as seguintes variáveis Window:

Nota: é possível utilizar um **tstep** negativo. Neste caso, **tmin** deve ser superior a **tmax**.

Variável	Descrição
tmin, **tmax**	Valores máximo e mínimo de **t** que serão calculados.
tstep	Incremento do valor **t**. As equações paramétricas são calculadas de acordo com os valores de: x(tmin) y(tmin) x(tmin+tstep) y(tmin+tstep) x(tmin+2(tstep)) y(tmin+2(tstep)) ... que não ultrapasse ... ... que não ultrapasse ... x(tmax) y(tmax)
xmin, **xmax**, **ymin**, **ymax**	Extremos da janela de visualização.
xscl, **yscl**	Distância entre as marcas nos eixos x e y.

Os valores predefinidos (definidos quando seleccionar **6:ZoomStd** no menu F2 **Zoom** da barra de ferramentas) são:

tmin = 0		xmin = ⁻10.	ymin = ⁻10.
tmax = 2π	(6.2831853... radianos or 360 graus)	xmax = 10.	ymax = 10.
tstep =π/24	(.1308996... radianos or 7.5 graus)	xscl = 1.	yscl = 1.

Pode ser necessário modificar os valores predefinidos para as variáveis **t** (**tmin**, **tmax**, **tstep**), para garantir a representação de um número suficiente de pontos.

Exploração de um gráfico

Da mesma forma que ocorre com a representação gráfica das funções, é possível explorar um gráfico com as ferramentas seguintes.

Sugestão: é possível calcular **x(t)** e **y(t)** durante uma operação de traçado, digitando o valor de **t** e premindo [ENTER]; é possível utilizar QuickCenter em qualquer momento durante um traçado, mesmo se o cursor ainda estiver no ecrã.

Ferramenta	Para gráficos de equações paramétricas:
Cursor de movimento livre	Funciona da mesma forma que nos gráficos de funções.
[F2] **Zoom**	Funciona da mesma forma que nos gráficos de funções, excepto: • Apenas afecta as variáveis Window **x** (**xmin**, **xmax**, **xscl**) e **y** (**ymin**, **ymax**, **yscl**). • Não afecta as variáveis Window **t** (**tmin**, **tmax**, **tstep**) a não ser que seleccione **6:ZoomStd** (que define **tmin = 0**, **tmax = 2**π, e **tstep =** π**/24**).

Ferramenta	Para gráficos de equações paramétricas:
F3 **Trace**	Permite deslocar o cursor ao longo de um gráfico com um incremento **tstep** de cada vez. • Quando iniciar uma operação de traçado, o cursor posiciona-se na primeira equação paramétrica seleccionada em **tmin**. • QuickCenter é válido para todas as direcções. Se deslocar o cursor para uma área não contida no ecrã (para cima ou para baixo, para a esquerda ou para a direita), prima ENTER para centralizar a janela de visualização na posição do cursor. • O enquadramento automático não está disponível. Se deslocar o cursor para além do limite direito ou esquerdo do ecrã, a TI-89 Titanium / Voyage™ 200 não enquadrará automaticamente a janela de visualização; no entanto, é possível utilizar QuickCenter.
F5 **Math**	Para os gráficos de equações paramétricas, estão disponíveis apenas as opções: **1:Value**, **6:Derivatives**, **9:Distance**, **A:Tangent**, e **B:Arc**. Estas ferramentas baseiam-se nos valores de **t**. Por exemplo: • **1:Value** mostra os valores de x e y para um valor **t** especificado. • **6:Derivatives** determina **dy/dx**, **dy/dt**, ou **dx/dt** num ponto definido para o valor **t** especificado.

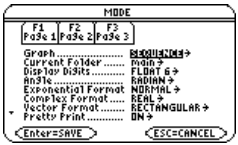
Representação gráfica de sucessões

Descrição das etapas para a representação gráfica de sucessões

Para representar sucessões graficamente, efectue o mesmo procedimento utilizado para funções **y(x)** como descrito no módulo *Representação gráfica de funções básicas*. Quaisquer diferenças são descritas nas páginas seguintes.

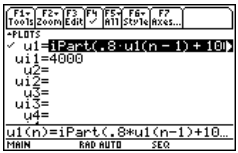
Representação gráfica de sucessões

1. Defina o modo **Graph** (MODE) para **SEQUENCE**. Defina também o modo **Angle**, se necessário.



2. Defina as sucessões e, se necessário, os valores iniciais no Y= Editor (◆ [Y=]).
3. Seleccione (F4) que sucessões definidas devem ser representadas graficamente. Não seleccione valores iniciais.

   Sugestão: para desactivar um gráfico estatístico, prima F5 **5** ou utilize F4.



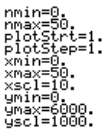
4. Defina o estilo de apresentação para uma sucessão.

   [2nd] [F6]
   [F6]

   Para as sucessões, o estilo predefinido é **Square**.

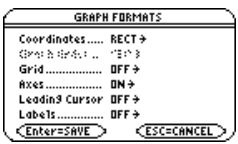
5. Defina a janela de visualização ([◆] [WINDOW]).

   [F2] **Zoom** muda também a janela de visualização.

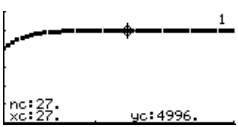


6. Mude o formato gráfico, se necessário.

   [F1] **9**
   — ou —
   [◆] [|]
   [◆] **F**



7. Represente as sucessões seleccionadas graficamente([◆] [GRAPH]).



Exploração do gráfico

No ecrã Graph, pode:

- Mostrar as coordenadas de qualquer pixel com o cursor de movimento livre ou as coordenadas de um ponto traçado pelo traçado de uma sucessão.

- Utilizar o menu F2 **Zoom** da barra de ferramentas para ampliar ou reduzir uma parte do gráfico.
- Utilizar o menu F5 **Math** da barra de ferramentas para calcular uma sucessão. Apenas **1:Value** fica disponível para sucessões.
- Representar as sucessões sobre os eixos **Time** (padrão), **Web** ou **Custom**.

Nota: é possível também calcular uma sucessão durante o traçado. Basta digitar o valor de n no teclado.

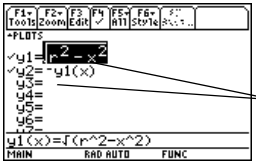
Diferenças entre a representação gráfica de funções e de sucessões

Este módulo assume que o leitor está familiarizado com a representação gráfica de funções **y(x)** como descrito no módulo *Representação gráfica de funções básicas*. Esta secção descreve as diferenças válidas para as sucessões.

Definição do modo Graph

Utilize MODE para configurar **Graph = SEQUENCE** antes de definir sucessões ou definir as variáveis Window. No Y= Editor e Window Editor, é possível introduzir informações apenas no modo Graph actual.

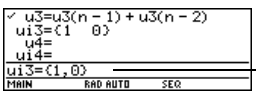
Definição de sucessões no Y= Editor



É possível definir sucessões de u1(n) a u99(n).

Utilize ui apenas em sucessões recursivas, que precisam de um ou mais valores iniciais.

Se uma sucessão precisar de mais de um valor inicial, introduza-os na forma de lista entre chavetas { }, com os elementos separados por vírgulas. É preciso utilizar uma lista para introduzir dois ou mais valores iniciais.



Introduza {1,0} mesmo que {1 0} apareça na lista de sucessões.

Se uma sucessão precisar de um valor inicial, mas este não for introduzido, aparece uma mensagem de erro durante a representação.

No Y= Editor, Axes permite seleccionar os eixos a utilizar na representação gráfica das sucessões. Opcionalmente, apenas para as sucessões, é possível seleccionar eixos diferentes para o gráfico. TIME é o eixo predefinido.

Eixos	Descrição
TIME	Representa n no eixo x e **u(n)** no eixo y.
WEB	Representa **u(n−1)** no eixo x e **u(n)** no eixo y.
CUSTOM	Permite seleccionar os eixos x e y.

O Y= Editor mantém uma lista de funções independentes para cada configuração do modo Graph. Por exemplo, suponha:

- No modo de representação gráfica FUNCTION, um conjunto de funções **y(x)** é definido. Pode alterar para o modo de representação gráfica SEQUENCE e definir um conjunto de sucessões **u(n)**.
- Quando voltar para o modo de representação gráfica FUNCTION, as suas funções **y(x)** ainda estão definidas no Y= Editor. Quando voltar ao modo de representação gráfica SEQUENCE, as suas sucessões **u(n)** ainda estão definidas.

Nota: é possível utilizar o comando **Define** no ecrã principal (consulte o módulo *Referência técnica*) para definir funções e equações em todos os modos de representação gráfica, independentemente do modo actual.

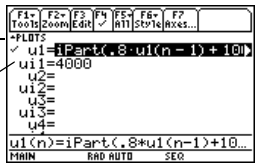
Selecção de sequências

Com os eixos TIME e WEB, a TI-89 Titanium / Voyage™ 200 representa graficamente apenas as sucessões seleccionadas. Se introduzir alguma sucessão que necessite de um valor inicial, é preciso introduzir o valor ui correspondente.

Nota: com os eixos TIME e CUSTOM, todas as sucessões definidas são calculadas, mesmo se não estiverem traçadas.

É possível seleccionar uma sucessão.

Não é possível seleccionar o seu valor inicial.



Com os eixos CUSTOM, uma sucessão personalizada é representada graficamente, independentemente de estar seleccionada.

Selecção do estilo de apresentação

Apenas os estilos **Line**, **Dot**, **Square** e **Thick** estão disponíveis para gráficos de sucessões. **Dot** e **Square** marcam apenas valores inteiros (em incrementos de **plotstep**) nos quais a sucessão é representada.

Variáveis Window

O Window Editor conserva um conjunto independente de variáveis Window para cada definição de modo Graph (como o Y= Editor mantém listas de função independentes). Os gráficos de sucessão utilizam as seguintes variáveis Window.

Variável	Descrição
nmin, nmax	Valores mínimo e máximo de n. As sucessões são calculadas em: u(nmin) u(nmin+1) u(nmin+2) ... que não ultrapasse … u(nmax)
plotstrt	Número do primeiro termo que será traçado (dependendo de plotstep). Por exemplo, para começar a representar com o segundo termo na sucessão, defina **plotstrt = 2**. O primeiro termo será calculado em **nmin**, mas não será traçado.

Variável	Descrição
plotstep	Valor do incremento de n para representação gráfica apenas. Este valor não afecta a forma como a sucessão é calculada, apenas os pontos que serão traçados. Por exemplo, suponha que **plotstep = 2**. A sucessão é calculada para cada inteiro consecutivo, mas é traçada apenas a cada dois inteiros.
xmin, **xmax**, **ymin**, **ymax**	Limites da janela de visualização.
xscl, **yscl**	Distância entre as marcas nos eixos x e y.

Nota: nmin e **nmax** precisam ser inteiros positivos, apesar de nmin poder ser zero ; **nmin**, **nmax**, **plotstrt** e **plotstep** precisam ser números inteiros ≥ 1. Se não forem números inteiros, serão arredondados para inteiros.

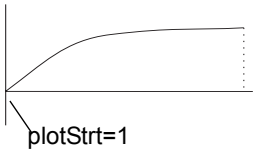
Os valores predefinidos (definidos quando seleccionar **6:ZoomStd** no menu F2 **Zoom** da barra de ferramentas) são:

nmin = 1.	xmin = ⁻10.	ymin = ⁻10.
nmax = 10.	xmax = 10.	ymax = 10.
plotstrt = 1.	xscl = 1.	yscl = 1.
plotstep = 1.		

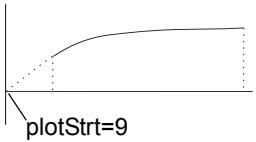
Pode ser necessário alterar os valores predefinidos das variáveis n e plot para garantir que seja uma quantidade suficiente de pontos.

Para entender como **plotstrt** afecta um gráfico, veja os exemplos seguintes de uma sucessãorecursiva.

Este gráfico é traçado a partir do primeiro termo.



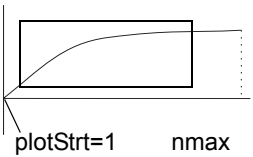
Este gráfico é traçado a partir do nono termo.



Nota: estes dois gráficos utilizam as mesmas variáveis Window, excepto **plotstrt**.

Com os eixos TIME (de Axes no Y= Editor), pode definir **plotstrt = 1** e ainda traçar apenas uma parte seleccionada da sucessão. É suficiente definir apenas uma janela de visualização que mostre a área do plano da coordenada que deseja visualizar.

Pode definir:



- **xmin** = primeiro valor de n a ser traçado
- **xmax** = **nmax** (apesar de outros valores poderem ser utilizados)
- **ymin** e **ymax** = valores esperados para a sucessão

Modificação do formato gráfico

O formato de Graph Order não está disponível.

- Com os eixos TIME ou CUSTOM, a representação de várias sucessões é traçada sempre simultaneamente.
- Com os eixos WEB, a representação de várias sucessões é traçada sempre sequencialmente.

Exploração de um gráfico

Como na representação de função, pode utilizar as seguintes ferramentas para explorar um gráfico. As coordenadas apresentadas aparecem na forma polar ou rectangular de acordo com o formato de gráfico definido.

Ferramenta	Para gráficos de sucessão:
Cursor de movimento livre	Funciona da mesma forma que em gráficos de função.
F2 **Zoom**	Funciona da mesma forma que em gráficos de função. • Apenas as variáveis Window **x** (**xmin**, **xmax**, **xscl**) e **y** (**ymin**, **ymax**, **yscl**) são afectadas. • As variáveis Window **n** e plot (**nmin**, **nmax**, **plotstrt**, **plotstep**) não são afectadas, a menos que **6:ZoomStd** seja seleccionado (o que define todas as variáveis Window com os seus valores predefinidos).

Ferramenta	Para gráficos de sucessão:
F3 **Trace**	De acordo com o eixo utilizado – TIME, CUSTOM ou WEB – Trace funciona de forma bem diferente. • Com os eixos TIME ou CUSTOM, é possível mover o cursor um plotstep de cada vez na sucessão. Para mover cerca de dez pontos traçados de uma vez, prima 2nd ▶ ou 2nd ◀. - Quando a operação de traçado é iniciada, o cursor posiciona-se sobre a primeira sucessão seleccionada, no número do termo especificado em **plotstrt**, mesmo se estiver fora da janela de visualização. - O QuickCenter é válido para todas as direcções. Se o cursor for movido para fora do ecrã (para a parte superior, inferior, esquerda ou direita), prima ENTER para centralizar a janela de visualização na posição do cursor. • Com os eixos WEB, o cursor de traçado segue a rede, não a sucessão. (Consulte “Utilização dos desenhos da rede” na página 134.)
F5 **Math**	Apenas **1:Value** está disponível para gráficos de sucessão. • Com os eixos TIME e WEB, o valor **u(n)** (representado por **yc**) aparece para um valor n especificado. • Com eixos CUSTOM, os valores que correspondem a x e y dependem dos eixos escolhidos.

É possível calcular uma sucessão durante o traçado, digitando um valor para n e premindo ENTER. É possível utilizar o QuickCenter durante um traçado, mesmo se o cursor ainda estiver no ecrã.

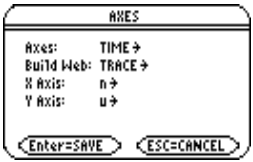
Definição de eixos para gráficos de tempo, de rede ou personalizados

Pode seleccionar eixos de tipos diferentes apenas para gráficos de sucessões. Exemplos destes tipos são apresentados posteriormente neste módulo.

Visualização da caixa de diálogo AXES

No Y= Editor, Axes:

- Dependendo da definição actual de Axes, algumas opções podem estar apagadas.
- Para sair sem fazer nenhuma alteração, prima [ESC].



Opção	Descrição
Axes	TIME — Traça **u(n)** no eixo y e n no eixo x. WEB — Traça **u(n)** no eixo y e **u(n–1)** no eixo x. CUSTOM — Permite seleccionar os eixos x e y.
Build Web	Activa apenas quando Axes = WEB, isto especifica se uma rede é traçada manualmente (TRACE) ou automaticamente (AUTO).
X Axis e **Y Axis**	Activa apenas quando Axes = CUSTOM. Permite seleccionar o valor ou a sucessão a traçar nos eixos x e y.

Para alterar uma definição qualquer, utilize o mesmo procedimento utilizado para alterar outras caixas de diálogo, como a caixa de diálogo MODE.

Utilização de gráficos de rede

Um gráfico de rede compara graficamente **u(n)** e **u(n−1)**, o que permite estudar o comportamento a longo prazo de uma sucessão recursiva. Os exemplos nesta secção ilustram também como o valor inicial pode afectar o comportamento de uma sucessão.

Funções válidas para gráficos de rede

Uma sucessão precisa satisfazer os seguintes critérios; caso contrário, não será representada de forma adequada nos eixos WEB. A sucessão:

- Precisa ser recursiva apenas com um nível de recursão; **u(n−1)**, não **u(n−2)**.
- Não pode fazer referência directa a n.
- Não pode fazer referência a outras sucessões.

Quando aparecer o ecrã Graph

Depois de seleccionar os eixos WEB e ver o ecrã Graph, a TI-89 Titanium / Voyage™ 200:

- Desenha uma recta de referência **y=x**.
- Representa as definições de sucessões seleccionadas como funções graficamente, com **u(n−1)** a ser a variável independente, convertendo de forma eficaz uma sucessão recursiva numa forma não recursiva para a representar graficamente.

Por exemplo, considere a sucessão u1(n) = $\sqrt{5 - u1(n-1)}$ e um valor inicial de **ui1=1**. A TI-89 Titanium / Voyage™ 200 representa a recta de referência **y=x** graficamente e, em seguida, y = $y = \sqrt{5 - x}$

Desenho da rede

Após a seqüência ser traçada, a rede pode ser exibida manual ou automaticamente, dependendo de como **Build Web** está definido na caixa de diálogo AXES.

Se Build Web =	A rede:
TRACE	Não aparece até premir F3. Em seguida, a rede é desenhada passo a passo à medida que o cursor de traçado se move (é necessário haver um valor inicial antes de utilizar Trace). **Nota:** com os eixos WEB, não é possível deslocar o cursor de traçado ao longo da sucessão como pode ser feito com outros modos de representação gráfica.
AUTO	É desenhada automaticamente. Pode premir F3 para traçar a rede e mostrar as suas coordenadas.

A rede:

1. Começa no eixo x no valor inicial ui (quando **plotstrt = 1**).
2. Move-se verticalmente (para cima ou para baixo) para a sucessão.
3. Move-se horizontalmente para a recta de referência **y=x**.
4. Repete este movimento vertical e horizontal até que **n=nmax**.

Nota: a rede começa em **plotstrt**. O valor de n é incrementado de 1 cada vez que a rede se move na sucessão (**plotstep** é ignorado).

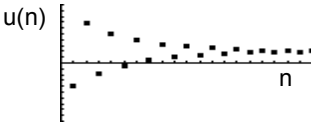
Exemplo: Convergência

1. No Y= Editor ([◆] [Y=]), defina u1(n) = ⁻ .8u1(n−1) + 3.6. Defina o valor inicial ui1 = ⁻4.
2. Defina **Axes = TIME**.
3. No Window Editor ([◆] [WINDOW]), defina as variáveis Window.

nmin=1	xmin=0	ymin=⁻10
nmax=25	xmax=25	ymax=10
plotstrt=1	xscl=1	yscl=1
plotstep=1		

4. Represente a sucessão graficamente ([◆] [GRAPH]).

   B Uma sucessão utiliza o estilo de apresentação **Square** por predefinição.



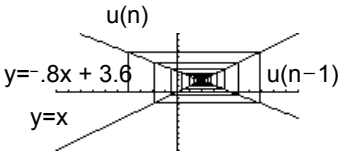
5. No Y= Editor, defina **Axes = WEB** e **Build Web = AUTO**.
6. No Window Editor ([◆] [WINDOW]), altere as variáveis Window.

nmin=1	xmin= ⁻10	ymin=⁻10
nmax=25	xmax=10	ymax=10
plotstrt=1	xscl=1	yscl=1
plotstep=1		

7. Volte a representar o gráfico da sucessão.

   As representações de rede aparecem sempre como rectas, independentemente do estilo de apresentação seleccionado.

   Sugestão: é possível mover o cursor para um valor n específico durante uma representação gráfica, digitando o valor e premindo [ENTER].



8. Prima [F3]. À medida que prime ▶, o cursor de traçado segue a rede. O ecrã mostra as coordenadas do cursor nc, xc e yc (onde xc e yc representam **u(n−1)** e **u(n)**).

À medida que traça valores maiores de nc, é possível ver xc e yc a aproximarem-se do ponto de convergência.

Sugestão: quando o valor nc muda, o cursor é posicionado na sucessão. Quando premir ▶, nc não varia, mas o cursor é posicionado sobre a recta de referência **y=x**.

Exemplo: Divergência

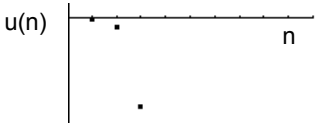
1. No Y= Editor ([◆] [Y=]), defina u1(n) = 3.2u1(n−1) − .8(u1(n−1))^2. Defina o valor inicial ui1 = 4.45.
2. Defina **Axes = TIME**.

3. No Window Editor ([◆] [WINDOW]), defina as variáveis Window.

nmin=0	xmin=0	ymin=⁻75
nmax=10	xmax=10	ymax=10
plotstrt=1	xscl=1	yscl=1
plotstep=1		

4. Represente a sucessão graficamente ([◆] [GRAPH]).

   Como a sucessão diverge rapidamente para valores negativos elevados, só são traçados alguns pontos.

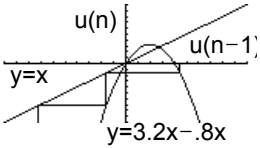


5. No Y= Editor, defina **Axes = WEB** e **Build Web = AUTO**.
6. No Window Editor ([◆] [WINDOW]), mude as variáveis Window.

nmin=0	xmin=⁻10	ymin=⁻10
nmax=10	xmax=10	ymax=10
plotstrt=1	xscl=1	yscl=1
plotstep=1		

7. Represente a sucessão graficamente de novo.

   O gráfico de rede mostra a rápida divergência da sucessão para valores negativos elevados.



Exemplo: Oscilação

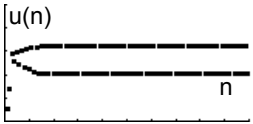
Este exemplo ilustra como o valor inicial pode afectar uma sucessão.

1. No Y= Editor ([◆] [Y=]), utilize a mesma sucessão definida no exemplo anterior: $u1(n) = 3.2u1(n-1) - .8(u1(n-1))^2$. Defina como valor inicial ui1 = 0.5.
2. Defina **Axes = TIME**.
3. No Window Editor ([◆] [WINDOW]), defina as variáveis Window.

nmin=1	xmin=0	ymin=0
nmax=100	xmax=100	ymax=5
plotstrt=1	xscl=10	yscl=1
plotstep=1		

4. Represente a sucessão graficamente ([◆] [GRAPH]).

   Nota: compare este gráfico com o exemplo de divergência. Esta é a mesma sucessão com um valor inicial diferente.

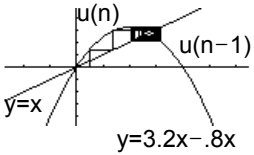


5. No Y= Editor, defina **Axes = WEB** e **Build Web = AUTO**.
6. No Window Editor ([◆] [WINDOW]), altere as variáveis Window.

nmin=1	xmin=2.68	ymin=4.7
nmax=100	xmax=6.47	ymax=47
plotstrt=1	xscl=1	yscl=1
plotstep=1		

7. Volte a representar o gráfico da sucessão.

   Nota: a rede move-se para uma órbita que oscila entre dois pontos estáveis.

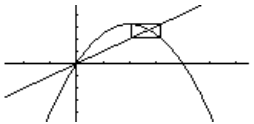


8. Prima F3. Em seguida, utiliza ▶ para traçar a rede.

   À medida que traça os valores maiores de nc, observe que xc e yc oscilam entre 2.05218 e 3.19782.

9. No Window Editor, defina **plotstrt=50**. Volte a representar a sequência graficamente.

   Nota: quando começar o gráfico de rede num termo posterior, a órbita de oscilação estável aparece mais claramente.



Utilização de gráficos personalizados

Os eixos CUSTOM proporcionam grande flexibilidade à representação gráfica de sucessões e são particularmente indicados para mostrar relações entre sucessões, como ilustrado no exemplo seguinte.

Exemplo: modelo predador-presa

Utilize o modelo predador-presa de biologia para determinar o número de coelhos e raposas que mantém a população em equilíbrio numa determinada região.

R = Número de coelhos

M = Taxa de crescimento de coelhos se não houver raposas (use .05)

K = Taxa de raposas que matam coelhos (use .001)

W = Número de raposas

G = Taxa de crescimento de raposas se houver coelhos (utilize .0002)

D = Taxa de mortalidade de raposas se não houver coelhos (utilize .03)

Rn = $R_{n-1}\ (1 + M - K\ W_{n-1})$

Wn = $W_{n-1}\ (1 + G\ R_{n-1} - D)$

1. No Y= Editor ([◆] [Y=]), defina as sucessões e os valores iniciais para $\mathbf{R}_n$ e $\mathbf{W}_n$.

   u1(n) = u1(n−1)∗ (1 + .05 −.001 ∗ u2(n−1))
   ui1 = 200
   u2(n) = u2(n−1) ∗ (1 + .0002 ∗ u1(n−1) −.03)
   ui2 = 50

   Nota: assuma que há inicialmente 200 coelhos e 50 raposas.

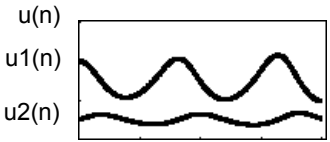
2. Defina **Axes = TIME**.

3. No Window Editor ([◆] [WINDOW]), defina as variáveis Window.

nmin=0	xmin=0	ymin=0
nmax=400	xmax=400	ymax=300
plotstrt=1	xscl=100	yscl=100
plotstep=1		

4. Represente a sucessão graficamente ([◆] [GRAPH]).

   Nota: utilize [F3] para representar individualmente o número de coelhos **u1(n)** e raposas **u2(n)** com o passar do tempo **(n)**.

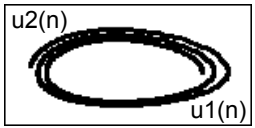


5. No Y= Editor, defina **Axes = CUSTOM**, **X Axis = u1** e **Y Axis = u2**.
6. No Window Editor ([◆] [WINDOW]), altere as variáveis Window.

nmin=0	xmin=84	ymin=25
nmax=400	xmax=237	ymax=75
plotstrt=1	xscl=50	yscl=10
plotstep=1		

7. Volte a representar a sucessão graficamente.

   Nota: utilize [F3] para traçar o número de coelhos (xc) e raposas (yc) durante o ciclo de 400 gerações.



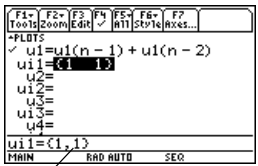
Utilização de uma sucessão para geração de tabela

As secções anteriores apresentaram a forma de representar graficamente uma sucessão. Pode utilizar uma sucessão para gerar uma tabela. Consulte o módulo *Tabelas* para obter informações detalhadas sobre tabelas.

Exemplo: Sucessão de Fibonacci

Numa seqüência de Fibonacci, os primeiros dois termos são 1 e 1. Cada termo sucessivo é a soma dos dois termos imediatamente precedentes.

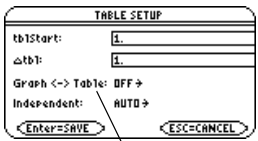
1. No Y= Editor ([◆] [Y=]), defina a sucessão e os valores iniciais como indicado.



É preciso introduzir {1,1}, apesar de {1 1} aparecer na lista de sucessão.

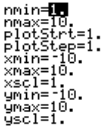
2. Defina os parâmetros da tabela ([◆] [TBLSET]) para:

   tblStart = 1
   Δtbl = 1
   Independent = AUTO

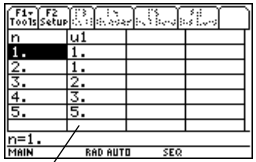


Este item fica oculto se os eixos TIME.

3. Defina as variáveis Window ([◆] [WINDOW]) de forma que **nmin** tenha o mesmo valor de **tblStart**.



4. Veja a tabela ([◆] [TABLE]).



A sucessão de Fibonacci está na coluna 2.

5. Percorra a tabela para baixo (⊖ ou [2nd] ⊖) para ver a sucessão.

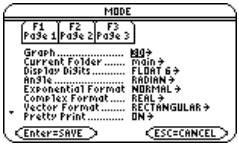
Representação gráfica de equações 3D

Descrição das etapas para a representação gráfica de equações 3D

Para representar as equações 3D graficamente, utilize o mesmo procedimento utilizado para funções **y(x)** como descrito no módulo *Representação gráfica de funções básicas*. As diferenças válidas para equações 3D são descritas nas páginas seguintes.

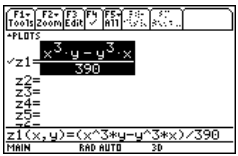
Representação gráfica de equações 3D

1. Defina o modo **Graph** (MODE) como **3D**. Defina também o modo **Angle**, se necessário.



2. Defina as equações 3D no Y= Editor (◆ [Y=]).
3. Seleccione com (F4) a equação que será representada graficamente. Pode seleccionar apenas uma equação 3D.

   Para desactivar um gráfico de dados estatísticos, prima F5 **5** ou utilize F4.



4. Defina o cubo de apresentação (◆ [WINDOW]).

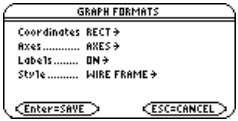
   Em gráficos tridimensionais, a janela de visualização é chamada de cubo de apresentação. [F2] **Zoom** também altera o cubo de apresentação.

```
eyeθ=20.
eyeϕ=70.
eyeψ=0.
xmin=-10.
xmax=10.
xgrid=14.
ymin=-10.
ymax=10.
ygrid=14.
zmin=-10.
zmax=10.
ncontour=5.
```

5. Mude o formato gráfico, se necessário.

   [F1] **9**
   – ou –
   [calc] ◆ [|]
   [keyboard] ◆ **F**

   Nota: para simplificar a visualização da orientação dos gráficos tridimensionais, active **Axes e Labels**.



6. Represente a equação seleccionada (◆ [GRAPH]).

   Nota: durante o cálculo e antes de mostrar o gráfico, o ecrã mostra o “percentual calculado.”

Exploração de um gráfico

No ecrã Graph, é possível:

- Traçar a equação.

- Utilizar o menu [F2] **Zoom** da barra de ferramentas para ampliar ou reduzir uma parte do gráfico. Algumas das opções do menu estão apagadas porque não estão disponíveis para gráficos 3D.
- Utilizar o menu [F5] **Math** da barra de ferramentas para calcular a equação num ponto especificado. Apenas **1:Value** está disponível para gráficos 3D.

Pode calcular também **z(x,y)** durante a representação gráfica. Digite o valor de x e prima [ENTER]; em seguida, digite o valor de y e prima [ENTER].

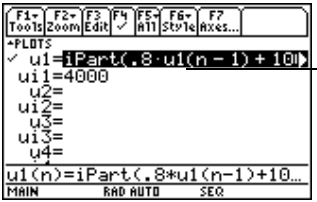
Diferenças entre as representações gráficas 3D e de funções

Este capítulo assume que o leitor já está familiarizado com a representação de funções **y(x)** como descrito no módulo *Representação gráfica de funções básicas*. Esta secção descreve as características específicas de equações 3D.

Definição do modo Graph

Utilize [MODE] para definir **Graph = 3D** antes de definir equações ou variáveis Window. Y= Editor e Window Editor permitem que informações sejam introduzidas apenas no modo Graph actual.

Definição de equações 3D no Y= Editor



Ë possível definir equações 3D de **z1(x,y)** a **z99(x,y)**.

O Y= Editor mantém uma lista independente de funções para cada definição de modo Graph. Suponha o exemplo seguinte:

- No modo de representação gráfica FUNCTION, o utilizador define um conjunto de funções **y(x)**. Pode alterar para o modo de representação 3D e, em seguida, configurar um conjunto de equações **z(x,y)**.
- Quando voltar para o modo de representação FUNCTION, as funções **y(x)** ainda estão configuradas no Y= Editor. De modo análogo, quando voltar ao modo de representação em 3D, as equações **z(x,y)** ainda estão configuradas.

Nota: é possível utilizar o comando **Define** a partir no ecrã principal (consulte o módulo *Referência técnica*) para configurar funções e equações para todos os modos de representação gráfica, independentemente do modo actual.

Selecção do estilo de apresentação

Como só é possível representar uma equação 3D de cada vez, os estilos de apresentação não estão disponíveis. No Y= Editor, o menu Style da barra de ferramentas permanece apagado.

Entretanto, para equações tridimensionais:

F1 **9**

– ou –

◆ I

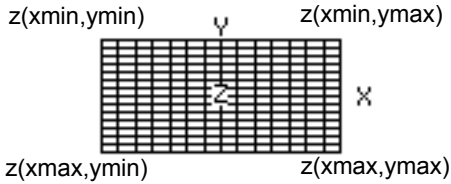
◆ **F**

podem ser utilizados para configurar o formato de estilo em WIRE FRAME ou HIDDEN SURFACE.

Variáveis Window

O Window Editor mantém um conjunto independente de variáveis Window para cada modo Graph (como Y= Editor mantém listas de funções). Os gráficos 3D utilizam as seguintes variáveis Window.

Variável	Descrição
eyeθ, eyeϕ, eyeψ	Ângulos (sempre expressos em graus) utilizados para ver o gráfico.
xmin, **xmax**, **ymin**, **ymax**, **zmin**, **zmax**	Limites do cubo de visualização.

Variável	Descrição
xgrid, **ygrid**	A distância entre **xmin** e **xmax** e entre **ymin** e **ymax** é dividida pelo número especificado de divisões da grelha. A equação **z(x,y)** é calculada em cada ponto da grelha quadriculada onde as suas rectas se interceptam. O valor do incremento ao longo de x e y é calculado como: $\text{incremento } x = \frac{xmax - xmin}{xgrid}$ $\text{incremento } y = \frac{ymax - ymin}{ygrid}$ O número de divisões da grelha quadriculada é **xgrid + 1** e **ygrid + 1**. Por exemplo, quando **xgrid = 14** e **ygrid = 14**, a grelha xy é composta por 225 (15 × 15) pontos. 
ncontour	Número de contornos uniformemente distribuídos ao longo da faixa de valores apresentados de z. (Consulte “Como são determinados os valores Z?” na página 155.)

Nota: se atribuir um valor fraccionário a **xgrid** ou **ygrid**, será arredondado para o número inteiro ≥ 1 mais próximo. O modo 3D não tem variáveis **scl** Window; assim, não é possível configurar marcas de divisão nos eixos.

Os valores predefinidos (configurados quando seleccionar **6:ZoomStd** no menu F2 **Zoom** da barra de ferramentas) são:

eyeθ = 20.	xmin = ⁻10.	ymin = ⁻10.	zmin = ⁻10.
eyeϕ = 70.	xmax = 10.	ymax = 10.	zmax = 10.
eyeψ = 0.	xgrid = 14.	ygrid = 14.	ncontour = 5.

Pode ser necessário atribuir valores maiores do que os valores predefinidos às variáveis grid (**xgrid**, **ygrid**) para garantir que sejam traçados pontos suficientes.

Nota: se atribuir valores maiores às variáveis da grelha quadriculada, a velocidade de representação do gráfico é reduzida.

Definição do formato gráfico

Os formatos Axes e Style são específicos do modo de representação gráfica 3D.

Exploração de um gráfico

Assim como na representação gráfica de funções, é possível explorar um gráfico com as seguintes ferramentas. As coordenadas aparecem na forma rectangular ou cilíndrica, conforme configurado no formato gráfico. Na representação gráfica tridimensional, as coordenadas cilíndricas aparecem quando:

F1 **9**

– ou –

[⌨] [◆] **F**

é utilizado para configurar **Coordinates = POLAR**.

Ferramenta	Para gráficos 3D:
Cursor de movimento livre	O cursor de movimento livre não está disponível.
[F2] **Zoom**	Funciona essencialmente da mesma forma que para gráficos de função, mas não se esqueça de que estão a ser utilizadas três dimensões em vez de duas. • Apenas os seguintes recursos de zoom estão disponíveis: **2:ZoomIn**, **3:ZoomOut**, **5:ZoomSqr**, **6:ZoomStd**, **A:ZoomFit**, **B:Memory**, **C:SetFactors** • Apenas as variáveis Window **x** (**xmin**, **xmax**), **y** (**ymin**, **ymax**) e **z** (**zmin**, **zmax**, **zscl**) são afectadas. • As variáveis Window **grid** (**xgrid**, **ygrid**) e **eye** (**eyeθ**, **eyeϕ**, **eyeψ**) não são afectadas, a menos que **6:ZoomStd** seja seleccionado (que restabelece o valor predefinido destas variáveis).

Ferramenta	Para gráficos 3D:
F3 **Trace**	Permite mover o cursor ao longo de uma linha da grelha quadriculada a partir de um ponto de grelha sobre a superfície tridimensional. • Quando uma representação gráfica é iniciada, o cursor aparece no ponto médio da grelha xy. • O QuickCenter está disponível. A qualquer momento durante uma representação gráfica, independentemente da posição do cursor, pode ser premido para centralizar o cubo de apresentação no cursor. • O movimento do cursor está restrito às direcções x e y. Não é possível mover o cursor além dos limites do cubo de apresentação configurados por **xmin**, **xmax**, **ymin** e **ymax**.
F5 **Math**	Apenas **1:Value** está disponível para gráficos 3D. Esta ferramenta mostra o valor z para um valor específico de x e y. Depois de seleccionar **1:Value**, digite o valor de x e prima ENTER. Em seguida, digite o valor de y e prima ENTER.

Nota: é também possível calcular **z(x,y)** durante uma representação gráfica. Digite o valor de x e prima ENTER; em seguida, digite o valor de y e prima ENTER.

Movimentação do cursor sobre uma superfície tridimensional

A forma com que o cursor se move ao longo de uma superfície tridimensional não é sempre clara. Os gráficos 3D possuem duas variáveis independentes (**x**,**y**) em vez de

uma e os eixos x e y possuem uma orientação diferente de outros modos de representação gráfica.

Como mover o cursor

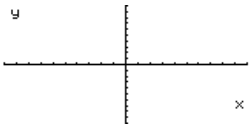
Numa superfície tridimensional, o cursor desloca-se sempre ao longo de uma recta da grelha quadriculada.

Tecla de cursor	Move o cursor para o próximo ponto da grelha na:
⊙	Direcção positiva de x
⊙	Direcção negativa de x
⊙	Direcção positiva de y
⊙	Direcção negativa de y

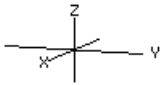
Nota: o cursor só pode ser movido dentro dos limites x e y configurados pelas variáveis Window **xmin**, **xmax**, **ymin** e **ymax**.

Apesar da simplicidade das regras, o movimento do cursor pode parecer confuso, se a orientação dos eixos não for observada.

Os eixos x e y possuem sempre a mesma orientação na representação gráfica 2D no ecrã Graph.

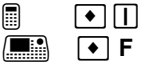


x e y possuem uma orientação diferente relativa ao ecrã Graph na representação gráfica tridimensional. Além disso, é possível girar e/ou elevar o ângulo de visualização.



eyeθ=20 eyeϕ=70 eyeψ=0
)

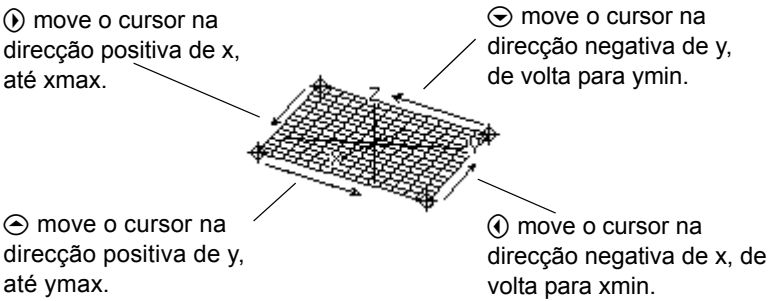
Para mostrar os eixos e os seus rótulos nos ecrãs de Y= Editor, Window Editor ou Graph, utilize:



Exemplo simples de movimentação do cursor

O gráfico seguinte mostra um plano inclinado que possui a equação z1(x,y) = ⁻(x + y) / 2. Suponha que tem de se deslocar à volta do limite apresentado.

Quando pressionar F3, o cursor de traçado aparece no meio da grelha xy. Utilize as teclas de cursor para mover o cursor para qualquer



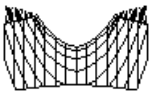
Com a apresentação e a rotulagem dos eixos, é possível entender com mais facilidade o padrão de movimento do cursor. Para a distância entre os pontos da grelha quadriculada diminuir, atribua um valor elevado às variáveis Window **xgrid** e **ygrid**.

Quando o cursor de traçado está num ponto interno do plano apresentado, move-se de um ponto da grelha para o próximo ponto ao longo de uma recta da grelha quadriculada. Não é possível movê-lo na diagonal pela grelha. Não se esqueça de que as rectas da grelha quadriculada podem não aparecer paralelas aos eixos.

Exemplo do cursor numa superfície oculta

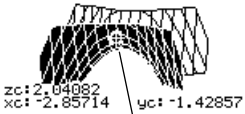
Em formas mais complexas, pode parecer que o cursor não está num ponto da grelha quadriculada. Esta é uma ilusão de óptica que ocorre quando o cursor está numa superfície oculta.

Por exemplo, considere o parabolóide hiperbólico
$z1(x,y) = (x^2 - y^2) / 3$. O gráfico
seguinte mostra a visualização ao longo do eixo y.



```
eyeθ=90.
eyeϕ=70.
eyeψ=0.
xmin=-10.
xmax=10.
xgrid=14.
ymin=-10.
ymax=10.
ygrid=14.
zmin=-10.
zmax=10.
ncontour=5.
```

Veja agora a mesma superfície em 10° do eixo x (eyeθ = 10).



O cursor pode ser movido de forma que não pareça estar sobre um ponto da grelha quadriculada.

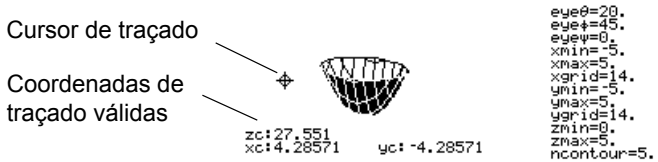
Se a parte frontal for removida, pode observar que o cursor está realmente sobre um ponto da grelha na parte traseira, antes oculta.

Nota: para separar a parte frontal da sela neste exemplo, defina **xmax=0** para mostrar apenas valores de x negativos.

Exemplo de um cursor “fora da curva”

Apesar de o cursor andar apenas numa recta da grelha, pode, muitas vezes, parecer não estar sobre a superfície tridimensional. Isto ocorre quando o eixo z é muito curto para mostrar **z(x,y)** para os valores correspondentes de x e y.

Por exemplo, suponha que o parabolóide $z(x,y) = x^2 + .5y^2$ seja traçado com as variáveis Window indicadas. É possível mover facilmente o cursor para uma posição qualquer, tal como:



Apesar de o cursor estar a deslocar-se sobre o parabolóide, parece estar fora da superfície porque as coordenadas de traçado:

- **xc** e **yc** estão dentro do cubo de apresentação.

  – mas –

- **zc** está fora do cubo de apresentação.

Nota: o QuickCenter permite centralizar o cubo de apresentação na posição do cursor. Basta premir [ENTER].

Quando **zc** estiver fora do limite z do cubo de apresentação, o cursor aparece em **zmin** ou **zmax** (apesar de o ecrã apresentar as coordenadas de traçado correctas).

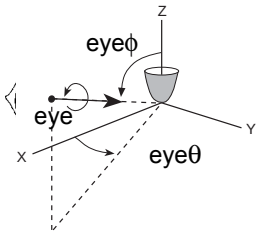
Rotação e/ou elevação com o ângulo de visualização

No modo de representação gráfica 3D, as variáveis de Window **eye**θ e **eye**ϕ permitem configurar os ângulos de visualização que determinam a linha de visão. Um nova variável Window, **eye**ψ, permite girar o gráfico à volta da linha de visão.

Como o ângulo de visualização é medido

O ângulo de visualização possui três componentes:

- **eye**θ — ângulo em graus a partir do eixo x positivo.
- **eye**ϕ — ângulo em graus a partir do eixo z positivo.
- **eye**ψ — ângulo em graus em que o gráfico é girado para a esquerda às volta da linha de visão configurada por **eye**θ e **eye**ϕ.
- Não introduza o símbolo °. Por exemplo, digite 20, 70 e 0.




```
eyeθ=20.
eyeϕ=70.
eyeψ=0.
xmin=-10.
xmax=10.
xgrid=14.
ymin=-10.
ymax=10.
ygrid=14.
zmin=-10.
zmax=10.
ncontour=5.
```

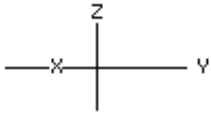
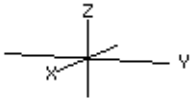
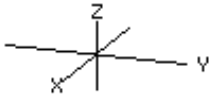
Não introduza o símbolo °.
Por exemplo, digite 20, 70

Nota: quando **eye**ψ**=0**, o eixo z estiver vertical no ecrã. Quando **eye**ψ**=90**, o eixo z gira 90 para a esquerda e fica na horizontal.

No Window Editor ([◆] [WINDOW]), introduza sempre **eye**θ, **eye**ϕ e **eye**ψ em graus, independentemente do modo de ângulo actual.

Consequência da mudança de eyeθ theta

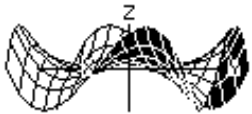
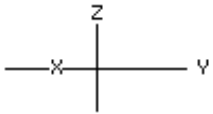
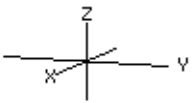
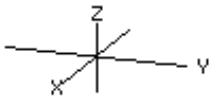
A apresentação do ecrã Graph está sempre orientada de acordo com o ângulo de visualização. Deste ponto de vista, é possível mudar **eye**θ para girar o ângulo de visualização à volta do eixo z.

z1(x,y) = ($x^3y - y^3x$) / 390		**Neste exemplo eyeθ = 20**
	eyeϕ = 90	
	eyeϕ = 70	
	eyeϕ = 50	

Nota: este exemplo incrementa **eye**θ de 30 em 30.

Consequências da mudança de eyeϕ phi

Quando alterar **eyeϕ**, o ângulo de exibição pode ser elevado acima do plano xy. Se **90 < eyeϕ < 270**, o ângulo de visualização está abaixo do plano xy.

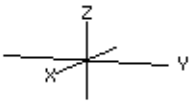
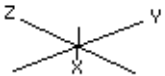
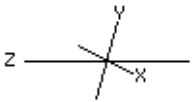
z1(x,y) = ($x^3y - y^3x$) / 390		**Neste exemplo eyeθ = 20**
	eyeϕ = 90	
	eyeϕ = 70	
	eyeϕ = 50	

Nota: este exemplo começa no plano xy (**eyeϕ = 90**) e **eyeϕ** é subtraído de 20 para elevar o ângulo de visualização.

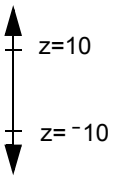
Consequências da mudança de eyeψ psi

A visualização do ecrã Graph está sempre orientada ao longo dos ângulos de visualização configurados por **eyeθ** e **eyeϕ**. É possível alterar **eyeψ** para o gráfico girar à volta da linha de visão.

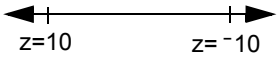
Nota: os eixos são expandidos ou contraídos durante a rotação para ajustar a largura e a altura do ecrã. Este procedimento provoca uma pequena distorção como mostrado no exemplo.

z1(x,y)=(x^3y-y^3x) / 390		**Neste exemplo, eyeθ=20 e eyeϕ=70**
	eyeψ = 0	
	eyeψ = 45	
	eyeψ = 90	

Quando **eyeψ=0**, o eixo z atinge a altura do ecrã.



Quando **eyeψ=90**, o eixo z atinge a largura do ecrã.



Conforme o eixo z gira 90°, o seu intervalo (⁻10 a 10 neste exemplo) expande-se, atingindo quase duas vezes seu comprimento original. De modo análogo, os eixos x e y expandem-se ou contraem-se.

No ecrã principal ou num programa

Os valores **eye** são guardados nas variáveis de sistema **eyeθ**, **eyeϕ** e **eyeψ**. É possível aceder a essas variáveis quando for necessário.

Para digitar ϕ ou ψ, prima [◆] [(] [alpha] **[F]** ou [◆] [(] [Y]. É também possível premir [2nd] [CHAR] e utilizar o menu Grego.

Para digitar ϕ ou ψ, prima [2nd] **G F** ou [2nd] **G Y**. É também possível premir [2nd] [CHAR] e utilizar o menu Grego.

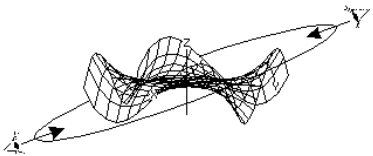
Animação interactiva de um gráfico 3D

Depois de traçar qualquer gráfico 3D, pode mudar o ângulo de visualização com o cursor.

Órbita de visualização

Quando utilizar ◀ e ▶ para animar um gráfico, considere como se estivesse a mover o ângulo de visualização ao longo da sua “órbita de visualização” à volta do do gráfico.

O movimento ao longo desta órbita pode fazer com que o eixo z oscile suavemente durante a animação.



Nota: a órbita de visualização afecta as variáveis eye de Window em quantidades diferentes.

Animação de um gráfico

Para:	Efectue o procedimento apresentado em seguida:
Animar o gráfico de forma incremental	Prima e solte o cursor rapidamente.
Mover ao longo da órbita de visualização:	◀ ou ▶
Mudar a elevação da órbita de visualização (aumenta ou diminui, principalmente, eyeϕ):	⊕ ou ⊖

Para:	Efectue o procedimento apresentado em seguida:
Animar o gráfico de forma contínua	Prima sem soltar o cursor durante um segundo, soltando-o em seguida. Para parar, prima [ESC], [ENTER], [ON] ou [◆] [␣] (espaço). Para interromper, prima [ESC], [ENTER], [ON], ou a barra de espaço.
Mudar entre 4 velocidades de animação (aumentar ou diminuir mudanças de incremento nas variáveis eye de Window)	Prima [+] ou [−].
Mudar o ângulo de visualização de um gráfico não animado para ver ao longo dos eixos x, y, ou z	Prima X, Y ou Z.
Voltar aos valores iniciais do ângulo eye	Prima 0 (zero).

Nota: se o gráfico aparecer na visualização expandida, volta para a visualização normal automaticamente quando premir uma tecla de cursor.

- Depois de animar o gráfico, pode parar e reiniciar a animação na mesma direcção, premindo:
  [ENTER] ou [alpha] [␣]
  [ENTER] ou barra de espaço
- Pode alternar para o próximo estilo de formato gráfico durante uma animação, premindo:
  [I]
  F

- Para ver um gráfico que mostra os ângulos de eye.

Animação de várias imagens gráficas

Também é possível animar um gráfico, guardando várias imagens gráficas e, em seguida, trocando-as (ou girando-as). Consulte “Animação de várias imagens gráficas” no módulo *Tópicos complementares de representação gráfica*. Este método fornece um maior controlo sobre os valores das variáveis de Window, particularmente **eye**ψ.

Mudança dos formatos dos eixos e de estilo

Com as predefinições, a TI-89 Titanium / Voyage™ 200 Graphing Calculator mostra as superfícies ocultas de um gráfico 3D, mas não os eixos. No entanto, o formato gráfico pode ser mudado a qualquer momento.

Apresentação da caixa de diálogo GRAPH FORMATS

No Y= Editor, no Window Editor ou no ecrã Graph:

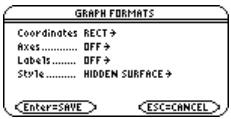
[F1] **9**

– ou –

[◆] [|]

[◆] **F**

- A caixa de diálogo mostra as definições actuais do formato gráfico.
- Para sair sem fazer alterações, prima [ESC].



Para alterar uma destas definições, utilize o mesmo procedimento utilizado para mudar outros tipos de caixas de diálogo, como a caixa de diálogo MODE.

Exemplos de definições de eixos

Para ver as definições válidas de **Axes**, realce a definição actual e prima ⊙.

$z1(x,y) = x^2+.5y^2$

- AXES — Mostra os eixos predefinidos xyz.

- BOX — Mostra os eixos da caixa tridimensional.

  Os limites da caixa são determinados pelas variáveis Window **xmin**, **xmax**, etc.

  Em muitos casos, a origem (0,0,0) está no interior da caixa, não numa extremidade. Por exemplo, se **xmin = ymin = zmin = ⁻10** e **xmax = ymax = zmax = 10**, a origem está no centro da caixa.

Nota: configurar Labels = ON é útil na apresentação de eixos 3D.

Exemplos de definições de estilo

Nota: a representação gráfica com WIRE FRAME é mais rápida e pode ser a mais adequada quando estiver a representar várias formas graficamente.

Para ver as definições válidas de Style, realce a definição actual e prima ⊙.

- WIRE FRAME — Mostra a forma tridimensional como um sólido transparente.

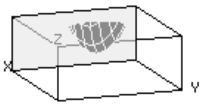
- HIDDEN SURFACES — Utiliza tons diferentes para diferenciar as duas faces de uma superfície tridimensional.

As secções posteriores deste módulo descrevem os níveis de contorno, contorno e fio e gráfico implícito.

Atenção para as possíveis ilusões de óptica

Os ângulos eye (variáveis Window **eye**θ, **eye**ϕ e **eye**ψ) utilizados na apresentação do gráfico podem causar ilusões de óptica, fazendo com que a perspectiva de um gráfico seja perdida. De uma forma geral, a maioria das ilusões de óptica ocorre quando os ângulos eye (de exibição) estão num quadrante negativo do sistema de coordenadas.

As ilusões de óptica podem ser mais frequentes com os eixos de caixas. Por exemplo, pode não estar imediatamente claro qual é a “face frontal” da caixa.

Visualização de cima do plano xy para baixo	Visualização de baixo do plano xy para cima
eyeθ = 20, eyeϕ = 55, eyeψ = 0	eyeθ = 20, eyeϕ = 120, eyeψ= 0
	

Nota: o primeiro dos dois exemplos mostra como os gráficos aparecem no ecrã. O segundo dos dois exemplos utiliza tons artificiais (não apresentados no gráfico) para indicar a face frontal da caixa.

Para minimizar o efeito de ilusões de óptica, utilize a caixa de diálogo GRAPH FORMATS para configurar Style = HIDDEN SURFACE.

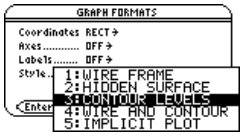
Traçados de contorno

Num traçado de contorno, uma linha é desenhada para ligar pontos adjacentes no gráfico 3D que possuem o mesmo valor de z. Esta secção aborda os estilos de formato gráfico CONTOUR LEVELS e WIRE AND CONTOUR.

Seleçcão do estilo de formato gráfico

No modo de representação gráfica 3D, defina uma equação e represente-a graficamente como faria com qualquer equação 3D, com a seguinte excepção: veja a caixa de diálogo GRAPH FORMATS, premindo [F1] **9** no Y= Editor, no Window Editor ou no ecrã Graph. Em seguida, defina:

Style = CONTOUR LEVELS
– ou –
Style = WIRE AND CONTOUR

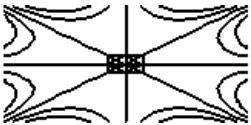
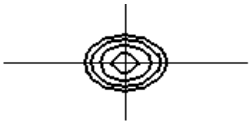
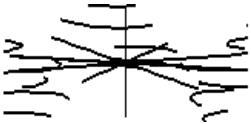
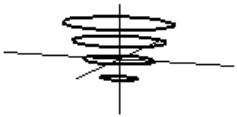
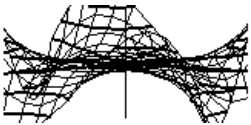


- Em CONTOUR LEVELS, só aparecem os contornos.
  - O ângulo de visualização é configurado inicialmente para que veja os contornos, olhando a partir do eixo z. Pode mudar o ângulo de visualização, se necessário.
  - O gráfico aparece na visualização expandida. Para alternar entre visualização expandida e normal, prima [×].
  - O formato Labels é configurado como OFF automaticamente.
- Em WIRE AND CONTOUR, os contornos são desenhados num modelo sem preenchimento (wire frame). O ângulo de visualização, a visualização (expandida ou normal) e o formato Labels permanecem nas definições configurações anteriores.

Nota:

- Pode alternar do estilo de formato de um gráfico para o próximo (ignorando IMPLICIT PLOT) no ecrã Graph, premindo:
  [I]
  F

- Premir:
  [calc] [|]
  [keyboard] **F**
  para seleccionar CONTOUR LEVELS não afecta o ângulo de visualização, a visualização e o formato Labels, como ocorre quando utiliza:
  [calc] [◆] [|]
  [keyboard] [◆] **F**

Estilo	**z1(x,y)=(x^{3}y–y^{3}x) / 390**	**z1(x,y)=x^2+.5y^2–5**
	Olhando a partir do eixo z	
CONTOUR LEVELS		
	Utilizando eyeθ=20, eyeϕ=70, eyeψ=0	
CONTOUR LEVELS		
WIRE AND CONTOUR		

Nota: estes exemplos utilizam os mesmos valores da variável x, y e z de Window como o cubo de apresentação **ZoomStd**. Se **ZoomStd** for utilizado, prima Z para olhar a partir

do eixo z. Não confunda os contornos com as linhas da grelha quadriculada. Os contornos são mais escuros.

Como os valores de Z são determinados?

Pode definir a variável ncontour de Window ([◆] [WINDOW]) para determinar o número de contornos que estarão distribuídos uniformemente ao longo do intervalo apresentado dos valores de z, onde:

$$\text{incremento} = \frac{\text{zmax} - \text{zmin}}{\text{ncontour} + 1}$$

```
eyeθ=20.
eyeϕ=70.
eyeψ=0.
xmin=-10.
xmax=10.
xgrid=14.
ymin=-10.
ymax=10.
ygrid=14.
zmin=-10.
zmax=10.
ncontour=5.
```

A predefinição é 5. Os valores são válidos de 0 a 20.

Os valores de z para os contornos são:

zmin + incremento
zmin + 2(incremento)
zmin + 3(incremento)
⋮
zmin + ncontour(incremento)

Se utilizar ncontour=5 e a janela de visualização comum (**zmin=⁻10** e **zmax=10**), o incremento é 3,333. Cinco contornos são desenhados para z=⁻6.666, ⁻3.333; 0; 3,333 e 6,666.

No entanto, não se esqueça de que um contorno não é desenhado para um determinado valor de z, se o gráfico 3D não estiver configurado para este valor de z.

Desenho de um contorno para os valores de Z de um ponto seleccionado interactivamente

Se um gráfico de contorno aparecer no momento, pode especificar um ponto no gráfico e desenhar um contorno para o valor de z correspondente.

1. Para ver o menu **Draw**, prima:
   [2nd] [F6]
   [F6]

2. Seleccione **7:Draw Contour**.
3. Ou:
   - Digite o valor de x do ponto e prima [ENTER], em seguida, digite o valor de y e prima [ENTER].

     – ou –
   - Mova o cursor para o ponto aplicável. (O cursor move-se ao longo das linhas da grelha quadriculada). Em seguida, prima [ENTER].

Por exemplo, suponha que o gráfico actual é $z1(x,y)=x^2+.5y^2–5$. Se especificar x=2 e y=3, é desenhado um contorno para z=3,5.

Nota: qualquer contorno existente continua no gráfico. Para remover os contornos predefinidos, vá para o Window Editor ([◆] [WINDOW]) e defina **ncontour=0**.

Desenho de contornos para valores específicos de Z

No ecrã Graph, vá para o menu Draw e seleccione **8:DrwCtour**. O ecrã principal aparece automaticamente com **DrwCtour** na linha de entrada. Em seguida, pode especificar um ou mais valores de z individualmente ou gerar uma sucessão de valores de z.

Alguns exemplos são:

DrwCtour 5	Desenha um contorno para z=5.
DrwCtour {1,2,3}	Desenha contornos para z=1, 2 e 3.
DrwCtour seq(n,n, 10,10,2)	Desenha contornos para uma sucessão de valores de z de ⁻10 a 10 em passos de 2 (⁻10, ⁻8, ⁻6, etc.).

Nota: para remover os contornos predefinidos, utilize [◆] [WINDOW] e defina **ncontour=0**.

Os contornos especificados são desenhados no gráfico 3D actual. (O contorno não será desenhado se o valor de z especificado estiver fora do cubo de apresentação ou se o gráfico 3D não estiver definido para aquele valor de z.)

Notas sobre traçados de contorno

Para um traçado de contorno:

- Pode utilizar as teclas de cursor para animar o traçado de contorno.
- Não é possível traçar ([F3]) os contornos. No entanto, pode traçar o modelo sem preenchimento (wire frame) como visto quando Style=WIRE AND CONTOUR.
- Pode demorar um pouco para calcular a equação.

- Devido ao possível longo tempo de cálculo, pode desejar experimentar primeiramente a sua equação 3D com Style=WIRE FRAME. O tempo de cálculo é muito menor. Em seguida, depois de ter os valores correctos da variável da janela, vá para a caixa de diálogo Graph Formats e defina Style=CONTOUR LEVELS ou WIRE AND CONTOUR.

  [◆] [|]

  [◆] **F**

Exemplo: contornos de uma superfície de módulo complexo

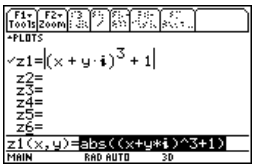
A superfície de módulo complexo dada por **z(a,b) = abs(f(a+b*i*))** mostra todas as raízes complexas de qualquer polinómio **y=f(x)**.

Exemplo

Neste exemplo, faça $f(x)=x^3+1$. Substitua a forma complexa geral $x+yi$ por x para expressar a equação de superfície complexa como $z(x,y)=abs((x+y\ i)^3+1)$.

1. Utilize [MODE] para configurar **Graph=3D**.
2. Prima [◆] [Y=] e defina a equação:

   z1(x,y)=abs((x+y∗*i*)^3+1)



3. Prima [◆] [WINDOW] e defina as variáveis de Window como mostrado.

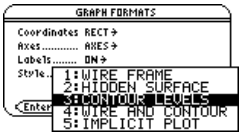
```
eyeθ=-90.
eyeϕ=0.
eyeψ=0.
xmin=-1.5
xmax=1.5
xgrid=14.
ymin=-1.5
ymax=1.5
ygrid=14.
zmin=-1.
zmax=2.
ncontour=10.
```

4. Veja a caixa de diálogo Graph Formats:

   [calculadora] [◆] [I]

   [calculadora] [◆] **F**

   Active os eixos, defina **Style = CONTOUR LEVELS** e volte ao Window Editor.



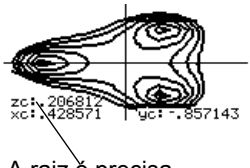
5. Prima [◆] [GRAPH] para representar a equação graficamente.

   Demorará um pouco para calcular um gráfico; portanto tenha paciência. Quando o gráfico aparecer, a superfície do módulo complexo toca o plano xy no ponto exacto das raízes complexas do polinómio:

$$-1,\ \frac{1}{2}+\frac{\sqrt{3}}{2}i,\ \text{et}\ \frac{1}{2}-\frac{\sqrt{3}}{2}i$$

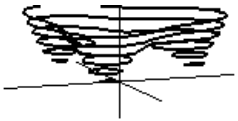
6. Prima [F3], e mova o cursor de traçado para a raiz no quarto quadrante.

   A coordenada permite estimar .428–.857i como a raiz.



A raiz é precisa quando z=0.

7. Prima [ESC]. Em seguida, utilize as teclas de cursor para animar o gráfico e vê-lo em diferentes ângulos eye.



Este exemplo mostra eyeθ=70, eyeϕ=70 e eyeψ=0.

Nota:

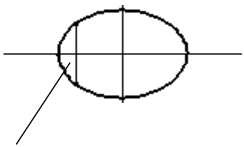
- Para obter estimativas mais precisas, aumente as variáveis **xgrid** e **ygrid** de Window. No entanto, este procedimento aumenta o tempo de cálculo do gráfico.
- Quando animar o gráfico, o ecrã muda para a visualização normal. Utilize [×] para alternar entre visualizações normal e expandida.

Traçados implícitos

Um traçado implícito é utilizado basicamente como uma forma de representar formas implícitas bidimensionais graficamente que não podem ser representadas no modo de representação gráfica de funções. Tecnicamente, um traçado implícito é um traçado de contorno 3D apenas com um único contorno desenhado para **z=0**.

Formas explícitas e implícitas

No modo de representação gráfica de função 2D, as equações possuem uma forma explícita **y=f(x)**, onde y é exclusivo para cada valor de x.



y não é único para cada x, assim não é possível representar esta função graficamente no modo

Muitas equações, entretanto, possuem uma forma implícita **f(x,y)=g(x,y)**, onde não se pode indicar explicitamente y em termos de x ou x em termos de y.

Através da utilização de traçados implícitos no modo de representação gráfica 3D, estas formas implícitas podem ser representadas graficamente sem calcular y ou x.

Disponha a forma implícita como uma equação configurada em zero.

$$f(x,y)–g(x,y)=0$$

No Y= Editor, introduza o lado não nulo da equação. Isto é válido porque um traçado implícito define automaticamente a equação como sendo igual a zero.

$$z1(x,y)=f(x,y)–g(x,y)$$

Por exemplo, dada a equação de elipse mostrada à direita, introduza a forma implícita em Y= Editor.

Se $x^2+.5y^2=30$, então $z1(x,y)=x^2+.5y^2–30$.

Nota: é possível também representar graficamente muitas formas implícitas se forem:

- Expressas como equações paramétricas.

- Divididas em funções explícitas.

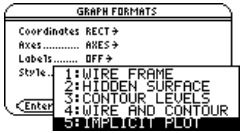
Selecção do estilo de formato gráfico

No modo de representação gráfica 3D, defina uma equação apropriada e represente-a graficamente como se faria com qualquer equação 3D, com a seguinte excepção. Veja a caixa de diálogo GRAPH FORMATS no Y= Editor, no Window Editor ou no ecrã Graph.

◆ |

◆ **F**

Em seguida, defina: **Style = IMPLICIT PLOT**



Nota: para mudar para os outros estilos de formato gráfico no ecrã Graph, prima:

|

F

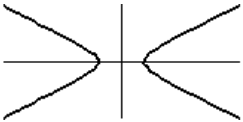
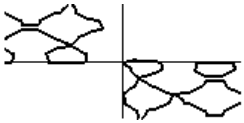
Para voltar a IMPLICIT PLOT, tem de utilizarr:

◆ |

◆ **F**

- O ângulo de visualização é inicialmente configurado para que veja o traçado, olhando a partir do eixo z. O ângulo de visualização pode ser mudado, se desejado.
- O traçado é mostrado na visualização expandida. Para alternar entre a visualização expandida e normal, prima ⊠.

- O formato Labels é configurado como OFF automaticamente.

Estilo	**$x^2-y^2=4$** **$z1(x,y)=x^2-y^2-4$**	**sin(x)+cos(y)= e(x y)** **z1(x,y)=sin(x)+cos(y)–e(x y)**
IMPLICIT PLOT		

Nota: Estes exemplos utilizam os mesmos valores de variável de Window x, y e z do cubo de apresentação **ZoomStd**. Se utilizar **ZoomStd**, prima Z para olhar a partir do eixo z.

Notas sobre traçados implícitos

Num traçado implícito:

- A variável **ncontour** de Window não afecta o traçado. Apenas o contorno **z=0** é desenhado, independentemente do valor de ncontour. O traçado apresentado mostra onde a forma implícita intercepta o plano xy.
- Pode utilizar as teclas de cursor para animar o traçado.
- Não é possível traçar (F3) o gráfico implícito. No entanto, pode traçar o gráfico sem preenchimento (wire frame) não visto da equação 3D.
- Pode demorar um pouco para calcular a equação.
- Devido ao tempo de cálculo possivelmente longo, pode desejar experimentar a sua equação 3D com Style=WIRE FRAME. O tempo de cálculo é muito menor. Em seguida, depois de ter certeza de que os valores das variáveis de Window estão

correctos, utilize:

[◆] [|]

[◆] **F**

e defina: Style=IMPLICIT PLOT.

Exemplo: traçado implícito de uma equação mais complexa

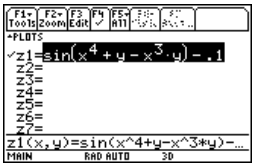
Pode utilizar o estilo de formato gráfico IMPLICIT PLOT para traçar e animar uma equação complexa que não pode ser representada graficamente de outra forma. Apesar de poder demorar mais tempo para calcular o gráfico, os resultados visuais podem justificar o tempo utilizado.

Exemplo

Represente a equação $\sin(x^4+y-x^3 y) = .1$ graficamente.

1. Utilize [MODE] para definir **Graph=3D**.
2. Prima [◆] [Y=] e defina a equação:

   z1(x,y)=sin(x^4+y–x^3y)–.1



3. Prima [◆] [WINDOW] e defina as variáveis de Window como mostrado.

```
eyeθ=-90.
eyeφ=0.
eyeψ=0.
xmin=-10.
xmax=10.
xgrid=14.
ymin=-10.
ymax=10.
ygrid=14.
zmin=-10.
zmax=10.
ncontour=5.
```

4. Prima:

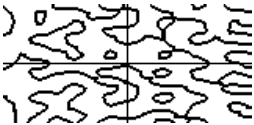
   [▦] [◆] [|]

   [▦] [◆] **F**

   active os eixos, defina **Style = IMPLICIT PLOT** e volte ao Window Editor.

5. Prima [◆] [GRAPH] para representar a equação graficamente.

   Demorará um pouco para calcular o gráfico, portanto, tenha paciência.



O gráfico mostra onde $\sin(x^4+y-x^3y) = .1$

6. Utilize as teclas de cursor para animar o gráfico e vê-lo em diferentes ângulos eye.

   Nota: para mais informações, aumente as variáveis **xgrid** e **ygrid** de Window. No entanto, este procedimento aumenta o tempo de cálculo do gráfico.

Nesta visualização expandida, este exemplo mostra eyeθ=$^-$127.85, eyeϕ=52.86 e eyeψ=$^-$18.26.

Nota: Quando animar um gráfico, o ecrã muda para visualização normal. Prima [×] para alternar entre visualizações normal e expandida.

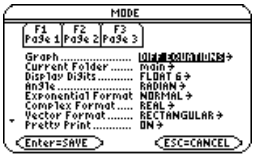
Representação gráfica de equações diferenciais

Descrição das etapas para a representação gráfica de equações diferenciais

Para representar graficamente as equações diferenciais, utilize as mesmas etapas gerais utilizadas para as funções **y(x)**, como descrito no módulo *Representação gráfica de funções básicas*. Todas as diferenças são descritas nas páginas seguintes.

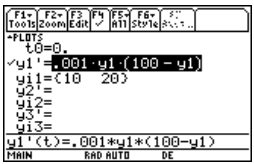
Representação gráfica de equações diferenciais

1. Defina o modo **Graph** (MODE) como **DIFF EQUATIONS**. Além disso, defina o modo **Angle** se necessário.

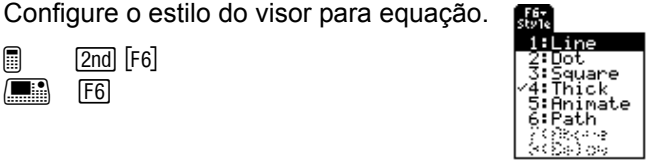


2. Defina as equações e, opcionalmente, as condições iniciais em Y= Editor (◆ [Y=]).
3. Seleccione (F4) quais equações definidas serão traçadas.

   Nota: para desactivar todos os gráficos de dados estatísticos, prima F5 **5** ou utilize F4 para desmarcar os gráficos de dados.

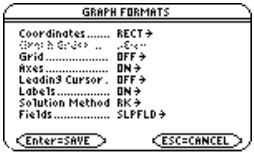


4. Configure o estilo do visor para equação.



[2nd] [F6]
[F6]

5. Configure o formato gráfico. **Solution Method** e **Fields** são únicos para as equações diferenciais.



[F1] **9**
— ou —
[◆] [I]
[◆] **F**

Nota: o formato **Fields** é crítico, dependendo da ordem da equação.

6. Configure os eixos de acordo com o caso, dependendo do formato de Fields.

[2nd] [F7]
[F7]

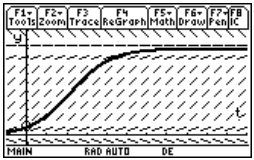
Nota: as definições válidas de **Axes** dependem do formato Fields.

7. Defina a janela de visualização ([◆] [WINDOW]).

```
t0=0.
tmax=10.
tstep=.1
tplot=0.
xmin=-10.
xmax=110.
xscl=10.
ymin=-10.
ymax=120.
yscl=10.
ncurves=0.
diftol=.001
fldres=20.
```

Nota: dependendo dos formatos de **Solution Method** e **Fields**, aparecem variáveis diferentes de Window. [F2] **Zoom** muda também a janela de visualização.

8. Represente graficamente as equações (◆ [GRAPH]).



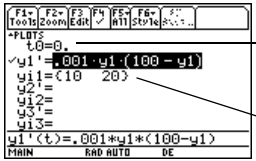
Diferenças entre a representação gráfica de funções e de equações diferenciais

Este módulo presume que o leitor já sabe como representar graficamente funções **y(x)** como descrito no módulo *Representação gráfica de funções básicas*. Esta secção descreve as diferenças.

Definição do modo gráfico

Utilize [MODE] para definir **Graph = DIFF EQUATIONS** antes de definir as equações diferenciais ou as variáveis Window. Y= Editor e Window Editor permitem introduzir informações apenas para a definição do modo Graph actual.

Definição de equações diferenciais em Y= Editor



Utilize t0 para especificar quando as condições iniciais ocorrem. É possível também definir t0 em Window Editor.

Utilize yi para especificar uma ou mais condições iniciais para a equação diferencial

É possível definir equações diferenciais de y1'(t) até y99'(t).

Nota: pode utilizar o comando **Define** a partir do ecrã principal para definir funções e equações.

Quando introduzir equações no Y= Editor, não utilize os formatos **y(t)** para fazer referência a resultados. Por exemplo:

Não utilize multiplicação implícita entre uma variável e parêntesis. Se utilizar, o conjunto será tratado como uma chamada de função.

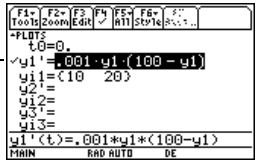
Introduza: y1' = .001y1∗(100−y1)
Não: y1' = .001y1(t)∗(100−y1(t))

Apenas as equações de 1ª ordem podem ser introduzidas no Y= Editor. Para representar graficamente equações de 2ª ordem ou de ordem superior, é preciso introduzi-las como um sistema de equações de 1ª ordem.

Para mais informações sobre a configuração de condições iniciais.

Selecção de equações diferenciais

Pode utilizar [F4] para seleccionar uma equação diferencial, mas não a sua condição inicial.



Importante: a selecção de y1' traçará o gráfico da curva de solução de y1 e não a derivada de y1', dependendo de como o eixo está configurado.

Selecção do estilo de apresentação

Com o menu Style, só estão disponíveis os estilos **Line**, **Dot**, **Square**, **Thick**, **Animate** e **Path**. Os estilos **Dot** e **Square** marcam apenas os valores discretos (em incrementos de **tstep**) nos quais é traçada uma equação diferencial.

[2nd] [F6]

[F6]

Definição de formatos gráficos

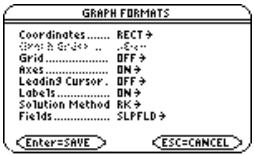
No Y= Editor, no Window Editor ou na janela Graph, prima:

F1 **9**

— ou —

◆ |

◆ **F**



Os formatos afectados pelas equações diferenciais são:

Formato gráfico	Descrição
Graph Order	Não está disponível.
Solution Method	Especifica o método usado para resolver as equações diferenciais. • RK — Método de Runge-Kutta. Para mais informações sobre o algoritmo utilizado neste método, consulte o módulo *Referência técnica*. • EULER — Método de Euler. • A escolha de um método permite optar por uma maior precisão ou velocidade. Geralmente, RK é mais preciso do que EULER, mas demora mais a encontrar a solução.

Formato gráfico	Descrição
Fields	Especifica se um campo será desenhado para a equação diferencial. • SLPFLD — Desenha um campo de inclinação para apenas uma equação de 1ª ordem, com t no eixo x e a solução no eixo y. • DIRFLD — Desenha um campo de direcção para apenas uma equação de 2ª ordem (ou sistema de duas equações de 1ª ordem), com eixos determinados pelas definições de eixos personalizados. • FLDOFF — Não mostra um campo. Isto é válido para equações de qualquer ordem, mas é necessário ser utilizado para equação de 3ª ordem ou superior. É preciso introduzir o mesmo número de condições iniciais para todas as equações no Y= Editor.

Importante: o formato gráfico de Fields é fundamental para se representar graficamente equações diferenciais com êxito.

Nota: se premir [ENTER] durante o desenho de um campo de inclinação ou direcção, o gráfico fará uma pausa depois de desenhar o campo, mas antes de traçar as soluções. Prima [ENTER] novamente para continuar. Para cancelar a representação gráfica, prima [ON].

Configuração de eixos

No Y= Editor, **Axes** pode ou não estar disponível, dependendo do formato gráfico actual.

Se estiver disponível, é possível seleccionar os eixos utilizados para representar graficamente as equações diferenciais. .

[2nd] [F7]

[F7]

Eixos	Descrição
TIME	Traça t no eixo x e y (as soluções das equações diferenciais seleccionadas) no eixo y.
CUSTOM	Permite seleccionar os eixos x e y.

Variáveis de Window

Os gráficos de equações diferenciais utilizam as seguintes variáveis de Window. Dependendo dos formatos gráficos de **Solution Method** e **Fields**, nem todas estas variáveis serão listadas no Window Editor ([◆] [WINDOW]) ao mesmo tempo..

Variável	Descrição
t0	Momento em que as condições iniciais introduzidas em Y= Editor ocorrem. É possível definir **t0** no Window Editor e Y= Editor. (Se definir **t0** no Y= Editor, **tplot** será definido automaticamente no mesmo valor.)

Variável	Descrição
tmax, **tstep**	Utilizado para determinar os valores de t onde as equações são traçadas: y'(t0) y'(t0+tstep) y'(t0+2∗tstep) ... sem exceder... y'(tmax) Se **Fields = SLPFLD**, **tmax** é ignorado. As equações são traçadas de **t0** até às duas extremidades do ecrã em incrementos de **tstep**.
tplot	O primeiro valor t traçado. Se não for um incremento **tstep**, a representação gráfica começa no próximo incremento de **tstep**. Em alguns casos, os primeiros pontos calculados e traçados iniciados em **t0** podem não ser interessantes visualmente. Se definir **tplot** maior que **t0**, é possível começar o traçado na área de interesse, o que diminui a duração da representação gráfica e evita encher desnecessariamente o ecrã Graph.

Nota: se **tmax < t0**, **tstep** terá que ser negativo. Se **Fields=SLPFLD**, **tplot** será ignorado e considerado como tendo o valor de **t0**.

Variável	Descrição
xmin, **xmax**, **ymin**, **ymax**	Limites da janela de visualização.
xscl, **yscl**	Distância entre as marcas em cada eixo, x e y.
ncurves	Número de curvas de solução (0 a 10) que serão desenhadas automaticamente se uma condição inicial não for especificada. Por predefinição, **ncurves = 0**.

Variável	Descrição
	Quando utilizar **ncurves**, **t0** é definido temporariamente no meio do ecrã e as condições iniciais são distribuídas uniformemente ao longo do eixo y, onde: $$incremento = \frac{ymax - ymin}{ncurves + 1}$$ Os valores de y para as condições iniciais são: ymin + incremento ymin + 2∗(incremento) ⋮ ymin + ncurves∗(incremento)
diftol	(apenas se **Solution Method = RK**) A tolerância utilizada pelo método de **RK** para ajudar a seleccionar um tamanho de passo para resolução da equação; tem de ser ≥1E⁻14.
fldres	(apenas se **Fields = SLPFLD** ou **DIRFLD**) Número de colunas (1 a 80) utilizado para desenhar um campo de direcção ou inclinação na largura máxima do ecrã.
Estep	(apenas se **Solution Method = EULER**) Iterações de Euler entre os valores **tstep**; tem de ser um número inteiro > 0. Para obter maior precisão, é possível aumentar **Estep** sem traçar pontos adicionais.
dtime	(apenas se **Fields = DIRFLD**) Ponto no tempo onde um campo de direcção é desenhado.

Os valores predefinidos (definidos quando seleccionar **6:ZoomStd** no menu da barra de ferramentas F2 **Zoom**) são:

t0 = 0.	xmin = ⁻1.	ymin = ⁻10.	ncurves = 0.
tmax = 10.	xmax = 10.	ymax = 10.	diftol = .001
tstep = .1	xscl = 1.	yscl = 1.	Estep = 1.
tplot = 0.			fldres = 14.
			dtime = 0.

Pode ser necessário mudar os valores predefinidos das variáveis t para assegurar que serão traçados pontos suficientes.

Variável de sistema fldpic

Quando um campo de direcção ou inclinação for desenhado, uma imagem do campo é guardada automaticamente numa variável do sistema chamada **fldpic**. Se realizar uma operação que represente graficamente as equações traçadas sem afectar o campo, a TI-89 Titanium / Voyage™ 200 Graphing Calculator reutiliza a imagem de **fldpic** em vez de fazer com que o campo seja desenhado novamente. Isto pode diminuir significantemente o tempo da nova representação gráfica.

fldpic é excluído automaticamente quando sair do modo de representação gráfica de equação diferencial ou aparecer um gráfico com **Fields = FLDOFF**.

Exploração de um gráfico

Como na representação gráfica de função, é possível explorar um gráfico com as ferramentas seguintes. Todas as coordenadas apresentadas aparecem na forma rectangular ou polar como definido no formato gráfico.

Ferramenta	Para gráficos de equações diferenciais:
Cursor de movimentação livre	Funciona exactamente como para gráficos de função.
F2 **Zoom**	Funciona exactamente como para gráficos de função. • Apenas as variáveis **x** (**xmin**, **xmax**, **xscl**) e **y** (**ymin**, **ymax**, **yscl**) de Window são afectadas. • As variáveis de Window **t** (**t0**, **tmax**, **tstep**, **tplot**) não são afectadas a menos que **6:ZoomStd** seja seleccionado (o que define todas as variáveis de Window para os valores predefinidos).

Ferramenta	Para gráficos de equações diferenciais:
F3 **Trace**	Permite mover o cursor ao longo da curva um **tstep** por vez. Para mover aproximadamente dez pontos traçados de uma única vez, prima 2nd ▷ ou 2nd ◁. Se introduzir condições iniciais no Y= Editor ou deixar que a variável **ncurves** de Window trace as curvas automaticamente, pode traçar as curvas. Se: 2nd F8 F8 for utilizado no ecrã Graph para seleccionar as condições iniciais interativamente, não é preciso traçar as curvas. QuickCenter é válido para todas as direcções. Se o cursor for movido para fora do ecrã (parte superior ou inferior, esquerda ou direita), prima ENTER para centralizar a janela de visualização no local do cursor. Utilize ▲ ou ▼ para ver os resultados de todas as curvas traçadas.
F5 **Math**	Apenas **1:Value** está disponível. • Com os eixos TIME, o valor da solução **y(t)** (representado por yc) aparece para um valor de t especificado. • Com os eixos CUSTOM, os valores correspondentes a x e y dependem dos eixos escolhidos.

Nota: é possível mover o cursor para um ponto particular durante um traçado, digitando um valor para t e premindo ENTER. É possível utilizar QuickCenter a qualquer momento numa representação gráfica, mesmo que o cursor ainda esteja no ecrã.

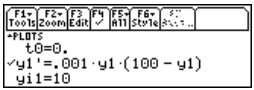
Definição das condições iniciais

É possível introduzir condições iniciais no Y= Editor, deixar que a TI-89 Titanium / Voyage™ 200 Graphing Calculator calcule as condições automaticamente ou seleccioná-las interactivamente no ecrã Graph.

Introdução de condições iniciais em Y= Editor

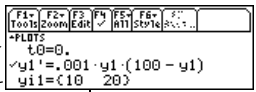
É possível especificar uma ou mais condições iniciais no Y= Editor. Para especificar mais de uma, introduza-as como uma lista entre chavetas { } e separada por vírgulas.

Para introduzir as condições iniciais para a equação **y1'**, utilize a linha **yi1**, etc.



Para especificar quando ocorrem as condições iniciais, utilize t0. Este é também o primeiro t calculado para o gráfico.

Para representar graficamente uma família de soluções, introduza uma lista de condições iniciais.

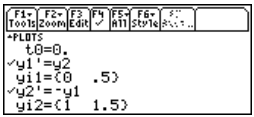


Introduza {10,20} mesmo que apareça {10 20}.

Para uma equação diferencial de segunda ordem ou superior, é preciso definir um sistema de equações de 1ª ordem em Y= Editor.

Nota: para mais informações sobre a definição de um sistema de equações de ordem mais elevada, consulte “Definição de um sistema para equações de ordem superior” .

Se introduzir condições iniciais, é preciso introduzir o mesmo número de condições iniciais para cada equação no sistema. Caso contrário, ocorrerá um erro de dimensão.



Se não introduzir uma condição inicial no Y= Editor

Se não introduzir as condições iniciais, a variável **ncurves** de Window (◆ [WINDOW]) especifica o número de curvas de solução representadas graficamente de forma automática. Por predefinição, **ncurves = 0**. É possível introduzir um valor de 0 a 10. Entretanto, o formato gráfico **Fields** e a definição **Axes** determinam se **ncurves** é utilizado.

Se Fields =	Então:
SLPFLD	Utilize **ncurves**, se não estiver definido em 0 para representar graficamente curvas.
DIRFLD	Ignore **ncurves**. Não represente nenhuma curva graficamente.
FLDOFF	Utilize **ncurves** se Axes = TIME (ou se Axes = Custom e o eixo x for t). Caso contrário, ocorre um erro de configuração Diff Eq.

Quando utilizar **ncurves**, **t0** é definido temporariamente no meio do ecrã Graph. Entretanto, o valor de **t0** como definido no Y= Editor ou Window Editor não é mudado.

Notas:

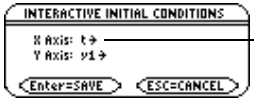
- Sem introduzir condições, utilize SLPFLD (com **ncurves=0**) ou DIRFLD para ver apenas um campo de direcção ou inclinação.

- SLPFLD é apenas para uma única equação de 1ª ordem. DIRFLD é apenas para uma equação de 2ª ordem (ou sistema de duas equações de 1ª ordem).

Selecção de uma condição inicial interactivamente a partir do ecrã Graph

Quando uma equação diferencial é representada graficamente (independente da curva de solução ser exibida ou não), é possível seleccionar um ponto no gráfico Graph e utilizá-lo como uma condição inicial.

Se Fields =	Efectue o procedimento apresentado em seguida:
SLPFLD – ou – DIRFLD	Prima: [2nd] [F8] [F8] Especifique uma condição inicial: • Mova o cursor para o ponto em questão e prima [ENTER]. – ou – • Para cada uma de duas coordenadas, digite um valor e prima [ENTER]. - Para SLPFLD (apenas para 1ª ordem), introduza valores para **t0** e **y(t0)**. - Para DIRFLD (apenas para 2ª ordem ou sistema de duas equações de 1ª ordem), introduza os valores para ambas as condições iniciais de **y(t0)**, onde **t0** é o valor definido no Y= Editor ou Window Editor. Um círculo indica a condição inicial e a curva de solução é desenhada.

Se Fields =	Efectue o procedimento apresentado em seguida:
FLDOFF	• Prima: [2nd] [F8] [F8] Um aviso aparece para seleccionar os eixos onde deseja introduzir condições iniciais.  t é uma selecção válida. Permitirá que seja especificado um valor para t0. As selecções serão utilizadas como os eixos para o gráfico. • É possível aceitar os padrões ou mudá-los. Em seguida, prima [ENTER]. • Especifique uma condição inicial conforme descrito para SLPFLD ou DIRFLD.

Nota: SLPFLD ou DIRFLD permite seleccionar as condições iniciais interactivamente, independentemente das condições iniciais serem ou não introduzidas no Y= Editor. FLDOFF permite seleccionar as condições iniciais interactivamente. Entretanto, se introduzir três ou mais equações, é preciso introduzir um único valor (não uma lista) como a condição inicial para cada equação no Y= Editor. Caso contrário, ocorre um erro de dimensão durante a representação gráfica.

Nota sobre o traçado de uma curva de solução

Quando introduz condições iniciais no Y= Editor ou deixa que **ncurves** represente graficamente curvas de solução automaticamente, pode utilizar [F3] para traçar as

curvas. Entretanto, não é possível traçar uma curva desenhada pela selecção de uma condição inicial interactivamente. Estas curvas são desenhadas, não traçadas.

Definição de um sistema para equações de ordem superior

No Y= Editor, é preciso introduzir todas as equações diferenciais como equações de 1ª ordem. Se existir uma equação da ordem n, é preciso transformá-la num sistema de n equações de 1ª ordem.

Transformação de uma equação num sistema de 1ª ordem

Um sistema de equações pode ser definido de várias formas, mas o método seguinte é um método geral

1. Reescreva a equação diferencial original conforme necessário.

   $y'' + y' + y = e^x$

   a) Resolva a derivada de ordem mais elevada.

   $y'' + y' + y = e^x$

   b) Expresse-a em termos de y e t.

   $y'' = e^t - y' - y$

   c) Apenas no lado direito da equação, substitua para eliminar todas as referências aos valores das derivadas.

No lugar de:	Coloque:
y	y1
y'	y2
y''	y3
y'''	y4
$y^{(4)}$	y5
⋮	⋮

$y'' = e^t - y2 - y1$

Não substitua no lado esquerdo neste momento.

Nota: para produzir uma equação de 1ª ordem, o lado direito tem de conter apenas variáveis não deriváveis.

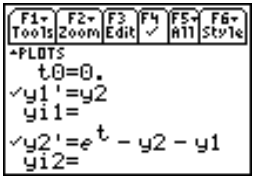
d) No lado esquerdo da equação, substitua o valor da derivada como mostrado abaixo.

No lugar de:	Coloque:
y'	y1’
y''	y2’
y'''	y3’
$y^{(4)}$	y4’
⋮	⋮

$y2' = e^t - y2 - y1$

2. Nas linhas correspondentes no Y= Editor, defina o sistema de equações como:

   y1' = y2
   y2' = y3
   y3' = y4
   – até –
   yn ' = a sua equação da ordem



Nota: Baseado nas substituições acima, as linhas de y' no Y= Editor representam:
y1' = y'
y2' = y''
etc.

Assim, a equação de 2ª ordem deste exemplo é introduzida na linha y2'.

Num sistema como este, a solução da equação y1' é a solução para a equação de ordem n. Pode querer desmarcar todas as outras equações no sistema.

Exemplo de uma equação de 2ª ordem

A equação diferencial de 2ª ordem **y"+y = 0** representa um oscilador harmónico simples. Transforme isto num sistema de equações para o Y= Editor. Em seguida, represente graficamente a solução com as condições iniciais **y(0) = 0** e **y'(0) = 1**.

Exemplo

1. Prima [MODE] e defina **Graph=DIFF EQUATIONS**.

2. Defina um sistema de equações para a equação de 2ª ordem.

   Reescreva a equação e faça as substituições necessárias.

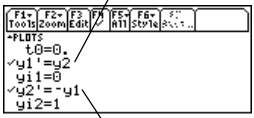
   y" + y = 0
   y" = ⁻y
   y" = ⁻y1
   y2' = ⁻y1

3. No Y= Editor ([◆] [Y=]), introduza o sistema de equações.

4. Introduza as condições iniciais: **yi1=0** e **yi2=1**

   Nota: t0 é o momento em que as condições iniciais ocorrem. É também o primeiro t calculado para o gráfico. Por predefinição, **t0=0**.

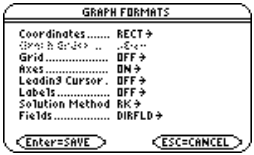
yi1 é a condição inicial para y(0).



yi2 é a condição inicial para y'(0).

5. Prima:
   [F1] **9**
   — ou —
   [calculadora] [◆] [|]
   [calculadora] [◆] **F**
   e defina **Axes = ON**, **Labels = OFF**, **Solution Method = RK** e **Fields = DIRFLD**.

   Importante: para equações de 2ª ordem, tem de definir **Fields=DIRFLD** ou **FLDOFF**.



6. No Y= Editor, prima:

   [2nd] [F7]

   [F7]

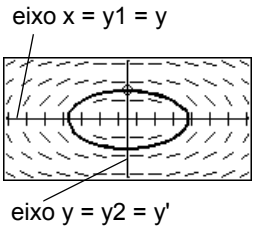
   e certifique-se de que **Axes = CUSTOM** com **y1** e **y2** como eixos.

   Importante: Fields=DIRFLD não pode traçar um eixo de tempo. Um erro **Invalid Axes** ocorre se **Axes=TIME** ou se t estiver definido como um eixo **CUSTOM**.

7. No Window Editor ([◆] [WINDOW]), defina as variáveis de Window.

t0=0	xmin=⁻2	ncurves=0
tmax=10	xmax=2	diftol=.001
tstep=.1	xscl=1	fldres=14
tplot=0	ymin=⁻2	dtime=0
	ymax=2	
	yscl=1	

8. Veja o ecrã Graph ([◆] [GRAPH]).

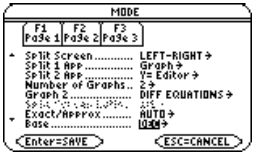


Se seleccionar **ZoomSqr** ([F2] **5**), é possível ver que a órbita do plano de fase é na verdade um círculo. Entretanto, **ZoomSqr** mudará as suas variáveis de Window.

Para examinar este oscilador harmónico mais detalhadamente, utilize o ecrã dividido para representar graficamente a maneira na qual y e y' mudam com relação ao tempo (t).

9. Prima [MODE] e mude as configurações do modo na **página 2** como mostrado. Em seguida, feche a caixa de diálogo **MODE**, que desenha novamente o gráfico.

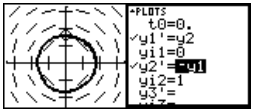
   Nota: para exibir gráficos diferentes nas duas partes de um ecrã dividido, pode utilizar o modo de dois gráficos.



10. Prima [2nd] [⊞] para alternar para o lado direito do ecrã dividido.

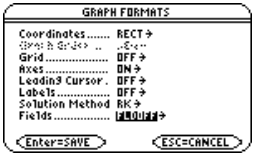
11. Utilize [F4] para seleccionar **y1'** e **y2'**.

    O lado direito utiliza as mesmas equações que o lado esquerdo. Entretanto, nenhuma equação é seleccionada inicialmente no lado direito.



12. Prima:
    [F1] **9**
    — ou —
    [◆] [|]
    [◆] **F**
    e defina **Fields = FLDOFF**.

    Importante: como **Fields=DIRFLD** não pode traçar um eixo de tempo, tem de mudar a definição de Fields. **FLDOFF** desliga todos os campos.



13. No Y= Editor, prima:
    [2nd] [F7]
    [F7]
    e certifique-se de que **Axes = TIME**.

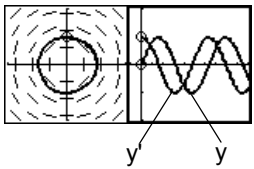
14. No Window Editor, altere **ymin** e **ymax** como mostrado à direita.

    Nota: quando entra no modo de dois gráficos, as variáveis de Window para o lado direito utilizam as predefinições.

ymin=⁻2.
ymax=2.

15. Prima [◆] [GRAPH] para ver o ecrã para o gráfico n.º 2.

    O lado esquerdo mostra a órbita do plano da fase. O lado direito mostra a curva de solução e a sua derivada.



16. Para voltar para um ecrã cheio do gráfico original, prima [2nd] [⊟] para alternar para o lado esquerdo. Em seguida, prima [MODE] e mude a definição de **Split Screen**.

Split Screen = FULL

Exemplo de uma equação de 3ª ordem

Para a equação diferencial de 3ª ordem **y'''+2y''+2y'+y = sin(x)**, escreva um sistema de equações para introduzir no Y= Editor. Em seguida, represente a solução como uma função do tempo. Utilize as condições iniciais **y(0) = 0**, **y'(0) = 1**, e **y''(0) = 1**.

Exemplo

1. Pressione [MODE] e defina **Graph=DIFF EQUATIONS**.

2. Defina um sistema de equações para a equação de 3ª ordem.

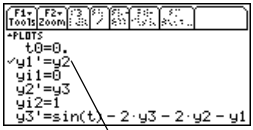
   Reescreva a equação e faça as substituições necessárias.

   y''' + 2y'' + 2y' + y = sin(x)
   y''' = sin(x) − 2y'' − 2y' − y
   y''' = sin(t) − 2y'' − 2y' − y
   y''' = sin(t) − 2y3 − 2y2 − y1
   y3' = sin(t) − 2y3 − 2y2 − y1

3. No Y= Editor ([◆] [Y=]), introduza o sistema de equações.

4. Introduza as condições iniciais: **yi1=0**, **yi2=1** e **yi3=1**

   Nota: t0 é o tempo em que as condições iniciais ocorrem. Por predefinição, **t0=0**.

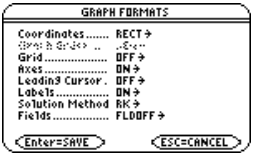


Importante: A solução da equação y1' é a solução da equação de 3ª ordem.

5. Certifique-se de que apenas **y1'** está seleccionado. Utilize [F4] para desmarcar todas as outras equações.

6. Prima:
   [F1] **9**
   — ou —
   [♦] [|]
   [♦] **F**
   e defina **Axes = ON**, **Labels = ON**, **Solution Method = RK** e **Fields = FLDOFF**.

   Importante: para equações de 3ª ordem ou superior, tem de definir **Fields=FLDOFF**. Caso contrário, um erro **Undefined variable** ocorre durante a representação gráfica.



7. No Y= Editor, prima:
   [2nd] [F7]
   [F7]
   e defina **Axes = TIME**.

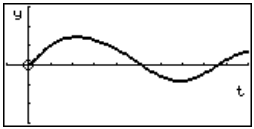
   Nota: com **Axes=TIME**, a solução da equação seleccionada é traçada em função do tempo (t).

8. No Window Editor ([♦] [WINDOW]), defina as variáveis de Window.

t0=0	xmin=⁻1	ncurves=0
tmax=10	xmax=10	diftol=.001
tstep=.1	xscl=1.	
tplot=0	ymin=⁻3	
	ymax=3	
	yscl=1	

9. Veja o ecrã Graph ([◆] [GRAPH]).

   Nota: para encontrar a solução num determinado momento, utilize [F3] para traçar o gráfico.



Definição de eixos para gráficos de tempo ou personalizados

A definição de eixos pode proporcionar uma grande flexibilidade na representação gráfica de equações diferenciais. Os eixos personalizados são particularmente eficazes para mostrar tipos de relações diferentes.

Visualização da caixa de diálogo AXES

No Y= Editor, prima:

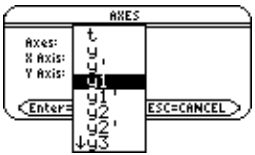
[2nd] [F7]

[F7]

Se **Fields = SLPFLD**, **Axes** não está disponível.

[2nd] [F7]

[F7]

Item	Descrição
Axes	TIME — Traça t no eixo x e y (soluções para todas as equações diferenciais seleccionadas) no eixo y. CUSTOM — Permite seleccionar os eixos x e y .
X Axis, **Y Axis**	Activo apenas quando Axes = CUSTOM; isto permite seleccionar o que se quer traçar nos eixos x e y. 
	t — tempo
	y — soluções (y1, y2 etc.) de todas as equações diferenciais seleccionadas
	y' — valores de todas as equações diferenciais seleccionadas (y1', y2' etc.)

Item	Descrição
	y1, y2 etc. — a solução da equação diferencial correspondente, independentemente da equação estar seleccionada
	y1', y2' etc. — o valor do lado direito da equação diferencial correspondente, independentemente da equação estar seleccionada

Nota: t não é válido para nenhum eixo (Axis) quando **Fields=DIRFLD**. Se seleccionar t, ocorre um erro Invalid axes durante o processo de representaçao gráfica.

Exemplo de eixos de tempo e personalizados

Utilize o modelo de biologia de predador-presa para determinar o número de coelhos e raposas necessários para manter o equilíbrio da população numa certa região. Represente graficamente a solução com os eixos de tempo e personalizados.

Modelo predador-presa

Utilize as duas equações diferenciais de 1ª ordem ligadas:

$y1' = -y1 + 0.1y1 * y2$ e $y2' = 3y2 - y1 * y2$

onde:

y1 = População de raposas

yi1 = População inicial de raposas (2)

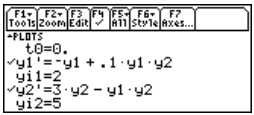
y2 = População de coelhos

yi2 = População inicial de coelhos (5)

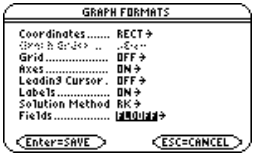
1. Utilize [MODE] para definir **Graph = DIFF EQUATIONS**.

2. No Y= Editor ([◆] [Y=]), defina as equações diferenciais e introduza as condições iniciais.

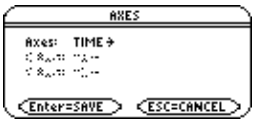
   Nota: para diminuir o tempo gasto na representação gráfica, limpe todas as outras equações no Y= Editor. Com **FLDOFF**, todas as equações são calculadas mesmo que não estejam seleccionadas.



3. Prima:
   [F1] **9**
   — ou —
   [◆] [I]
   F
   Defina **Axes = ON**, **Labels = ON**, **Solution Method = RK** e **Fields = FLDOFF**.



4. No Y= Editor, prima:
   [2nd] [F7]
   [F7]
   Defina **Axes = TIME**.



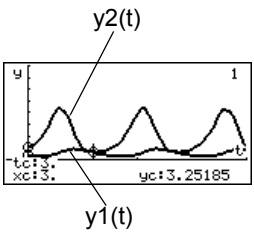
5. No Window Editor ([◆] [WINDOW]), defina as variáveis de Window.

t0=0	xmin=⁻1	ncurves=0
tmax=10	xmax=10	diftol=.001
tstep=π/24	xscl=5	
tplot=0	ymin=⁻10	
	ymax=40	
	yscl=5	

6. Represente graficamente as equações diferenciais ([◆] [GRAPH]).

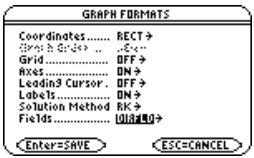
7. Prima [F3] para traçar. Em seguida, prima **3** [ENTER] para ver o número de raposas (**yc** para **y1**) e coelhos (**yc** para **y2**) em **t=3**.

   Nota: utilize ⊕ e ⊖ para mover o cursor de traçado entre as curvas para **y1** e **y2**.



8. Volte ao Y= Editor. Em seguida, prima:
   [F1] **9**
   — ou —
   [calc] [◆] [|]
   [keyboard] [◆] **F**
   Defina **Fields = DIRFLD**.

   Nota: neste exemplo, **DIRFLD** é utilizado para duas equações diferenciais relacionadas que não representam uma equação de 2ª ordem.



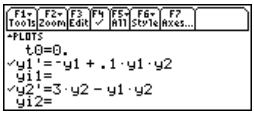
9. Prima:

   [2nd] [F7]

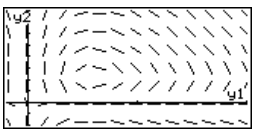
   [F7]

   Confirme se os eixos estão definidos como mostrado.

10. No Y= Editor, limpe as condições iniciais para **yi1** e **yi2**.



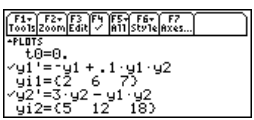
11. Volte ao ecrã Graph, que exibe apenas o campo de direcção.



12. Para representar graficamente uma família de soluções, volte ao Y= Editor e introduza as condições iniciais como mostrado abaixo.

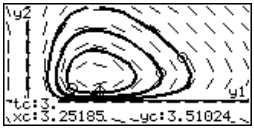
    yi1={2,6,7} e **yi2={5,12,18}**

    Nota: utilize uma lista para especificar mais de uma condição inicial.



13. Volte ao ecrã Graph, que mostra uma curva para cada par de condições iniciais.

14. Prima [F3] para traçar. Em seguida, prima **3** [ENTER] para ver o número de raposas (**xc**) e coelhos (**yc**) em **t=3**.



Como **t0=0** e **tmax=10**, é possível traçar no intervalo 0 ≤ t ≤ 10.

Nota: utilize ⊕ e ⊖ para mover o cursor de traçado de uma curva de condição inicial para outra.

Exemplo: comparação entre RK e Euler

Considere um modelo de crescimento logístico **dP/dt = 001∗P∗(100−P)**, com a condição inicial **P(0) = 10**. Utilize a instrução **BldData** para comparar os pontos de representação gráfica calculados pelos métodos da solução de **RK** e **Euler**. Em seguida, trace estes pontos ao longo de um gráfico com a solução exacta da equação.

Exemplo

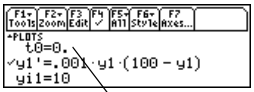
1. Prima [MODE] e defina **Graph=DIFF EQUATIONS**.

2. Expresse a equação de 1ª ordem em termos de **y1'** e **y1**.

y1'=.001y1∗(100−y1)

Não utilize multiplicação implícita entre a variável e o parêntesis. Se o fizer, será tratada como uma chamada de função.

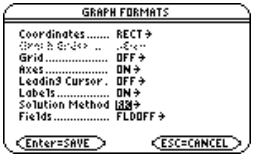
3. Introduza a equação no Y= Editor (◆ [Y=]).
4. Introduza a condição inicial: **yi1=10**



t0 é o tempo em que a condição inicial ocorre. Por predefinição, t0=0.

5. Prima:
   [F1] **9**
   — ou —
   [calculadora] ◆ [|]
   [teclado] ◆ **F**
   Defina **Solution Method = RK** e **Fields = FLDOFF**.

   Nota: para diminuir o tempo gasto na representação gráfica, limpe todas as outras equações no Y= Editor. Com **FLDOFF**, todas as equações são calculadas mesmo se não estiverem seleccionadas.



6. No Window Editor (◆ [WINDOW]), defina as variáveis de Window.

t0=0.	xmin=⁻1.	ncurves=0.
tmax=100.	xmax=100.	diftol=.001
❶ tstep=1.	xscl=1.	
tplot=0.	ymin=⁻10.	
	ymax=10	
	yscl=1.	

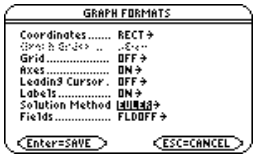
❶ **Importante:** mude o valor de **tstep** de .1 (predefinição) para 1. Caso contrário, **BldData** calcula muitas linhas para a variável de dados e ocorre um erro de dimensão.

7. No ecrã principal
   [HOME]
   [◆] [CALC HOME]
   utilize **BldData** para criar uma variável de dados com os pontos de representação gráfica de **RK**.

BldData rklog

8. Volte ao Y= Editor, prima:
   [F1] **9**
   — ou —
   [◆] [|]
   [◆] **F**
   Defina **Solution Method = EULER**.

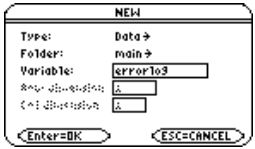
   Nota: não é necessário representar graficamente a equação antes de utilizar **BldData**. Para mais informações sobre **BldData**, consulte o módulo *Referência técnica*.



9. Volte ao ecrã principal principal e utilize **BldData** para criar uma variável de dados com os pontos de representação gráfica de **Euler**.

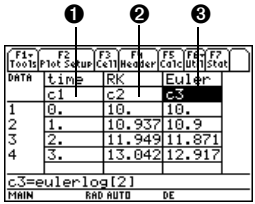
BldData eulerlog

10. Utilize o Matrix and Data Editor para criar uma nova variável de dados chamada **errorlog**.



Nota: errorlog permite combinar os dados em **rklog** e **eulerlog** para que os dois conjuntos de dados possam ser visualizados lado a lado.

11. Nesta nova variável de dados, defina os cabeçalhos das colunas **c1**, **c2** e **c3** para fazerem referência aos dados em **rklog** e **eulerlog**. Além disso, introduza os títulos de coluna como mostrado.

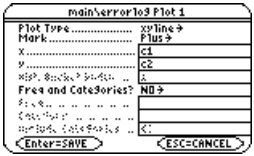


❶ c1=rklog[1] ou c1=eulerlog[1]
❷ c2=rklog[2]
❸ c3= eulerlog[2]

Para definir um cabeçalho de coluna, mova o cursor para a coluna desejada, prima [F4], digite a expressão de referência (como **rklog[1]** para **c1**) e prima [ENTER].

Nota: rklog[1] e **rklog[2]** referem-se às colunas 1 e 2 em **rklog**, respectivamente. De modo análogo com **eulerlog[2]**.

12. No Matrix and Data Editor, prima [F2]. Em seguida, prima [F1] e defina **gráfico 1** para os dados de **RK**, como mostrado à direita.



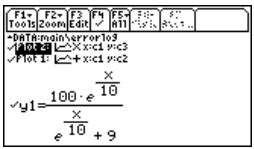
13. Defina o **gráfico 2** para os dados de **Euler**. Utilize os valores mostrados à direita.

Plot Type=xyline
Mark=Cross
x=c1
y=c3

14. Volte ao Y= Editor, prima [MODE] e defina **Graph = FUNCTION**.

15. A solução exacta da equação diferencial é dada abaixo. Introduza-a como y1.

    y1 = (100∗e^(x/10))/(e^(x/10)+9)



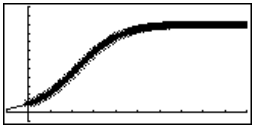
É possível utilizar ⊕ para percorrer para cima para visualizar os gráficos 1 e 2.

16. No Window Editor, defina as variáveis de Window.

xmin=⁻10	ymin=⁻10.	xres=2.
xmax=100	ymax=120.	
xscl=10	yscl=10.	

17. Veja o ecrã Graph ([◆] [GRAPH]).

    Nota: a linha indistinta no gráfico indica as diferenças entre os valores de **RK** e **Euler**.

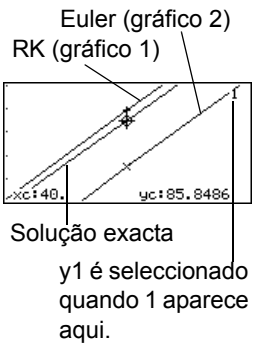


18. No Window Editor, aumente as variáveis Window para que as diferenças possam ser examinadas mais detalhadamente.

xmin=39.7	ymin=85.5	xres=2
xmax=40.3	ymax=86	
xscl=.1	yscl=.1	

19. Volte ao ecrã Graph.

20. Prima [F3] para traçar e, em seguida, prima ⊕ ou ⊖ até que **y1** seja seleccionado. (1 aparece no lado superior direito.) Em seguida, introduza 40.



Quando mover o cursor de traçado para traçar cada solução para **xc = 40**, descobrirá que:

- A solução exacta (**y1**) é 85,8486, arredondada para ter seis dígitos.
- A solução de **RK** (**gráfico 1**) é 85,8952.
- A solução de **Euler** (**gráfico 2**) é 85,6527.

Pode utilizar o Matrix and Data Editor para abrir a variável de dados errorlog e ir para **time = 40**.

Exemplo da função deSolve( )

A função **deSolve( )** permite resolver muitas equações diferenciais de 1ª e de 2ª ordem com exactidão.

Exemplo

Para obter uma solução geral, utilize a sintaxe seguinte. Para uma solução particular, consulte o módulo *Referência técnica*.

deSolve(*1aOu2aOrdemODE, varIndependente, varDependente***)**

Utilize a equação diferencial de 1ª ordem logística descrita no exemplo na página 176 para encontrar a solução geral para y com relação a t.

deSolve(y' = 1/1000 y∗(100−y),t,y**)**

Para obter ', digite [2nd] ['].

Não utilize multiplicação implícita entre uma variável e o parêntesis. Se o fizer, o conjunto será tratado como uma chamada de função.

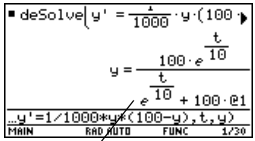
Notas:

- Para obter a precisão máxima, utilize 1/1000 em vez de .001. O número de ponto flutuante pode introduzir erros de arredondamento.

- Este exemplo não envolve representação gráfica. Por isso, pode utilizar qualquer modo Graph.

Antes de utilizar **deSolve( )**, limpe todas as variáveis t e y existentes ou ocorre um erro.

1. No ecrã principal
   [HOME]
   [◆] [CALC HOME]
   utilize **deSolve( )** para encontrar a solução geral.



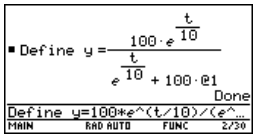
@1 representa uma constante. Pode obter uma constante diferente (@2,etc.).

2. Utilize a solução para definir uma função.

   a) Prima ⊕ para realçar a solução na área do histórico. Em seguida, prima [ENTER] para colar a solução automaticamente na linha de entrada.

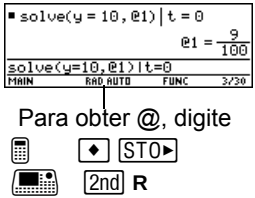
   b) Insira a instrução **Define** no início da linha. Em seguida, prima [ENTER].

   Nota: prima [2nd] ◀ para ir para o início da linha de entrada.

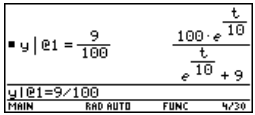


3. Para uma condição inicial **y=10** com **t=0**, utilize **solve( )** para encontrar a constante @1.

   Nota: se for obtida uma constante diferente (@2 etc.), encontre a solução para a constante.



4. Calcule a solução geral (**y**) com a constante **@1=9/100** para obter a solução particular mostrada.



Pode também utilizar **deSolve( )** para calcular este problema directamente. Introduza:

deSolve(y' = 1/1000 y∗(100−y) e y(0)=10,t,y**)**

Solução de problemas com o formato gráfico Fields

Se tiver dificuldades na representação gráfica de uma equação diferencial, esta secção pode ajudá-lo a corrigir o problema. Muitos problemas podem estar relacionados à configuração do formato gráfico Fields.

Definição do formato gráfico Fields

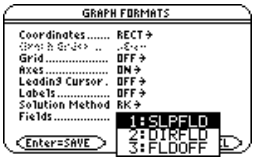
No Y= Editor, no Window Editor ou no ecrã Graph, prima:

F1 **9**

— ou —

▦ ◆ I

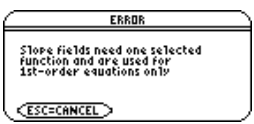
▦ ◆ **F**



Qual a ordem da equação que está a ser representada graficamente?

Se a equação for:	Configurações de campos válidos serão:
1ª ordem	SLPFLD ou FLDOFF
2ª ordem (sistema de duas equações de 1ª ordem)	DIRFLD ou FLDOFF
3ª ordem ou superior (sistema de três ou mais equações de 1ª ordem)	FLDOFF

Como **Fields = SLPFLD** é a configuração predefinida, uma mensagem de erro comum aparece à direita.

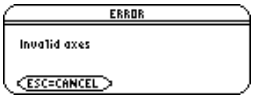
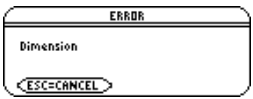


Se obtiver esta ou qualquer outra mensagem de erro:

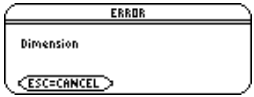
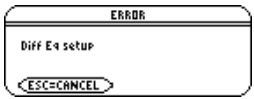
- Utilize a tabela anterior para encontrar as configurações válidas de Fields para a ordem da sua equação. Mude-a para a configuração válida.
- No caso de uma configuração particular de Fields, verifique o seguinte para obter informações válidas para a configuração.

Fields=SLPFLD

No Y= Editor	Utilize F4 para seleccionar apenas uma equação de 1ª ordem. É possível introduzir várias equações, mas apenas uma por vez pode ser seleccionada. A equação seleccionada não pode referir-se a qualquer outra equação no Y= Editor. Por exemplo: Se **y1'=y2**, ocorre um erro Undefined variable na representação gráfica.
No ecrã Graph	Se o campo de inclinação for desenhado sem o traçado de nenhuma curva de solução.

No Y= Editor	Introduza um sistema válido de duas equações de 1ª ordem. Para mais informações sobre a definição de um sistema válido para uma equação de 2ª ordem, consulte o exemplo de uma equação de 2ª ordem. Configure **Axes = CUSTOM**: [2nd] [F7] [F7] Se **Axes = TIME**, ocorre um erro Invalid axes quando traçar um gráfico.  Se introduzir condições iniciais no Y= Editor, as equações referenciadas pelos eixos personalizados devem ter o mesmo número de condições iniciais. Caso contrário, ocorre um erro de dimensão na representação gráfica. 
Com eixos personalizados	Defina eixos que sejam válidos para o seu sistema de equações. Não seleccione t para qualquer eixo. Caso contrário, ocorre um erro Invalid axes na representação gráfica. Os dois eixos têm de fazer referência a equações diferentes no seu sistema de equações. Por exemplo, y1 x y2 é válido, mas y1 x y1' gera um erro Invalid axes.

No ecrã Graph	Se desenhar o campo de direcção, mas não traçar nenhuma curva, introduza as condições iniciais no Y= Editor ou seleccione uma interactivamente no ecrã Graph. Se tiver introduzido condições iniciais, seleccione **ZoomFit**: [F2] [alpha] **A** [F2] **A** A variável de Window **ncurves** é ignorada com DIRFLD. As curvas predefinidas não são desenhadas automaticamente.
Notas	Com DIRFLD, as equações referenciadas pelos eixos personalizados determinam as equações que são representadas graficamente, independentemente das equações seleccionadas no Y= Editor. Se o sistema de equações fizer referência a t, o campo de direcção (não as curvas traçadas) é desenhado com relação a um tempo específico, que é definido pela variável de Window dtime.

No Y= Editor	Se introduzir uma equação de 2ª ordem ou ordem superior, introduza-a como um sistema válido de equações. Todas as equações (seleccionadas ou não) devem ter o mesmo número de condições iniciais. Caso contrário, ocorre um erro de dimensão na representação gráfica.  Para configurar **Axes = TIME** ou **CUSTOM**, prima: 2nd [F7] F7
Com eixos personalizados	Se X Axis não for t, tem de introduzir pelo menos uma condição inicial para cada equação no Y= Editor (independentemente de a equação estar seleccionada ou não). Caso contrário, ocorre um erro Diff Eq setup durante a representação gráfica. 

No ecrã Graph	Se não for traçada nenhuma curva, defina uma condição inicial. Se já tiver introduzido as condições iniciais no Y= Editor, seleccione **ZoomFit**: [F2] [alpha] **A** [F2] **A** Uma equação de 1ª ordem pode parecer diferente com FLDOFF e não com SLPFLD. Isto é porque FLDOFF utiliza as variáveis de Window **tplot** e **tmax**, que são ignoradas com SLPFLD.
Notas	Em equações de 1ª ordem, utilize FLDOFF e **Axes = Custom** para traçar eixos que não são possíveis com SLPFLD. Por exemplo, é possível traçar t x y1' (onde SLPFLD traça t x y1). Se introduzir várias equações de 1ª ordem, é possível traçar uma equação ou a sua solução vs. uma outra, especificando-as como sendo os eixos.

Se utilizar o ecrã da tabela para ver equações diferenciais

Pode utilizar o ecrã da tabela para ver os pontos num gráfico de equação diferencial. No entanto, a tabela pode mostrar equações diferentes das representadas. A tabela mostra apenas as equações seleccionadas, independentemente das equações estarem traçadas com as suas configurações actuais de **Fields** e **Axes**.

Tabelas

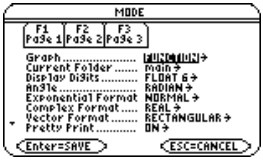
Visão geral do procedimento para gerar uma tabela

O procedimento abaixo explica como gerar uma tabela de valores para uma ou mais funções. As páginas seguintes explicam como definir os parâmetros da tabela e apresentá-los.

Geração de uma tabela

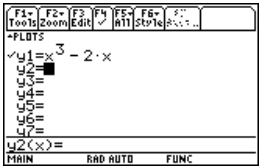
1. Defina o modo **Graph** e, se necessário, o modo **Angle** (MODE).

   Nota: as tabelas não estão disponíveis no modo de representação gráfica 3D.



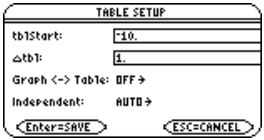
2. Defina as funções no Y= Editor (◆ [Y=]).
3. Seleccione com (F4) as funções definidas que deseja mostrar na tabela.

   Nota: para mais informações sobre como definir e seleccionar as funções com o Y= Editor, consulte o módulo *Representação gráfica de funções básicas*.

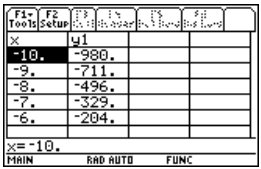


4. Defina os parâmetros iniciais da tabela ([◆] [TBLSET]).

   Nota: pode especificar uma tabela automática baseada em valores iniciais ou que corresponda a um gráfico ou a uma tabela manual (ask).



5. Veja a tabela ([◆] [TABLE]).



Exploração da tabela

No ecrã Table é possível:

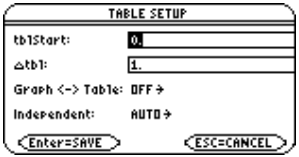
- Deslocar-se pela tabela para ver os valores contidos noutras páginas.
- Realçar uma célula para ver o seu valor completo.
- Modificar os parâmetros de configuração da tabela. Se modificar o valor inicial ou o incremento da variável independente, é possível aumentar ou diminuir a tabela para ver com mais ou menos precisão os valores.
- Modificar as dimensões da célula.
- Editar as funções seleccionadas.
- Criar ou editar uma tabela manualmente para ver apenas alguns valores específicos da variável independente.

Configuração dos parâmetros de uma tabela

Para configurar os parâmetros iniciais de uma tabela, utilize a caixa de diálogo TABLE SETUP. Depois de ver a tabela, pode utilizar esta caixa de diálogo para modificar os parâmetros.

Apresentação da caixa de diálogo TABLE SETUP

Para mostrar a caixa de diálogo TABLE SETUP, prima [◆] [TBLSET]. Pode também premir [F2] no ecrã Table.



Parâmetro de definição	Descrição
tblStart	Se **Independente = AUTO** e **Graph < - > Table = OFF**, este parâmetro especifica o valor inicial da variável independente.
Δtbl	Se **Independent = AUTO** e **Graph < - > Table = OFF**, este parâmetro especifica o valor de incremento da variável independente. **Δtbl** pode ser positivo ou negativo, mas não zero.

Parâmetro de definição	Descrição
Graph < - > Table	Se **Independent = AUTO**: OFF — A tabela baseia-se nos valores introduzidos por **tblStart** e **Δtbl.** ON — A tabela baseia-se nos mesmos valores da variável independente utilizados para representar graficamente as funções no ecrã Graph. Estes valores dependem das variáveis Window definidas no Window Editor e das dimensões do ecrã dividido.
Independent	AUTO — A TI-89 /Voyage™ 200 gera automaticamente uma série de valores para a variável independente, baseada em **tblStart**, **Δtbl**, e **Graph < - > Table**. ASK — Permite criar manualmente uma tabela, inserindo valores específicos para a variável independente.

Nota: a tabela inicia-se com **tblStart;** no entanto, os valores precedentes podem aparecer através da utilização da tecla ⊕.

Que parâmetros de configuração utilizar

Para gerar:	tblStart	Δtbl	Graph < - > Table	Independent
Uma tabela automática				
• Baseada em valores iniciais	valor	valor	**OFF**	**AUTO**
• Correspondente ao ecrã Graph	–	–	**ON**	**AUTO**

Para gerar:	tblStart	Δtbl	Graph < – > Table	Independent
Uma tabela manual	–	–	**–**	**ASK**

Nota: "–" significa que qualquer valor introduzido para este parâmetro é ignorado pelo tipo de tabela indicado.

No modo de representação gráfica SEQUENCE, utilize números inteiros para **tblStart** e **Δtbl**.

Modificação dos parâmetros de configuração

Na caixa de diálogo **TABLE SETUP**:

1. Utilize ⊖ e ⊕ para realçar o valor ou configuração que deseja modificar.
2. Especifique o novo valor ou configuração.

Para modificar:	Operação:
tblStart ou **Δtbl**	Digite o novo valor; o valor é cancelado quando iniciar a digitação. — ou — Prima ◀ ou ▶ para remover o realce e modifique o valor existente.

Para modificar:	Operação:
Graph < - > Table ou **Independent**	Prima ◀ ou ▶ para ver um menu com valores válidos e: • Desloque o cursor para realçar a definição e prima [ENTER]. — ou — • Prima o número correspondente à definição desejada.

Nota: para cancelar um menu ou sair de uma caixa de diálogo sem guardar as modificações, prima [ESC] em vez de [ENTER].

3. Depois de modificar os valores ou as configurações desejados, prima [ENTER] para guardar as modificações e feche a caixa de diálogo.

No ecrã principal ou num programa

Os parâmetros da tabela podem ser configurados no ecrã principal ou num programa. É possível:

- Guardar os valores directamente nas variáveis de sistema **tblStart** e **Δtbl**. Consulte “Armazenamento e recuperação dos valores de variáveis” — no módulo *Funcionamento da calculadora*.
- Definir **Graph < - > Table** e Independent com a função **setTable**. Consulte o módulo *Referência técnica*.

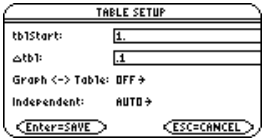
Apresentação de uma tabela de forma automática

Se **Independent = AUTO** na caixa de diálogo TABLE SETUP, uma tabela é gerada automaticamente quando aparecer o ecrã **Table**. Se **Graph < - > Table = ON**, os valores da tabela correspondem aos valores do ecrã Graph. Se **Graph < - > Table = OFF**, a tabela baseia-se nos valores introduzidos em **tblStart** e **Δtbl**.

Antes de iniciar

Defina e seleccione as funções desejadas no Y= Editor (◆ [Y=]). Este exemplo utiliza a função **y1(x) = x^3−x/3**.

Introduza os parâmetros iniciais da tabela (◆ [TBLSET]).



Apresentação do ecrã Table

Para ver o ecrã Table, prima ◆ [TABLE].

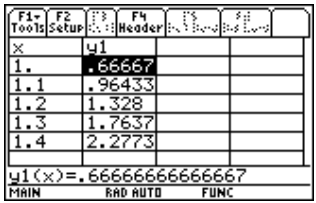
O cursor realça célula que contém o valor inicial da variável independente. O cursor pode ser deslocado para qualquer célula que contenha um valor.

A primeira coluna mostra os valores da variável independente.

As outras colunas mostram os valores correspondentes das funções seleccionadas no Y= Editor.

A linha de cabeçalho contém os nomes da variável independente (x) e das funções

A linha de entrada mostra o valor completo da célula realçada.



Nota: para voltar ao valor inicial, prima ⊕ ou 2nd ⊕.

Para deslocar o cursor:	**Prima:**
Uma célula de cada vez	⊖, ⊕, ⊕, ou ⊖
Uma página de cada vez	2nd e ⊖, ⊕, ⊕, ou ⊖

A linha de cabeçalho e a primeira coluna são fixas e não é possível deslocá-las para fora do ecrã.

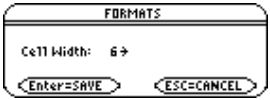
- Se for para cima ou para baixo, os nomes da variável e da função aparecem sempre na parte superior do ecrã.
- Se for para direita ou para a esquerda, os valores da variável independente aparecem sempre na parte esquerda do ecrã.

Modificação da largura da célula

A largura da célula determina o número máximo de dígitos e símbolos que podem aparecer dentro da célula (número de casas decimais, sinal de menos e símbolo "E" de notação científica). Todas as células de uma tabela têm a mesma largura.

Nota: a largura predefinida das células é **6**.

Para modificar a largura da célula no ecrã **Table**:



1. Prima [F1] **9**
   — ou —
   [◆] [|]
   [◆] **F**
2. Prima ⊕ ou ⊖ para ver o menu com as larguras válidas (**3–12**).
3. Posicione o cursor para realçar um número e prima [ENTER]. Se os números forem compostos por um único dígito, é possível digitá-lo e premir [ENTER].)
4. Prima [ENTER] para fechar a caixa de diálogo e actualizar a tabela.

Apresentação dos números numa célula

Sempre que for possível, os números são reproduzidos de acordo o modo de apresentação actual (Display Digits, Exponential Format, etc.). O número pode ser arredondado, se necessário. Todavia:

- se o tamanho do número exceder à largura da célula, o número é arredondado e apresentado em notação científica.
- se o tamanho da célula for muito pequeno para aparecer em notação científica, aparece “...”.

Nota:

- se uma função não estiver definida para um valor determinado, aparece undef.
- Utilize [MODE] para definir o modo de apresentação.

A predefinição **Display Digits = FLOAT 6**. Esta definição de modo mostra um número até seis dígitos, mesmo se a largura da célula puder conter outros. As outras definições influenciam analogamente a apresentação dos números.

			Largura da célula:	
Precisão máxima	**3**	**6**	**9**	**12**
1.2345678901	1.2	1.2346	1.23457	1.23457*
⁻123456.78	...	⁻1.2E5	⁻123457.	⁻123457.*
.000005	...	5.E⁻6	.000005	.000005
1.2345678E19	...	1.2E19	1.2346E19	1.23457E19*
⁻1.23456789012E⁻200	...	...	⁻1.2E⁻200	⁻1.2346E⁻200

***Nota:** dependendo do modo de apresentação, alguns valores não aparecem com a precisão máxima, mesmo se a célula os puder conter.

Nota: para ver um número com a precisão máxima, realce a célula e observe a linha de entrada.

Se os resultados forem números complexos

Uma célula mostra o máximo possível de um número complexo (conforme o modo de apresentação actual) seguido por "..." no final da porção apresentada.

Se realçar uma célula com um número complexo, a linha de entrada mostra a parte real e imaginária com um máximo de 4 dígitos cada uma (FLOAT 4).

Edição de uma função seleccionada

É possível modificar uma função seleccionada numa tabela sem utilizar o Y= Editor.

1. Posicione o cursor numa célula da coluna que contém a função desejada. A linha de cabeçalho da tabela contém os nomes das funções (**y1**, etc.).
2. Prima [F4] para posicionar o cursor na linha de entrada, onde aparece a função realçada.

   Nota: este recurso permite ver uma função sem sair da tabela.
3. Realize as modificações necessárias.
   - Digite a nova função, que cancela a função anterior.

     — ou —
   - Prima [CLEAR] para cancelar a função anterior e digite a nova.

     — ou —
   - Prima ◀ ou ▶ para remover o realce e modifique a função.

   Nota: para cancelar as modificações e posicionar o cursor na tabela, prima [ESC] em vez de [ENTER].

4. Prima [ENTER] para guardar a função modificada e actualize a tabela. A função também é guardada no Y= Editor.

Modificação dos parâmetros de configuração

Depois de gerar uma tabela automática, é possível modificar os seus parâmetros de configuração conforme necessário.

Prima [F2] ou [◆] [TBLSET] para ver a caixa de diálogo TABLE SETUP; realize as modificações.

Construção de uma tabela manual (Ask)

Se **Independent = ASK** na caixa de diálogo TABLE SETUP, a TI-89 / Voyage™ 200 permite criar uma tabela manual, introduzindo uma série de valores específicos para a variável independente.

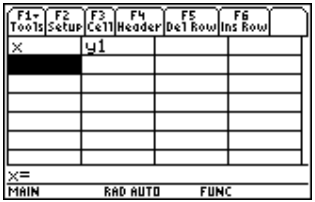
Visualizaçã do ecrã Table

Para ver o ecrã Table, prima [◆] [TABLE] ou [APPS] **5**.

Se definir **Independent = ASK** (com [◆] [TBLSET]) antes de ver uma tabela pela primeira vez, aparece uma tabela em branco. O cursor realça a primeira célula da coluna da variável independente.

A linha de cabeçalho contém os nomes da variável independente (x) e as funções seleccionadas (y1).

Introduza um valor aqui.



Se aparecer primeiro uma tabela automática e, em seguida, definir **Independent = ASK**, a tabela continua a mostrar os mesmos valores. Todavia não é possível ver outros valores, deslocando o ecrã para cima ou para baixo.

Introdução ou edição do valor de uma variável independente

É possível inserir um valor apenas na coluna 1 (variável independente).

1. Posicione o cursor para realçar a célula que deseja introduzir ou modificar.
   - Se iniciar com uma tabela vazia, os valores podem ser inseridos apenas em células consecutivas (linha 1, linha 2, etc.). Não é possível ignorar uma célula (linha 1, linha 3).
   - É possível modificar um valor contido numa célula da coluna 1.
2. Prima [F3] para posicionar o cursor na linha de entrada.
3. Digite o novo valor ou a nova expressão, ou modifique o valor existente.
4. Prima [ENTER] para introduzir o valor na tabela e actualizar os valores da função correspondente.

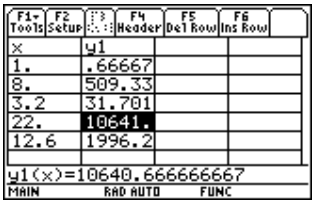
   Nota: para introduzir um novo valor numa célula, não é preciso premir [F3]; basta começar a digitá-lo.

O cursor volta à célula onde foi efectuada a introdução. Utilize a tecla ⊖ para passar para a próxima linha.

Introduza os valores em qualquer ordem numérica.

Introduza um novo valor aqui.

Mostra o valor completo da célula evidenciada.



Nota: neste exemplo é possível deslocar o cursor para a coluna 2, mas os valores devem ser introduzidos apenas na coluna 1.

Introdução de uma lista na coluna da variável independente

1. Posicione o cursor para realçar uma célula da coluna da variável independente.
2. Prima F4 para posicionar o cursor na linha de entrada.
3. Digite uma série de valores entre chavetas { } e separados por vírgulas. Por exemplo:

   x={1,1.5,1.75,2}

   É possível inserir uma variável na lista ou uma expressão cujo resultado se converta numa lista.

   Nota: os valores contidos na coluna da variável independente aparecem na forma de lista (que podem ser modificados).

4. Prima [ENTER] para deslocar os valores para a coluna da variável independente. A tabela é actualizada para mostrar os valores da função correspondente.

Adição, cancelamento de eliminação

Para:	Operação:
Inserir uma nova linha em cima de uma linha especificada	Realce uma célula na linha especificada e prima: [2nd] [F6] [F6] A nova linha fica indefinida (undef) até que digite um valor para a variável independente.
Cancelar uma linha	Realce uma célula na linha e prima [F6]. Se realçar uma célula na coluna da variável independente, é possível premir [←].
Apagar a tabela inteira (mas não as funções seleccionadas em Y=)	Prima [F1] **8**. Após o pedido de confirmação, prima [ENTER].

Largura da célula e formatos de apresentação

A apresentação dos números numa tabela depende de numerosos factores.

No ecrã principal ou num programa

A variável de sistema **tblInput** contém uma lista de todos os valores da variável independente inseridos na tabela, inclusive aqueles que não estão actualmente

apresentados. Esta variável **tblInput** também é utilizada para tabelas automáticas, mas contém apenas os valores da variável independente actualmente apresentados.

Antes de ver uma tabela, é possível guardar uma lista de valores directamente na variável de sistema **tblInput**.

Tópicos adicionais da representação gráfica

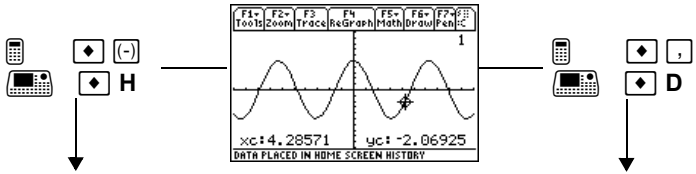
Recolha de pontos de dados de um gráfico

É possível guardar conjuntos de valores de coordenadas e/ou resultados matemáticos para análise posterior no ecrã Graph. As informações podem ser guardadas como uma matriz de linha única (vector) no ecrã principal ou como pontos de dados numa variável de dados do sistema que pode ser aberta no Data/Matrix Editor.

Recolha de pontos

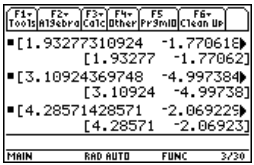
1. Veja o gráfico. (Este exemplo mostra **y1(x)=5∗cos(x)**.)
2. Veja as coordenadas ou resultados matemáticos que deseja recolher.
3. Para guardar as informações no ecrã principal ou na variável *sysData*.
   [◆] [(-)] (Ecrã principal) ou [◆] [,] (variável *sysData*)
   [◆] **H** (Ecrã principal) ou [◆] **D** (variável *sysData*)
4. Repita o processo, se necessário.

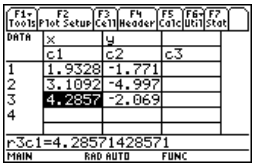
   Nota: para ver as coordenadas ou resultados matemáticos, trace uma função com [F3] ou realize uma operação com [F5] **Math** (como a obtenção do valor **Minimum** ou **Maximum**). Pode também utilizar o cursor de movimento livre.



As coordenadas apresentadas são adicionadas à área do histórico do ecrã principal (mas não à linha de entrada) como um vector ou matriz de linha

As coordenadas apresentadas são guardadas numa variável de dados chamada *sysData*, que pode ser aberta no Data/Matrix Editor.





Nota: utilize os ecrãs divididos para ver um gráfico e o ecrã principal ou o Data/Matrix Editor ao mesmo tempo.

Notas sobre a variável SysData

- Se premir:

  D

  - Se a variável *sysData* não existir, é criada na pasta **MAIN**.
  - Se *sysData* já existir, os novos dados são concatenados no fim dos dados existentes. Os cabeçalhos de coluna ou títulos existentes (das colunas afectadas) são apagados; os títulos são substituídos por títulos adequados aos novos dados.

- A variável *sysData* pode ser apagada, excluída etc, como qualquer outra variável de dados, mas não pode ser protegida.
- Se no ecrã Graph existir uma função ou um gráfico estatístico que faça referência ao conteúdo actual de *sysData*, não funcionará.

Representação gráfica de uma função definida no ecrã principal

Em muitos casos, uma função ou expressão pode ser criada no ecrã principal e, só posteriormente, ser representada graficamente. É possível copiar uma expressão para Y= Editor ou representá-la graficamente no ecrã principal sem utilizar oY= Editor.

O que é variável independente "original"?

No Y= Editor, tem de definir todas as funções em termos da variável independente "original" do modo Graph actual.

Modo Graph	Variável independente original
Function	x
Parametric	t
Polar	θ
Sequence	n
3D	x, y

Modo Graph	Variável independente original
Equação diferencial	t

Cópia do ecrã principal para Y= Editor

Uma expressão apresentada no ecrã principal pode ser copiada para oY= Editor através de um dos seguintes métodos.

Método	Descrição
Copiar e colar	1. Coloque a expressão realçada no ecrã principal. Prima [F1] e seleccione **5:Copy**. 2. Veja o Y= Editor, realce a função desejada e prima [ENTER]. 3. Prima [F1] e seleccione **6:Paste**. Em seguida, prima [ENTER]. **Nota:** Em vez de utilizar [F1] **5** ou [F1] **6** para copiar e colar, utilize: [◆] [COPY] ou [◆] [PASTE]. [◆] **C** (copy) ou [◆] **V** (colar).
[STO▸]	Guarda a expressão num nome de função Y=. 2x^3+3x^2−4x+12→y1(x) Utilize o nome completo da função: y1(x) em vez de y1. **Nota:** para copiar uma expressão da área do histórico do ecrã principal para a linha de entrada, utilize a função de colar automático ou copiar e colar.

Método	Descrição
Comando **Define**	Define a expressão como uma função Y= definida pelo utilizador. `Define y1(x)=2x^3+3x^2−4x+12` **Nota:** o comando **Define** está disponível no menu [F4] da barra de ferramentas do ecrã principal.
[2nd] [RCL]	Se a expressão já estiver guardada numa variável: 1. Veja o Y= Editor, realce a função desejada e prima [ENTER]. 2. Prima [2nd] [RCL]. Digite o nome da variável que contém a expressão e prima duas vezes [ENTER]. **Importante:** para recuperar a variável de uma função como **f1(x)**, digite apenas a parte inicial **f1** e não o nome completo da função. 3. Prima [ENTER] para guardar a expressão recuperada na lista de função do Y= Editor. **Nota:** [2nd] [RCL] é útil quando uma expressão é guardada numa variável ou função que não corresponde a Y= Editor, como **a1** ou **f1(x)**.

Representação gráfica no ecrã principal

O comando **Graph** permite representar uma expressão graficamente no ecrã principal sem utilizar o Y= Editor. Diferente de Y= Editor, **Graph** permite especificar uma

expressão em termos de qualquer valor independente, sem levar em conta o modo actual de representação gráfica.

Se a expressão estiver em termos:	Utilize o comando Graph como mostrado neste exemplo:
Da variável independente original	Graph 1.25x∗cos(x) Na representação gráfica de uma função, x é a variável original.
De uma variável independente não original	Graph 1.25a∗cos(a),a Especifique a variável independente; caso contrário, pode ser obtido um erro.

Nota: Graph utiliza as configurações actuais da variável Window e está disponível no menu F4 da barra de ferramentas do ecrã principal.

O **Graph** não funciona com gráficos de sucessões ou de equações diferenciais. Para gráficos de equações paramétricas, polares e tridimensionais, utilize as seguintes variações:

No modo de representação PARAMETRIC:	**Graph** *xExpr*, *yExpr*, *t*
No modo de representação POLAR:	**Graph** *expr*, θ
No modo de representação 3D:	**Graph** *expr*, *x*, *y*

Nota: para criar uma tabela no ecrã principal, utilize o comando **Table**. É parecido com **Graph**. Ambos partilham as mesmas expressões.

O **Graph** não copia a expressão para o Y= Editor. Em vez disso, suspende temporariamente todas as funções seleccionadas no Y= Editor. Pode traçar, efectuar zoom ou mostrar e editar expressões de **Graph** no ecrã Table, assim como as funções do Y= Editor.

Limpeza do ecrã Graph

Sempre que executar **Graph**, a nova expressão é adicionada às existentes. Para limpar os gráficos:

- Execute o comando **ClrGraph** (disponível no menu F4 **Other** da barra de ferramentas do ecrã principal).

  – ou –

- Veja o Y= Editor. A próxima vez que o ecrã Graph for apresentado, utilizará as funções seleccionadas no Y= Editor.

Outros benefícios das funções definidas pelo utilizador

Uma função definida pelo utilizador pode ser definida em termos de qualquer variável independente. Por exemplo:

Definida em termos de “aa”.

```
Define f1(aa)=1.25aa*cos(aa)
Graph f1(x)
```

Faz referência à função através da variável independente original.

e:

```
Define f1(aa)=1.25aa*cos(aa)
f1(x)→y1(x)
```

Representação gráfica de uma função definida por partes

Para representar graficamente uma função definida por partes, tem de definir primeiro a função, especificando os limites e as expressões de cada parte. A função **when** é extremamente útil para funções de duas partes. Para três ou mais partes, pode ser mais fácil criar uma função definida pelo utilizador com várias instruções.

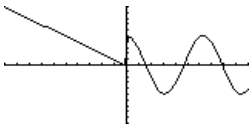
Utilização da função When

Para definir uma função de duas partes, utilize a sintaxe:

when(*condição*, *ExpressãoVerdadeira*, *ExpressãoFalsa***)**

Por exemplo, suponha que deseja representar graficamente uma função com duas partes.

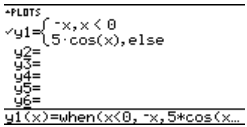
Quando:	Utilize a expressão:
$x < 0$	$-x$
$x \geq 0$	$5 \cos(x)$



No Y= Editor:

A função aparece no modo Pretty Print nesta forma.

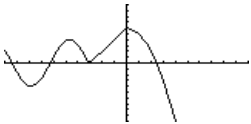
Introduza a função nesta forma.



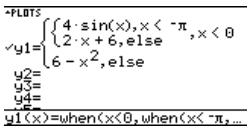
Para três ou mais partes, pode utilizar a função **when** várias vezes.

Nota: Para digitar **when**, digite-o ou utilize **CATALOG**..

Quando:	**Utilize a expressão:**
$x < -\pi$	$4 \sin(x)$
$x \geq -\pi$ and $x < 0$	$2x + 6$
$x \geq 0$	$6 - x^2$



Em Y= Editor



onde:

```
y1(x)=when(x<0,when(x< - π,4∗sin(x),2x+6),6−x^2)
```

Esta função é válida quando x<0.

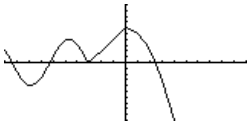
As funções com vários when tornam-se rapidamente complexas e difíceis de visualizar.

Utilização de uma função definida pelo utilizador com várias instruções

Para três ou mais partes, o utilizador pode definir uma função com várias instruções.

Por exemplo, considere a função de três partes apresentada anteriormente.

Quando:	Utilize a expressão:
$x < -\pi$	4 sin(x)
$x \geq -\pi$ and $x < 0$	2x + 6
$x \geq 0$	$6 - x^2$



Nota: para mais informações sobre semelhanças e diferenças entre funções e programas, consulte o módulo *Programação*.

Uma função definida pelo utilizador com várias instruções pode ter algumas estruturas de controlo e tomadas de decisão (**If**, **ElseIf**, **Return**, etc.) utilizadas em programação. Se criar a estrutura de uma função, pode ser útil visualizá-la na forma de bloco.

```
❶ Func
    If x<-π Then
    Return 4*sin(x)
  ElseIf x>=-π and x<0 Then
    Return 2x+6
  Else
    Return 6-x^2
  EndIf
❶ EndFunc
```

❶ **Func** e **EndFunc** designam o início e o fim de uma função.

Deve introduzir uma função com várias instruções no ecrã principal ou no Y= Editor numa única linha.

Utilize dois pontos (:) para separar cada instrução.

```
Func:If x< ⁻π Then:Return 4∗sin(x): ... :EndIf:EndFunc
```

No Y= Editor:

Apenas Func aparece numa função com várias partes.

Introduza uma função de várias partes numa linha. Inclua dois pontos para

No ecrã principal ou num programa

Pode utilizar o comando **Define** para criar uma função definida pelo utilizador com várias instruções no ecrã principal.

Estão disponíveis informações sobre como copiar uma função do ecrã principal para o Y= Editor.

O Program Editor (no módulo *Programação*) permite criar uma função definida pelo utilizador. Por exemplo, utilize o Program Editor para criar uma função denominada **f1(xx)**. No Y= Editor, defina **y1(x) = f1(x)**.

Representação gráfica de uma família de curvas

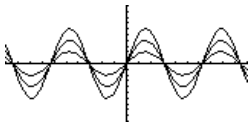
A introdução de uma lista numa expressão permite representar uma função para cada valor da lista. (Não é possível representar graficamente uma família de curvas nos modos de representação gráfica SEQUENCE e 3D.)

Exemplos de utilização do Y= Editor

Introduza a expressão **{2,4,6} sin(x)** e represente graficamente as funções.

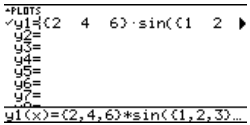
Nota: agrupe os elementos de uma lista entre chavetas (2nd [{] e 2nd [}]) e separe-os com vírgulas.

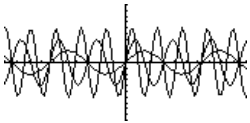
```
*PLOTS
✓y1={2  4  6}·sin(x)
 y2=
 y3=
 y4=
 y5=
 y6=
 y7=
y1(x)={2,4,6}*sin(x)
```



Representação gráfica
de três funções:
2 sin(x), 4 sin(x), 6 sin(x)

Introduza a expressão **{2,4,6} sin({1,2,3} x)** e represente graficamente as funções.





Representação gráfica de três funções:
2 sin(x), 4 sin(2x), 6 sin(3x)

Nota: as vírgulas aparecem na linha de entrada, mas não na lista da função.

Exemplo da utilização do comando Graph

De modo análogo, é possível utilizar o comando **Graph** no ecrã principal ou num programa.

```
graph {2,4,6}sin(x)
graph {2,4,6}sin({1,2,3}x)
```

Representação gráfica simultânea com listas

Quando o formato gráfico estiver definido como **Graph Order = SIMUL**, as funções são representadas graficamente em grupos de acordo com o número de elementos da lista.

```
*PLOTS
✓y1={2  4  6}·sin(x)
✓y2={1  2  3}·x+4
✓y3=cos(x)
```

Para as funções deste exemplo, a TI-89 Titanium / Voyage™ 200 Graphing Calculator representa graficamente três grupos.

- 2 sin(x), x+4, cos(x)
- 4 sin(x), 2x+4
- 6 sin(x), 3x+4

As funções dentro de cada grupo são representadas graficamente de forma simultânea, mas os grupos são representados graficamente na forma sequencial.

Nota: para configurar formatos gráficos nos ecrã do Y= Editor, Window Editor ou Graph, prima:

◆ |

◆ **F**

Representação gráfica de uma família de curvas

Se premir ⊖ ou ⊕, move o cursor de representação gráfica para a próxima curva ou a curva anterior da mesma família antes de passar para a próxima função ou a função anteriormente seleccionada.

Utilização do modo de dois gráficos

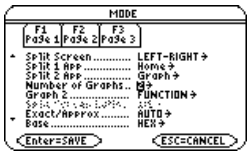
No modo de dois gráficos, os recursos de funções gráficas da TI-89 Titanium / Voyage™ 200 Graphing Calculator são duplicados, fornecendo dois cálculos para gráficos independentes. O modo de dois gráficos está disponível apenas no modo de ecrãs divididos. Para mais informações sobre os ecrãs divididos, consulte o módulo *Ecrãs divididos*.

Definição do modo

Várias definições de modo afectam o modo de dois gráficos, mas apenas duas são indispensáveis. Ambas estão na página 2 da caixa de diálogo **MODE**.

1. Prima [MODE]. Em seguida, prima [F2] para ver a página 2.

2. Defina os seguintes modos necessários.



   - **Split Screen = TOP-BOTTOM** ou **LEFT-RIGHT**
   - **Number of Graphs = 2**

3. Opcionalmente, é possível definir os seguintes modos.

   Pagina 1:
   - **Graph** = modo **Graph** para a parte superior ou esquerda do ecrã dividido

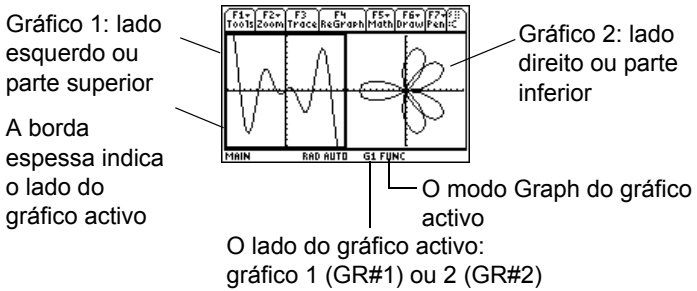
Pagina 2:

- **Split 1 App** = aplicação para a parte superior ou esquerda
- **Split 2 App** = aplicação para a parte inferior ou direita
- **Graph 2** = modo **Graph** para a parte da direita ou inferior
- **Split Screen Ratio** = dimensões relativas aos dois ecrãs (Voyage™ 200)

4. Prima [ENTER] para fechar a caixa de diálogo.

Ecrã de dois gráficos

Um ecrã de dois gráficos é semelhante a um ecrã normal dividido.



Recursos gráficos independentes

Ambos os gráficos 1 e 2 possuem recursos independentes:

- Modos Graph (FUNCTION, POLAR etc.). Outros modos como **Angle**, **Display Digits** etc. são partilhados e afectam os dois gráficos.
- As variáveis de Window Editor.
- Os parâmetros de configuração de tabela e ecrãs Table.
- Os formatos gráficos como **Coordinates**, **Axes** etc.
- Os ecrãs de gráficos.
- Y= Editors. Os dois gráficos partilham definições de gráficos estatísticos e funções comuns.

  Nota: Y= Editor só é completamente independente quando os dois gráficos possuem modos de representação gráfica diferente, como descrito abaixo.

As aplicações gráficas independentes (Y= Editor, ecrã Graph etc.) podem ser apresentadas nos dois lados do ecrã ao mesmo tempo.

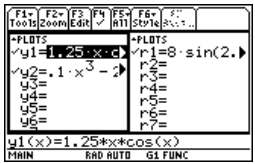
As aplicações não gráficas (ecrã principal, Matrix and Data Editor, etc.) são partilhadas e podem aparecer apenas num dos lados num determinado momento.

Y= Editor no modo de dois gráficos

Mesmo no modo de dois gráficos, só existe um Y= Editor, que mantém uma única lista de funções para cada definição do modo Graph. Entretanto, se os dois lados utilizarem

o mesmo modo de representação gráfica, cada um dos lados pode seleccionar funções diferentes da única lista.

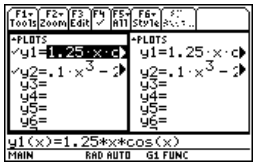
- Quando os dois lados utilizarem modos de representação gráfica diferentes, cada lado apresenta uma lista diferente de funções.



- Quando os dois lados utilizarem o mesmo modo de representação gráfica, os lados apresentam a mesma lista de funções.
  - É possível utilizar [F2] para seleccionar gráficos estatísticos e funções diferentes (indicados por ✓) para cada lado.
  - Se definir um estilo de apresentação para uma função, é utilizado nos dois lados.

    [2nd] [F6]

    [F6]



- Suponha que os gráficos 1 e 2 estão definidos para representação gráfica de uma função. Apesar de os dois lados terem a mesma lista de funções, pode seleccionar (✓) funções diferentes para representar graficamente.

Nota: as alterações (redefinição de função, mudança de estilo etc.) feitas no Y= Editor activo não são reflectidas no lado inactivo até que seja alterado.

Utilização de ecrãs divididos

Para mais informações sobre os ecrãs divididos, consulte o módulo *Ecrãs divididos*.

- Para alternar de um lado gráfico para outro, prima [2nd] [⊞] (segunda função de [APPS]).
- Para ver aplicações diferentes:
  - Alterne para o lado apropriado do gráfico e mostre a aplicação como faria normalmente.

    – ou –
  - Utilize [MODE] para mudar **Split 1 App** e/ou **Split 2 App**.
- Para sair do modo de dois gráficos:
  - Utilize [MODE] para definir **Number of Graphs = 1** ou saia dos ecrãs divididos através da definição de **Split Screen = FULL**.

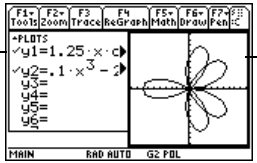
    – ou –
  - Prima duas vezes [2nd] [QUIT] para sair do modo de ecrãs divididos e voltar para ao ecrã principal no tamanho máximo.

  Nota: as aplicações não gráficas (como o ecrã principal) não podem aparecer simultaneamente nos dois lados.

Não se esqueça de que os dois lados são independentes

No modo de dois gráficos, os dois lados podem parecer estar relacionados quando, de facto, não estão. Por exemplo:

No gráfico 1, Y= Editor lista as funções y(x).



No gráfico 2, o gráfico polar utiliza equações r(θ) não apresentadas.

No ecrã principal ou num programa

Depois de definir o modo de dois gráficos, as operações relacionadas com os gráficos passam a referir-se ao lado do gráfico activo. Por exemplo:

```
10→xmax
```

afecta o gráfico 1 ou o gráfico 2, dependendo do lado que estiver activo quando o comando for executado.

Para activar alternadamente um lado ou outro, prima [2nd] [⇄] ou utilize a função **switch**, **switch(1)** ou **switch(2)**.

Traçado de uma função ou da inversa de uma função num gráfico

Para efeitos de comparação, pode representar graficamente uma função sobre outra já representada. Normalmente, esta função traçada é uma variação do gráfico existente. A inversa de uma função pode ser também representada. (Estas operações não estão disponíveis para representações gráficas tridimensionais.)

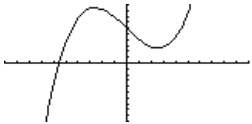
Traçado de funções ou equações polares ou paramétricas

Execute **DrawFunc**, **DrawParm** ou **DrawPol** no ecrã principal ou num programa. Não é possível representar uma função ou equação de forma interactiva no ecrã Graph.

DrawFunc *expressão*
DrawParm *expressão1*, *expressão2* [,*tmin*] [,*tmax*] [,*tstep*]
DrawPol *expressão* [,θ*min*] [,θ*max*] [,θ*step*]

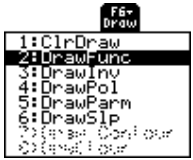
Por exemplo:

1. Defina **y1(x)=.1x^3–2x+6** no Y= Editor e represente graficamente a função.



2. No ecrã Graph, prima:
   [2nd] [F6]
   [F6]
   e seleccione **2:DrawFunc**.

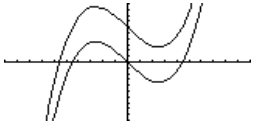
   Para ver o ecrã principal e colocar **DrawFunc** na linha de entrada, prima:
   [2nd] [F6] **2**
   [F6] **2**



3. No ecrã principal, especifique a função que deseja representar.

```
DrawFunc y1(x)−6
```

4. Prima [ENTER] para traçar a função no ecrã Graph.



Não é possível ampliar, traçar ou realizar uma operação matemática numa função traçada.

Nota: para apagar a função desenhada, prima, [F4]
– ou –
[calc] [2nd] [F6] e seleccione **1:ClrDraw**.
[TI-84] [F6] e seleccione **1:ClrDraw**.

Traçado da inversa de uma função

Execute **DrawInv** no ecrã principal ou num programa. Não é possível traçar a inversa de uma função de forma interactiva no ecrã Graph.

DrawInv *expressão*

Por exemplo, utilize o gráfico de **y1(x)=. 1x^3–2x+6**, como mostrado acima.

1. No ecrã Graph, prima:
   [2nd] [F6]
   [F6]
   e seleccione **3:DrawInv**.

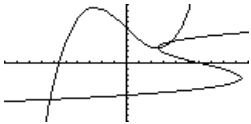
   Para ver o ecrã principal e colocar **DrawInv** na linha de entrada, prima:
   [2nd] [F6] **3**
   [F6] **3**

2. No ecrã principal, especifique a função inversa.

   DrawInv y1(x)

3. Prima [ENTER].

   A função inversa é traçada como **(y,x)** em vez de **(x,y)**.



Traçado de uma recta, uma circunferência ou um rótulo de texto num gráfico

Pode traçar um ou mais objectos no ecrã Graph para comparações. Por exemplo, represente uma recta horizontal para mostrar que dois pontos de um gráfico possuem o mesmo valor de y. (Alguns objectos não estão disponíveis para representações tridimensionais.)

Eliminação de todos os objectos traçados

Um objecto traçado não é parte do gráfico. É traçado “sobre ele” e permanece no ecrã até ser apagado.

No ecrã principal:

- [2nd] [F6]
  [F6]
  e seleccione **1:ClrDraw**.

  – ou –

- Prima [F4] para voltar a traçar o gráfico.

Nota: é possível também introduzir **ClrDraw** na linha de entrada do ecrã principal.

As operações que fazem com que a função Smart Graph trace novamente o gráfico (como mudar as variáveis Window ou desmarcar uma função no Y= Editor) podem ser utilizadas com o mesmo objectivo.

Traçado de um ponto ou uma recta com a mão livre

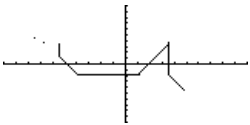
No ecrã Graph:

1. [2nd] [F7]
   [F7]
   e seleccione **1:Pencil**.
2. Mova o cursor para a posição desejada.

Para traçar:	Operação:
Um ponto (tamanho do pixel)	Prima [ENTER].
Uma recta com a mão livre	Prima sem soltar [↑] e mova o cursor para traçar a recta. Prima sem soltar [⊙], movendo o cursor para desenhar a linha. Para interromper o desenho da linha, solte [↑] ou [⊙].

Nota: se traçar uma recta com a mão livre, o cursor pode ser movido na diagonal.

O modo “lápis” continua activo após a recta ou ponto ser desenhado.



- Para continuar a desenhar, mova o cursor para um outro ponto.
- Para sair, prima [ESC].

Nota: se um traçado for feito sobre um pixel branco, é desenhado um ponto ou uma recta preta. Se o traçado for feito sobre um pixel preto, um ponto ou recta branca (que pode funcionar como uma borracha de apagar) é desenhada.

Remoção de partes individuais de um objecto traçado

No ecrã Graph:

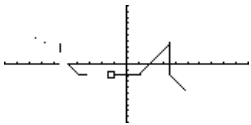
1. [2nd] [F7]
   [F7]
   e seleccione **2:Eraser**. O cursor aparece como uma pequena caixa.

2. Mova o cursor para a posição desejada.

Para remover:	Operação:
A área sobre a caixa	Prima [ENTER].
Ao longo de uma recta traçada com a mão livre	Prima sem soltar [↑][F1] movendo o cursor para apagar a linha. Prima sem soltar [ ], movendo o cursor para apagar a linha. Para sair, solte [↑] ou [ ].

Nota: esta técnica remove também partes de funções representadas graficamente.

O modo de “remover” permanece activo após a remoção ter sido feita.



- Para continuar a remover, mova o cursor da caixa para uma outra posição.
- Para sair, prima [ESC].

Traçado de uma recta entre dois pontos

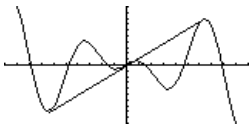
No ecrã Graph:

1. [2nd] [F7]
   [F7]
   e seleccione **3:Line**.
2. Mova o cursor para o primeiro ponto e prima [ENTER].

3. Mova para o segundo ponto e prima [ENTER]. (Conforme a movimentação é feita, a recta é traçada a partir do primeiro ponto até o cursor.)

   Nota: utilize [2nd] para mover o cursor em incrementos maiores; [2nd] ▶, etc.

O modo de "recta" permanece activo após a recta ser desenhada.



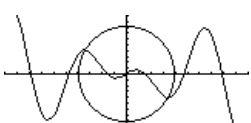
- Para traçar uma outra recta, mova o cursor para a nova posição inicial.
- Para sair, prima [ESC].

Traçado de uma circunferência

No ecrã Graph:

1. [2nd] [F7]
   [F7]
   e seleccione **4:Circle**.

2. Mova o cursor para o centro da circunferência e prima [ENTER].
3. Mova o cursor para definir o raio e prima [ENTER].



   Nota: utilize [2nd] para mover o cursor em incrementos maiores; [2nd] ▶, etc.

Traçado de uma recta horizontal ou vertical

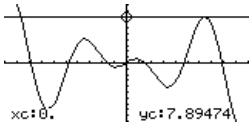
No ecrã Graph:

1. [2nd] [F7]
   [F7]
   e seleccione **5:Horizontal** ou **6:Vertical**. Uma recta horizontal ou vertical e um cursor intermitente aparecem no ecrã.

   Se a recta aparecer inicialmente sobre um eixo, pode ser difícil vê-la. Entretanto, o cursor intermitente pode ser facilmente notado.

2. Utilize as teclas de cursor para mover a recta para a posição apropriada. Em seguida, prima [ENTER].

   O modo "recta" permanece após a recta ser traçada.

   - Para continuar, mova o cursor para uma outra posição.
   - Para sair, prima [ESC].

   Nota: utilize [2nd] para mover o cursor em incrementos maiores; [2nd] ▶, etc.

Traçado de uma recta tangente

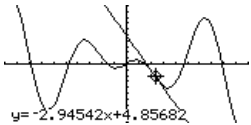
Para traçar uma recta tangente, utilize o menu [F5] **Math** da barra de ferramentas. No ecrã Graph:

1. Prima [F5] e seleccione **A:Tangent**.

2. Se necessário, utilize ⊖ e ⊕ para seleccionar a função desejada.

3. Mova o cursor para o ponto de tangência e prima [ENTER].

   A recta tangente é traçada e a sua equação é apresentada.



   Nota: para definir o ponto de tangência, pode também digitar o valor de x e premir [ENTER].

Traçado de uma recta com base num ponto e numa inclinação

Para traçar uma recta através de um ponto especificado com uma inclinação especificada, execute o comando **DrawSlp** no ecrã principal ou num programa. Utilize a sintaxe:

DrawSlp *x*, *y*, *inclinação*

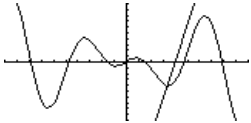
É também possível aceder a **DrawSlp** a partir do ecrã Graph.

1. [2nd] [F6]
   [F6]
   e seleccione **6:DrawSlp**. Este procedimento alterna para o ecrã principal e coloca **DrawSlp** na linha de entrada.

2. Complete o comando e prima [ENTER].

```
DrawSlp 4,0,6.37
```

A TI-89 Titanium / Voyage™ 200 Graphing Calculator alterna automaticamente para o ecrã **Graph** e traça a recta.

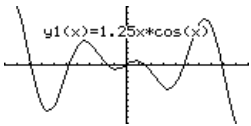


Digitação de rótulos de texto

No ecrã Graph:

1. [2nd] [F7]
   [F7]
   e seleccione **7:Text**.
2. Posicione o cursor de texto no local que desejar começar a digitar.
3. Digite o rótulo do texto.

   O modo “texto” permanece activo após o texto ter sido digitado.

   - Para continuar, mova o cursor para outra posição.
   - Para sair, prima [ENTER] ou [ESC].

   Nota: o cursor de texto indica o canto superior esquerdo do próximo carácter a ser digitado.

No ecrã principal ou num programa

Há comandos disponíveis para traçar todos os objectos descritos nesta secção. Há ainda comandos (como **PxlOn**, **PxlLine**, etc.) que permitem traçar objectos, especificando a posição exacta dos pixéis no ecrã.

Para obter uma lista desses comandos de traçamento disponíveis, consulte “Traçado de um gráfico no ecrã Graph” no módulo *Programação*.

Armazenamento e abertura da imagem gráfica

Uma imagem do ecrã Graph actual pode ser guardada numa variável **PICTURE** (ou **PIC**). Posteriormente, é possível abrir a variável e ver a imagem. Este procedimento guarda apenas a imagem, não as configurações do gráfico utilizadas para gerar a imagem.

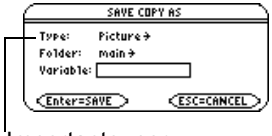
Armazenamento da imagem completa na Graph

Uma imagem inclui qualquer função traçada, eixos, marcas e objectos desenhados. A imagem não inclui indicadores de limites inferior e superior, as mensagens nem as coordenadas do cursor.

Veja o ecrã Graph. Em seguida:

1. Prima [F1] e seleccione **2:Save Copy As**.

2. Especifique o tipo (**Picture**), a pasta e um nome exclusivo para a variável.
3. Prima [ENTER]. Depois de digitar dados numa caixa de entrada como **Variable**, tem de premir duas vezes [ENTER].



Importante: por predefinição Type = GDB (para base de dados de gráfico). Tem de definir

Armazenamento de uma parte do ecrã Graph

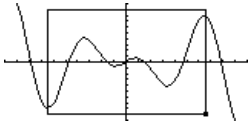
É possível definir uma caixa rectangular que delimite uma parte do ecrã Graph que deseje guardar.

1. [2nd] [F7]
   [F7]
   e seleccione **8:Save Picture**.

   Um rectângulo aparece à volta da margem externa do ecrã.

   Nota: não é possível guardar uma parte de um gráfico tridimensional.

2. Seleccione o primeiro canto do rectângulo, movendo as partes superior e esquerda. Em seguida, prima [ENTER].

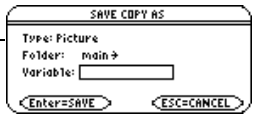


   Nota: utilize ⊖ e ⊕ para mover a parte superior ou inferior e utilize ⊕ e ⊖ para mover os lados.

3. Seleccione o segundo canto, movendo as partes inferior e direita. Em seguida, prima [ENTER].

4. Especifique a pasta e um nome exclusivo de variável.

5. Prima [ENTER]. Depois de digitar os dados numa caixa de entrada como **Variable**, tem de premir duas vezes [ENTER].



Nota: quando uma parte de um gráfico é guardada, Type é definido automaticamente como Picture.

Abertura de uma imagem gráfica

Quando abrir uma imagem gráfica, sobrepõe-se ao ecrã Graph actual. Para ver apenas a imagem a abrir, utilize o Y= Editor para desmarcar todas as outras funções antes de a abrir.

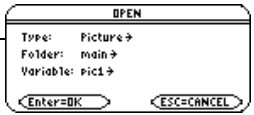
No ecrã Graph:

1. Prima [F1] e seleccione **1:Open**.

2. Seleccione o tipo (**Picture**), a pasta e a variável da imagem do gráfico que deseja abrir.

   Nota: se não aparecer nenhum nome de variável na caixa de diálogo, é porque não há imagens gráficas na pasta.

3. Prima [ENTER].



Importante: por predefinição, Type = GDB (para o base de dados gráfico). Certifique-se de definir Type = Picture.

Uma imagem gráfica é um objecto de desenho. Não é possível traçar nenhuma curva sobre uma imagem.

Para imagens armazenadas de uma parte do ecrã Graph

Quando premir [F1] e seleccionar **1:Open**, a imagem é sobreposta a partir do canto superior esquerdo do ecrã Graph. Se a imagem guardada for de uma parte do ecrã Graph, pode parecer deslocada em relação ao gráfico sobre a imagem.

Para especificar o pixel do ecrã a utilizar como o vértice do canto superior esquerdo, consulte “No ecrã principal ou num programa” na página 6.)

Remoção de uma imagem gráfica

As variáveis Picture não desejadas ocupam desnecessariamente memória da calculadora. Para remover uma variável, utilize o ecrã VAR-LINK ([2nd] [VAR-LINK]) como descrito no módulo *Gestão da memória e das variáveis*.

Num programa ou no ecrã principal

Para guardar (armazenar) e abrir (recuperar) uma imagem gráfica, utilize os comandos **StoPic**, **RclPic**, **AndPic**, **XorPic** e **RplcPic** como descrito no módulo *Referência técnica* .

Para ver uma série de imagens gráficas como uma animação, utilize o comando **CyclePic**. Veja o exemplo desta operação no comando CyclePic.

Animação de uma série de imagens gráficas

Como descrito anteriormente neste módulo, é possível guardar a imagem de um gráfico. O comando **CyclePic** permite ver uma série de imagens gráficas para criar uma animação.

Comando CyclePic

Antes de utilizar **CyclePic**, tem de dispor de imagens gráficas que tenham como base o mesmo nome e sejam numeradas sequencialmente, a partir do 1 (como img1, img2, img3. . .).

Para circular entre as imagens, utilize a sintaxe:

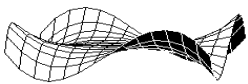
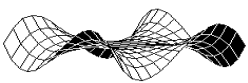
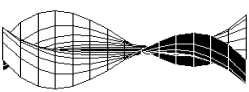
CyclePic *NomeCadeiaImg*, *n* [,*espera*] [,*ciclos*] [,*direção*]
❶ ❷ ❸ ❹ ❺

❶ nome base das imagens, entre aspas, como "img"
❷ nº de imagens a circular
❸ segundos entre as imagens
❹ nº de vezes a repetir o ciclo

Exemplo

Este programa (denominado **cyc**) ilustra uma animação com 10 visualizações de um gráfico tridimensional, tendo cada visualização sofrido uma rotação de 10° à volta do eixo Z. Para mais informações sobre estes comandos, consulte o módulo *Referência*

técnica . Para mais informações sobre a utilização do Program Editor, consulte o módulo *Programação*.

Listagem do programa	Gráficos (alternados) do programa
:cyc()	
:Prgm	
:local I	
:◉Set mode and Window variables	
:setMode(“graph”,”3d”)	
:70→eyeϕ	
:⁻10→xmin	
:10→xmax	
:14→xgrid	
:⁻10→ymin	
:10→ymax	
:14→ygrid	
:⁻10→zmin	
:10→zmax	
:1→zscl	
:◉Define the function	
:(x^3*y–y^3*x)/390→z1(x,y)	
:◉Generate pics and rotate	
:For i,1,10,1	
: i*10→eyeθ	
: DispG	
: StoPic #("pic" & string(i))	
:EndFor	
:◉Display animation	
:CyclePic "pic",10,.5,5,⁻1	
:EndPrgm	

Comentários começam com ◉. Prima:

[◆] [)]

[2nd] **X**

Nota: este programa leva vários minutos a executar devido à sua complexidade.

Depois de introduzir este programa no Program Editor, vá para o ecrã principal e introduza **cyc( )**.

Armazenamento e abertura de uma base de dados gráficos

Uma base de dados gráficos é o conjunto de todos os elementos que definem o gráfico. Se guardar o banco de dados gráficos como uma variável GDB permite que o gráfico seja recriado posteriormente, sempre que a variável da base de dados guardada for aberta.

Elementos de uma base de dados gráficos

Uma base de dados gráficos é composta por:

- Configurações de modo ([MODE]) para **Graph**, **Angle**, **Complex Format** e **Split Screen** (apenas se o modo de dois gráficos estiver a ser utilizado).
- Todas as funções em Y= Editor ([◆] [Y=]), incluindo os estilos de apresentação e as funções seleccionadas.
- Os parâmetros de Table ([◆] [TBLSET]), as variáveis Window ([◆] [WINDOW]) e os formatos gráficos
  [F1] **9**

– ou –

◆ |

◆ F

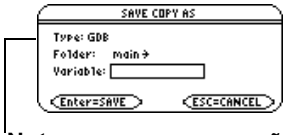
O base de dados gráficos não inclui objectos traçados ou gráficos estatísticos.

Nota: os elementos dos dois gráficos são guardados numa única base de dados no modo de dois gráficos.

Armazenamento da base de dados do gráfico actual

No Y= Editor, no Window Editor, no ecrã Table ou Graph:

1. Prima F1 e seleccione **2:Save Copy As**.
2. Especifique a pasta e um nome de variável exclusivo.
3. Prima ENTER. Depois de digitar os dados numa caixa de entrada como Variable, tem de premir duas vezes ENTER.



Nota: se começar no ecrã Graph, utilize Type=GDB.

Abertura de uma base de dados gráficos

Precaução: quando abrir a base de dados gráficos, as suas informações substituem as informações da base de dados actual, que são perdidas. Para evitar que isto ocorra, guarde a base de dados gráficos actual antes de abrir outra base de dados.

No Y= Editor, no Window Editor, no ecrã Table ou Graph:

1. Prima [F1] e seleccione **1:Open**.
2. Seleccione a pasta e a variável que contêm a base de dados gráficos que deseja abrir.
3. Prima [ENTER].

Nota: se começar no ecrã Graph, utilize Type=GDB.

Remoção de uma base de dados gráficos

As variáveis GDB não utilizadas ocupam desnecessariamente a memória da calculadora. Para remover as variáveis, utilize o ecrã VAR-LINK ([2nd] [VAR-LINK]) descrito no módulo *Gestão da memória e das variáveis*.

Num programa ou no ecrã principal

É possível guardar (armazenar) e abrir (recuperar) uma base de dados gráficos com os comandos **StoGDB** e **RclGDB** como descrito no módulo *Referência técnica*.

Ecrãs divididos

Definição e saída do modo de ecrãs divididos

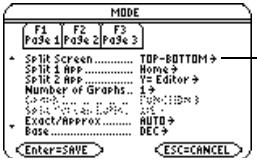
Para definir o modo de ecrãs divididos, utilize a caixa de diálogo MODE para especificar as definições deste modo. Depois de dividir o ecrã, permanece assim até que seja alterado novamente.

Definição do modo de ecrãs divididos

1. Prima [MODE] para ver a caixa de diálogo **MODE**.
2. Como os modos relacionados com os ecrãs divididos estão descritos na segunda página da caixa de diálogo **MODE**, pode:
   - Utilizar ⊖ para ir para baixo.

     — ou —
   - Premir [F2] para ver página 2.
3. Defina o modo **Split Screen** como uma das definições abaixo. Para mais informações sobre os procedimentos utilizar uma definição de modo, consulte o módulo *Funcionamento da calculadora*.

Definições de ecrãs divididos
TOP-BOTTOM
LEFT-RIGHT



Quando definir Split Screen = TOP-BOTTOM ou LEFT-RIGHT, os modos previamente apagados como Split 2 App ficam activos.

Definição das aplicações iniciais

Antes de premir [ENTER] para fechar a caixa de diálogo MODE, é possível utilizar os modos **Split 1 App** e **Split 2 App** para seleccionar as aplicações que deseja utilizar.

Modo	Especifica a aplicação:
Split 1 App	No lado esquerdo ou superior dos ecrãs divididos.
Split 2 App	No lado direito ou inferior dos ecrãs divididos.

Se **Split 1 App** e **Split 2 App** estiverem definidos para a mesma aplicação, a TI-89 / Voyage™ 200 sai do modo de ecrãs divididos e mostra a aplicação no ecrã com o tamanho máximo.

É possível abrir aplicações diferentes depois da apresentação dos ecrãs divididos.

Nota: no modo de duas representações gráficas descrito no módulo *Tópicos complementares de representação gráfica*, a mesma aplicação pode estar nos dois lados dos ecrãs divididos.

Outros modos que afectam um dos ecrãs divididos

Modo	Descrição
Number of Graphs **Nota:** deixe definido em 1 a menos que tenha lido a secção correspondente ao assunto no módulo *Tópicos complementares de representação gráfica*.	Permite configurar e mostrar dois conjuntos independentes de gráficos. Este é um recurso avançado de representação gráfica descrito em “Utilização do modo de duas representações gráficas” no módulo *Tópicos complementares de representação gráfica*.

Ecrãs divididos e coordenadas de pixel

A TI-89 / Voyage™ 200 possui comandos que utilizam coordenadas de pixel para representar rectas, circunferências, etc. no ecrã Graph. A tabela seguinte mostra como as configurações dos modos **Split Screen** e **Split Screen Ratio** afectam o número de pixéis disponíveis no ecrã Graph.

Nota:

- para obter uma lista dos comandos de desenho, consulte “Desenho no ecrã Graph” no módulo *Programação*.
- devido à borda à volta da aplicação activa, os ecrãs divididos possuem uma área de apresentação menor do que um ecrã de tamanho máximo.

TI-89:

		Split 1 App		Split 2 App	
Split	**Ratio**	**x**	**y**	**x**	**y**
FULL	N/A	0 – 158	0 – 76	N/A	N/A
TOP–BOTTOM	1:1	0 – 154	0 – 34	0 – 154	0 – 34
LEFT–RIGHT	1:1	0 – 76	0 – 72	0 – 76	0 – 72

Voyage™ 200:

		Split 1 App		Split 2 App	
Split	**Ratio**	**x**	**y**	**x**	**y**
FULL	N/A	0 – 238	0 – 102	N/A	N/A
TOP–BOTTOM	1:1	0 – 234	0 – 46	0 – 234	0 – 46
	1:2	0 – 234	0 – 26	0 – 234	0 – 68
	2:1	0 – 234	0 – 68	0 – 234	0 – 26
LEFT–RIGHT	1:1	0 – 116	0 – 98	0 –116	0 – 98
	1:2	0 – 76	0 – 98	0 – 156	0 – 98
	2:1	0 – 156	0 – 98	0 – 76	0 – 98

Saída do modo de ecrãs divididos

Método 1: Prima [MODE] para ver a caixa de diálogo **MODE**. Em seguida, defina **Split Screen = FULL**. Quando premir [ENTER] para fechar a caixa de diálogo, o ecrã de tamanho máximo mostra a aplicação especificada em **Split 1 App**.

Método 2: Prima duas vezes [2nd] [QUIT] para ver o ecrã principal no tamanho máximo.

Quando desligar a TI-89 / Voyage™ 200

A desactivação da TI-89 / Voyage™ 200 não determina a saída do modo de ecrãs divididos.

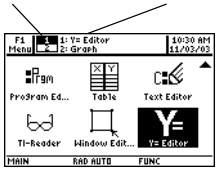
Se a TI-89 / Voyage™ 200 for desligada:	Quando ligar a TI-89 / Voyage™ 200 novamente:
Com [2nd] [OFF].	O modo de ecrãs divididos ainda permanece activo, ainda que o ecrã principal apareça no lugar da aplicação que estava activa quando premiu [2nd] [OFF].
Com APD™ ou [◆] [OFF].	O modo de ecrãs divididos mantém-se inalterado.

Indicadores de estado dos ecrãs divididos no ambiente de trabalho das Apps

Para voltar ao ambiente de trabalho das Apps, prima APPS. O estado dos ecrãs divididos aparece na parte superior do ambiente de trabalho das Apps com os nomes das Apps abertas e as partes do ecrã onde aparece cada App

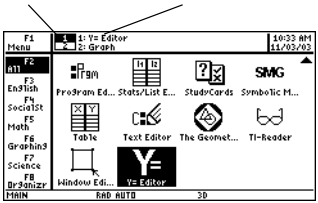
Nota: o ambiente de trabalho das Apps aparece sempre na vista de ecrã completo.

Indicador de ecrãs Nomes das Apps



Indicadores de ecrãs divididos no ambiente de trabalho das Apps da TI-89 Titanium

Indicador de ecrãs Nomes das Apps



Indicadores de ecrãs divididos no ambiente de trabalho das Apps da Voyage™ 200

Indicador de ecrãs divididos	Descrição
1 2	Ecrã dividido de cima para baixo • **1** indica a aplicação que aparece na parte superior do ecrã. • **2** indica a aplicação que aparece na parte inferior do ecrã. O número realçado indicada a parte activa do ecrã dividido.
1 2	Ecrã dividido da esquerda para a direita • **1** indica a aplicação que aparece na parte esquerda do ecrã. • **2** indica a aplicação que aparece na parte inferior do ecrã. O número realçado indicada a parte activa do ecrã dividido.

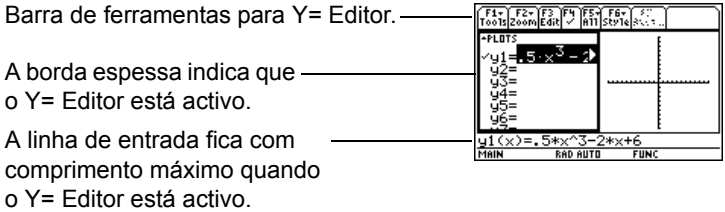
Selecção da aplicação activa

Num ecrã dividido, apenas uma das duas aplicações pode estar activa num determinado momento. É possível alternar facilmente entre as aplicações existentes ou abrir uma outra aplicação.

Aplicação activa

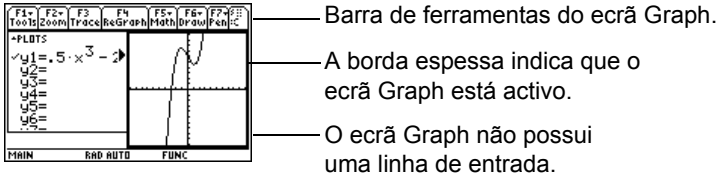
- A aplicação activa é indicada por uma borda espessa.

- A barra de ferramentas e a linha de estado com a largura do ecrã em tamanho máximo estão associadas à aplicação activa.
- Para as aplicações que possuem uma linha de entrada (como o ecrã principal e Y= Editor), a linha de entrada é da largura do ecrã em tamanho máximo apenas quando aquela aplicação está activa.



Alternância entre aplicações

Prima [2nd] [+] (segunda função de [APPS]) para alternar de uma aplicação para outra.



Abertura de uma aplicação

Método 1:

1. Utilize [2nd] [⊞] para alternar para a aplicação que deseja substituir.
2. Utilize [APPS] ou [◆] (como [◆] [WINDOW]) para seleccionar a nova aplicação.

Se seleccionar uma aplicação apresentada, a TI-89 /Voyage™ 200 alternará para a respectiva aplicação.

Método 2:

3. Prima [MODE] e, em seguida, [F2].
4. Modifique **Split 1 App** e/ou **Split 2 App**.

   Se definir **Split 1 App** e **Split 2 App** para a mesma aplicação, a TI-89 / Voyage™ 200 sai do modo de ecrãs divididos e mostra a aplicação no ecrã com tamanho máximo.

Nota: no modo de duas representações gráficas descrito no módulo *Tópicos complementares de representação gráfica*, a mesma aplicação pode estar presente nas duas partes do ecrã dividido.

Utilização de 2nd QUIT para ver o ecrã principal

Sugestão: se premir duas vezes [2nd] [QUIT], sai do modo de ecrãs divididos.

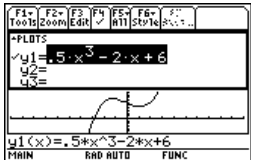
Se o ecrã principal:	**Se premir [2nd] [QUIT]:**
Não estiver a ser mostrado	O ecrã principal é aberto em vez da aplicação activa.

Se o ecrã principal:	Se premir 2nd [QUIT]:
Estiver a ser mostrado, mas não for a aplicação activa	Alterna para o ecrã principal, que passa a ser a aplicação activa.
For a aplicação activa	Sai do modo de ecrãs divididos e mostra um ecrã principal em tamanho máximo.

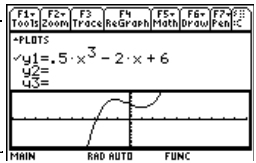
Se utilizar a divisão de cima para baixo

Se seleccionar a divisão TOP-BOTTOM, não se esqueça de que a linha de entrada e a barra de ferramentas estão sempre associadas à aplicação activa. Por exemplo:

A linha de entrada é para o Y= Editor activo, *não para o ecrã* Graph.



A barra de ferramentas é para o ecrã activo Graph, *não* para



Nota: ambas as divisões **Top-Bottom** e **Left-Right** utilizam os mesmos métodos de selecção de uma aplicação.

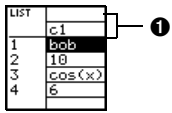
Matriz de dados/matrizes

Visão geral das variáveis de lista, dados e matrizes

Para utilizar o Data/Matrix Editor de uma forma eficaz, é indispensável compreender o significado das variáveis de lista, matriz e dados.

Variável de lista

Uma lista é constituída por uma série de itens (números, expressões ou uma sequência de caracteres) denominados elementos, que podem estar ou não relacionados. No Data/Matrix Editor, uma variável de lista:



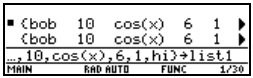
- Aparece como uma única coluna de elementos, cada uma das quais numa célula separada.
- Deve ser contínua; a lista não pode conter célula vazias.
- Pode conter até 999 elementos.

❶ As células de título e de cabeçalho da coluna não são guardadas como parte da lista.

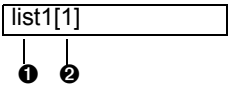
Uma variável de lista que contenha mais de uma coluna de elementos, converte-se automatica-mente numa variável de dados.

No ecrã principal (ou em qualquer outra aplicação em que uma lista possa ser utilizada), é possível introduzir uma lista entre chavetas { } com uma série de elementos separados por vírgulas.

Enquanto na linha de entrada é preciso separar os elementos com vírgulas, estes aparecem separados por espaços na área do histórico.



Para referir um determinado elemento da lista, utilize o formato apresentado à direita.



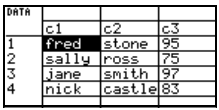
❶ Nome da variável de lista
❷ Número do elemento (ou número do índice)

Nota: depois de criar uma lista no Data/Matrix Editor, é possível utilizá-la em qualquer aplicação (por exemplo: o ecrã principal).

Variável de dados

Uma variável de dados é basicamente um conjunto de listas que podem ou não estar relacionadas. No Data/Matrix Editor, uma variável de dados:

- Pode conter até 99 colunas.
- Pode conter até 999 elementos em cada coluna. Conforme o tipo de dados, é possível ter colunas com comprimentos diferentes.



- Deve ser composta por colunas contínuas - células em branco, ou vazias não são admitidas numa coluna.

Nota: as colunas devem ter o mesmo comprimento no cálculo estatístico.

O comando **NewData** permite criar uma variável de dados constituída por listas existentes no ecrã principal ou num programa.

```
NewData data1,list1,list2
```

❶ ❷

❶ Nome da variável de dados que deseja criar;
❷ Nomes das listas de variáveis existentes

O ecrã principal não permite ver directamente uma variável de dados, apenas colunas ou elementos especificados.

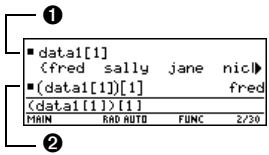
❶ ❷

```
data1[1]
(data1[1])[1]
```

❸ ❹

❶ Nome da variável de dados
❷ Número da coluna
❸ Número da coluna
❹ Número do elemento na coluna

Por exemplo:

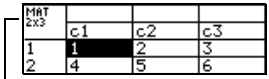


❶ Mostra a coluna 1 da variável data1.
❷ Mostra o elemento 1 na coluna 1 da variável de data1.

Variável de matriz

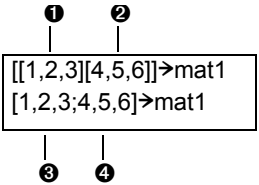
Uma matriz é um conjunto rectangular de elementos. Para criar uma matriz no Data/Matrix Editor, tem de especificar o número de linhas e de colunas (que poderá ser modificado em seguida, acrescentando ou cancelando linhas e colunas). No Data/Matrix Editor, uma variável de matriz:

- É parecida com uma variável de dados; todavia as colunas devem ter o mesmo comprimento.
- O valor inicial de cada célula é 0; este valor pode ser substituído pelo valor desejado.



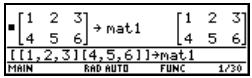
Mostra o tamanho da matriz.

Para guardar uma matriz a partir do ecrã principal ou de um programa mediante um dos métodos equivalentes mostrados à direita, utilize STO▸.



❶ linha 1
❷ linha 2
❸ linha 1
❹ linha 2

Mesmo que introduza a matriz como descrito acima, será apresentada no formato convencional na área do histórico.



Uma matriz criada no Data/Matrix Editor pode ser utilizada em qualquer aplicação (por exemplo: o ecrã principal).

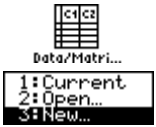
Nota: utilize parêntesis rectos para referir-se a um determinado elemento da matriz. Por exemplo: introduza **mat1[2,1]** para aceder ao 1º elemento da 2º linha.

Início de uma sessão do Data/Matrix Editor

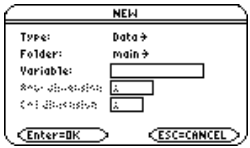
No Data/Matrix Editor, é possível criar uma variável nova, continuar a utilizar a variável actual (aquela apresentada da última vez que utilizou o Data/Matrix Editor) ou abrir uma variável existente.

Criação de uma nova variável de dados, matriz ou lista

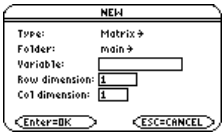
1. Prima [APPS] e seleccione Data/Matrix Editor.
2. Seleccione **3:New**.



3. Especifique as informações relativas à nova variável.



Item	Permite:	
Type	Seleccionar o tipo de variável que deseja criar. Prima ▶ para ver o menu com os tipos disponíveis.	1:Data 2:Matrix 3:List

Item	Permite:
Folder	Seleccionar a pasta onde deseja memorizar a nova variável. Prima ⊙ para ver o menu das pastas existentes. Para mais informações sobre as pastas, consulte o módulo *Ecrã principal da calculadora*.
Variable	Digitar o nome da nova variável. Se introduzir um nome já existente, quando premir [ENTER] aparece uma mensagem de erro. Quando premir [ESC] ou [ENTER] para aceitar o erro, a caixa de diálogo **NEW** aparece novamente.
Row dimension e Col dimension	Se Type = Matrix, digite o número de linhas e de colunas da matriz. 

Nota: se não for digitado um nome de variável, a TI-89 Titanium / Voyage™ 200 Graphing Calculator apresenta o ecrã principal.

4. Prima [ENTER] (depois de digitar numa caixa de entrada como **Variable**, prima duas vezes [ENTER]), para criar e mostrar uma variável vazia no Data/Matrix Editor.

Utilização da variável actual

É possível ir do Data/Matrix Editor para outras aplicações. Para voltar à variável apresentada quando deixou o Data/Matrix Editor, reinicie o Data/Matrix Editor e seleccione **1:Current**.

Criação de uma nova variável no Data/Matrix Editor

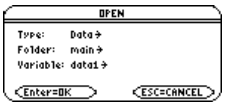
No Data/Matrix Editor:

1. Prima F1 e seleccione **3:New**.
2. Especifique o tipo, a pasta e o nome da variável. Em caso de matrizes, especifique também o número de linhas e de colunas.

Abertura de uma outra variável

É possível abrir outras variáveis em qualquer momento.

1. No Data/Matrix Editor, prima F1 e selcecione **1:Open**.
   – ou –
   Em qualquer aplicação, reinicie o Data/Matrix Editor e seleccione **2:Open**.
2. Seleccione o tipo, a pasta e a variável que deseja abrir.
3. Prima ENTER.



Nota: Variable apresenta a primeira variável existente por ordem alfabética. Se não houver variáveis, não é apresentado nada.

Nota sobre o cancelamento de uma variável

Todas as variáveis são guardadas automaticamente no Data/Matrix Editor; isto produz um acumular de variáveis que consomem memória.

Para cancelar uma variável, utilize o ecrã VAR-LINK (2nd [VAR-LINK]). Para mais informações sobre o ecrã VAR-LINK, consulte o módulo *Gestão da memória e das variáveis*.

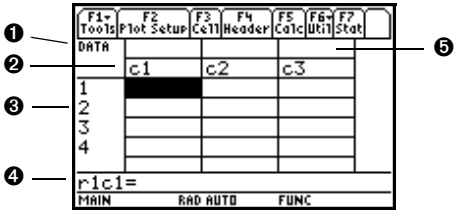
Introdução e visualização dos valores de uma célula

Quando criar uma variável, o Data/Matrix Editor está inicialmente em branco (em caso de variáveis de listas ou dados) ou apresenta uma série de zeros (em caso de matrizes). Quando abrir uma variável existente, aparecem os valores correspondentes. Neste caso, é possível introduzir valores adicionais ou modificar os valores existentes.

Ecrã do Data/Matrix Editor

A ilustração seguinte mostra o ecrã do Data/Matrix Editor em branco. Quando este ecrã aparece, o cursor realça a célula da linha 1, coluna 1.

❶ Tipo de variável
❷ Cabeçalho da coluna
❸ Número de linhas
❹ Número de linha e de coluna da célula realçada
❺ Célula de cabeçalho de coluna utilizada para digitar o título de cada coluna



Quando introduzir valores, a linha de entrada mostra o valor completo da célula realçada.

Nota: utilize a célula de cabeçalho situada na parte superior de cada coluna para identificar a informação contida nessa coluna.

Introdução ou edição de um valor numa célula

É possível introduzir qualquer tipo de expressão numa célula (número, variável, função, sucessão, etc.).

1. Realce com o cursor a célula onde deseja introduzir ou modificar um valor.
2. Prima [ENTER] ou [F3] para posicionar o cursor na linha de entrada.
3. Digite um novo valor ou modifique o valor existente.
4. Prima [ENTER] para introduzir o valor na célula realçada.

Se premir [ENTER], o cursor automaticamente realça a próxima célula para que possa continuar a introduzir ou modificar valores. Contudo, a direcção de deslocamento do cursor depende do tipo de variável.

Nota: para introduzir um novo valor, é possível digitá-lo sem premir [ENTER] ou [F3]. Todavia, tem de utilizar [ENTER] ou [F3] modificar um valor existente.

Tipo de variável	Se premir [ENTER], o cursor desloca-se:
Lista ou Dados	Para baixo, para a célula da linha seguinte.
Matriz	Para a direita, para a célula da coluna seguinte. Quando chegar à última célula de uma linha, o cursor desloca-se automaticamente para a primeira célula da linha seguinte. Este procedimento permite introduzir valores na linha 1, linha 2, etc.

Deslocação do cursor no Editor

Para deslocar o cursor:	Prima:
Uma célula por vez	⊖, ⊕, ⊙, ou ⊘
Uma página por vez	[2nd] e ⊖, ⊕, ⊙, ou ⊘
Vá para a linha 1 da coluna actual ou para a última linha que contenha os dados de qualquer coluna do ecrã. Se o cursor estiver na última linha ou passar, [◆] ⊖ irá para a linha 999.	[◆] ⊕ ou [◆] ⊖
Vá para a coluna 1 ou para a última coluna que contenha os dados. Se o cursor estiver na última coluna ou passar dela, [◆] ⊙ irá para a coluna 99.	[◆] ⊘ ou [◆] ⊙

Nota: também é possível utilizar ⊖ ou ⊕ para introduzir um valor na linha de entrada.

Quando deslocar o texto para cima ou para baixo, a linha de cabeçalho permanece fixa na parte superior do ecrã para que os números das colunas estejam sempre visíveis. Se o deslocamento for para a direita ou para a esquerda, os números da linha permanecem à esquerda do ecrã para que estejam sempre visíveis.

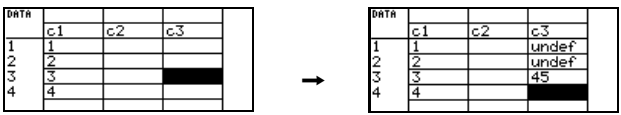
Como introduzir valores automaticamente nas linhas e colunas

Quando introduzir um valor numa célula, o cursor desloca-se para a próxima célula. Todavia, é possível posicionar o cursor em qualquer célula para introduzir um valor, pois a TI-89 Titanium / Voyage™ 200 Graphing Calculator ajusta automaticamente os espaços em branco.

- Numa variável de lista, uma célula vazia fica indefinida até a introdução de um valor.

Nota: uma variável de lista com mais de uma coluna, é convertida automaticamente numa variável de dados.

- Numa variável de dados, as célula vazias de uma coluna são tratadas da mesma forma que as variáveis de lista. Todavia, os espaços vazios entre as colunas permanecem vazios.



- Quando introduzir um valor numa célula de uma matriz não incluída nos limites actuais, as linhas e/ou colunas adicionais são automaticamente acrescentadas à matriz para incluir a nova célula. O valor zero é atribuído às outras células contidas nas novas linhas e/ou colunas.

MAT 2x3	c2	c3	c4
1	2	3	
2	5	6	
3			
4			

→

MAT 3x4	c2	c3	c4
1	2	3	0
2	5	6	0
3	0	0	12
4			

Nota: embora na criação de uma matriz especifique o seu tamanho, é possível acrescentar linhas e/ou colunas facilmente.

Modificação da largura da célula

A largura da célula determina o número de caracteres que podem ser exibidos nela. Para modificar a largura da célula no Data/Matrix Editor:

1. Para ver a caixa de diálogo **FORMATS**, prima:
   [F1] **9**
   – ou –
   [◆] [|]
   [◆] **F**

Cell width é o número máximo de caracteres que podem aparecer numa célula.
Todas as células têm a mesma largura.

Nota: não se esqueça de que para ver um número com precisão, é possível realçar a célula e verificar a linha de entrada.

2. Depois de realçar o valor actual de **Cell Width**, prima ⊕ ou ⊖ para ver um menu de dígitos (de **3** a **12**).
3. Realce o número desejado com o cursor e prima [ENTER]. (Para os números de um único dígito, é possível digitá-lo e premir [ENTER].)
4. Prima [ENTER] para fechar a caixa de diálogo.

Cancelamento de uma ou de todas as colunas

Este procedimento permite cancelar o conteúdo de uma coluna, mas não a coluna.

Para cancelar:	Operação:
Uma coluna	1. Desloque o cursor para uma célula qualquer da coluna. 2. Prima: [2nd] [F6] [F6] e seleccione **5:Clear Column**. (Este item não está disponível para uma matriz.)
Todas as colunas	Prima [F1] e seleccione **8:Clear Editor**. Quando for pedida a confirmação, prima [ENTER] (ou [ESC] para cancelar a operação).

Nota: numa variável de lista ou de dados, as colunas vazias não contém informações. Numa matriz, as colunas vazias contém zeros.

Definição do cabeçalho da coluna com uma expressão

Numa variável de lista ou numa coluna de uma variável de dados, é possível introduzir no cabeçalho da coluna uma função que gera automaticamente uma lista de elementos. Numa variável de dados, a coluna também pode ser definida em função de uma outra.

Introdução da definição do cabeçalho

No Data/Matrix Editor:

1. Posicione o cursor em qualquer célula da coluna e prima [F4].

   – ou –

   Posicione o cursor na célula de cabeçalho (**c1, c2**, etc.) e prima [ENTER].

   Nota:

   - Não é preciso premir [ENTER] para digitar uma nova definição ou para substituir a definição existente. Todavia, tem de pressionar [ENTER] para modificar a definição existente.
   - Para ver a definição existente, prima [F4] ou posicione o cursor na célula de cabeçalho e observe a linha de entrada.

2. Digite a nova expressão que substitui a existente.

   Se utilizar as teclas [F4] ou [ENTER] [8][⇥] [÷] [⇥][×][⇥] [⌫], o cursor desloca-se para a linha de entrada e realça a eventual definição existente; também é possível:

   - Prima [CLEAR] para cancelar a expressão realçada e introduzir uma outra.
     – ou –

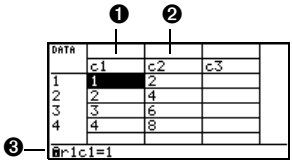
- Prima ◀ ou ▶ para remover o realce e modificar a expressão existente.

Nota: prima [ESC] antes de [ENTER] para anular eventuais modificações,.

É possível utilizar uma expressão que:	**Por exemplo:**
Gere uma série de números.	c1=seq(x^2,x,1,5) c1={1,2,3,4,5}
Refira-se a uma outra coluna.	c2=2∗c1 c4=c1∗c2–sin(c3)

Nota: a função seq está descrita no módulo *Referência técnica*. Referir-se a uma coluna vazia provoca uma mensagem de erro (a não ser que **Auto-calculate = OFF.** .

3. Prima [ENTER], ▼, ou ▲ para guardar a definição e actualizar as colunas.



❶ c1=seq(x,x,1,7); ❷ c2=2∗c1; ❸ Não é possível modificar directamente uma célula bloqueada (🔒), pois está definida pelo cabeçalho da coluna.

Nota: nas variáveis de dados, a definição de um cabeçalho é guardada quando sair do Data/Matrix Editor. Nas variáveis de lista, as definições não são guardadas (apenas os valores das células resultantes).

Cancelamento da definição de um cabeçalho

1. Posicione o cursor numa célula qualquer da coluna e prima [F4].

   – ou –

   Posicione o cursor na célula de cabeçalho (**c1, c2**, etc.) e prima [ENTER].
2. Prima [CLEAR] para cancelar a expressão realçada.
3. Prima [ENTER], ⊖, ou ⊕.

Utilização de listas existentes como colunas

Suponha que tem uma ou mais listas e deseja utilizá-las como colunas numa variável de dados.

No:	Operação:
Data/Matrix Editor	Utilize [F4] para definir o cabeçalho na coluna apropriada. Tome como referência a lista existente. Por exemplo: c1=list1
Ecrã ou num programa	Utilize o comando **NewData** como descrito no módulo *Referência técnica*. Por exemplo:: **NewData** *datavar*, *list1* [, *list2*] [, *list3*] ... ❶ ❷

No:	Operação:
❶ Variável de dados. Se esta variável já existir, será redefinida segundo as listas especificadas. ❷ Variáveis de lista existentes a copiar para as colunas na variável de dados.	

Nota: se dispuser do acessório opcional CBL 2™ ou CBR™, utilize estes procedimentos para as listas agrupadas. Utilize [2nd] [VAR-LINK] para ver as variáveis de listas existentes.

Preenchimento de uma matriz com uma lista

Não é possível utilizar o Data/Matrix Editor para preencher uma matriz com uma lista. Todavia, é possível utilizar o comando **list▶mat** no ecrã principal ou num programa. Para mais informações, consulte o módulo *Referência técnica*.

Função de cálculo automático

Para as variáveis de lista e de dados, o Data/Matrix Editor dispõe de uma função de cálculo automático. Por predefinição, Auto-calculate = ON. Portanto, quando realizar modificações que afectem a definição de cabeçalho (ou em qualquer coluna relativa à definição do cabeçalho), todas as definições são calculadas de novo automaticamente. Por exemplo:

- Se a definição de cabeçalho for modificada, a nova definição é aplicada automaticamente.
- Se o cabeçalho da coluna 2 estiver definido como c2=2∗c1, a modificação feita na coluna 1 reflete-se automaticamente na coluna 2.

Para activar e desactivar a função de cálculo automático no Data/Matrix Editor:

1. Prima:
   [F1] **9**

   – ou –

   [calc] [◆] [|]
   [calc] [◆] **F**

2. Defina **Auto-Calculate** em **OFF** ou **ON**.
3. Prima [ENTER] para fechar a caixa de diálogo.

Se **Auto-calculate = OFF** e se forem feitas modificações conforme descritas acima, as definições do cabeçalho não serão calculadas até que **Auto-calculate = ON**.

Nota: pode ser conveniente programar **Auto-calculate = OFF** para fazer várias modificações sem tornar a calcular todas as vezes, introduzir uma definição como c1=c2+c3 antes de inserir as colunas 2 e 3, ou ignorar os erros contidos numa definição até que sejam corrigidos.

Utilização das funções Shift e CumSum no cabeçalho de uma coluna

Na definição de um cabeçalho, é possível utilizar as funções shift e **cumSum** como descritas abaixo. Essas descrições diferem ligeiramente daquelas contidas no módulo *Referência técnica*. Esta secção descreve a utilização destas funções com o **Data/Matrix Editor**, enquanto o módulo *Referência técnica* fornece uma descrição geral do ecrã principal ou programa.

Utilização da função Shift

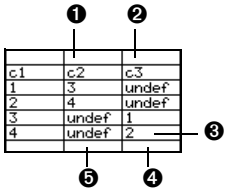
A função **shift** copia uma coluna base e desloca-a para cima ou para baixo, um número especificado de elementos. Utilize F4 para definir um cabeçalho de coluna com a sintaxe:

shift (*column* [,*integer*]**)**

❶ ❷

❶ Coluna utilizada como base para o deslocamento.
❷ Número de elementos a deslocar (positivo, para cima; negativo, para

Por exemplo: para deslocar a coluna dois elementos para cima ou para baixo:



❶ c2=shift(c1,2)
❷ c3=shift(c1,-2)
❸ As colunas deslocadas têm o mesmo comprimento da coluna base column (c1).
❹ Os últimos dois elementos de c1 deslocam-se para baixo e desaparecem na parte inferior; os elementos indefinidos deslocam-se para cima.
❺ Os dois primeiros elementos de c1 desaparecem na parte superior; os elementos indefinidos deslocam-se para a parte inferior.

Nota: para introduzir “shift”, digite-o no teclado ou seleccione-o em CATALOG.

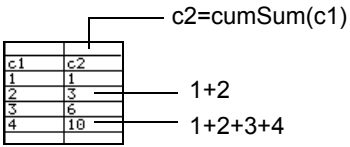
Utilização da função CumSum

A função **cumSum** efectua a soma acumulada, o somatório, dos elementos de uma coluna base. Utilize [F4] para definir o cabeçalho da coluna com a seguinte sintaxe:

cumSum (*column***)**

Coluna utilizada como coluna base para a soma acumulada.

Por exemplo:



Nota: para introduzir "**cumSum**", digite-a, seleccione-a em CATALOG ou prima [2nd] [MATH] e seleccione-a no submenu List.

Classificação de colunas

Depois de introduzir informações numa variável de lista, dados ou matriz, é possível classificar uma coluna específica por ordem alfabética ou numérica. As colunas podem ser classificadas a partir de uma coluna "chave".

Classificação de uma única coluna

No Data/Matrix Editor:

1. Posicione o cursor numa célula qualquer da coluna.
2. Prima:
   [2nd] [F6]
   [F6]
   e seleccione **3:Sort Column**.

   Os números são ordenados por ordem crescente.

   As sequências de caracteres são ordenadas por ordem alfabética.

C1	C1
fred →	75
sally	82
chris →	98
jane	chris
75 →	fred
98	jane
82	sally

Classificação de todas as colunas a partir de uma coluna “chave”

Suponha que deseja classificar uma base de dados na qual cada coluna da mesma linha contém informações interrelacionadas (como o nome, o apelido e as notas das

provas dos alunos). Neste caso, a classificação de uma única coluna anularia a relação existente entre as colunas.

No Data/Matrix Editor:

1. Posicione o cursor numa célula qualquer da coluna “chave”.

   Neste exemplo, desloque o cursor para a segunda coluna (**c2**) para classificar por apelido.

   Nota: para as variáveis de **lista** equivale a classificar uma única coluna.

c1	c2	c3
fred	stone	95
sally	ross	75
jane	smith	97
nick	castle	93

2. Prima:

   [2nd] [F6]

   [F6]

   e seleccione **4:Sort Col**, adjust all.

   Nota: este item do menu não estará disponível quando houver uma coluna bloqueada.

c1	c2	c3
nick	castle	93
sally	ross	75
jane	smith	97
fred	stone	95

Para utilizar este procedimento numa variável de dados:

- Todas as colunas devem ter o mesmo comprimento.
- Nenhuma coluna poderá estar bloqueada (definida no cabeçalho de coluna por uma função). Se o cursor estiver posicionado numa coluna bloqueada, o símbolo 🔒 aparece no início da linha de entrada.

Armazenamento de uma cópia de uma variável de lista, dados ou matriz

É possível guardar uma cópia das variáveis de lista, matriz ou dados. Também é possível copiar uma lista para uma variável de dados ou seleccionar uma coluna de uma variável de dados e copiá-la para uma lista.

Tipos válidos de cópia

É possível copiar:	Para:
Uma lista	Uma lista ou dados
Dados	Dados
Uma coluna de dados	Uma lista
Uma matriz	Uma matriz

Nota: uma lista converte-se automaticamente numa variável de dados, quando introduzir mais de uma coluna de informação.

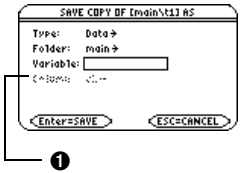
Procedimento

No Data/Matrix Editor:

1. Veja a variável que deseja copiar.

2. Prima [F1] e seleccione **2:Save Copy As**.

3. Na caixa de diálogo:



   - Seleccione **Type** e **Folder** para a cópia.
   - Digite o nome da variável para a cópia.
   - Quando disponível, seleccione a coluna a partir da qual deseja copiar.

   Nota: se digitar o nome de uma variável existente, o seu conteúdo será substituído.

   ❶ O item Column fica inactivo, até que copie uma coluna de dados para uma lista. A informação da coluna não é utilizada para outros tipos de cópia.

4. Prima [ENTER] (depois de escrever numa caixa de entrada como Variable, prima duas vezes [ENTER]).

Para copiar uma coluna de dados para uma lista

Uma variável de dados pode conter várias colunas, enquanto que uma variável de lista só pode conter uma. Para copiar uma variável de dados para uma lista, tem de seleccionar a coluna que deseja copiar.

❶ Variável de lista onde será copiada a coluna de dados.

❷ Coluna de dados que será copiada para a lista. Por predefinição, aqui aparece a coluna onde o cursor está posicionado.

Representação gráfica de dados estatísticos

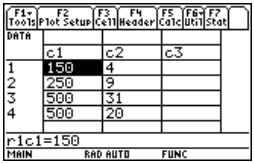
Visão geral do procedimento de análise estatística

Esta secção contém uma visão geral do procedimento utilizado para a execução de cálculos estatísticos ou representação de gráficos estatísticos. Para mais informações, consulte as páginas seguintes.

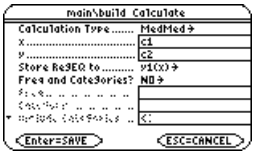
Cálculo e representação de dados estatísticos

1. Defina **Graph** (MODE) para **FUNCTION**.
2. Introduza os dados estatísticos no Data/Matrix Editor.

   Nota: para mais informações sobre a introdução de dados no Data/Matrix Editor, consulte no módulo Editor de Dados/Matrizes.

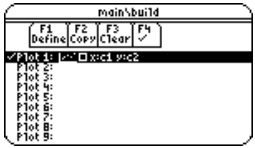


3. Realize cálculos estatísticos para encontrar as variáveis estatísticas ou adaptar os dados a um modelo (F5).



4. Defina e seleccione os gráficos estatísticos (F2 e então F1).

   Nota: o Y= Editor pode também ser utilizado para definir e seleccionar gráficos estatísticos e funções **y(x)**.



5. Defina a janela de visualização (◆ [WINDOW]).

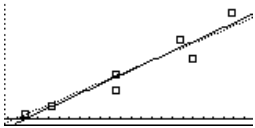
6. Altere o formato do gráfico, se necessário.

   F1 **9**

   — ou—

   ◆ |

   ◆ **F**



7. Represente os gráficos estatísticos e as funções seleccionadas (◆ [GRAPH]).

Exploração dos gráficos estatísticos

No ecrã Graph, é possível:

- Mostrar as coordenadas de qualquer pixel com o cursor de movimento livre ou de qualquer ponto pelo traçado gráfico.
- Utilizar o menu F2 **Zoom** da barra de ferramentas para ampliar ou reduzir uma parte do gráfico.
- Utilizar o menu F5 **Math** da barra de ferramentas para analisar uma função qualquer (mas não os gráficos) que pode ser representada graficamente.

Realização de cálculos estatísticos

No Data/Matrix Editor, utilize o menu F5 **Calc** da barra de ferramentas para realizar cálculos estatísticos. É possível analisar estatísticas de uma ou duas variáveis, ou realizar vários tipos de análise de regressão.

Caixa de diálogo Calculate

É preciso abrir uma variável de dados, pois o Data/Matrix Editor não realiza cálculos estatísticos com variáveis de lista ou matriz.

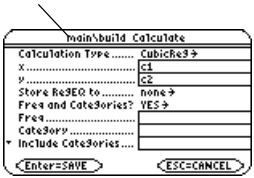
No Data/Matrix Editor:

1. Prima F5 para ver a caixa de diálogo **Calculate**.

   Neste exemplo, todos os itens estão activos. Na sua calculadora, só estão activos os itens válidos para as definições actuais de **Calculation Type** e **Use Freq and Categories?**.

   Nota: se um item não for válido para a definição actual, aparece atenuado. O cursor não pode ser posicionado num item atenuado.

Percurso da variável de dados

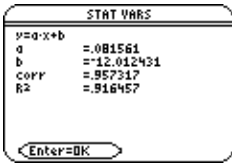


2. Especifique a definição adequada para os itens ativos.

Item	Descrição
Calculation Type	Seleccione o tipo de cálculo.
x	Digite o número da coluna (**C1**, **C2**, etc.) utilizada para os valores da variável independente x no Data/Matrix Editor.
y	Digite o número da coluna utilizada para os valores da variável dependente y. Esta operação é necessária em todos os **Calculation Types** excepto **OneVar**.
Store RegEQ to	Se Calculation Type for uma análise de regressão, é possível seleccionar o nome de uma função **(y1(x)**, **y2(x)**, etc.). Desta forma, a equação de regressão pode ser guardada e apresentada no Y= Editor.
Use Freq and Categories?	Seleccione **NO** ou **YES**. Observe que Freq, Category, e Include Categories só estão activas se Use Freq and Categories? = YES.
Freq	Digite o número da coluna que contém um valor de “ponderação”, um peso, para cada ponto de dados. Se o número da coluna não for introduzido, todos os pontos de dados terão o mesmo valor de peso (1).
Category	Digite o número da coluna que contém um valor de categoria para cada ponto de dados.
Include Categories	Se especificar uma coluna Category, é possível utilizar este item para limitar o cálculo aos valores de categoria especificados. Por exemplo: se especificar {1,4}, o cálculo utilizará apenas os dados com valores de categoria 1 ou 4.

3. Prima [ENTER] (depois de digitar numa caixa de entrada, prima duas vezes [ENTER]).

   Os resultados aparecem no ecrã **STAT VARS**. O formato depende de **Calculation Type**. Por exemplo:

Para Calculation Type = OneVar	ParaCalculation Type = LinReg
	
Quando aparece ▼ em vez de =, é possível procurar	

   Nota: os pontos de dados não definidos (apresentados como **undef**), são ignorados nos cálculos estatísticos

4. Para fechar o ecrã **STAT VARS**, prima [ENTER].

Apresentação do ecrã STAT VARS

O menu Stat da barra de ferramentas do Data/Matrix Editor volta a apresentar os resultados do cálculo anterior (a não ser que os dados da memória tenham sido cancelados).

[2nd] [F7]

[F7]

Os resultados anteriores são cancelados:

- Quando modificar ou dados ou a definição Calculation Type.
- Quando abrir outra ou a mesma variável de dados (se o cálculo for relativo a uma coluna de uma variável de dados). Os resultados também são cancelados quando sair e voltar a abrir o Data/Matrix Editor com uma variável de dados.
- Quando a pasta actual (se o cálculo for relativo a uma variável de lista da pasta anterior).

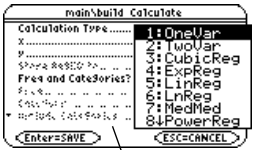
Tipos de cálculos estatísticos

Como descrito na secção anterior, a caixa de diálogo Calculate permite especificar o cálculo estatístico que deseja realizar. Esta secção contém uma descrição mais detalhada dos tipos de cálculos.

Selecção do tipo de cálculo

Na caixa de diálogo Calculate (F5), realce a definição do estado actual de **Calculation Type** e prima ▶.

É possível seleccionar os tipos disponíveis no menu.



Um item atenuado não é válido para o tipo de cálculo actual.

Tipo de cálculo	Descrição
OneVar	Estatística de uma única variável — Calcula as variáveis estatísticas. (Consulte “Variáveis estatísticas” na página 248.)
TwoVar	Estatísticas de duas variáveis — Calcula as variáveis estatísticas. (Consulte “Variáveis estatísticas” na página 248.)
CubicReg	Regressão cúbica — Adapta os dados a um polinómio de terceiro grau $\mathbf{y=ax^3+bx^2+cx+d}$. É preciso ter pelo menos quatro pontos de dados. • Com quatro pontos, a equação é um polinômio. • Com cinco pontos ou mais, é uma regressão polinomial.
ExpReg	Regressão exponencial — Adapta os dados a uma equação do tipo $\mathbf{y=ab^x}$ (onde a é a intersecção com o eixo y) com o método de mínimos quadrados e os valores transformados **x** e **ln(y)**.
LinReg	Regressão linear — Adapta os dados a uma equação do tipo **y=ax+b** (onde a é a inclinação e b é a intersecção com o eixo y) com o método de mínimos quadrados x e y.
LnReg	Regressão logarítmica — Adapta os dados a uma equação do tipo **y=a+b ln(x)** com o método de mínimos quadrados e os valores transformados **ln(x)** e **y**.
Logistic	Regressão logística — Ajusta os dados ao modelo **y=a/(1+b∗e^(c∗x))+d** e actualiza todas as variáveis estatísticas do sistema.

Tipo de cálculo	Descrição
MedMed	Mediana — Adapta os dados ao modelo **y=ax+b** (onde a é a inclinação e b é a intersecção com o eixo y) com a recta mediana-mediana que faz parte da técnica da recta resistente. Os valores **medx1**, **medy1**, **medx2**, **medy2**, **medx3**, e **medy3** são calculados e guardados nas variáveis, mas não apresentados no ecrã STAT VARS.
PowerReg	Regressão de potência — Adapta os dados ao modelo de equação**y=ax**^{b} com o método dos mínimos quadrados e os valores transformados **ln(x)** e **ln(y)**.
QuadReg	Regressão quadrática — Adapta os dados ao polinómio de segundo grau **y=ax**^{2}**+bx+c**. É preciso ter pelo menos três pontos de dados. • Com três pontos, a equação é um polinómio. • Com quatro pontos ou mais, é uma regressão polinomial.
QuartReg	Regressão de quarto grau — Adapta os dados ao polinómio de quarto grau **y=ax**^{4}**+bx**^{3}**+cx**^{2}**+ dx+e**. É preciso ter pelo menos cinco pontos de dados. • Com cinco pontos, a equação é um polinómio.Com seis pontos ou mais, é uma regressão polinomial.
SinReg	Regressão sinoidal — Calcula a regressão sinusoidal e actualiza todas as variáveis estatísticas do sistema. O resultado é sempre em radianos, independentemente da definição do modo de ângulo.

Nota: em **TwoVar** e nos cálculos de regressão, as colunas especificadas para x e y (e opcional, Freq ou Category), devem ter o mesmo tamanho.

No ecrã principal ou num programa

Utilize o comando correspondente ao cálculo que deseja realizar. Os nomes dos comandos coincidem com o de Calculation Type. Para mais informações sobre os comandos, consulte o módulo *Referência técnica*.

Importante: estes comandos realizam cálculos estatísticos, mas não mostram automaticamente os resultados. Para mostrar os resultados, utilize o comando **ShowStat**.

Variáveis estatísticas

Os resultados dos cálculos estatísticos são guardados em variáveis. Para aceder a estas variáveis, digite o nome da variável ou utilize o ecrã VAR-LINK como descrito no módulo *Gestão da memória e das variáveis*. Todas as variáveis estatísticas são canceladas quando modificar ou trocar o tipo de cálculo. As outras condições determinam o cancelamento das variáveis.

Variáveis calculadas

As variáveis estatísticas são guardadas como variáveis de sistema. Todavia **regCoef** e **regeq** são tratadas, respectivamente, como uma variável de lista e uma variável de função.

	Uma Var	Duas Var	Regressões
média dos valores de x	$\bar{x}$	$\bar{x}$	
soma dos valores de x	Σx	Σx	

	Uma Var	Duas Var	Regressões
soma dos valores de x^2	Σx^2	Σx^2	
desvio padrão da amostra de x	Sx	Sx	
desvio padrão da população de x	σx	σx	
número de pontos de dados	nStat	nStat	
média dos valores de y		$\bar{y}$	
soma dos valores de y		Σy	
soma dos valores de y^2		Σy^2	
desvio padrão da amostra de y		Sy	
desvio padrão da população de y		σy	
soma dos valores de $x*y$		Σxy	
mínimo dos valores de x	minX	minX	
máximo dos valores de x	maxX	maxX	
mínimo dos valores de y		minY	
máximo dos valores de y		maxY	
primeiro quartil	q1		
mediana	medStat		
terceiro quartil	q3		
equação de regressão			regeq
coeficientes de regressão (a, b, c, d, e)			regCoef

	Uma Var	Duas Var	Regressões
coeficiente de correlação ††			corr
coeficiente de determinação ††			R^2
valor de resumo (apenas MedMed) †			medx1, medy1, medx2, medy2, medx3, medy3

†† **corr** é definido exclusivamente para uma regressão linear, enquanto **R^2** é definido para todas as regressões de polinómios.

Nota:

- se **regeq** é 4x + 7, então **regCoef** é {4 7}. Para aceder ao coeficiente "*a*" (o primeiro elemento da lista), utilize um índice, como por exemplo **regCoef[1]**.
- o primeiro quartil é a mediana dos pontos situados entre **minX** e **medStat**, e o terceiro quartil é o ponto médio compreendido entre **medStat** e **maxX**.

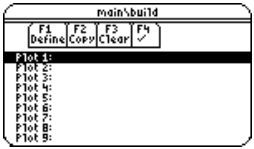
Definição de um gráfico estatístico

No Data/Matrix Editor, é possível utilizar os dados introduzidos para definir vários tipos de gráficos estatísticos. Podem ser definidos até 9 gráficos simultaneamente.

Procedimento

No Data/Matrix Editor:

1. Prima F2 para ver o ecrã **Plot Setup**. Inicialmente nenhum gráfico está definido.



2. Realce o gráfico que deseja definir com o cursor.

3. Prima F1 para definir o gráfico.

   Neste exemplo, todos os itens estão activos. Na sua calculadora, só estão activos os itens válidos para as definições actuais de **Plot Type** and **Use Freq and Categories?**.

   Nota: se um item não for válido para a definição actual, aparecerá atenuado. O cursor não pode ser posicionado num item atenuado.

Nome do percurso da variável de dados

4. Especifique a definição adequada para os itens activos.

Item	Descrição
Plot Type	Seleccione o tipo de gráfico.(Consulte “Tipos de gráficos estatísticos” na página 252.)
Mark	Seleccione o símbolo utilizado para representar os pontos de dados: Box (☐), Cross (x), Plus (+), Square (■), ou Dot (•).

Item	Descrição
x	Digite o número da coluna no Data/Matrix Editor (**C1**, **C2**, etc.) utilizado para os valores da variável independente x.
y	Digite o número da coluna utilizada para os valores da variável dependente y. Só fica activa se Plot Type = Scatter ou xyline.
Hist. Bucket Width	Especifique a largura das barras do histograma. (Consulte “Histograma” na página 253.)
Use Freq and Categories?	Seleccione **NO** ou **YES**. Não se esqueçam de que Freq, Category, e Include Categories só ficam activas se Use Freq and Categories? = YES. (Freq está activo se Plot Type = Box Plot ou Histogram.)
Freq	Digite o número da coluna que contém um valor de “ponderação”, um peso, para cada ponto de dados. Se o número da coluna não for introduzido, todos os pontos de dados terão o mesmo valor de peso (1).
Category	Digite o número da coluna que contém um valor de categoria para cada ponto de dados.
Include Categories	Se especificar uma coluna Category, é possível utilizar este item para limitar o cálculo aos valores de categoria especificados. Por exemplo: se especificar {1,4}, o cálculo utilizará apenas os dados com valores de categoria 1 ou 4.

Nota:

- No Data/Matrix Editor, os gráficos definidos com números de coluna utilizam sempre a última variável de dados, mesmo não tendo sido utilizada para criar a definição.

- Para utilizar uma variável de lista existente para x, y, Freq, ou Category, digite o nome da lista em vez do número da coluna.

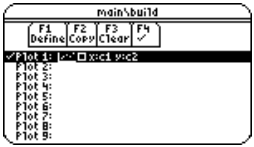
- Consulte o exemplo de utilização de Freq, Category, e Include Categories.

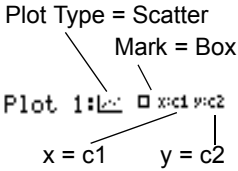
5. Prima [ENTER] (depois de digitar numa caixa de entrada, prima duas vezes [ENTER]).

   O ecrã **Plot Setup** volta a aparecer.

   O gráfico definido é automaticamente seleccionado para a representação gráfica.

   Observe a definição resumida do gráfico.





Nota: os pontos de dados não definidos (apresentados com **undef**) são ignorados quando realizar um gráfico estatístico.

Seleccionar e anular a selecção de um gráfico

Em Plot Setup, realce o gráfico e prima [F4] para activar ou desactivar o gráfico. Um gráfico estatístico permanece realçado quando:

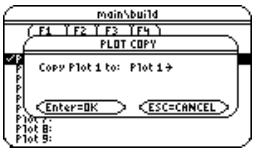
- O modo do gráfico é modificado. (Os gráficos estatísticos não são representados no modo 3D.)

- Um comando Graph é executado.
- Uma variável diferente no Data/Matrix Editor é aberta.

Cópia da definição de um gráfico

Em **Plot Setup**:

1. Realce o gráfico e pressione F2.
2. Prima ▶ e seleccione o número do gráfico que deseja copiar.
3. Prima ENTER.



Nota: quando seleccionar o gráfico original (✓), selecciona também a cópia.

Cancelamento da definição do gráfico

Em Plot Setup, realce o gráfico e prima F3. Para voltar a definir um gráfico existente, não é necessário cancelá-lo antes, pois é possível modificar a definição existente. Para evitar a representação do gráfico, anule-o.

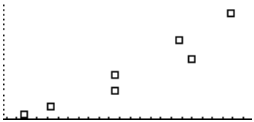
Tipos de gráficos estatísticos

Durante a definição de um gráfico estatístico como descrito na secção anterior, o ecrã Plot Setup permite seleccionar o tipo de gráfico. Esta secção contém uma descrição mais detalhada dos tipos de gráficos disponíveis.

Scatter

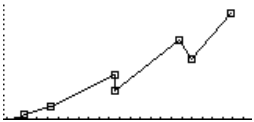
Os pontos de dados x e y são representados como pares de coordenadas. Portanto, as colunas ou listas indicadas para x e y devem ter o mesmo tamanho.

- Os pontos representados aparecem com o símbolo seleccionado como Mark.
- Se necessário, é possível especificar a mesma coluna ou lista para x e y.



xyline

É um gráfico scatter no qual os pontos de dados são representados e ligados segundo a ordem em que x e y aparecem.



Antes da representação, é possível classificar as colunas no Data/Matrix Editor.

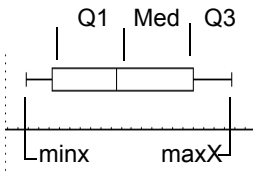
[2nd] [F6] **3** ou [2nd] [F6] **4**

[F6] **3** ou [F6] **4**

Box Plot

Representa os dados de uma variável em relação aos pontos de dados máximo e mínimo (**minX** and **maxX**) do conjunto.

- A caixa está definida pelo primeiro quartil (**Q1**), pela mediana (**Med**) e pelo terceiro quartil (**Q3**).
- As marcas prolongam-se de **minX** a **Q1** e de **Q3** a **maxX**.



- Se seleccionar vários gráficos de caixas, estes são representados um por cima do outro segundo a ordem de numeração.
- Utilize NewPlot para mostrar dados estatísticos como um traçado de caixa modificada.
- Seleccione Mod Box Plot como Plot Type quando definir um traçado no Data/Matrix Editor.

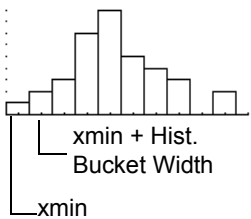
Um traçado modificado de caixa exclui os pontos fora do intervalo [**Q1**−**X, Q3+X**], onde X é definido como **1.5 (Q3**−**Q1)**. Estes pontos atípicos são traçados individualmente além das margens do traçado da caixa com a marca seleccionada.

Histogram

Representa os dados de uma variável num histograma. O eixo x é dividido em segmentos da mesma largura, denominados cubos ou barras. A altura das barras (o seu valor y) indica a quantidade de pontos de dados incluídos no intervalo da barra.

- Durante a definição do gráfico, é possível especificar a largura de cada barra mediante **Hist. Bucket Width** (o valor predefinido é 1).
- O ponto de dados situado na extremidade da barra é colocado na barra à direita.

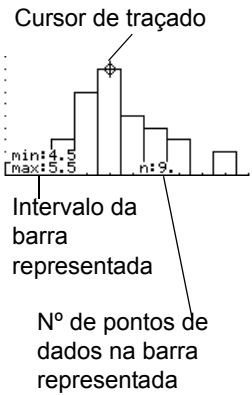
$$\text{Número de barras} = \frac{xmax - xmin}{\text{Hist. Bucket Width}}$$



- **ZoomData** (F2 **9** do ecrã Graph, Y= Editor, ou Window Editor) ajusta **xmin** e **xmax** para incluir todos os pontos de dados, porém não ajusta o eixo y.

- Utilize [◆] [WINDOW] para definir **ymin = 0** e **ymax** = número de pontos de dados previsto na barra maior, mais alta.

- Quando traçar um histograma ([F3]), o ecrã apresenta as informações relativas à barra traçada.



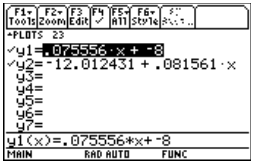
Utilização do Y= Editor com gráficos estatísticos

As secções anteriores descreveram o procedimento para definir e seleccionar gráficos estatísticos no Data/Matrix Editor. Os gráficos estatísticos também podem ser definidos e seleccionados no Y= Editor.

Visualização da lista dos gráficos estatísticos

Prima [◆] [Y=] para ver Y= Editor. Os nove gráficos estatísticos estão colocados na parte superior do ecrã (não visíveis) por cima das funções **y(x)**. Todavia, o indicador PLOTS fornece algumas informações.

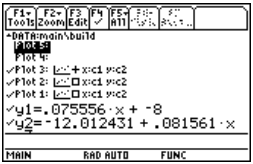
Por exemplo: PLOTS 23 indica que foram seleccionados os gráficos 2 e 3.



Para ver a lista dos gráficos estatísticos, utilize ⊕ para deslocar os valores acima das funções **y(x)**.

Se realçar um gráfico, este apresenta a variável de dados que será utilizada no gráfico.

Se um gráfico estiver definido, mostra a mesma notação resumida do ecrã Plot Setup.



No Y= Editor, é possível realizar praticamente as mesmas operações que em qualquer função **y(x)** nos gráficos estatísticos.

Nota: no Data/Matrix Editor, os gráficos definidos com números de coluna sempre utilizam a última variável de dados, mesmo que não tenha sido utilizada para criar a definição.

Para:	Operação:
Modificar a definição de um gráfico	Realce o gráfico e prima [F3]. Aparece o ecrã de definição apresentado no Data/Matrix Editor.
Seleccionar ou anular um gráfico	Realce o gráfico e prima [F4].
Desactivar todos os gráficos e/ou funções	Prima [F5] e seleccione o item aplicável. Pode seleccionar o item apropriado. Este menu também pode ser utilizado para activar todas as funções.

Nota: não pode utilizar [2nd] [F6]; [F6] para definir um estilo de visualização do traçado. Entretanto, a definição do traçado permite seleccionar a marca utilizada para o traçado.

Para representar gráficos e funções Y=

Se necessário, é possível seleccionar e representar gráficos estatísticos e funções **y(x)** simultaneamente.

Representação e traçado de um gráfico estatístico definido

Depois de introduzir os pontos de dados e definir os gráficos estatísticos, é possível representar os gráficos seleccionados com os métodos utilizados no Y= Editor (como descrito no módulo *Representação gráfica de funções básicas).*

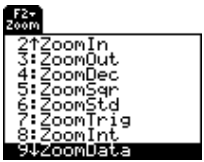
Definição da janela de visualização

Os gráficos estatísticos aparecem de acordo com o gráfico actual e utilizam as variáveis definidas no Window Editor.

Utilize [◆] [WINDOW] para ver o Window Editor. Proceda de uma das seguintes maneiras:

- Introduza os valores apropriados.
  — ou —
- Seleccione **9:ZoomData** no menu [F2] **Zoom** da barra de ferramentas. (É possível utilizar qualquer opção zoom, mas **ZoomData** é a mais indicada para os gráficos estatísticos.)

ZoomData estabelece a janela de visualização para que mostre todos os pontos de dados.

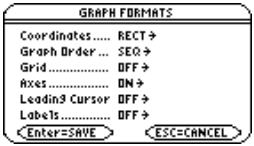


Nos histogramas e gráficos de caixas, são ajustados apenas **xmin** e **xmax**. Se a parte superior do histograma não aparece, trace o histograma para encontrar o valor de **ymax**.

Nota: F2 **Zoom** está disponível no Y= Editor, no Window Editor e no ecrã Graph.

Alteração do formato do gráfico

Prima:
F1 **9**
— ou —
◆ |
◆ **F**
no Y= Editor, no Window Editor ou no ecrã Graph.



Em seguida, altere as configurações necessárias.

Traçado de um gráfico estatístico

No ecrã Graph, prima F3 para traçar um gráfico com o cursor. O movimento do cursor depende do Plot Type.

Tipo de gráfico	Descrição
Scatter ou xyline	O traçado inicia no primeiro ponto de dados.
Box plot	O traçado inicia na mediana. Prima ◀ para traçar em **Q1** e **minX**. Pressione ▶ pata traçar em **Q3** e **maxX**.
Histogram	O cursor desloca-se a partir do ponto central superior de cada barra, iniciando pela primeira barra da esquerda.

Nota: quando mostrar um gráfico estatístico, o ecrã Graph não se enquadra automaticamente, se o traçado do cursor ultrapassar a margem direita ou esquerda do ecrã. Para centralizar o ecrã no cursor de traçado, prima [ENTER].

Se premir ⊕ ou ⊖ para passar para outro gráfico ou função **y(x)**, o cursor desloca-se para o ponto inicial ou actual do gráfico (e não para o pixel mais próximo).

Utilização de frequências e categorias

Para determinar a forma de análise dos dados, é possível utilizar valores de frequência e/ou de categorias. Os valores de frequência permitem "ponderar" determinados pontos de dados. Os valores de categoria permitem analisar um subconjunto de pontos de dados.

Exemplo de uma coluna de frequência

Numa variável de dados, é possível utilizar qualquer coluna do Data/Matrix Editor para atribuir um valor de frequência (ou de ponderação) aos pontos de dados de cada linha. Um valor de frequência deve ser um número inteiro ≥ 0 se Calculation Type = OneVar ou MedMed ou se Plot Type = Box Plot. Nos outros tipos de cálculos ou gráficos estatísticos, o valor de frequência pode ser um número qualquer ≥ 0.

Por exemplo: suponha que introduziu as notas das provas de um estudante, onde:

- O peso da nota da prova feita na metade do semestre é o dobro do peso das outras provas.
- O peso da prova final é o triplo.

No Data/Matrix Editor, é possível introduzir as notas e os valores de frequência em duas colunas.

Notas das provas	Valores de frequência
c1	c2
85	1
97	1
92	2
89	1
91	1
95	3

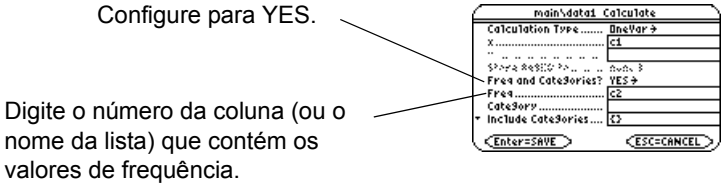
As notas ponderadas são equivalentes aos valores da coluna apresentada à direita.

c1	
85	
97	
92	❶
92	❶
89	
91	
95	❷
95	❷
95	❷

❶ Frequência é 2
❷ Frequência é 3

Nota: um valor de frequência 0 elimina o ponto de dados da análise.

Para utilizar os valores de frequência, especifique a coluna de frequência durante a realização do cálculo estatístico ou a definição do gráfico. Por exemplo:



Nota: é possível utilizar os valores de frequência de uma variável de lista em vez de uma coluna.

Exemplo de uma coluna de categoria

Numa variável de dados, é possível utilizar qualquer coluna para especificar um valor de categoria (ou subconjunto) dos pontos de dados de cada linha. O valor de categoria pode ser um número qualquer.

Suponha que introduziu as notas da prova de um grupo composto por estudantes da 7a. e 8a. séries. O objectivo é analisar as notas do grupo todo, mas também incluir uma análise por categorias, por exemplo: alunas da 7a. série, alunos da 7a. série, alunos e alunas da 8a. série, etc.

Em primeiro lugar, é preciso determinar os valores de categoria que deseja utilizar.

Valor de categoria	Utilizado para indicar:
1	alunas da 7a. série
2	alunos da 7a. série
3	alunas da 8a. série
4	alunos da 8a. série

Nota: não é preciso determinar um valor de categoria para a classe inteira, assim como estabelecer os valores de categoria para todos os estudantes da 7a. e 8a. séries, pois são uma combinação de categorias diferentes.

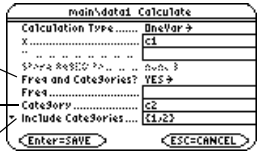
No Data/Matrix Editor, é possível introduzir as notas e os valores de categoria em duas colunas.

Notas das provas	**Valores de categoria**
c1	c2
85	1
97	3
92	2
88	3
90	2
95	1
79	4
68	2
92	4
84	3
82	1

Para utilizar os valores de categoria, especifique a coluna e os valores de categoria que deseja incluir na análise durante a realização do calculo estatístico ou defina o gráfico estatístico.

Defina para YES.

Digite o número da coluna (ou o nome da lista) que contém os valores de categoria



Digite os valores de categoria entre chavetas{ }, separados por vírgulas - não digite um número de coluna ou um nome de lista.

Nota: é possível utilizar os valores de categoria de uma variável de lista em vez de uma coluna.

Para analisar:	Inclua as categorias:
alunas da 7a. série	{1}
alunos da 7a. série	{2}
alunas e alunos da 7a. série	{1,2}
alunas da 8a. série	{3}
alunos da 8a. série	{4}
alunas e alunos da 8a. série	{3,4}
todas as alunas (7a. e 8a. séries)	{1,3}
todos os alunos (7a. e 8a. séries)	{2,4}

Nota: para analisar a turma inteira, deixe em branco a caixa de entrada Category. Os valores de categoria são ignorados.

Acessório opcional CBL 2™ ou CBR™

O sistema Calculator-Based Laboratory™ (CBL 2) e o Calculator-Based Ranger™ (CBR) são acessórios opcionais fornecidos à parte, que permitem recolher dados de experiências reais de vários tipos. Os programas para o CBL 2 e para o CBR da TI-89 Titanium / Voyage™ 200 estão disponíveis no site Web da TI education.ti.com.

Armazenamento de dados do CBL 2™

Os dados recolhidos com o sistema CBL 2 são inicialmente guardados na própria unidade CBL 2, de onde deverão ser recuperados (e transferidos para a TI-89 Titanium / Voyage™ 200) com o comando **Get**, que está descrito no módulo *Referência técnica*.

Embora os conjuntos dos dados recuperados possam ser memorizados em diversos tipos de variáveis (lista, real, matriz, imagem), a utilização das variáveis de lista facilita a realização de cálculos estatísticos.

Quando transferir a informação recolhida para a TI-89 Titanium / Voyage™ 200, é possível especificar os nomes das variáveis de lista que deseja utilizar.

Por exemplo: O CBL 2 pode ser utilizado para recolher dados relativos à temperatura durante um período de tempo. Durante a transferência dos dados, suponha que são guardados:

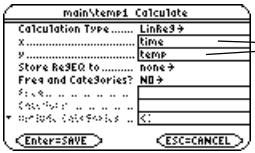
- Os dados relativos à temperatura numa variável de lista denominada *temp*.
- Os dados relativos ao intervalo de tempo numa variável de lista denominada *time*.

Depois de guardar a informação da CBL 2 na TI-89 Titanium / Voyage™ 200, as variáveis de lista CBL 2 podem ser utilizadas de duas formas diferentes.

Nota: para mais informações sobre a utilização de CBL 2 e a recuperação de dados para a TI-89 Titanium / Voyage™ 200, consulte o manual fornecido com a unidade CBL 2.

Referência às listas CBL 2™

Durante a realização de cálculos estatísticos ou durante a definição de um gráfico estatístico, é possível referir-se de forma explícita às variáveis de lista CBL 2. Por exemplo:



Digite o nome da variável de lista CBL em vez de número de coluna.

Criação de uma variável de dados com as listas CBL 2™

É possível criar variáveis de dados que incluam as variáveis de lista CBL 2 necessárias.

- No ecrã principal ou num programa, utilize o comando **NewData**.

NewData *dataVar*, *list1* [,*list2*] [,*list3*] ...

Nomes das variáveis de lista CBL. Na nova variável de dados, list1 será copiada na coluna 1, list 2 na 2, etc.

Nome da variável de dados que deseja criar.

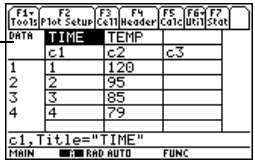
Por exemplo:

NewData *temp1*, *time*, *temp*

cria uma variável de dados denominada *temp1* onde time está posicionado na coluna 1 e *temp* na coluna 2.

- No Data/Matrix Editor, crie uma nova variável de dados vazia com o nome apropriado. Para cada lista CBL 2 que deseja incluir, defina um cabeçalho de coluna com o nome da lista.

Por exemplo: defina a coluna 1 como time, a coluna 2 como temp



Nota: para definir ou cancelar um cabeçalho de coluna, utilize F4. Para mais informações, consulte o módulo *Editor de Dados/Matrizes*.

Neste momento, as colunas associam-se às listas CBL 2. Na eventualidade de modificar as listas, as colunas serão automaticamente actualizadas; o cancelamento das listas determina a perda dos dados.

Para tornar a variável de dados independente das listas CBL 2, apague o cabeçalho de cada coluna. A informação permanecerá na coluna, que não estará mais associada à lista CBL 2.

CBR™

É também possível utilizar o Calculator-Based Ranger™ (CBR) para explorar a relação matemática e científica entre distância, velocidade, aceleração e tempo com os dados recolhidos das actividades desenvolvidas.

Programação

Execução de um programa existente

Um programa criado (como descrito nas secções seguintes deste módulo) pode ser executado a partir do ecrã principal. O resultado do programa, caso exista, aparece no ecrã Program I/O numa caixa de diálogo ou no ecrã Graph.

Execução de um programa

No ecrã principal:

1. Digite o nome do programa.

2. É sempre preciso digitar um par de parêntesis depois do nome.

   Alguns programas requerem um ou mais argumentos.

   Nota: utilize [2nd] [VAR-LINK] para listar as variáveis **PRGM** existentes. Realce uma variável e prima [ENTER] para colar o seu nome na linha de entrada.

3. Prima [ENTER].

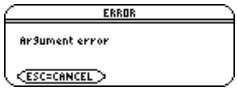
```
prog1()
```
└ Se argumentos não forem necessários

```
prog1(x,y)
```
└ Se argumentos forem necessários

Nota: os argumentos especificam os valores iniciais de um programa. (Consulte “Obtenção de valores num programa” na página 7.)

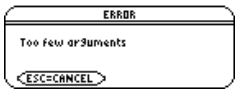
Quando executar um programa, a TI-89 Titanium / Voyage™ 200 Graphing Calculator verifica automaticamente se há erros. Por exemplo, a mensagem seguinte aparece se:

- Não houver ( ) depois do nome do programa.



Esta mensagem de erro aparece, se:

- Não introduzir a quantidade esperada de argumentos, caso sejam necessários.



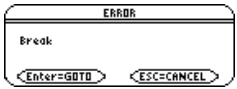
Para cancelar a execução do programa se ocorrer um erro, prima [ESC]. Depois de corrigir o problema, o programa pode ser executado novamente.

Nota: a TI-89 Titanium / Voyage™ 200 verifica também se ocorrem erros durante a execução do próprio programa.

“Interrupção” de um programa

O indicador BUSY aparece na linha de estado durante a execução de um programa.

Prima [ON] para interromper a execução do programa. Em seguida, aparece uma mensagem.



- Para ver o programa no Program Editor, prima [ENTER]. O cursor aparece no comando onde ocorreu a interrupção.
- Para cancelar a execução de um programa, prima [ESC].

Onde aparece o resultado do programa?

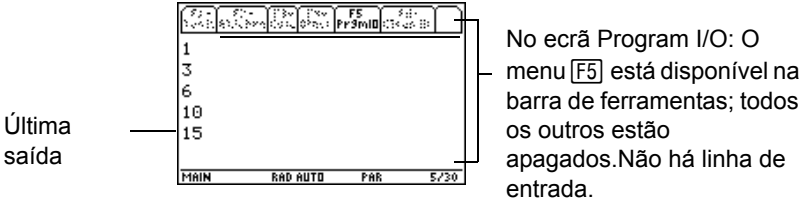
A TI-89 Titanium / Voyage™ 200 mostra automaticamente as informações no ecrã apropriado de acordo com os comandos do programa.

- A maior parte dos comandos de entrada e saída utiliza o ecrã Program I/O. (Os comandos de entrada pedem ao utilizador para introduzir informações.)
- Os comandos relacionados com o ecrã Graph utilizam geralmente o ecrã Graph.

Quando o programa terminar, a TI-89 Titanium / Voyage™ 200 mostra o último ecrã.

Ecrã Program I/O

Os resultados novos aparecem debaixo dos eventuais resultados precedentes (que podem ter sido mostrados anteriormente por este programa ou por um programa diferente) no ecrã Program I/O. Quando a página estiver cheia, os resultados mais antigos saem pela parte superior e desaparecem do ecrã.



Nota: para limpar resultados anteriores, introduza o comando **ClrIO** no seu programa. É possível também executar **ClrIO** a partir do ecrã principal.

Quando um programa é interrompido no ecrã Program I/O, é preciso compreender que não é o ecrã principal (apesar dos dois ecrãs serem parecidos). O ecrã Program I/O só é utilizado para mostrar resultados ou solicitar entrada pelo utilizador. Não é possível fazer cálculos neste ecrã.

Nota: se os cálculos não aparecerem no ecrã principal depois da execução de um programa, pode ser que o ecrã actual seja o ecrã Program I/O.

Saída do ecrã Program I/O

No ecrã Program I/O:

- Prima F5 para alternar entre o ecrã principal e o ecrã Program I/O.)

  – ou –

- Prima ESC, 2nd [QUIT] ou
  HOME
  ◆ [CALC HOME]
  para ver o ecrã principal.
  – ou –

- Veja qualquer outro ecrã da aplicação (com [APPS], [◆] [Y=] etc.).

Início de uma sessão do Program Editor

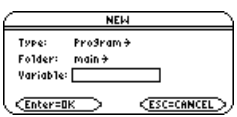
Sempre que iniciar o Program Editor, é possível continuar o programa ou função actual (apresentado da última vez que o Program Editor foi utilizado), abrir um programa ou função existente, ou iniciar um novo programa ou função.

Início de um novo programa ou função

1. Prima [APPS] e, em seguida, seleccione **Program Editor**.

2. Seleccione **3:New**.

3. Especifique as informações apropriadas para o novo programa ou função.

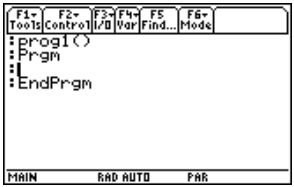


Item	Permite:
Type	Escolher se deseja criar um novo programa ou função.
Folder	Seleccionar a pasta onde o programa ou função será guardado. Para mais informações sobre as pastas, consulte o módulo *Ecrã principal da calculadora*.

Item	Permite:
Variable	Digitar o nome de uma variável para o programa ou para a função. Se especificar uma variável já existente, aparece uma mensagem de erro quando premir [ENTER]. Quando premir [ESC] ou [ENTER] depois do erro, a caixa de diálogo **NEW** aparece novamente.

4. Prima [ENTER] (depois de digitar dados numa caixa de entrada como **Variable**, é preciso premir duas vezes [ENTER]) para ver um “modelo” vazio.

Este é o modelo para um programa. As funções possuem um modelo parecido.



É possível utilizar o Program Editor conforme descrito nas secções seguintes deste módulo.

Nota: um programa (ou função) é guardado automaticamente à medida que é digitado. Não é preciso guardá-lo manualmente quando sair do Program Editor, iniciar ou abrir um outro programa.

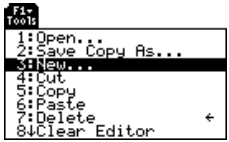
Continuação do programa actual

É possível sair do Program Editor e ir para outra aplicação a qualquer momento. Para voltar ao programa ou função apresentado quando sair do Program Editor, reinicie o Program Editor e seleccione **1:Current**.

Início de um novo programa a partir do Program Editor

Para sair do programa ou da função actual e iniciar um novo:

1. Prima F1 e seleccione **3:New**.
2. Especifique o tipo, a pasta e a variável para o novo programa ou função.
3. Pressione duas vezes ENTER.



Abertura de um programa anterior

Pode abrir um programa ou função criado anteriormente a qualquer momento.

1. No **Program Editor**, prima F1 e seleccione **1:Open**.

   – ou –

   Noutra aplicação, reinicie o Program Editor e seleccione **2:Open**.

2. Seleccione o tipo, a pasta e a variável apropriados.
3. Prima ENTER.

Nota: por predefinição, Variable mostra o primeiro programa ou função existente por ordem alfabética.

Cópia de um programa

Em alguns casos, pode ser preciso copiar um programa ou função para editar a cópia, enquanto o original é mantido.

1. Veja o programa ou função que deseja copiar.
2. Prima [F1] e seleccione **2:Save Copy As**.
3. Especifique a pasta e a variável para a cópia.
4. Prima duas vezes [ENTER].

Observação sobre remoção de um programa

Como todas as sessões do Program Editor são automaticamente guardadas, pode acumular programas e funções anteriores, que ocupam espaço na memória.

Para excluir programas e funções, utilize o ecrã VAR-LINK ([2nd] [VAR-LINK]). Para mais informações sobre VAR-LINK, consulte no módulo *Gestão da memória e das variáveis*.

Descrição da introdução de um programa

Um programa é uma série de comandos executados por ordem sequencial (apesar de alguns comandos alterarem o fluxo do programa). De uma forma geral, qualquer instrução que possa ser executada a partir do ecrã principal pode ser incluída num programa. A execução de um programa continua até chegar ao fim do programa ou de um comando **Stop**.

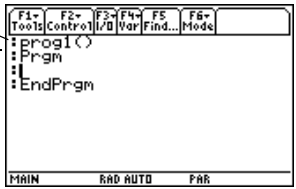
Introdução e edição de linhas de programa

Num modelo em branco, é possível introduzir comandos para o seu programa novo.

Nome do programa especificado quando um novo programa é criado.

Introduza os comandos do seu programa entre **Prgm** e **EndPrgm**.

Todas as linhas do programa começam com dois pontos.



Nota: utilize as teclas de cursor para percorrer o programa para introduzir ou editar comandos. Utilize [◆] ⊕ ou [◆] ⊖ para ir para o início ou o fim de um programa, respectivamente.

É possível introduzir e editar comandos de programa no Program Editor com as mesmas técnicas utilizadas para introduzir e editar texto no Text Editor. Consulte “Introdução e edição de texto” no módulo *Editor de Texto*.

Depois de digitar cada linha de programa, prima [ENTER] para inserir uma nova linha vazia e continuar a introdução de nova linha. Uma linha de programa pode ser maior do que uma linha do ecrã; neste caso, passa automaticamente para a próxima linha do ecrã.

Nota: a introdução de um comando não executa o comando. Só é executado quando o programa for executado.

Introdução de linhas com vários comandos

Para introduzir mais de um comando numa única linha, separe-os com dois pontos, premindo [2nd] [:].

Introdução de comentários

Um símbolo (◙) de comentário permite introduzir um comentário num programa. Quando o programa for executado, todos os caracteres à direita de ◙ são ignorados.

```
  :prog1()
  :Prgm
❶ :◙Displays sum of 1 thru n
  :Request "Enter an integer",n
❷ :expr(n)→n:◙Convert to numeric expression
  :------
```

❶ Descrição do programa.
❷ Descrição de **expr**.

Nota: utilize comentários para introduzir informações que sejam úteis para alguém que leia o código do programa posteriormente.

Para introduzir o símbolo de comentário, prima:

- [♦] [)] / [2nd] **X**

  – ou –

- Prima [F2] e seleccione **9:◙**

Controlo do fluxo de um programa

Quando um programa é executado, as linhas do programa são executadas por ordem sequencial. Entretanto, alguns comandos alteram o fluxo do programa. Por exemplo:

- As estruturas de controlo como os comandos **If...EndIf** utilizam um teste condicional para decidir que parte do programa deve ser executada.
- As estruturas de repetição, ou loop, como **For...EndFor**, repetem um grupo de comandos.

Utilização do retrocesso

Os programas mais complexos que utilizam **If...EndIf** e estruturas de repetição, como **For...EndFor**, são mais fáceis de ler e entender quando utiliza o retrocesso.

```
:If x>5 Then
:  Disp "x is > 5"
:Else
:  Disp "x is < or = 5"
:EndIf
```

Apresentação de resultados calculados

Num programa, os resultados calculados não aparecem a menos que utilize um comando de saída. Esta é uma diferença importante entre a realização de um cálculo no ecrã principal e num programa.

Num programa, os resultados destes cálculos não aparecem (embora apareçam no ecrã principal).

```
:12*6
:cos(π/4)
:solve(x^2-x-2=0,x)
```

Os comandos de saída, como **Disp**, mostram resultados durante a execução de um programa.

```
:Disp 12*6
:Disp cos(π/4)
:Disp solve(x^2-x-
2=0,x)
```

A apresentação do resultado de um cálculo não guarda o resultado. Se quiser aceder ao resultado posteriormente, guarde-o numa variável.

```
:cos(π/4)→maximum
:Disp maximum
```

Nota: a lista dos comandos de saída está disponível.

Introdução de valores num programa

Para introduzir valores num programa, pode:

- Pedir ao utilizador que guarde um valor (com STO▸) em determinadas variáveis antes da execução do programa.
  O programa pode utilizar estas variáveis durante a execução.

- Introduzir os valores directamente no programa.

```
:Disp 12*6
:cos(π/4)→maximum
```

- Incluir comandos de entrada que peçam ao utilizador para introduzir os valores necessários à execução do programa.

```
:Input "Enter a value",i
:Request "Enter an
integer",n
```

- Pedir ao utilizador para transferir um ou mais valores para o programa quando for executado.

```
prog1(3,5)
```

Nota: a lista de comandos de entrada está disponível.

Exemplo de transferência de valores para um programa

O programa seguinte desenha uma circunferência no ecrã Graph e, em seguida, desenha uma recta horizontal pelo ponto máximo da circunferência. Tem de transferir três valores para o programa: as coordenadas x e y do centro da circunferência e o raio r.

- Quando escrever o programa no Program Editor:

Introduza os dois ( ) ao lado do nome do programa, especifique as variáveis que serão utilizadas para guardar os valores transmitidos.

Não se esqueça de que o programa contém também comandos que configuram o ecrã Graph.

```
:circ(x,y,r) ❶
:Prgm
:FnOff
:ZoomStd
:ZoomSqr
:Circle x,y,r
:LineHorz y+r
:EndPrgm
```

❶ Apenas circ( ) é inicialmente apresentado no modelo em branco; não se esqueça de editar esta linha.

Nota: neste exemplo, não é possível utilizar circle como nome do programa porque entra em conflito com o nome do comando.

Antes de desenhar a circunferência, o programa desactiva todas as funções Y= Editor seleccionadas, mostra uma janela de visualização predefinida, convertendo-a em “quadrada”.

- Para executar o programa a partir do ecrã principal:

  O utilizador precisa especificar os valores aplicáveis como argumentos dentro de ( ).

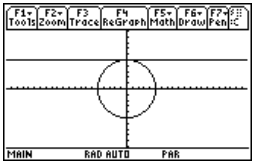
  circ(0,0,5)

  Transmitido para r

  Transmitido para y

  Transmitido para x

  Os argumentos são transmitidos para o programa por ordem.



Nota: este exemplo assume que o utilizador introduz valores que podem ser mostrados pela janela de visualização configurada por **ZoomStd** e **ZoomSqr**.

Descrição da introdução de uma função

Uma função criada no Program Editor é bastante parecida com as funções e instruções utilizadas geralmente a partir do ecrã principal.

Por que criar uma função definida pelo utilizador?

As funções (assim como programas) são ideais quando é necessário realizar um cálculo ou um procedimento várias vezes. A função só precisa ser escrita uma vez e pode ser utilizada quantas vezes for necessário. No entanto, as funções possuem algumas vantagens sobre os programas.

- É possível criar funções que expandem as funções integradas da TI-89 Titanium / Voyage™ 200 Graphing Calculator. As funções criadas são utilizadas da mesma forma que as integradas.
- As funções devolvem valores que podem ser representados graficamente ou introduzidos numa tabela. O mesmo não ocorre com programas.
- Pode utilizar uma função (mas não um programa) dentro de uma expressão. Por exemplo: **3∗func1(3)** é válido, mas não **3∗prog1(3)**.
- Pelo facto de transmitir argumentos para as funções, é possível escrever funções genéricas que não estejam vinculadas a nomes específicos de variáveis.

Nota: pode criar uma função no ecrã principal, mas o Program Editor é mais conveniente para funções complexas com várias linhas.

Diferenças entre funções e programas

Este manual utiliza algumas vezes a palavra comando, referindo-se de forma geral a instruções e funções. No entanto, quando escrever uma função, é imprescindível diferenciar instruções das funções.

Uma função definida pelo utilizador:

- Pode utilizar apenas as seguintes instruções. Todas as outras são inválidas.

Cycle	**Define**	**Exit**
For...EndFor	**Goto**	**If...EndIf** (all forms)
Lbl	**Local**	**Loop...EndLoop**
Return	**While...EndWhile**	→ (STO▸ tecla)

- Pode utilizar todas as funções integradas na TI-89 Titanium / Voyage™ 200, excepto:

  setFold **setGraph** **setMode**
  setTable **switch**

- Pode fazer referência a qualquer variável; no entanto, pode guardar um valor apenas numa variável local.
  - Os argumentos utilizados para transmitir valores para funções são automaticamente tratados como variáveis locais. Se for necessário guardar valores noutras variáveis, será preciso declará-las dentro da função como variáveis locais.
- Não pode chamar um programa como uma subrotina, mas pode chamar uma outra função definida pelo utilizador.
- Não pode definir um programa.
- Não pode definir uma função global, mas pode definir uma função local.

Nota: para mais informações sobre variáveis locais.

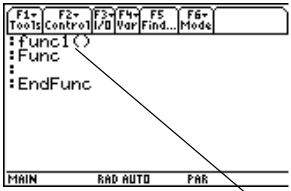
Introdução de uma função

Quando criar uma nova função no Program Editor, a TI-89 Titanium / Voyage™ 200 mostra um “modelo” vazio.

Nota: utilize as teclas de cursor percorrer a função para introduzir ou editar comandos.

Nome da função especificado quando uma nova função é criada.

Introduza os seus comandos entre **Func** e **EndFunc**.



Todas as linhas de função começam com um sinal de dois pontos.

Não se esqueça de editar esta linha para incluir todos os argumentos necessários. Não se esqueça de utilizar nomes de argumentos na definição que nunca

Se a função requerer entrada, tem de transmitir um ou mais valores para a função. (Uma função definida pelo utilizador só pode guardar variáveis locais e não pode utilizar instruções que solicitem a introdução de dados pelo utilizador.)

Como devolver o valor de uma função

Há duas formas de devolver o valor de uma função:

- Na última linha da função (antes de **EndFunc**), calcule o valor a devolver.

```
:cube(x)
:Func
:x^3
:EndFunc
```

- Utilize **Return**. Esta instrução é útil para sair de uma função e devolver o valor a partir de qualquer ponto intermédio da função.

```
:cube(x)
:Func
:If x<0
:  Return 0
:x^3
:EndFunc
```

Nota: este exemplo calcula o cubo se **x≥0**; caso contrário, devolve um `0` como resultado.

O argumento x é tratado automaticamente como uma variável local. No entanto, se o exemplo precisar de outra variável, a função precisaria de a declarar como local com o comando Local.

Há uma instrução implícita **Return** no final da função. Se a última linha não for uma expressão, ocorre um erro.

Exemplo de uma função

A função seguinte calcula a raiz de índice x de y ($\sqrt[x]{y}$). Tem de transmitir os dois valores para a função: x e y.

Nota: pelo facto de x e y serem variáveis locais para a função, não são afectadas por qualquer variável x ou y existente.

Função chamada a partir do ecrã principal	Função definida no Program Editor

3→x:125→y
4∗xroot(3,125) 20
5
	`:xroot(x,y)`
	`:Func`
	`:y^(1/x)`
	`:EndFunc`

Chamada de um programa a partir de um outro

Um programa pode chamar outro programa como uma subrotina. A subrotina pode ser externa (um programa independente) ou interna (incluída no programa principal). As subrotinas são úteis quando um programa precisa de repetir o mesmo grupo de comandos em vários locais diferentes.

Chamada de um programa independente

Para chamar um programa independente, utilize a sintaxe utilizada para executar o programa no ecrã principal.

```
:subtest1()
:Prgm
:For i,1,4,1
:  subtest2(i,i*1000)
:EndFor
:EndPrgm
```

```
:subtest2(x,y)
:Prgm
:  Disp x,y
:EndPrgm
```

Chamada de uma subrotina interna

Para definir uma subrotina interna, utilize o comando **Define** com **Prgm...EndPrgm**. Como uma subrotina tem de ser definida antes de ser chamada, recomendamos que defina subrotinas no início do programa principal.

Uma subrotina interna é chamada e executada da mesma forma que um programa independente.

```
  :subtest1()
  :Prgm
❶ :local subtest2
❷ :Define subtest2(x,y)=Prgm
⋮ :  Disp x,y
❷ :EndPrgm
  :●Beginning of main program
  :For i,1,4,1
❸ :  subtest2(i,I*1000)
  :EndFor
  :EndPrgm
```

❶ Declara a subrotina como uma variável local.

❷ Define a subrotina.

❸ Chama a subrotina.

Nota: utilize o menu F4 **Var** da barra de ferramentas do Program Editor para introduzir os comandos **Define** e **Prgm...EndPrgm**.

Observações sobre a utilização de subrotinas

Quando terminar a subrotina, a execução devolve ao programa chamado. Para sair de uma subrotina a qualquer momento, utilize o comando **Return**.

Uma subrotina não pode aceder às variáveis locais declaradas no programa chamado. De modo analógo, o programa principal não pode aceder às variáveis locais declaradas numa subrotina.

Os comandos **Lbl** são locais para os programas em que estão localizados. No entanto, um comando **Goto** no programa principal não pode desviar-se para um rótulo existente numa subrotina ou vice-versa.

Utilização de variáveis num programa

De um modo geral, os programas tratam as variáveis da mesma forma que o utilizador as utiliza no ecrã principal. No entanto, o “fim” da variável afecta a forma como é guardada e acedida.

Fim das variáveis

Fim	Descrição
Variáveis do sistema (globais)	São variáveis com nomes reservados que são criadas automaticamente para guardar dados sobre o estado da TI-89 Titanium / Voyage™ 200 Graphing Calculator. Por exemplo, as variáveis Window (**xmin**, **xmax**, **ymin**, **ymax** etc.) estão globalmente disponíveis a partir de qualquer pasta. • É possível fazer referência a estas variáveis apenas com o nome da variável, independentemente da pasta actual. • Um programa não pode criar variáveis do sistema, mas pode utilizar os seus valores e (na maior parte dos casos) guardar novos valores.

Fim	Descrição
Variáveis da pasta	São variáveis que são guardadas numa determinada pasta. • Se utilizar apenas o nome de variável, é guardada na pasta actual. Por exemplo: 5→start • Se a referência for feita apenas ao nome da variável, tem de estar na pasta actual. Caso contrário, não será encontrada (mesmo se a variável existir noutra pasta). • Para guardar ou aceder a uma variável guardada noutra pasta, é necessário especificar o caminho. Por exemplo: 5→class\start (class = Nome de pasta; start = Nome de variável) Depois de o programa terminar, todas as variáveis de pasta criadas pelo programa continuam a existir e ocupar a memória.
Variáveis locais	São variáveis temporárias existentes apenas durante a execução do programa. Quando o programa terminar, as variáveis locais são excluídas automaticamente. • Para criar uma variável local num programa, utilize o comando **Local** para declarar a variável. • Uma variável local é tratada como única mesmo se houver uma variável de pasta existente com o mesmo nome. • A utilização de variáveis locais é ideal para guardar valores temporários, que não se deseja guardar.

Nota: se um programa tiver variáveis locais, uma função representada graficamente não pode aceder-lhes. Por exemplo:

```
Local a
5→a
Graph a*cos(x)
```

pode apresentar um erro ou um resultado inesperado (se a variável a existir na pasta actual).

Erros da Circular Definition

Quando calcular uma função definida pelo utilizador ou executar um programa, é possível especificar um argumento que inclua a mesma variável que foi utilizada para definir a função ou criar o programa. No entanto, para evitar erros da Circular Definition, é necessário atribuir um valor para as variáveis x ou i, que são utilizadas no cálculo da função ou na execução do programa. Por exemplo:

```
❶ x+1→x

  – ou–

  For i,i,10,1
❶ Disp i
  EndFor
```

❶ Aparece uma mensagem de erro de `Circular Definition`, se x ou i não tiver um valor. O erro não ocorre se x ou i já tiver um valor atribuído.

Comandos relacionados com as variáveis e funções

Comando	Descrição
STO▸ tecla	Guarda um valor numa variável. Assim como no ecrã principal, prima STO▸ para introduzir um símbolo →.
Archive	Move as variáveis especificadas de RAM para a memória de arquivo de dados do utilizador.
BldData	Permite criar uma variável de dados baseada nas informações gráficas introduzidas no Y=Editor, Window Editor, etc.
CopyVar	Copia o conteúdo de uma variável
Define	Define uma variável de programa (subrotina) ou uma função dentro de um programa.
DelFold	Exclui uma pasta. Todas as variáveis dentro da pasta precisam ser excluídas antes.
DelType	Elimina as variáveis não arquivadas do tipo especificado em todas as pastas.
DelVar	Exclui uma variável.
getFold	Devolve o nome da pasta actual.
getType	Devolve uma cadeia de caracteres que indica o tipo de dados (EXPR, LIST etc.) de uma variável determinada.
isArchiv()	Indica se a variável está arquivada ou não.
isLocked()	Indica se a variável está bloqueada ou não.
isVar()	Indica se a variável está na tabela de símbolos ou não.
Local	Declara uma ou mais variáveis como locais.

Comando	Descrição
Lock	Bloqueia uma variável, impedindo-a de ser excluída ou alterada por engano, até que seja desbloqueada novamente.
MoveVar	Move uma variável de uma pasta para outra.
NewData	Cria uma variável de dados com colunas que consistem de uma série ou de listas especificadas.
NewFold	Cria uma nova pasta.
NewPic	Cria uma variável de imagem com base numa matriz.
Rename	Muda o nome de uma variável.
Unarchiv	Move uma variável especificada da memória de arquivo de dados do utilizador para a RAM.
Unlock	Desbloqueia uma variável bloqueada.

Nota: os comandos **Define**, **DelVar** e **Local** estão disponíveis a partir do menu F4 **Var** da barra de ferramentas do Program Editor.

Utilização de variáveis locais em funções ou programas

Uma variável local é uma variável temporária que existe apenas enquanto uma função definida pelo utilizador está a ser calculada ou um programa definido pelo utilizador está a ser executado.

Exemplo de uma variável local

O programa seguinte mostra um loop **For...EndFor** (que é discutido posteriormente neste módulo). A variável *i* é o contador do loop. Quase sempre a variável *i* só é utilizada enquanto o programa estiver a ser executado.

```
❶ :Local I
  :For i,0,5,1
  :  Disp I
  :EndFor
  :Disp i
```

❶ Declara a variável i como local.

Nota: sempre que puder, utilize as variáveis locais nos casos em que a variável seja utilizada apenas dentro de um programa e não precise de ser guardada depois de o programa parar.

Se declarar a variável *i* como variável local, será automaticamente eliminada quando o programa parar de forma a que não ocupe espaço em memória.

O que provoca uma mensagem de erro Undefined Variable?

Uma mensagem de erro **Undefined** variable aparece quando uma função definida pelo utilizador for calculada ou um programa definido pelo utilizador for executado, fazendo referência a uma variável local não inicializada (cujo valor não foi atribuído).

Este exemplo é uma função de múltiplas instruções, e não um programa. As quebras de linhas são mostradas aqui, mas o texto deve ser digitado na linha de entrada como uma

linha contínua, como por exemplo: **Define fact(n)=Func:Local…** onde as reticências indicam que o texto da linha de entrada continua fora do ecrã.

Por exemplo:

```
  Define fact(n)=Func:
❶ Local m:
  While n>1:
    n*m→m: n-1→n:
  EndWhile:
  Return m:
  EndFunc
```

❶ Não foi atribuído um valor inicial para a variável local m.

No exemplo seguinte, a variável local m existe independente de qualquer variável m que exista fora da função.

As variáveis locais devem ser inicializadas

Todas as variáveis locais devem receber um valor inicial antes de serem referenciadas.

```
  Define fact(n)=Func:
❶ Local m: 1→m:
  While n>1:
    n*m→m: n-1→n:
  EndWhile:
  Return m:
  EndFunc
```

❶ 1 está guardado como valor inicial de m.

A TI-89 Titanium / Voyage™ 200 Graphing Calculator não pode utilizar uma variável local para realizar cálculos simbólicos.

Como executar cálculos simbólicos

Se quiser que uma função ou programa execute cálculos simbólicos, tem de utilizar uma variável global em vez de uma variável local. No entanto, deve ter certeza de que essa variável global não existe fora do programa. Os métodos seguintes podem ajudar.

- Faça referência a um nome de variável global, normalmente com dois ou mais caracteres, que não exista fora da função ou do programa.
- Inclua **DelVar** na função ou no programa para eliminar a variável global, caso exista, antes de lhe fazer referência. (**DelVar** não exclui variáveis bloqueadas ou arquivadas.)

Operações com cadeias de caracteres

As cadeias de caracteres são utilizadas para introduzir e apresentar o texto. Podem ser introduzidas directamente ou guardadas numa variável.

Como as cadeias de caracteres são utilizadas

Uma cadeia de caracteres é uma sequência de caracteres entre "aspas". Na programação, as cadeias de caracteres permitem que o programa mostre informações ou peça ao utilizador para realizar uma determinada acção. Por exemplo:

```
Disp "The result is",answer

– ou –

Input "Enter the angle in degrees",ang1

– ou –

"Enter the angle in degrees”→str1
Input str1,ang1
```

Alguns comandos de entrada (como **InputStr**) guardam automaticamente os dados de entrada do utilizador como uma cadeia de caracteres e não exigem que o utilizador utilize aspas.

Não é possível fazer cálculos matemáticos com cadeias de caracteres, mesmo que pareçam uma expressão numérica. Por exemplo, a cadeia de caracteres "`61`" representa os caracteres "`6`" e "`1`", não o número `61`.

Apesar de não ser possível utilizar cadeias de caracteres como "`61`" ou "`2x+4`" em cálculos, é possível converter uma cadeia de caracteres numa expressão numérica com o comando **expr**.

Comandos de cadeias de caracteres

Nota: para mais informações sobre a sintaxe de todos os comandos e funções da TI-89 Titanium / Voyage™ 200, consulte o módulo *Referência técnica*.

Comando	Descrição
#	Converte uma cadeia de caracteres num nome de variável. Esta operação chama-se conversão indirecta.
&	Une (relaciona) duas cadeias de caracteres.
char	Devolve o carácter que corresponde a um código de carácter específico. Este comando é o oposto do comando **ord**.
dim	Devolve o número de caracteres (tamanho) de uma cadeia de caracteres.
expr	Converte uma cadeia de caracteres numa expressão numérica e executa-a. Este comando é o oposto do comando **string**. **Importante:** alguns comandos de entrada do utilizador guardam o valor introduzido como cadeias de caracteres. Antes de realizar uma operação matemática com um valor introduzido por um destes comandos, é preciso convertê-lo numa expressão numérica.
format	Devolve uma expressão na forma de uma sequência de caracteres baseada no modelo de formato (fixo, científico, engenharia, etc.)
inString	Verifica se uma cadeia de caracteres possui uma subcadeia de caracteres especificada. Se contiver, **inString** volta a posição do carácter onde a primeira ocorrência da subcadeia começa.

Comando	Descrição
left	Devolve um número especificado de caracteres a partir do lado esquerdo (início) de uma cadeia de caracteres.
mid	Devolve um número especificado de caracteres a partir de uma posição qualquer de uma cadeia de caracteres.
ord	Devolve o código do primeiro carácter dentro de uma cadeia de caracteres. Este comando é o oposto do comando **char**.
right	Devolve um número especificado de caracteres a partir do lado direito (final) de uma cadeia de caracteres.
rotate	Desloca os caracteres de uma sequência. A predefinição é ¯1 (desloca um carácter para a direita).
shift	Desloca os caracteres de uma sequência e substitui-os por espaços. A predefinição é ¯1 (desloca um carácter para a direita e o substitui por um espaço). Exemplos: shift("abcde",2)⇒"cde " e shift("abcde")⇒" abcd"
string	Converte uma expressão numérica numa cadeia de caracteres. Este comando é o oposto do comando **expr**.

Testes condicionais

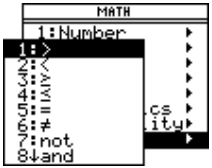
Os testes condicionais permitem aos programas tomarem decisões. Por exemplo, o programa pode decidir a acção a realizar de acordo com o resultado de um teste (verdadeiro ou falso). Os testes condicionais são efectuados com estruturas de controlo como **If...EndIf** e estruturas de repetição como **While...EndWhile** (descritas posteriormente neste módulo).

Introdução de um operador

- Digite o operador directamente a partir do teclado.
  – ou –

- Prima [2nd] [MATH] e seleccione **8:Test**. Em seguida, seleccione o operador a partir do menu.

  – ou –



- Veja as funções internas. Prima:
  [CATALOG]
  [2nd] [CATALOG]
  Os operadores de teste estão listados junto da linha inferior do menu [F2] Built-in.

Operadores relacionais

Os operadores relacionais permitem comparar dois valores num teste condicional. Os valores podem ser números, expressões, listas ou matrizes (mas precisam coincidir em tipo e dimensão).

Operador	Verdadeiro se:	Exemplo
>	Maior que	a>8
<	Menor que	a<0
≥	Maior ou igual que	a+b≥100
≤	Menor ou igual que	a+6≤b+1
=	Igual	list1=list2

Operador	Verdadeiro se:	Exemplo
≠	Diferente de	mat1≠mat2

Nota: pode digitar no teclado:
>= para ≥
<= para ≤
/= para ≠
(Para obter o carácter /, prima [÷].)

Operadores booleanos

Os operadores booleanos permitem combinar resultados de dois testes.

Operador	Verdadeiro se:	Exemplo
and	Os dois testes são verdadeiros	a>0 and a≤10
or	Pelo menos um teste é verdadeiro	a≤0 or b+c>10
xor	Um teste é verdadeiro e o outro é falso	a+6<b+1 xor c<d

A função Not

A função **not** inverte o resultado de um teste de verdadeiro para falso e vice-versa. Por exemplo:

not x>2	é verdadeiro	x≤2
	falso se	x>2

Nota: se **not** for utilizado a partir do ecrã principal, aparece como **~** na área do histórico. Por exemplo, **not x>2** aparece como **~(x>2)**.

Utilização de If, Lbl e Goto para controlar o fluxo de programa

As estruturas **If...EndIf** utilizam um teste condicional para decidir se um ou mais comandos serão executados. Os comandos **Lbl** (rótulo) e **Goto** são utilizados para desviar (ou pular) de uma posição do programa para outra.

Menu F2 Control da barra de ferramentas

Para introduzir a estrutura **If...EndIf**, utilize o menu F2 **Control** da barra de ferramentas do Program Editor.

O comando **If** está disponível directamente a partir do menu F2.

Para visualizar um submenu que liste todas as estruturas **If**, seleccione **2:If...Then**.

Quando seleccionar uma estrutura como **If...Then...EndIf**, um modelo é inserido na posição do cursor.

```
:If | Then ❶
:EndIf
```

❶ O cursor é posicionado de forma a que um teste condicional possa ser introduzido.

Comando If

Para executar apenas um comando se um teste condicional for verdadeiro, utilize a forma geral:

```
  :If x>5
❶ :   Disp "x is greater than 5"
❷ :Disp x
```

❶ Executado apenas se x>5; for omitido, caso contrário.
❷ Mostra o valor de x, independentemente do seu valor.

Neste exemplo, é preciso guardar um valor para x antes de executar o comando **If**.

Nota: utilize o retrocesso para tornar os seus programas mais fáceis de ler e compreender.

Estruturas If...Then...EndIf

Para executar vários comandos se um teste condicional for verdadeiro, utilize a estrutura:

```
  :If x>5 Then
❶ :  Disp "x is greater than 5"
❶ :  2*x→x
❷ :EndIf
  :Disp x
```

❶ Executado apenas se x>5.
❷ Mostra o valor de:
- 2x se x>5
- x se x≤5

Nota: EndIf indica o fim do bloco iniciado por **Then**, que é executado se a condição for verdadeira.

Estruturas If...Then...Else... EndIf

Para executar um grupo de comandos se um teste condicional for verdadeiro e um grupo diferente se a condição for falsa, utilize esta estrutura:

```
  :If x>5 Then
❶ :  Disp "x is greater than 5"
❶ :  2*x→x
  :Else
❷ :  Disp "x is less than or
❷    equal to 5"
  :  5*x→x
  :EndIf
❸ :Disp x
```

❶ Executado apenas se x>5.
❷ Executado apenas se x≤5.
❸ Mostra o valor de:
- 2x se x>5
- 5x se x≤5

Estruturas If...Then...ElseIf... EndIf

Uma forma mais complexa do comando **If** permite que uma série de condições sejam testadas. Suponha que o seu programa pede ao utilizador um número correspondente a uma dentro quatro opções. Para testar cada uma das opções (**If Choice = 1**, **If Choice = 2**...), utilize a estrutura **If...Then...ElseIf...EndIf**.

Consulte o módulo *Referência técnica* para obter informações e ver um exemplo.

Os comandos Lbl e Goto

Também é possível controlar o fluxo do programa com os comandos **Lbl** (rótulo) e **Goto**.

Utilize o comando **Lbl** para atribuir um nome a uma posição particular dentro do programa.

Lbl *NomedoRótulo*

└ nome a atribuir para esta posição (utilize a convenção de criação de nomes de variáveis)

O comando **Goto** é utilizado em qualquer parte do programa para o desviar para um local correspondente ao rótulo especificado.

Goto *NomedoRótulo*

└ especifica para que comando **Lbl** deve desviar

Como o comando **Goto** é incondicional (sempre desvia para o rótulo especificado), é utilizado frequentemente com um comando **If** para que possa ser especificado um desvio condicional. Por exemplo:

```
  :If x>5
❶ :  Goto GT5
❷ :Disp x
  :--------
  :--------
  :Lbl GT5
  :Disp "The number was > 5”
```

❶ Se x>5, desvia directamente para o rótulo GT5.
❷ Neste exemplo, o programa precisaria de incluir comandos (como **Stop**) para impedir que Lbl GT5 fosse executado se x≤5.

Utilização de loops para repetir um grupo de comandos

Para repetir o mesmo grupo de comandos sucessivamente, utilize um loop. Estão disponíveis vários tipos de loops. Cada tipo possui uma forma diferente de sair do loop, que se baseia num teste condicional.

Menu F2 Control da barra de ferramentas

Para introduzir a maioria dos comandos relacionados com loop, utilize o menu F2 **Control** do Program Editor.

Quando seleccionar um loop, o comando loop e o seu comando **End** correspondente são inseridos na posição do cursor.

```
:For | ❶
:EndFor
```

❶ Se o loop precisar de argumentos, o cursor será posicionado após o comando.

Em seguida, pode introduzir os comandos que serão executados dentro do loop.

Nota: um comando de loop indica o início do loop. O comando **End** correspondente ao início do loop indica o fim do loop.

Loops For...EndFor

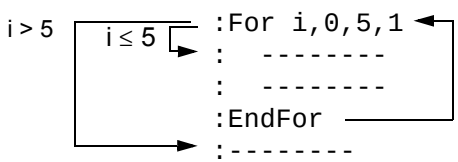
Um loop **For...EndFor** utiliza um contador para controlar o número de vezes que o loop é repetido. A sintaxe do comando **For** é:

Nota: o valor final pode ser inferior ao valor inicial, mas o incremento precisa de ser negativo.

For(*variável*, *início*, *fim*[, *incremento*]**)**
❶ ❷ ❸ ❹

❶ variável utilizada como contador
❷ valor do contador utilizado na primeira vez que **For** é executado
❸ sai do loop quando *variável* excede este valor

Quando a instrução **For** for executada, o valor da variável é comparado ao valor final. Se a variável não excede o valor final, as instruções dentro do loop serão executadas; caso contrário, o controlo do programa desviar-se-á para o comando após o **EndFor**.



Nota: o comando **For** incrementa automaticamente a variável do contador para que o programa possa sair do loop após um certo número de repetições.

Na última instrução do loop (**EndFor**), o controlo do programa volta para o comando **For**, onde o valor da variável é incrementado e comparado com o valor *final*.

Por exemplo:

```
  :For i,0,5,1
❶ :  Disp I
  :EndFor
❷ :Disp i
```

❶ Mostra 0, 1, 2, 3, 4 e 5.
❷ Mostra 6. Quando a *variável* atingir o valor 6, o loop não é mais executado.

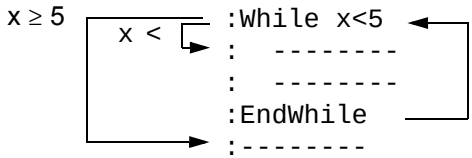
Nota: é possível declarar a variável do contador como local, se não for necessário guardar a variável depois de o programa terminar.

Loops While...EndWhile

O loop **While...EndWhile** repete um bloco de comandos enquanto uma condição especificada for verdadeira. A sintaxe do comando **While** é:

While *condição*

Quando a instrução **While** é executada, a condição é calculada. Se a condição for verdadeira, o loop será executado; caso contrário, o controlo do programa desviar-se-á para o comando após **EndWhile**.



Nota: o comando **While** não muda a condição automaticamente. É preciso incluir comandos que permitam que o programa saia do loop.

Na última instrução do loop (**EndWhile**), o controlo do programa volta para o comando **While**, onde a condição é recalculada.

Para que o loop possa ser executado pela primeira vez, a condição precisa ser inicialmente verdadeira.

- Todas as variáveis referenciadas na condição têm de ser definidas antes do comando **While**. (Os valores podem ser integrados no programa ou introduzidos pelo utilizador.)
- O loop tem de conter comandos que mudam os valores contidos na condição, fazendo com que seja falsa num determinado momento. Caso contrário, a condição será sempre verdadeira e o programa não sairá do loop (chamado de loop infinito).

Por exemplo:

```
❶ :0→x
  :While x<5
❷ :  Disp x
❸ :  x+1→x
  :EndWhile
❹ :Disp x
```

❶ Inicialmente define x.
❷ Mostra 0, 1, 2, 3 e 4.
❸ Incrementa x.
❹ Mostra 5. Quando x atingir 5, o loop não é mais executado.

Loops Loop...EndLoop

Um **Loop...EndLoop** gera um loop infinito, que se repete infinitamente. O comando **Loop** não possui argumento.

```
:Loop
:  --------
:  --------
:EndLoop
:--------
```

Geralmente, o loop contém comandos que permitem ao programa sair do loop. Os comandos mais utilizados são: **If**, **Exit**, **Goto** e **Lbl** (rótulo). Por exemplo:

```
  :0→x
  :Loop
  :  Disp x
  :  x+1→x
❶ :  If x>5
  :    Exit
  :EndLoop
❷ :Disp x
```

❶ Um comando **If** verifica a condição.
❷ Sai do loop e desvia para este ponto quando x atingir 6.

Nota: o comando **Exit** sai do loop actual.

Neste exemplo, o comando **If** pode estar em qualquer posição no loop.

Posição do comando If:	Comportamento do loop:
Na primeira instrução do loop	O loop só será executado, se a condição for verdadeira.
Na última instrução do loop	O loop será executado pelo menos uma vez e só se repete enquanto a condição for verdadeira.

Para sair do loop, o comando **If** poderia utilizar também o comando **Goto** para transferir o controlo do programa para um comando **Lbl** (rótulo) especificado.

Repetição imediata de um loop

O comando **Cycle** transfere imediatamente o controlo do programa para a próxima iteração de um loop (antes da iteração actual se completar). Este comando funciona com **For...EndFor**, **While...EndWhile** e **Loop...EndLoop**.

Loops Lbl e Goto

Apesar de os comandos **Lbl** (rótulo) e **Goto** não serem estritamente comandos de loop, podem ser utilizados para criar um loop infinito. Por exemplo:

```
:Lbl START
:  --------
:  --------
:Goto START
:--------
```

Assim como **Loop...EndLoop**, o loop deve conter comandos que permitam ao programa sair do loop.

Configuração da TI-89 Titanium / Voyage™ 200

Os programas podem conter comandos que alterem a configuração da TI-89 Titanium / Voyage™ 200 Graphing Calculator. Como as mudanças de modo são particularmente úteis, o menu **Mode** da barra de ferramentas do Program Editor simplifica a introdução da sintaxe correcta do comando **setMode**.

Comandos de configuração

Comando	Descrição
getConfg	Devolve uma lista das características da calculadora.
getFold	Devolve o nome da pasta actual.
getMode	Devolve a definição actual de um modo especificado.
getUnits	Devolve uma lista das unidades predefinidas.
getFold	Define a pasta actual.
setGraph	Define um formato gráfico especificado (**Coordinates**, **Graph Order** etc.).

Comando	Descrição
setMode	Define todos os modos, excepto **Current Folder**.
setTable	Define um parâmetro de configuração de tabela específico (**tblStart**, **Δtbl** etc.)
setUnits	Define as unidades predefinidas dos resultados exibidos.
switch	Define a janela activa num ecrã dividido ou devolve o número da janela activa.

Nota: As cadeias de caracteres de parâmetro/modo utilizadas nas funções **setMode( )**, **getMode( )**, **setGraph( )** e **setTable( )** não são traduzidas para outros idiomas quando utilizadas num programa. Consulte o módulo *Referência técnica*.

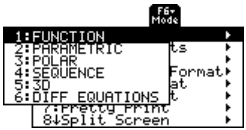
Introdução do comando SetMode

No Program Editor:

1. Posicione o cursor no ponto em que deseja inserir o comando **setMode**.

2. Prima:
   [2nd] [F6]
   [F6]
   para ver uma lista de modos.

   Nota: o menu **Mode** não permite definir o modo **Current Folder**. Para definir este modo, utilize o comando **setFold**.



3. Seleccione um modo para ver um menu das configurações válidas.

4. Seleccione uma configuração.

A sintaxe correcta é inserida `:setMode("Graph","FUNCTION")` no seu programa.

Obtenção de dados introduzidos pelo utilizador e apresentação do resultado

Apesar de ser possível integrar valores num programa (ou previamente guardados em variáveis), um programa pode solicitar que o utilizador introduza informações durante a sua execução. De modo análogo, um programa pode mostrar informações como o resultado de um cálculo.

Menu F3 I/O da barra de ferramentas

Para introduzir a maior parte dos comandos de entrada/saída mais utilizados, utilize o menu F3 **I/O** da barra de ferramentas do Program Editor.

Para ver um submenu com os comandos adicionais, seleccione **1:Dialog**.

Comandos de entrada

Comando	Descrição
getKey	Devolve o código da próxima tecla premida. Consulte o módulo *Referência técnica* para obter informações sobre código de tecla.
Input	Pede ao utilizador que introduza uma expressão. A expressão é tratada da forma como é introduzida. Por exemplo: • Uma expressão numérica é tratada como uma expressão. • Uma expressão entre "aspas" é tratada como uma cadeia de caracteres. **Input** pode também mostar o ecrã Graph e permitir que o utilizador actualize as variáveis xc e yc (rc e θc no modo polar) pelo posicionamento do cursor gráfico.
InputStr	Pede ao utilizador que introduza uma expressão. A expressão é sempre tratada como uma cadeia de caracteres; o utilizador não tem de colocar a expressão entre "aspas".
PopUp	Mostra uma caixa de menu pendente e permite seleccionar um item.
Prompt	Pede ao utilizador que introduza uma série de expressões. Assim como com **Input**, cada expressão é tratada de acordo com a forma como foi introduzida.
Request	Mostra uma caixa de diálogo que pede ao utilizador para introduzir uma expressão. **Request** trata sempre a expressão introduzida como uma cadeia de caracteres.

Nota: uma cadeia de caracteres não pode ser utilizada num cálculo. Para converter um cálculo numa expressão numérica, utilize o comando **expr**.

Comandos de saída

Comando	Descrição
ClrIO	Limpa o ecrã Program I/O.
Disp	Mostra uma expressão ou cadeia de caracteres no ecrã Program I/O. **Disp** também pode mostrar o conteúdo actual do ecrã Program I/O sem mostrar informações adicionais.
DispG	Mostra o conteúdo actual do ecrã Graph.
DispHome	Mostra o conteúdo actual do ecrã principal.
DispTbl	Mostra o conteúdo actual do ecrã Table.
Output	Mostra uma expressão ou cadeia de caracteres no ecrã Program I/O começado por coordenadas especificadas.
Format	Formata o modo no qual as informações numéricas são mostradas.
Pause	Suspende a execução do programa até o utilizador premir [ENTER]. Por exemplo, é útil para permitir a visualização de uma expressão. A pausa permite que o utilizador leia o resultado e decida quando continuar.
Text	Mostra uma caixa de diálogo com uma cadeia específica de caracteres.

Nota:

- Num programa, a realização de um cálculo não mostra o resultado. É preciso utilizar um comando de saída.

- Depois de **Disp** e **Output**, o programa continua imediatamente. Pode desejar adicionar um comando **Pause**.

Comandos da interface gráfica do utilizador

Comando	Descrição
Dialog... **EndDlog**	Define um bloco de programa (composto dos comandos **Title**, **Request** etc) que mostra uma caixa de diálogo.
Toolbar... **EndTbar**	Define um bloco de programa (composto pelos comandos Title, Item etc) que substitui os menus da barra de ferramentas. A barra de ferramentas redefinida só tem efeito enquanto o programa é executado e até o utilizador seleccionar um item. Em seguida, a barra original aparece novamente.
CustmOn... **CustmOff**	Activa ou remove uma barra de ferramentas personalizada.
Custom... **EndCustm**	Define um bloco de programa que mostra uma barra de ferramentas personalizada quando o utilizador prime [2nd] [CATALOG]. A barra de ferramentas permanece activa até que o utilizador prima [2nd] [CATALOG] novamente ou mude as aplicações.
DropDown	Mostra um menu instantâneo dentro de uma caixa de diálogo.
Item	Mostra um item de menu de uma barra de ferramentas redefinida.

Comando	Descrição
Request	Cria uma caixa de entrada dentro de uma caixa de diálogo.
Text	Mostra uma cadeia de caracteres dentro de uma caixa de diálogo.
Title	Mostra o título de uma caixa de diálogo ou de um menu dentro de uma barra de ferramentas.

Nota:

- Quando um programa que configura uma barra de ferramentas personalizada for executado, a barra de ferramentas ficará disponível mesmo quando o programa terminar.

- **Request** e **Text** são comandos isolados que podem ser também utilizados do lado de fora de uma caixa de diálogo ou de um bloco de programa da barra de ferramentas.

Criação de um menu personalizado

O menu personalizado da TI-89 Titanium / Voyage™ 200 Graphing Calculator permite criar o seu próprio menu na barra de ferramentas. Um menu personalizado pode conter qualquer função, instrução ou conjunto de caracteres disponível. A TI-89 Titanium / Voyage™ 200 tem um menu personalizado predefinido que pode ser modificado ou redefinido.

Activação e desactivação do menu personalizado

Quando criar um menu personalizado, este pode ser activado ou desactivado manualmente pelo utilizador, ou automaticamente pelo programa.

Para:	Efectue o procedimento apresentado em seguida:
Activar o menu personalizado	No ecrã principal ou em qualquer outra aplicação: • Prima 2nd [CATALOG]. No ecrã principal ou num programa: • Execute o comando **CustmOn**.
Desactivar o menu personalizado	Em qualquer aplicação: • Prima 2nd [CATALOG] novamente. – ou – • Vá para uma aplicação diferente. No menu personalizado predefinido do ecrã principal: 1. Seleccione o menu **Tools**: 2nd [F7] F7 Seleccione **3:CustmOff**. Este procedimento **CustmOff** na linha de entrada. 2. Prima ENTER. utilize apenas **CustmOff** num programa. É também possível utilizar **CustmOff** num programa.

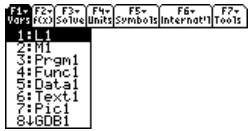
CustmOff

Nota: quando activar o menu personalizado, este substitui o menu da barra de ferramentas normal. Excepto se tiver criado um menu personalizado diferente, aparece o menu personalizado predefinido.

Definição de um menu personalizado

Para criar um menu personalizado, utilize a seguinte estrutura geral.

:
Custom
: **Title** *título do menu F1*
: **Item** *item 1*
: **Item** *item 2*
: …
: **Title** *título do menu F2*
: …
: **Title** *título do menu F3*
: …



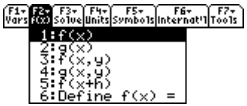
Nota: quando o utilizador seleccionar um item do menu, o texto definido por aquele comando **Item** é colado na posição actual do cursor.

Por exemplo

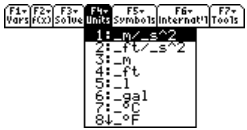
```
:Custom
:Title "Vars"
:Item "L1":Item "M1":Item "Prgm1":Item "Func1":Item "Data1"
:Item "Text1":Item "Pic1":Item "GDB1":Item "Str1"
❶ :Title "f(x)"
:Item "f(x)":Item "g(x)":Item "f(x,y)":Item "g(x,y)"
```

```
❶  :Item "f(x+h)":Item "Define f(x) ="
   :Title "Solve"
   :Item "Solve(":Item " and ":Item "{x,y}"
   :Item "Solve( and ,{x,y})"
❷  :Title "Units"
   :Item "_m/_s^2":Item "_ft/_s^2":Item "_m":Item "_ft":Item "_l"
❷  :Item "_gal":Item "_\o\C":Item "_\o\F":Item "_kph":Item "_mph"
   :Title "Symbols"
   :Item "#":Item "\beta\":Item "?":Item "~":Item "&"
   :Title "Internat'l"
   :Item "\e`\":Item "\e'\":Item "\e^\":Item "\a`\"
   :Item "\u`\":Item "\u^\":Item "\o^\":Item "\c,\":Item "\u..\"
   :Title "Tools"
   :Item "ClrHome":Item "NewProb":Item "CustmOff"
   :EndCustm
   :CustmOn
```

Nota: o menu personalizado seguinte pode ser ligeiramente diferente do menu predefinido da sua calculadora.



❶



❷

Nota: veja como "_\o\C" e "_\o\F" aparecem como °C e °F no menu. Da mesma forma, veja os caracteres internacionais acentuados.

Para modificar o menu personalizado predefinido, utilize **3:Restore custom default** (como descrito abaixo) para obter os comandos para o menu predefinido. Copie esses

comandos, utilize o Program Editor para criar um novo programa e copie para o programa em branco. Em seguida, modifique os comandos, conforme necessário.

Nota: este procedimento insere todos os comandos numa única linha. Não tem de os dividir em linhas separadas.

Só pode criar e utilizar um menu personalizado de cada vez. Se precisar de mais, escreva um programa separado para cada menu personalizado. Em seguida, execute o programa para o menu que precisa.

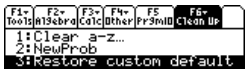
Restauração do menu personalizado predefinido

Para restaurar a predefinição:

1. No menu normal do ecrã principal (não o menu personalizado), seleccione **Clean Up**.
   [2nd] [F6]
   [F6]

2. Seleccione **3:Restore custom default**.

   Este procedimento cola os comandos utilizados para criar o menu predefinido na linha de entrada.



3. Prima [ENTER] para executar os comandos e restaurar a predefinição.

Quando restaurar a predefinição, qualquer menu personalizado anterior será apagado. Se o menu anterior tiver sido criado com um programa, poderá executar o programa novamente, se quiser reutilizar o menu posteriormente.

Criação de uma tabela ou um gráfico

Para criar uma tabela ou um gráfico baseado numa ou mais funções ou equações, utilize os comandos listados nesta secção.

Comandos de tabela

Comando	Descrição
DispTbl	Mostra o conteúdo actual do ecrã Table.
setTable	Define os parâmetros da tabela Independent ou Graph <−> Table. (Para definir os outros dois parâmetros de tabela, pode guardar os valores apropriados nas variáveis do sistema **tblStart** e **Δtbl**.)
Table	Constrói e mostra uma tabela baseada numa ou mais expressões ou funções.

Comandos gráficos

Comando	Descrição
ClrGraph	Apaga todas as funções ou expressões que foram representadas graficamente com o comando **Graph**.
Define	Cria uma função definida pelo utilizador.
DispG	Mostra o conteúdo actual do ecrã Graph.
FnOff	Desmarca todas as funções Y= (ou apenas as especificadas).

Comando	Descrição
FnOn	Selecciona todas as funções Y= (ou apenas as especificadas).
Graph	Representa graficamente uma ou mais expressões especificadas com o modo actual de representação gráfica.
Input	Mostra o ecrã Graph e permite actualizar as variáveis xc e yc (rc e θc no modo polar) pelo posicionamento do cursor gráfico.
NewPlot	Cria uma nova definição de gráfico estatístico.
PlotsOff	Desmarca todos os gráficos de dados estatísticos (ou apenas os especificados).
PlotsOn	Selecciona todos os gráficos de dados estatísticos (ou apenas os especificados).
setGraph	Muda as definições de vários formatos gráficos (**Coordinates**, **Graph Order** etc.).
setMode	Define o modo Graph, além de outros modos.
Style	Define o estilo de apresentação de uma função.
Trace	Permite que um programa trace um gráfico.
ZoomBox – até – **ZoomTrig**	Realiza todas as operações de **Zoom** disponíveis a partir do menu F2 da barra de ferramentas ou no Y= Editor, no Window Editor e no ecrã Graph.

Nota: para mais informações sobre a utilização de **setMode.**

Comandos da base de dados e imagem gráfica

Comando	Descrição
AndPic	Mostra o ecrã Graph e sobrepõe uma imagem gráfica guardada com a lógica **AND**.
CyclePic	Anima uma série de imagens gráficas guardadas.
NewPic	Cria uma variável de imagem gráfica baseada numa matriz.
RclGDB	Restaura todas as definições guardadas numa base de dados gráficos.
RclPic	Mostra o ecrã Graph e sobrepõe uma imagem gráfica guardada com a lógica **OR**.
RplcPic	Limpa o ecrã Graph e exibe uma imagem gráfica guardada.
StoGDB	Guarda as definições gráficas actuais numa variável da base de dados gráficos.
StoPic	Copia o ecrã Graph (ou uma parte rectangular especificada) para uma variável de imagem gráfica.
XorPic	Mostra o ecrã Graph e sobrepõe uma imagem gráfica guardada com a lógica **XOR**.

Nota: para mais informações sobre as imagens gráficas e a base de dados, consulte também o módulo *Tópicos complementares de representação gráfica*.

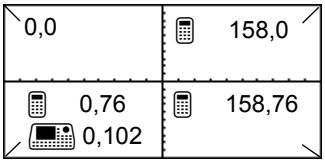
Desenho no ecrã Graph

Os comandos apresentados nesta secção permitem criar objectos de desenho no ecrã Graph.

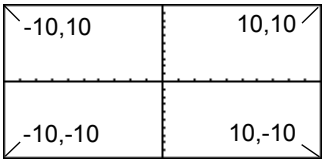
Comparação entre coordenadas de pixel e do ponto

Quando desenhar um objecto, é possível utilizar um dos dois sistemas de coordenadas para especificar uma posição no ecrã.

- **Coordenadas de pixel** — Referem-se aos pixéis que constituem fisicamente o ecrã. Estes são independentes da janela de visualização porque o ecrã tem:
  159 (0 a 158) pixéis de largura e 77 (0 a 76) pixéis de altura.
  239 (0 a 238) pixéis de largura e 103 (0 a 102) pixéis de altura.
- **Coordenadas do ponto** — Referem-se às coordenadas válidas para a janela de visualização activa (conforme definido no Window Editor).



Coordenadas de pixel (independentes da janela de visualização)



Coordenadas de ponto (para janela de visualização predefinida)

Nota: para mais informações sobre as coordenadas de pixel no ecrã dividido, consulte o módulo *Ecrãs divididos*.

Muitos comandos de desenho possuem duas formas: uma para coordenadas de pixel e outra para coordenadas do ponto.

Nota: os comandos de pixel iniciam com **PxI**, como **PxIChg**.

Eliminação de objectos desenhados

Comando	Descrição
ClrDraw	Apaga todos os objectos desenhados do ecrã Graph.

Desenho de um ponto ou de um pixel

Comando	Descrição
PtChg ou **PxIChg**	Alterna (inverte) um pixel nas coordenadas especificadas. **PtChg**, que utiliza coordenadas do ponto, afecta o pixel mais próximo do ponto especificado. Se o pixel estiver desactivado, será activado. Se o pixel estiver activado, será desactivado.
PtOff ou **PxIOff**	Desactiva (apaga) um pixel nas coordenadas especificadas. **PtOff**, que utiliza coordenadas de ponto, afecta o pixel mais próximo do ponto especificado.
PtOn ou **PxIOn**	Activa (mostra) um pixel nas coordenadas especificadas. **PtOn**, que utiliza coordenadas do ponto, afecta o pixel mais próximo do ponto especificado.
PtTest ou **PxITest**	Devolve verdadeiro ou falso para indicar se a coordenada especificada está activada ou desactivada, respectivamente.

Comando	Descrição
PtText ou **PxlText**	Mostra uma cadeia de caracteres nas coordenadas especificadas.

Desenho de rectas e circunferências

Comando	Descrição
Circle ou **PxlCrcl**	Desenha, apaga ou inverte uma circunferência com um centro e um raio especificados.
DrawSlp	Desenha uma recta com uma inclinação especificada, passando por um ponto especificado.
Line ou **PxlLine**	Desenha, apaga ou inverte uma recta entre dois pares de coordenadas.
LineHorz ou **PxlHorz**	Desenha, apaga ou inverte uma recta horizontal numa coordenada de linha especificada.
LineTan	Desenha uma recta tangente a uma expressão especificada num ponto especificado. (Este comando desenha apenas a recta tangente, não a expressão.)
LineVert ou **PxlVert**	Desenha, apaga ou inverte uma recta vertical numa coordenada de coluna especificada.

Expressões de desenho

Comando	Descrição
DrawFunc	Desenha uma expressão especificada.

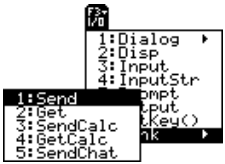
Comando	Descrição
DrawInv	Desenha o inverso de uma expressão especificada.
DrawParm	Desenha uma equação paramétrica com expressões especificadas como as suas componentes x e y.
DrawPol	Desenha uma expressão polar especificada.
DrwCtour	Desenha contornos no modo de representação gráfica em 3D.
Shade	Desenha duas expressões e utiliza tons diferentes para diferenciar as áreas onde *expressão1* < *expressão2*.

Acesso a outra TI-89 Titanium / Voyage™ 200, ao CBL 2 ou ao CBR

Se ligar duas TI-89 Titanium / Voyage™ 200 Graphing Calculator (descrito no módulo *Ligação e Actualização*), os seus programas podem transmitir variáveis entre si. Se uma TI-89 Titanium / Voyage™ 200 estiver ligada a um Calculator-Based Laboratory™ (CBL 2™) ou ao Calculator-Based Ranger™ (CBR™), um programa na TI-89 Titanium / Voyage™ 200 pode aceder ao CBL 2 ou ao CBR.

Menu F3 I/O da barra de ferramentas

Utilize o menu F3 **I/O** da barra de ferramentas do Program Editor para introduzir os comandos nesta secção.



1. Prima F3 e seleccione **8:Link**.
2. Seleccione um comando.

Acesso a outra TI-89 Titanium / Voyage™ 200

Quando ligar duas calculadoras TI-89 Titanium / Voyage™ 200, uma delas actua como unidade receptora e a outra como unidade transmissora.

Comando	Descrição
GetCalc	Executado na unidade de recepção. Configura a unidade para receber uma variável pela porta de I/O. • Depois de a unidade de recepção executar **GetCalc**, a unidade de transmissão tem de executar **SendCalc**. • Depois de a unidade de transmissão executar **SendCalc**, a variável transmitida será guardada na unidade de recepção (no nome de variável especificado por **GetCalc**).
SendCalc	Executado na unidade de transmissão. Envia uma variável para unidade de recepção pela porta de I/O. • Antes de a unidade de transmissão executar **SendCalc**, a unidade de recepção tem executar **GetCalc**.

Comando	Descrição
SendChat	Executado na unidade de transmissão com uma alternativa geral a **SendCalc**. Útil se a unidade de recepção for uma TI-92 (ou para um programa de troca de messagens que permite utilizar uma TI-92, TI-92 Plus ou de uma Voyage™ 200).

Nota: para ver um programa de exemplo que sincronize as unidades de recepção e transmissão de forma a que **GetCalc** e **SendCalc** sejam executados na sequência adequada, consulte “Transmissão de variáveis sob o controlo de programa” no módulo *Ligação e Actualização*.

Acesso a um sistema CBL 2 ou CBR

Para mais informações, consulte o manual das unidades CBL 2 ou CBR.

Comando	Descrição
Get	Obtém uma variável do sistema CBL 2 ou CBR ligado e guarda-a armazena na TI-89 Titanium / Voyage™ 200.
Send	Envia uma variável de lista da TI-89 Titanium / Voyage™ 200 para o sistema CBL 2 ou CBR.

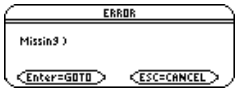
Depuração de programas e tratamento de erros

Depois de escrever um programa, pode utilizar algumas técnicas para localizar e corrigir erros. É possível também criar um comando de tratamento de erros integrado no programa.

Erros do tempo de execução

O primeiro passo na depuração do seu programa é executá-lo.
A TI-89 Titanium / Voyage™ 200 Graphing Calculator verifica automaticamente se há erros de sintaxe em todos os comandos executados. Se ocorrer um erro, uma mensagem indica a natureza do erro.

- Para ver o programa no Program Editor, prima [ENTER]. O cursor aparece junto ao erro.



- Para cancelar a execução do programa e voltar ao ecrã principal, prima [ESC].

Se o programa permitir seleccionar uma das opções, execute o programa e teste todas as opções.

Técnicas de depuração

As mensagens de erro do tempo de execução podem localizar erros de sintaxe, mas não erros na lógica de programação. As técnicas seguintes podem ser úteis.

- Durante a fase de testes, não utilize variáveis locais para verificar os valores das variáveis depois de o programa terminar. Depois de a depuração terminar, declare as variáveis em questão como sendo locais.
- Insira temporariamente os comandos **Disp** e **Pause** no programa para ver valores de variáveis importantes.
  - **Disp** e **Pause** não podem ser utilizados numa função definida pelo utilizador. Para transformar temporariamente uma função num programa, mude **Func** e **EndFunc** para **Prgm** e **EndPrgm**. Utilize **Disp** e **Pause** para depurar o programa.

Em seguida, remova **Disp** e **Pause** e transforme o programa novamente em função.

- Para confirmar que um loop é executado o número correcto de vezes, mostre a variável do contador ou os valores do teste condicional.
- Para confirmar que uma subrotina é executada, mostre mensagens como `Entering subroutine` e `Exiting subroutine` no início e no fim da subrotina.

Comandos de tratamento de erro

Comando	Descrição
Try...EndTry	Define um bloco de programa que permite que o programa execute um comando e, se necessário, se restaure de um erro gerado por aquele comando.
ClrErr	Limpa o estado de erro e define o número de erro da variável de sistema Errornum como zero.
PassErr	Passa um erro para o nível seguinte do bloco **Try...EndTry**.

Exemplo: Utilização de abordagens alternativas

A apresentação introdutória no módulo *Previews* mostra um programa que pede que ao utilizador introduza um número inteiro, some todos os inteiros de 1 até o número introduzido e mostre o resultado. Esta secção apresenta outras abordagens que podem ser utilizadas para alcançar o mesmo objectivo.

Exemplo 1

Este exemplo utiliza **InputStr** para entrada, um loop **While...EndWhile** para calcular o resultado e **Text** para mostrar o resultado.

```
  :prog1()
  :Prgm
❶ :InputStr "Enter an integer",n
❷ :expr(n)→n
  :0→temp:1→I
❸ :While i≤n
⋮ :  temp+i→temp
⋮ :  i+1→I
❸ :EndWhile
❹ :Text "The answer is "&string(temp)
  :EndPrgm
```

❶ Pede dados de entrada no ecrã Program I/O.
❷ Converte a cadeia de caracteres introduzida com **InputStr** numa expressão.
❸ Cálculo no loop.
❹ Mostra a saída numa caixa de diálogo.

Nota: Para obter ≤, digite [◆] [0] (zero). Para obter **&**, prima:
[◆] [×] (vezes)
[2nd] **H**

Exemplo 2

Este exemplo utiliza **Prompt** para entrada, **Lbl** e **Goto** para criar um loop e **Disp** para mostrar o resultado.

```
  :prog2()
  :Prgm
❶ :Prompt n
  :0→temp:1→I
❷ :Lbl top
⋮ :  temp+i→temp
⋮ :  i+1→I
⋮ :  If i≤n
❷ :    Goto top
❸ :Disp temp
  :EndPrgm
```

❶ Pede dados de entrada no ecrã Program I/O.
❷ Cálculo no loop.
❸ Mostra a saída no ecrã Program I/O.

Nota: como **Prompt** devolve n como um número, não é preciso utilizar **expr** para converter n.

Exemplo 3

Este exemplo utiliza **Dialog...EndDlog** para criar caixas de diálogo para entrada e saída de dados. Utiliza **Loop...EndLoop** para calcular o resultado.

```
  :prog3()
  :Prgm
❶ :Dialog
⋮ :  Title "Enter an integer"
⋮ :  Request "Integer",n
❶ :EndDlog
❷ :expr(n)→n
  :0→temp:0→I
❸ :Loop
⋮ :  temp+i→temp
⋮ :  i+1→I
⋮ :  If i>n
⋮ :    Exit
❸ :EndLoop
❹ :Dialog
⋮ :  Title "The answer is"
⋮ :  Text string(temp)
❹ :EndDlog
  :EndPrgm
```

❶ Define uma caixa de diálogo para a entrada de dados.
❷ Converte a cadeia de caracteres introduzida com **Request** numa expressão.
❸ Cálculo no loop.
❹ Define uma caixa de diálogo para a saída.

Exemplo 4

Este exemplo utiliza as funções integradas da TI-89 Titanium / Voyage™ 200 para calcular o resultado sem utilizar o loop.

```
  :prog4()
  :Prgm
❶ :Input "Enter an integer",n
❷ :sum(seq(i,i,1,n))→temp
❸ :Disp temp
  :EndPrgm
```

❶ Pede dados de entrada no ecrã Program I/O.
❷ Calcula a soma.
❸ Mostra a saída no ecrã Program I/O.

Nota: como **Input** devolve n como um número, não é preciso utilizar **expr** para converter n.

Função	Utilizada neste exemplo para:
seq	Gerar a sequência de números inteiros de **1** a **n**. **seq(***expressão, var, baixo, alto [,passo]***)** ❶ ❷ ❸ ❸ ❹ ❶ variável que será incrementada ❷ expressão utilizada para gerar a sequência ❸ valores inicial e fim de *var* ❹ incremento para *var* ; se omitido, utiliza 1

Função	Utilizada neste exemplo para:
sum	Soma os inteiros na lista gerados por seq.

Programas em linguagem Assembly

É possível executar programas escritos para a TI-89 Titanium / Voyage™ 200 em linguagem assembly. Os programas nessa linguagem geralmente têm um processamento muito mais rápido e permitem um maior controlo do que os programas de sequências de teclas que escreve com o Program Editor interno.

Onde obter programas em linguagem Assembly

Os programas em linguagem assembly, assim como programas de batimentos de teclas, estão disponíveis no site Web da Texas Instruments em education.ti.com.

Os programas disponíveis neste site fornecem funções ou características adicionais que não estejam integradas na TI-89 Titanium ou Voyage™ 200. Verifique o site Web da Texas Instruments para obter informações actualizadas.

Depois de transferir um programa da Web para o computador, utilize um cabo USB cable ou um cabo TI-GRAPH LINK™ e o software TI Connect para enviar o programa para a TI-89 Titanium / Voyage™ 200.

Para mais informações sobre a instalação de Flash Apps, consulte education.ti.com/guides.

Nota sobre TI-GRAPH LINK

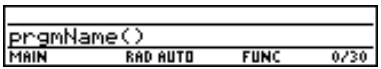
Se tiver um cabo TI-GRAPH LINK™ e o software da TI-89 ou TI-92 Plus, não se esqueça de o software TI-GRAPH LINK não é compatível com a TI-89 Titanium ou a Voyage™ 200. No entanto, o cabo funciona com todas as unidades. Utilize o software TI Connect no computador.

Pode adquirir cabos da computador para unidade e de unidade para unidade na loja TI em education.ti.com/buy.

Execução de um programa em linguagem Assembly

Depois de guardar um programa em linguagem assembly para a TI-89 Titanium / Voyage™ 200 na sua unidade, pode executar o programa a partir do ecrã principal da mesma forma como ocorre com qualquer outro programa.

- Se o programa exigir um ou mais argumentos, digite-os dentro dos ( ). Consulte a documentação do programa para obter informaçöes sobre os argumentos obrigatórios.



- Se o programa não estiver na pasta actual, não se esqueça de especificar o nome do caminho.

Pode chamar um programa em linguagem assembly a partir de um outro programa como subrotina, eliminá-lo ou utilizá-lo da mesma forma como qualquer outro programa.

Atalhos para executar um programa

Pode utilizar atalhos do teclado no ecrã principal para executar até seis programas definidos pelo utilizador ou programas em linguagem assembly. No entanto, os programas devem ter os seguintes nomes.

No ecrã principal, prima:	Para executar um programa, se existir:
◆ 1	kbdprgm1( )
⋮	⋮
◆ 6	kbdprgm6( )

Os programas devem estar guardados na pasta `MAIN`. Além disso, não é possível utilizar um atalho para executar um programa que exija um argumento.

Se tiver um programa com um nome diferente e o quiser executar através de um atalho no teclado, copie ou renomeie o programa existente para **kbdprgm1( )**, etc.

Não pode editar um programa em linguagem Assembly

Não pode utilizar a TI-89 Titanium / Voyage™ 200 para editar um programa em linguagem assembly. O Program Editor interno não abrirá programas em linguagem assembly.

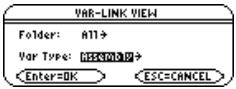
Apresentação de uma lista de programas em linguagem Assembly

Para listar os programas em linguagem assembly guardados na memória:

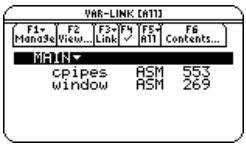
1. Veja o ecrã **VAR-LINK** (2nd [VAR-LINK]).

2. Prima [F2] **View**.

3. Seleccione a pasta em questão (ou All) e configure **Var Type = Assembly**.



4. Prima [ENTER] para ver a lista de programas em linguagem assembly.



Nota: o tipo de dados dos programas em linguagem Assembly é `ASM`.

Para mais informações sobre como escrever um programa em linguagem Assembly

As informações necessárias para ensinar um programador principiante a escrever programas em linguagem assembly estão fora do âmbito deste livro. No entanto, se tiver alguma experiência com a linguagem assembly, consulte o site Web da TI (education.ti.com) para obter informações específicas sobre como aceder aos recursos da TI-89 Titanium / Voyage™ 200.

A TI-89 Titanium / Voyage™ 200 inclui também o comando **Exec** que executa uma cadeia composta por uma série de op-codes do 68000 da Motorola. Esses códigos funcionam como uma outra forma de programa em linguagem assembly. Consulte as informações disponíveis no site Web da TI.

Nota: é necessário utilizar um computador para escrever programas em linguagem Assembly. Não pode criar programas em linguagem Assembly a partir do teclado da calculadora.

Advertência: Exec permite aceder a toda a potencialidade do microprocessador. Esteja consciente de que pode facilmente cometer um erro que venha a parar a calculadora e que provoque a perda dos seus dados. Sugerimos que faça uma cópia de segurança do conteúdo da calculadora antes de tentar utilizar o comando **Exec**.

Editor de texto

Início de uma secção com o Text Editor

Quando iniciar o Text Editor, pode começar uma nova secção de texto, continuar a secção actual (a secção apresentada pela última vez que o Text Editor foi utilizado) ou abrir uma secção anterior.

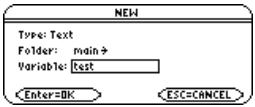
Início de nova secção

1. Prima [APPS] e, em seguida, seleccione **Text Editor**.

2. Seleccione **3:New**.

   Aparece a caixa de diálogo **NEW**.

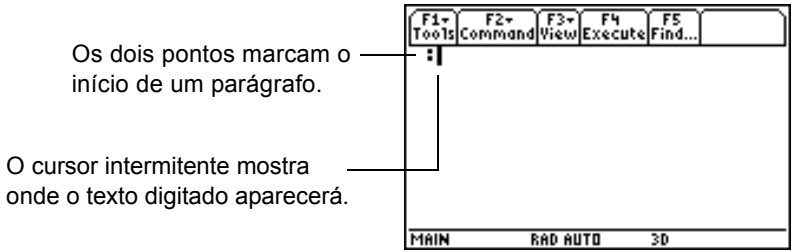
3. Especifique uma pasta e uma variável de texto que deseja utilizar para guardar a nova secção.



Item	Descrição
Type	É automaticamente definido como `Text` e não pode ser mudado.

Item	Descrição
Folder	Mostra a pasta onde a variável de texto será guardada. Para mais informações sobre pastas, consulte o módulo *Ecrã principal da calculadora*. Para utilizar uma pasta diferente, prima ⊙ para ver um menu de pastas existentes. Em seguida, seleccione uma pasta.
Variable	Digite um nome de variável. Se uma variável já existente for especificada, uma mensagem de erro aparece quando premir [ENTER]. Quando [×] [÷][7] [ESC] ou [ENTER] para reconhecer o erro, a caixa de diálogo **NEW** aparece novamente.

4. Prima [ENTER] (depois de digitar dados numa caixa de entrada, como **Variable**, tem de premir duas vezes [ENTER]) para mostrar um ecrã vazio do Text Editor.



O Text Editor poderá ser utilizado agora como descrito nas secções seguintes deste módulo.

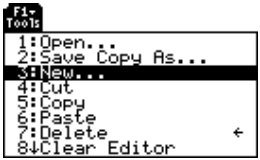
Nota: a secção é guardada automaticamente à medida que digita. Por esta razão, não é necessário guardá-la manualmente quando deixar o Text Editor, quando começar uma nova secção ou abrir uma secção anterior.

Continuação da secção actual

É possível sair do Text Editor e ir para uma outra aplicação a qualquer momento. Para voltar à secção apresentada quandio sair do Text Editor, reinicie o Text Editor e seleccione **1:Current**.

Início de uma nova secção no Text Editor

Para terminar a secção actual do Text Editor e começar uma nova:



1. Prima [F1] e seleccione **3:New**.
2. Especifique uma pasta e uma variável de texto para a nova secção.
3. Prima duas vezes [ENTER].

Abertura de uma secção anterior

Pode abrir uma secção do Text Editor a qualquer momento.

1. No Text Editor, prima [F1] e seleccione **1:Open**.

   — ou —

   Em qualquer aplicação, reinicie o Text Editor e seleccione **2:Open**.

2. Seleccione a pasta e a variável de texto apropriadas.
3. Prima [ENTER].

Nota: por predefinição, **Variable** mostra a primeira variável de texto existente em ordem alfabética.

Cópia de uma secção

Em alguns casos, pode desejar copiar uma secção para que a cópia seja editada e o original mantido.

1. Mostra a secção que deseja copiar.
2. Prima [F1] e seleccione **2:Save Copy As**.
3. Especifique a pasta e a variável de texto da secção copiada.
4. Prima duas vezes [ENTER].

Observação sobre a exclusão de uma secção

Todas as secções do Text Editor são guardadas automaticamente, por esta razão é possível acumular um número razoável de secções precedentes, que ocupam espaço na memória.

Utilize o ecrã VAR-LINK ([2nd] [VAR-LINK]) para excluir a variável de texto da secção que deseja eliminar. Para mais informações sobre VAR-LINK, consulte no módulo *Gestão da memória e das variáveis*.

Introdução e edição de texto

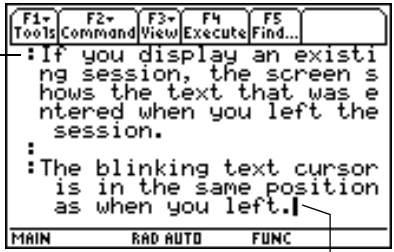
Depois de iniciar uma secção no Text Editor, pode introduzir e editar texto. De uma forma geral, podem ser utilizadas as mesmas técnicas utilizadas na introdução e na edição de informações na linha de entrada do ecrã principal.

Digitação de texto

Quando uma nova secção é criada no Text Editor, aparece um ecrã vazio. Quando abrir uma secção anterior ou voltar à secção actual, aparece o texto existente na secção.

Todos os parágrafos do texto começam com um espaço e dois pontos.

O espaço inicial é utilizado em scripts de comando e em relatórios.



Cursor de texto

Não é necessário premir [ENTER] no fim de cada linha. No fim de uma linha, o próximo carácter digitado é quebrado na linha seguinte. Só prima [ENTER] quando desejar iniciar um novo parágrafo.

Quando chegar à parte inferior do ecrã, as linhas anteriores desaparecem no topo do ecrã.

Se utilizar um cabo USB e o software TI Connect™ com a TI-89 Titanium ou um cabo de computador-unidade TI-GRAPH LINK™ e o software TI Connect™ com o Voyage™ 200 Graphing Calculator, pode utilizar o teclado do computador para digitar um ficheiro de texto e enviar esse ficheiro para a TI-89 Titanium / Voyage™ 200. Este é util se necessitar de criar um ficheiro de texto volumoso.

Para mais informações sobre como obter cabos ou software TI Connect™, visite o site Web da TI em education.ti.com, ou contacte a Texas Instruments em TI-Cares™.

Nota:

- Utilize as teclas de cursor para percorrer uma secção ou para posicionar o cursor de texto.
- Prima [2nd] ⊕ ou [2nd] ⊖ para deslocar um ecrã para cima ou para baixo de cada vez e [◆] ⊕ ou [◆] ⊖ para ir para o topo ou base da secção de texto.

Digitação de caracteres alfabéticos

Para:	Na TI-89, prima:	Na Voyage™ 200, prima:
Digitar um único carácter alfabético minúsculo.	[alpha] e a tecla da letra (a linha de estado mostra ◢)	a tecla da letra
Digitar um único carácter alfabético maiúsculo.	[↑] e a tecla da letra (a linha de estado mostra ⬆)	[↑] e a tecla da letra (a linha de estado mostra ⬆)

Para:	Na TI-89, prima:	Na Voyage™ 200, prima:
Digitar um espaço.	[alpha] [␣] (função alfabética da tecla [(-)])	barra de espaço
Activar o alpha-lock de minúsculas.	[2nd] [a-lock] (a linha de estado mostra ▬a▬)	(nenhuma ação necessária)
Activar o ALPHA-lock de maiúsculas.	[↑] [a-lock] (a linha de estado mostra ▬A▬)	[2nd] [CAPS]
Desactivar qualquer alpha-lock.	[alpha] (desactiva o bloqueio de maiúsculas e minúsculas)	[2nd] [CAPS] (desactiva o bloqueio de maiúsculas)

Nota: na TI-89, não precisa de [alpha] ou de alpha-lock para digitar x, y, z ou t. Mas tem de utilizar [↑] ou ALPHA-lock de maiúsculas para X, Y, Z ou T. Na TI-89, alpha-lock fica sempre desactivado quando a aplicação é modificada, como por exemplo, ir do Text Editor para o ecrã principal.

Na TI-89, enquanto houver um dos tipos de alpha-lock activado:

- Para digitar um ponto, vírgula ou outro carácter que seja a função primária de uma tecla, é necessário desactivar o alpha-lock.
- Para digitar um carácter de segunda função como [2nd] [{], não é necessário desactivar o alpha-lock. Depois de digitar o carácter, alpha-lock permanece activo.

Exclusão de caracteres

Para excluir:	Prima:
O carácter à esquerda do cursor	[←] ou [F1] **7**
O carácter à direita do cursor	[◆] [DEL] (é o mesmo que [◆] [←])
Todos os caracteres à direita do cursor até o fim do parágrafo	[CLEAR]
Todos os caracteres do parágrafo (independente da posição do cursor no parágrafo)	[CLEAR] [CLEAR]

Nota: se não houver caracteres à direita do cursor, [CLEAR] exclui todo o parágrafo.

Realce do texto

Para:	Operação:
parar o texto	Mova o cursor para o início ou o fim do texto Prima sem soltar [↑] prima: • ⊖ ou ⊕ para realçar os caracteres à esquerda ou à direita do cursor. • ⊖ ou ⊕ para realçar todos os caracteres até a posição do cursor na próxima linha ou na linha anterior.

Nota: para retirar o texto realçado sem substituir ou excluir, mova o cursor.

Substituição ou exclusão do texto realçado

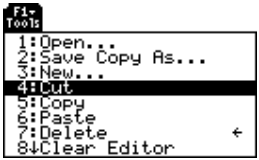
Para:	Operação:
Substituir o texto realçado	Digite o novo texto.
Excluir o texto realçado	Prima [←].

Operação de corte, cópia e colagem de texto

As operações de corte e cópia colocam o texto realçado na área de transferência da TI-89 / Voyage™ 200. A operação de corte remove o texto da sua posição actual (permite mover o texto) e a operação de cópia mantém o texto original.

1. Realce o texto que deseja mover ou copiar.

2. Prima [F1].



3. Seleccione o item apropriado do menu.
   - Para mover o texto, seleccione **4:Cut**.
     — ou—
   - Para copiar o texto, seleccione **5:Copy**.

   Nota: pode premir:
   [◆] [CUT], [◆] [COPY], [◆] [PASTE]
   [◆] **X**, [◆] **C**, [◆] **V**
   para cortar, copiar e colar sem utilizar o menu [F1] da barra de ferramentas.

4. Mova o cursor de texto para a posição onde deseja inserir o texto.
5. Prima [F1] e, em seguida, seleccione **6:Paste**.

Este procedimento permite cortar, copiar e colar texto:

- Dentro da mesma secção de texto.
- De uma secção de texto para outra. Depois de cortar ou copiar texto para uma secção, abra a outra secção e, em seguida, cole o texto.

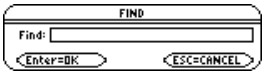
- De uma secção de texto para uma aplicação diferente. Por exemplo, pode colar texto na linha de entrada do ecrã principal.

Localização de texto

No Text Editor:

1. Posicione o cursor de texto em qualquer local que preceda o texto que deseja localizar. Todas as operações de localização começam na posição actual do cursor.

2. Prima [F5].



3. Digite o texto a ser localizado.

   A pesquisa não é sensível a maiúsculas. Por exemplo: CASO, caso ou Caso produzem o mesmo efeito.

   Nota: a caixa de diálogo **FIND** retém o último texto pesquisado introduzido. É possível digitar sobre ele ou editá-lo.

4. Prima duas vezes [ENTER].

Se o texto pesquisado:	O cursor:
É encontrado	Vai para o início do texto localizado.
Não é encontrado	Não se move.

Inserção ou sobreposição de caracteres

Por predefinição, a TI-89 / Voyage™ 200 está no modo de inserção. Para alternar entre os modos de inserção e sobreposição, prima [2nd] [INS].

Se a TI-89 / Voyage™ 200 está no:	O próximo carácter que for digitado:
Insert mode O cursor fica fino entre os caracteres	Será inserido na posição do cursor.
Overtype mode O cursor realça o carácter	Substituirá o carácter realçado.

Nota: observe a forma do cursor para ver que modo está a ser utilizado, inserção ou sobreposição.

Limpeza do conteúdo do Text Editor

Para excluir todos os parágrafos existentes e mostrar um ecrã de texto vazio, prima [F1] e, em seguida, seleccione **8:Clear Editor**.

Introdução de caracteres especiais

Pode utilizar o menu CHAR para seleccionar qualquer carácter especial numa lista. É possível também digitar alguns caracteres no teclado. Para ver os caracteres disponíveis no teclado, é possível ver um mapa com os caracteres e suas respectivas teclas.

Selecção de caracteres no menu CHAR

1. Prima [2nd] [CHAR].
2. Seleccione a categoria desejada.

   Um menu lista os caracteres daquela categoria.
3. Seleccione um carácter. Pode ser necessário percorrer o menu.

   Nota: para caracteres acentuados, seleccione International. Os caracteres internacionais frequentemente utilizados estão disponíveis também a partir do menu personalizado ([2nd] [CUSTOM]).

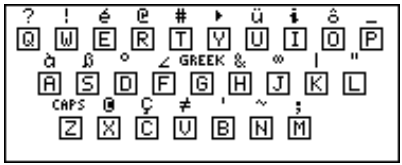
↓ indica que é possível fazer o texto rolar.

Apresentação do mapa do teclado

O mapa do teclado mostra os atalhos que permitem a introdução de determinados caracteres especiais e letras gregas no teclado. Mostra também atalhos para outros recursos da calculadora.

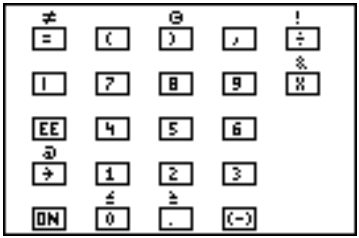
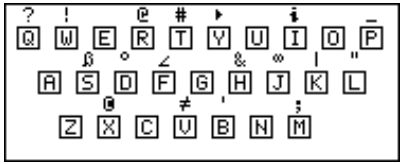
O mapa do teclado não mostra todos os atalhos disponíveis. Consulte a parte de dentro da capa e da contracapa deste manual para obter uma lista completa das teclas de atalho.

Para aceder aos atalhos, prima a tecla [2nd]. Alguns caracteres especiais estão marcados no teclado, mas a maioria não está..

Na TI-89:	**Na Voyage™ 200:**
Prima [◆] [EE] para ver o mapa do teclado. Prima [ESC] para sair do mapa.	Prima [◆] [KEY] para ver o mapa do teclado. Prima [ESC] para sair do mapa.
TI-89 Mapa do teclado	 Voyage™ 200 Mapa do teclado
Para aceder aos atalhos da TI-89, prima primeiro a tecla [◆].	Para aceder aos atalhos da Voyage™ 200, prima primeiro a tecla [2nd]. Alguns caracteres especiais estão marcados no teclado, mas a maioria não.

Atalho dos recursos do mapa de teclado da TI-89:	Atalho dos recursos do mapa de teclado da Voyage™ 200:
GREEK (◆ (.) — Acede ao conjunto de caracteres grecos (descritos posteriormente nesta secção). SYSDATA (◆ ,) — Copia as coordenadas do gráfico actual para a variável de sistema sysdata. FMT (◆ |) — Mostra a caixa de diálogo FORMATS. KBDPRGM1 – 9 (◆ 1 a ◆ 9) — Se tiver programas definidos pelo utilizador, ou em linguagem assembly, de nomes kbdprgm1() a kbdprgm9(), estes atalhos executam o programa correspondente. OFF (◆ [OFF]) — Semelhante a 2nd [OFF] exceto: • Pode utilizar ◆ [OFF] se aparecer uma messagem de erro. • Desliga a TI-89 de forma que volta à aplicação actual da próxima vez que for ligada. HOMEDATA (◆ (-)) — Copia as coordenadas do gráfico actual para a área do histórico do ecrãa principal.	GREEK (2nd **G**) — Acede ao conjunto de caracteres grecos (descritos posteriormente nesta secção). CAPS (2nd [CAPS])— Activa e desactiva Caps Lock. Accent marks — (é, ü, ô, à, ç, e ~) serão adicionados na próxima letra premida (descritos posteriormente nesta secção).

Digitação de símbolos especiais no teclado

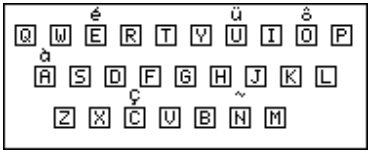
Na TI-89:	Na Voyage™ 200:
Prima [◆] e, em seguida, a tecla do símbolo. Por exemplo: [◆] [×] (vezes) aparece &.	Prima [2nd] e, em seguida, a tecla do símbolo. Por exemplo: [2nd] H aparece &.
	
Estes símbolos especiais não são afectados pela condição de alpha-lock (activado ou desactivado).	Estes símbolos especiais não são afectados pela condição de caps lock (activado ou desactivado).

Nota: para ajudar a encontrar as teclas desejadas, este mapa mostra apenas símbolos especiais.

Digitação de caracteres acentuados do teclado Voyage™ 200

Premir uma tecla de acento não mostra a letra digitada acentuada. O acento será adicionado na próxima letra premida.

1. Prima 2nd e, em seguida, a tecla do acento desejado.



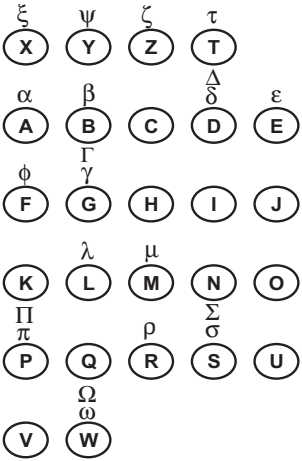
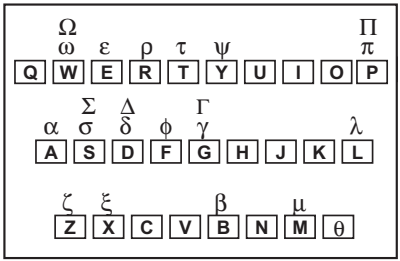
Nota: para ajudar a encontrar as teclas desejadas, este mapa mostra apenas as teclas acentuadas, cedilha e til.

2. Prima a tecla da letra que deseja acentuar.
   - É possível acentuar letras minúsculas e maiúsculas.
   - Um acento só pode ser adicionado às letras que são válidas com aquele acento.

Acento	Letras válidas (minúscula ou maiúscula)	Exemplos
´	A, E, I, O, U, Y	é, É
¨	A, E, I, O, U, y (but not Y)	ü, Ü
ˆ	A, E, I, O, U	ô, Ô
`	A, E, I, O, U	à, À
ç	C	ç, Ç
~	A, O, N	ñ, Ñ

Digitação de letras gregas no teclado

Pressione uma combinação de teclas para aceder ao conjunto de caracteres grego. Em seguida, seleccione o carácter no teclado para introduzir uma letra grega.

Na TI-89:	Na Voyage™ 200:
Prima [◆] [(] para aceder ao conjunto de caracteres grego.	Prima [2nd] **G** para aceder ao conjunto de caracteres grego.
	

Nota: quando pressionar uma combinação de teclas que não acedem a uma letra grega, obterá a letra normal dessa tecla. Nenhuma calculadora mostra um mapa das letras gregas. Os mapas mostrados aqui são utilizados apenas para referência.

Várias teclas permitem aceder às letras gregas maiúsculas e minúsculas. Por exemplo:

Na TI-89:	Na Voyage™ 200:
Prima [◆] [(] para aceder o conjunto de caracteres grego. Prima [◆] [(] [alpha] + letter para aceder às letras gregas minúsculas. Exemplo: [◆] [(] [alpha] [W] aparece ω Prima [◆] [(] [↑] + letter para aceder às letras gregas maiúsculas. Exemplo: [◆] [(] [↑] [W] aparece Ω	Prima [2nd] **G** para aceder ao conjunto de caracteres grego. Prima [2nd] **G** + letter para aceder às letras gregas minúsculas. Exemplo: [2nd] **G W** aparece ω Prima [2nd] **G** [↑] + letter para aceder às letras gregas maiúsculas. Exemplo: [2nd] **G** [↑] **W** aparece Ω

As teclas exactas premidas na TI-89 dependem se alpha-lock estiver activado ou desactivado. Por exemplo:

Na TI-89, se:	Então:
Alpha-lock estiver desactivado.	[◆] [(] **X** ou [◆] [(] [alpha] **X** aparece ξ. ([alpha] não é obrigatório para X, Y, Z ou T.) [◆] [(] [alpha] **W** aparece ω. [◆] [(] [↑] **W** aparece Ω. ([↑] é utilizado para as letras maiúsculas.)
Alpha-lock de minúsculas ([2nd] [a-lock]) está activado.	[◆] [(] **X** aparece ξ. [◆] [(] **W** aparece ω. [◆] [(] [↑] **W** aparece Ω.
ALPHA-LOCK de maiúsculas ([↑] [a-lock]) está activada.	[◆] [(] **X** aparece ξ. [◆] [(] **W** aparece Ω. [◆] [(] [↑] **W** aparece Ω.

Importante: quando premir [alpha] na TI-89 para aceder a uma letra grega enquanto alpha-lock estiver activado, alpha-lock será desactivado.

Lista com todos os caracteres especiais

Para aceder a uma lista com todos os caracteres especiais, consulte o módulo *Referência técnica*.

Introdução e execução de um script de comandos

Um script de comandos permite utilizar o Text Editor para digitar uma série de linhas de comandos que podem ser executadas a qualquer momento no ecrã principal. Este procedimento permite criar scripts interactivos, onde vários comandos são predefinidos numa série e executados individualmente.

Inserção de uma marca de comando

No Text Editor:

1. Posicione o cursor na linha de comando.

2. Prima F2 para ver o menu Command da barra de ferramentas.

3. Seleccione **1:Command**.

   C aparece no início da linha de texto (à esquerda dos dois pontos).

   Nota: esta operação não insere uma nova linha de comando, simplesmente marca uma linha existente como sendo uma linha de comando.

4. Digite um comando exactamente como faria no ecrã principal.

   A linha só pode conter o comando sem texto adicional.

   Nota: pode marcar uma linha como sendo um comando antes ou depois de digitar os comandos na linha.

Pode digitar vários comandos numa linha, mas devem estar separados por dois pontos.

Exclusão de uma marca de comando

Para excluir apenas a marca **C**; mantendo o texto do comando.

1. Posicione o cursor num ponto qualquer da linha marcada.

2. Prima F2 e seleccione **4:Clear command**.

Execução de um comando

Para executar um comando, é preciso marcar a linha com **C**. Uma linha sem a marca de comando **C** não será executada.

1. Posicione o cursor num ponto qualquer da linha de comando.
2. Prima F4.

O comando é copiado para a linha de entrada no ecrã principal e executado. O ecrã principal aparece temporariamente durante a execução do comando e, em seguida, o Text Editor aparece novamente.

Depois da execução, o cursor passa para a próxima linha do script, permitindo a execução de outros comandos.

Sugestão: prima HOME; ◆ [CALC HOME] ou utilize os ecrãs divididos para acompanhar o resultado no ecrã principal.

Divisão dos ecrãs Text Editor e principal

Os ecrãs divididos permitem ver o seu script de comandos e, ao mesmo tempo, acompanhar o resultado de um comando executado.

Para:	Prima:
Dividir o ecrã	F3 e seleccione **1:Script view**.

Para:	Prima:
Voltar a usar o Text Editor em tamanho máximo	[F3] e seleccione **2:Clear split**.

É possível utilizar [MODE] para definir os ecrãs divididos manualmente. Entretanto, [F3] define o par Text Editor/ecrã principal muito mais facilmente do que [MODE].

- A aplicação activa está indicada por uma borda espessa. (Por predefinição, o Text Editor é a aplicação activa.)
- Para alternar entre o Text Editor e o ecrã principal, prima [2nd] [⊞] (segunda função de [APPS]).

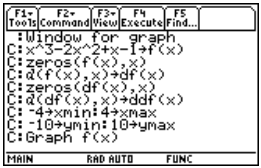
Criação de um script a partir de dados digitados no ecrã principal

No ecrã principal, é possível guardar todas as entradas da área do histórico numa variável de texto, que são guardadas automaticamente no formato de script. Desta forma, a variável de texto pode ser aberta no Text Editor e os dados digitados podem ser executados como comandos.

Para mais informações, consulte “Armazenamento das entradas no ecrã principal como script no Text Editor” no módulo *Ecrã principal da calculadora*.

Exemplo

1. Digite o seu script. Prima [F2] e seleccione **1:Command** para marcar as linhas de comando.
2. Prima [F3] e seleccione **1:Script view**.

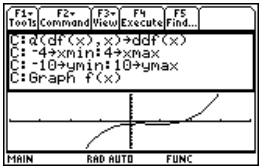


3. Mova o cursor para a primeira linha de comando. Em seguida, prima [F4] para executar o comando.

   Nota: alguns comandos demoram mais para serem executados. Aguarde até que o indicador Busy desapareça antes de premir [F4] novamente.

4. Continue a utilizar [F4] para executar cada comando e pare imediatamente antes do comando Graph.
5. Execute o comando Graph.

   Nota: neste exemplo, o comando Graph mostra o ecrã Graph no lugar do ecrã principal.



6. Prima [F3] e seleccione **2:Clear split** para voltar ao ecrã dividido do Text Editor.

Solucionador numérico

Visualização do solucionador e introdução a uma equação

Depois de ver o solucionador numérico, comece a introduzir a equação que deseja calcular.

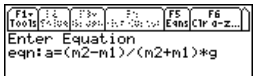
Visualização do solucionador numérico

Para ver o solucionador numérico, prima APPS.

O ecrã do solucionador numérico mostra a última equação introduzida, se houver.

Introdução de uma equação

Na linha **eqn:**, digite a sua equação.



Pode:	Por exemplo:
Digitar uma equação directamente.	**a=(m2−m1)/(m2+m1)∗g** **a+b=c+sin(d)**

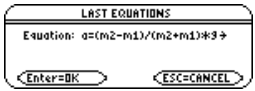
Pode:	Por exemplo:
Referir-se a uma função ou equação definida em qualquer lugar. **Nota:** • Não utilize nomes de função do sistema (como **y1(x)** ou **r1(θ)**) como variáveis simples (**y1** ou **r1**). • Tome cuidado com a multiplicação implícita. Por exemplo, **a(m2+m1)** é tratada como referência a uma função, não como **a∗(m2+m1)**.	Suponha que **y1(x)** tenha sido definida num dos seguintes locais: • Y= Editor: **y1(x)=1.25x∗cos(x)** – ou – • Ecrã principal: **Define y1(x)=1.25x∗cos(x)** No solucionador numérico, seria introduzido: **y1(x)=0** ou **y1(t)=0**, O argumento não tem que coincidir com o argumento utilizado na etc.
Digitar uma expressão sem um sinal =. **Nota:** quando as variáveis são definidas, pode definir **exp** ou calculá-la.	**e+f−ln(g)** Depois de premir [ENTER], a expressão é definida igual a uma variável do sistema chamada exp e introduzida como: exp=e+f−ln(g)
Recuperar uma equação introduzida previamente ou abra uma equação guardada. **Nota:** depois de premir [ENTER], a equação actual é guardada automaticamente na variável do sistema **eqn**.	Consulte o trecho correspondente mais adiante nesta secção.

Recuperação de equações introduzidas anteriormente

As equações introduzidas recentemente (até 11 com a predefinição) são mantidas na memória. Para recuperar uma destas equações:

1. No ecrã do solucionador numérico, prima [F5].

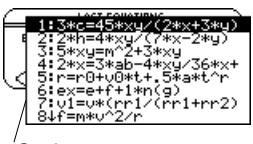
   Uma caixa de diálogo mostra a equação introduzida mais recentemente.



2. Seleccione uma equação.

   - Para seleccionar a equação apresentada, prima [ENTER].
   - Para seleccionar uma equação diferente, prima ⊙ para ver uma lista. Em seguida, seleccione a equação desejada.

   Nota: é possível especificar quantas equações são mantidas. No solucionador numérico, prima [F1] e seleccione **9:Format** (ou utilize [calc] [◆] [I]; [keyboard] [◆] **F**). Em seguida, seleccione um número de 1 a 11.



Cada equação aparece só uma vez na lista. Se a mesma equação for re-introduzida 5 vezes, aparecerá apenas uma vez.

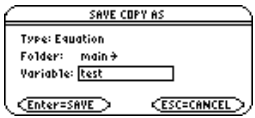
3. Prima [ENTER].

Armazenamento de equações para utilização futura

Como o número de equações que pode ser recuperado com [F5] **Eqns** é limitado, uma equação particular pode não ser mantida indefinidamente.

Para guardar a equação actual para utilização futura, guarde-a numa variável.

1. No ecrã do solucionador numérico, prima [F1] e seleccione **2:Save Copy As**.



2. Especifique uma pasta e um nome de variável para a equação.
3. Prima [)] [⇥][e^x] [÷][5][÷][e^x] [ENTER].

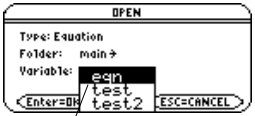
Nota: uma variável de equação tem um tipo de dados EXPR como mostrado nos ecrãs MEMORY e VAR-LINK.

Abertura de uma equação guardada

Para abrir uma variável de equação previamente guardada:

1. No ecrã do solucionador numérico, prima [F1] e seleccione **1:Open**.

2. Seleccione a pasta apropriada e a variável da equação.
3. Prima [ENTER].



A variável eqn contém a equação actual; sempre aparece por ordem alfabética na

Definição de variáveis conhecidas

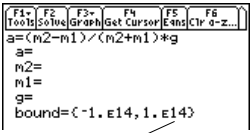
Depois de digitar uma equação no solucionador numérico, introduza os valores apropriados para todas as variáveis, excepto para a incógnita.

Definição da lista de variáveis

Depois de digitar a equação na linha **eqn:**, prima [ENTER] ou ⊖.

O ecrã lista as variáveis pela ordem em que aparecem na equação. Se uma variável já estiver definida, aparece o valor. Pode editar estes valores de variável.

Nota: se uma variável existente estiver bloqueada ou arquivada, não é possível editar o seu valor.

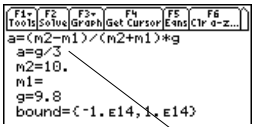


A solução precisa estar dentro dos limites especificados, que podem ser editados.

Para todas as variáveis, excepto para a que se deseja encontrar o valor, introduza um número ou expressão.

Observações e erros comuns

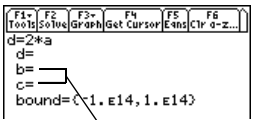
- Se definir uma variável:
  - Em termos de uma outra variável na equação, tal variável precisa ser definida primeiro.
  - Em termos de uma outra variável que não está na equação, tal variável precisa ter um valor; não podendo estar indefinida.
  - Como uma expressão, é calculada quando se move o cursor para fora da linha. O cálculo da expressão precisa resultar num número real



Como a está definido em termos de g, é preciso definir g antes de a. Quando o cursor é movido para uma outra linha, g/3 é calculado.

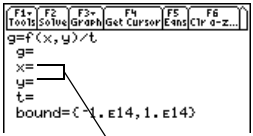
- Se a equação contiver uma variável definida em termos de outras variáveis, tais variáveis são listadas.

  Nota: quando atribuir um valor a uma variável no solucionador numérico, a variável é definida globalmente. Continua a existir depois de sair do solucionador.



Se a variável a foi definida previamente como b+c→a, b e c são listadas em vez de a.

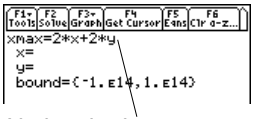
Caso seja feita referência a uma função definida anteriormente, todas as variáveis utilizadas como argumentos na chamada de função são listadas, não as variáveis utilizadas para definir a função.



Se f(a,b) foi definido previamente como √(a^2+b^2) e a equação contém

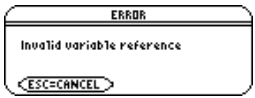
- Se uma equação contiver uma variável do sistema (**xmin**, **xmax**, etc.), esta variável não é listada. O solucionador utiliza o valor existente da variável do sistema.

  Nota: não é possível calcular o valor de uma variável do sistema que não seja exp. Além disso, se a equação contiver uma variável do sistema, não é possível utilizar [F3] para a representar graficamente.



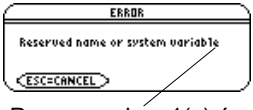
Na janela de visualização, xmax=10.

- Apesar de poder utilizar uma variável do sistema na equação, um erro ocorre se [F3] for utilizado para representar a solução graficamente.



- Se o erro mostrado à direita for visto, exclua o valor da variável introduzido. Em seguida, edite a equação para utilizar uma variável diferente.

  Nota: Este erro ocorre, se utilizar um nome reservado incorrectamente ou for feita uma referência a uma função do sistema indefinida como uma variável simples sem parêntesis.



Por exemplo, y1(x) é indefinido e y1 é utilizado.

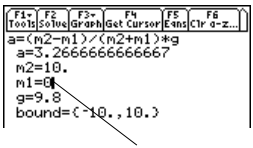
Edição da equação

No solucionador numérico, prima ⊕ até que o cursor esteja na equação. O ecrã muda automaticamente para mostrar apenas a linha **eqn:**. Faça as suas mudanças e, em seguida, prima [ENTER] ou ⊖ para voltar para a lista de variáveis.

Especificação de um valor suposto inicial e/ou limites (opcional)

Para encontrar uma solução mais rapidamente ou encontrar uma solução particular (se várias soluções existirem), pode opcionalmente:

- Introduzir um valor suposto inicial para a incógnita. O valor inicial precisa estar dentro dos limites especificados.
- Introduzir os limites inferior e superior próximos da solução.



Em termos de limites, também pode introduzir variáveis ou expressões que são calculadas com os valores apropriados (**bound={lower,upper}**) ou uma variável de lista válida que contenha uma lista de dois elementos (**bound=list**). Os limites devem ser dois elementos de ponto flutuante, sendo o primeiro deles inferior ou igual ao segundo.

Nota: seleccione um valor suposto inicial.

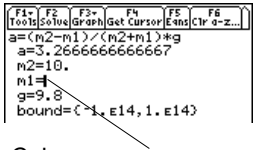
Cálculo de incógnita

Depois de digitar uma equação no solucionador numérico e introduzir valores para as incógnitas, pode realizar o cálculo da incógnita.

Cálculo de uma solução

Com todas as variáveis conhecidas definidas:

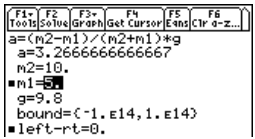
1. Mova o cursor para a variável incógnita.



Coloque o cursor na variável que deseja calcular.

2. Prima F2 **Solve**.

   Um ▪ marca a solução e **left−rt**. O ▪ desaparece quando edita um valor, move o cursor para a equação, ou deixa o solucionador.



Nota: Para interromper (parar) um cálculo, prima [ON]. A incógnita mostra o valor em teste quando a pausa ocorreu.

A solução e os valores introduzidos permitem calcular os lados esquerdo e direito da equação separadamente. O **left−rt** mostra a diferença, que indica a exactidão da solução. Quanto menor o valor, mais exacta será a solução. Se a solução for exacta, **left−rt=0**.

Se:	Operação:
Deseja encontrar outros valores	Edite os valores da variável ou da equação.
Deseja encontrar uma solução diferente para uma equação com várias soluções	Introduza um valor suposto inicial e/ou um novo conjunto de limites próximos da outra solução.
A mensagem abaixo é vista: ERROR Excessive iteration ESC=CANCEL	Prima [ESC]. A variável incógnita mostra o valor em teste quando o erro ocorreu. • O valor **left−rt** pode ser pequeno o suficiente para aceitar o resultado. • Caso contrário, introduza um conjunto diferente de limites.

Nota: Um processo iterativo é utilizado para calcular uma equação. Se o processo iterativo não convergir numa solução, ocorre este erro.

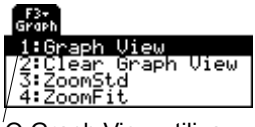
Representação gráfica da solução

É possível representar a solução de uma equação graficamente depois de definir as variáveis conhecidas, antes ou depois de encontrar o valor da incógnita. A representação gráfica de soluções permite ver quantas soluções existem e utilizar o cursor para seleccionar limites e um valor suposto inicial precisos.

Visualização do gráfico

No solucionador numérico, deixe o cursor sobre a incógnita. Prima [F3] e seleccione:

1:Graph View
– ou –
3:ZoomStd
– ou –
4:ZoomFit



O Graph View utiliza os valores actuais da variável Window actual.

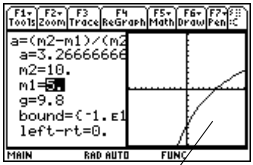
Para mais informações sobre

O gráfico aparece num ecrã dividido, onde:

- A incógnita é traçada no eixo x.
- **left−rt** é traçado no eixo y.

As soluções para a equação existem em **left−rt=0**, onde o gráfico cruza o eixo x.

Nota: Para mais informações, consulte o módulo *Ecrãs divididos*.



As definições actuais de formato de gráfico são utilizadas.

É possível explorar o gráfico com o cursor de movimento livre, traçado, ampliação, etc., como descrito no módulo *Representação gráfica de funções básicas*.

Como o gráfico afecta várias definições

Quando utilizar o solucionador numérico para ver um gráfico:

- Os seguintes modos são mudados automaticamente para estas definições:

Modo	Definição
Graph	FUNCTION Todas as funções seleccionadas no Y= Editor não serão representadas graficamente.
Split Screen	LEFT-RIGHT
Number of Graphs	1

Nota: se utilizar previamente definições de modo diferentes, seria necessário seleccionar novamente estas definições manualmente.

- Todos os gráficos estatísticos são desmarcados.
- Depois de sair do solucionador numérico, o ecrã Graph pode continuar a mostrar a solução da equação, ignorando quaisquer funções seleccionadas em Y=. Neste caso, veja Y= Editor e, em seguida, volte para ao ecrã Graph. Além disso, o gráfico é redefinido quando muda o modo Graph ou usa **ClrGraph** no ecrã principal (F4 **5**) ou num programa.

Selecção de um novo valor suposto inicial a partir de Graph

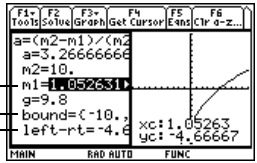
Para utilizar o cursor gráfico para seleccionar um valor suposto inicial:

1. Mova o cursor (o de movimento livre ou de traçado) para o ponto que deseja utilizar como o novo valor suposto.

2. Utilize [2nd] [⊞] para activar o ecrã do solucionador numérico.
3. Certifique-se de que o cursor está na incógnita, e prima [F4].

   Nota: a coordenada do cursor xc é o valor da variável incógnita, e yc é o valor de **left−rt**.

4. Prima [F2] para resolver a equação.



[F4] define o valor xc do cursor gráfico como um valor suposto inicial e o valor yc como left−rt.

Deslocação para um ecrã completo

No ecrã dividido:

- Para ver o ecrã completo do solucionador numérico, utilize [2nd] [⊞] para activar o ecrã do solucionador, prima [F3] e, em seguida, seleccione **2:Clear Graph View**.

  – ou –

- Para ver o ecrã principal, prima duas vezes [2nd] [QUIT].

Cancelamento de variáveis antes de deixar o solucionador numérico

Quando resolver uma equação, as suas variáveis continuam a existir depois de sair do solucionador numérico. Se a equação contiver variáveis de um único carácter, os seus

valores podem afectar inadvertidamente cálculos simbólicos posteriores. Antes de sair do solucionador numérico, pode desejar:

1. Premir:
   [2nd] [F6]
   [F6]
   para limpar todas as variáveis de um único carácter na pasta actual.
2. Premir [ENTER] para confirmar a acção.
3. O ecrã volta à linha **eqn:** do solucionador.

Nota: sempre que desejar apagar as variáveis de um carácter listadas no solucionador, utilize: [2nd] [F6]; [F6].

Bases numéricas

Introdução e conversão de bases numéricas

Independentemente do modo Base, é preciso sempre utilizar o prefixo apropriado quando introduzir um número binário ou hexadecimal.

Introdução de um número binário ou hexadecimal

Para introduzir um número binário, utilize a forma:

0b *Númerobinário* (por exemplo: **0b11100110**)

- *Número binário com até 32 dígitos*
- Zero, não a letra O, e a letra b

Para introduzir um número hexadecimal, utilize a forma:

0h *Númerohexadecimal* (por exemplo **0h89F2C**)

- *Número hexadecimal com até 8 dígitos*
- Zero, não a letra O, e a letra h

Nota: pode digitar o **b** ou o **h** no prefixo, assim como os caracteres hexadecimais **A – F**, em maiúsculas ou minúsculas.

Se um número for introduzido sem prefixo **0b** ou **0h**, como 11, será tratado como número decimal. Se o prefixo **0h** for omitido num número hexadecimal com **A – F** , toda ou parte da entrada será tratada como uma variável.

Conversão entre bases numéricas

Utilize o operador de conversão ▸.

integerExpression ▸ **Bin** *integerExpression* ▸ **Dec** *integerExpression* ▸ **Hex**	Para obter ▸, prima [2nd] [▸]. Além disso, é possível seleccionar conversões de base no menu MATH/Base.

Por exemplo, para converter 256 de decimal em binário:

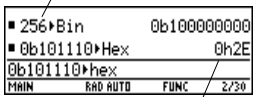
256 ▸ Bin

Nota: Se o valor não for um número inteiro, um erro Domain será exibido.

Para converter 101110 de binário em hexadecimal:

0b101110 ▸ Hex

Para uma entrada binária ou hexadecimal, é preciso utilizar o prefixo 0b ou 0h.



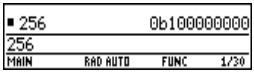
Os resultados utilizam o prefixo 0b ou 0h para identificar a base.

Método alternativo para conversões

Em vez de utilizar ▶, é possível:

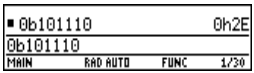
1. Utilizar [MODE] para definir o modo **Base** na base que deseja converter. (Consulte “Definição do modo Base para resultados apresentados” na página 333.)

Se o modo Base = BIN:



2. Digitar o número que deseja converter (com o prefixo correcto) e premir [ENTER] [8][9] [÷][(] [ᵇ] [×] [8][(] [×][⇥] .

Se o modo Base = HEX:

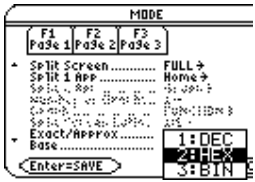


Realização de operações matemáticas com números hexadecimais ou binários

Em qualquer operação com números inteiros, pode introduzir um número hexadecimal ou binário. Os resultados aparecem de acordo com o modo Base. Entretanto, os resultados estão restritos a certos limites de tamanho quando Base = HEX ou BIN.

Definição do modo Base para os resultados apresentados

1. Prima [MODE] [F2] para ver a **página 2** no ecrã **MODE**.
2. Vá para o modo **Base**, prima ⊙, e seleccione a definição desejada.
3. Prima [ENTER] para fechar o ecrã **MODE**.

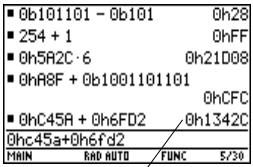


O modo **Base** controla apenas o formato apresentado dos resultados inteiros.

Nota: o modo Base afecta apenas o resultado. É preciso sempre utilizar o prefixo 0h ou 0b para introduzir um número hexadecimal ou binário.

Os resultados em forma de fracção ou de ponto flutuante são sempre mostrados em forma decimal.

Se modo Base = HEX:



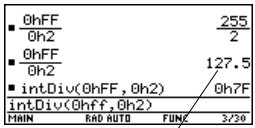
O prefixo 0h no resultado identifica a

Divisão quando Base = HEX ou BIN

Quando Base=HEX ou BIN, o resultado de uma divisão aparece na forma hexadecimal ou binária apenas se o resultado for um número inteiro.

Para assegurar que a divisão produz sempre um número inteiro, utilize **intDiv( )** em vez de ÷.

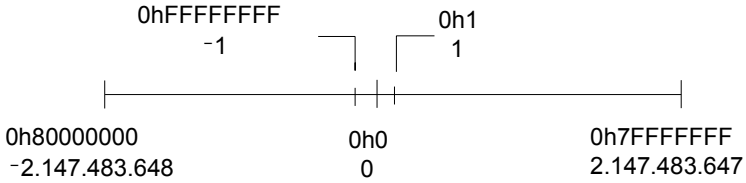
Se modo Base = HEX:



Prima ON ENTER para ver o resultado na forma APPROXIMATE.

Limitações de tamanho quando Base = HEX ou BIN

Quando Base=HEX ou BIN, um resultado inteiro é guardado internamente como um número binário de 32 bits com sinal, que utiliza o intervalo (mostrado em hexadecimal e decimal):



Se a magnitude de um resultado for muito grande para ser guardada na forma binária de 32 bits com sinal, uma operação de módulo simétrica trará o resultado para o intervalo. Qualquer número maior que 0h7FFFFFFF é afectado. Por exemplo, de 0h80000000 a 0hFFFFFFFF tornam-se números negativos.

Comparação ou manipulação de bits

Os operadores e funções seguintes permitem comparar ou manipular bits num número binário. É possível introduzir um número inteiro em qualquer base numérica. Tais entradas são convertidas automaticamente em número binário para operação de bits, e os resultados aparecem de acordo com o modo Base.

Operações booleanas

Operador com sintaxe	Descrição
not inteiro	Devolve o complemento de um, onde cada bit é invertido.
(-) *inteiro*	Devolve o complemento de dois, que é o complemento de um +1.
inteiro1 **and** *inteiro2*	Numa comparação **and** bit a bit, o resultado é 1 se os dois bits são 1; caso contrário, o resultado é 0. O valor devolvido representa o resultado da operação binária.
inteiro1 **or** *inteiro2*	Numa comparação **or** bit a bit, o resultado é 1 se algum bit é 1; o resultado é 0 apenas se os dois bits são 0. O valor devolvido representa o resultado da operação binária.
inteiro1 **xor** *inteiro2*	Numa comparação **xor** bit a bit, o resultado é 1 se algum bit (mas não ambos) é 1; o resultado é 0 se ambos os bits são 0 ou ambos os bits são 1. O valor devolvido representa o resultado da operação binária.

Nota: pode seleccionar estes operadores a partir do menu MATH/Base. Consulte o módulo *Referência técnica* neste manual para ver um exemplo da utilização dos operadores.

Suponha que introduz:

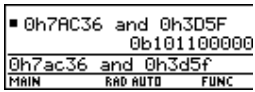
0h7AC36 **and** 0h3D5F

Internamente, os números inteiros hexadecimais são convertidos num número binário de 32 bits com sinal.

Em seguida, os bit correspondentes são comparados.

Se modo Base = HEX:

Se modo Base = BIN:



0h7AC36 = 0b00000000000000111101011000011011 0
and **and**
0h3D5F 0b00000000000000000001111010101111 1
0b00000000000000000001011000001011 0 = 0h2C16

Os zeros à esquerda não são mostrados no resultado.

Nota: Se for introduzido um número inteiro muito grande para ser guardado na forma binária de 32 bits com sinal, uma operação de módulo simétrico trará o valor para o intervalo.

O resultado aparece de acordo com o modo Base.

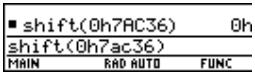
Rotação e deslocamento de bits

Função com sintaxe	Descrição
rotate(*inteiro***)** – ou – **rotate(***inteiro,nºDeRotações***)**	Se *nºDeRotações* for: • omitido — bits giram uma vez para a direita (predefinição é ⁻1). • negativo — bits giram o número especificado de vezes para a direita. • positivo — bits giram o número especificado de vezes para a esquerda. Numa rotação para a direita, o bit mais à direita ocupa o lugar do bit mais à esquerda; e vice-versa numa rotação para a esquerda.
shift(*inteiro***)** – ou – **shift(***inteiro,nºDeDeslocamentos***)**	Se *nºDeDeslocamentos* for: • omitido — os bits deslocam-se uma vez para a direita (predefinição é ⁻1). • negativo — os bits deslocam-se o número especificado de vezes para a direita. • positivo — os bits deslocam-se o número especificado de vezes para a esquerda. No deslocamento para a direita, o bit mais à direita cai e 0 ou 1 é inserido na posição do bit mais à esquerda. No deslocamento para a esquerda, o bit mais à esquerda cai e 0 é inserido na posição do bit mais à direita.

Suponha que introduz:

shift(0h7AC36**)**

Se modo Base = HEX:



Internamente, o número inteiro hexadecimal é convertido num número binário de 32 bits com sinal.

Se modo Base = BIN:

Em seguida, o deslocamento é aplicado ao número binário.

Todos os bits são deslocados para a direita.

7AC36 = 0b00000000000000111101011000011011 0

Insere 0 se o bit mais à esquerda é 0, ou 1 se o bit mais à esquerda é 1.

Caiu

b00000000000000111101011000011011 = 0h3D61B

Os zeros à esquerda não são mostrados no resultado.

O resultado aparece de acordo com o modo Base.

Nota: se for introduzido um número inteiro muito grande para ser guardado na forma binária de 32 bits com sinal, uma operação de módulo simétrico trará o valor para o intervalo.

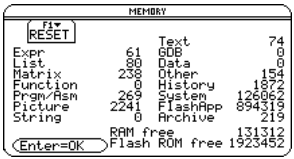
Gestão da memória e das variáveis

Controlo e reinicialização da memória

O ecrã **MEMORY** mostra a quantidade de memória (em bytes) utilizada por todas as variáveis de cada tipo de dados, independentemente dessas variáveis estarem armazenadas na RAM ou no arquivo de dados do utilizador. Este ecrã pode também ser utilizado para reinicializar a memória.

Apresentação do ecrã MEMORY

Prima [2nd] [MEM]. O ecrã seguinte refere-se a uma TI-89 Titanium / Voyage™ 200 Graphing Calculator. (Os números do ecrã **MEMORY** podem ser diferentes dos apresentados em seguida.)



Prgm/Asn: Inclui os programas escritos para a TI-89 Titanium / Voyage™ 200 além de quaisquer programas em linguagem assembly que tiverem sido carregados
History: Tamanho dos pares armazenados na área do histórico do ecrã principal.
FlashApp: Tamanho das aplicações Flash.
RAM free: Espaço livre em RAM.
Flash ROM free: Espaço livre na Flash ROM.

Nota: Para visualizar o tamanho individual das variáveis e determinar se estão no arquivo de dados do utilizador, utilize o ecrã **VAR-LINK**.

Para fechar o ecrã, prima [ENTER]. Para reinicializar a memória, efectue o procedimento seguinte.

Reinicialização da memória

No ecrã **MEMORY**:

1. Prima F1.
2. Seleccione o item desejado.

Item	Descrição
RAM	**1:All RAM:** Reinicializar a RAM apaga todos os dados e programas da RAM. **2:Default:** Reinicializa todas as variáveis de sistema e modos para suas configurações originais de fábrica. Isto não afecta as variáveis, funções ou pastas definidas pelo utilizador.
Flash ROM	**1:Archive:** Reinicializar o Arquivo apaga todos os dados e programas da Flash ROM. **2:Flash Apps:** Reinicializar as Flash Apps apaga todas as aplicações Flash da Flash ROM. **3:Both:** Reinicializar ambos apaga todos os dados, programas e aplicações Flash da Flash ROM.
All Memory	Reinicializar excluirá todos os dados, programas e aplicações Flash da RAM e da Flash ROM.

Importante: Para excluir variáveis individualmente (ao invés de todas), utilize **VAR-LINK**.

3. Depois do pedido de confirmação, prima [ENTER].

   A TI-89 Titanium / Voyage™ 200 apresenta uma mensagem após a conclusão da operação de reinicialização.

   Nota: para anular a operação de reinicialização, prima [ESC] em vez de [ENTER].

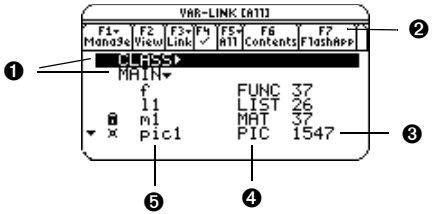
4. Prima [ENTER] para confirmar a mensagem.

Apresentação do ecrã VAR-LINK

O ecrã **VAR-LINK** contém as variáveis e as pastas definidas actualmente. Este módulo explica como manipular as variáveis e/ou as pastas apresentadas no ecrã.

Apresentação do ecrã VAR-LINK

Prima [2nd] [VAR-LINK]. Por predefinição, o ecrã **VAR-LINK** apresenta todas as variáveis definidas pelo utilizador em todas as pastas juntamente com todos os tipos de dados.



❶ Nomes das pastas (ordem alfabética)
❷ Mostra as aplicações Flash instaladas
❸ Tamanho em bytes

❹ Tipo de dados
❺ Nomes de variáveis (ordem alfabética)

Isto...	Indica isto...
▸	Visão reduzida da pasta. (à direita do nome da pasta)
▾	Visão expandida da pasta (à direita do nome da pasta).
▾	Pode percorrer para ver mais variáveis e/ou pastas (no canto inferior esquerdo do ecrã)
✓	Se seleccionado com F4.
🔒	Bloqueado
⌘	Arquivado

Para percorrer os dados da lista:

- Prima ⊖ ou ⊕. (Utilize 2nd ⊖ ou 2nd ⊕ para percorrer uma página de cada vez.)
  – ou –
- Digite um carácter. Se houver nomes de variáveis começadas por este carácter, o cursor posiciona-se no primeiro nome.

Nota: digite algumas vezes o mesmo carácter para ver os nomes que começam com tal carácter.

Tipos de variáveis indicados em VAR-LINK

Tipo	Descrição
ASM	Programa em linguagem assembly

Tipo	Descrição
DATA	Dados
EXPR	Expressão (inclui os valores numéricos)
FUNC	Função
GDB	Base de dados gráficos
LIST	Lista
MAT	Matriz
PIC	Imagem de um gráfico
PRGM	Programa
STR	Cadeia
TEXT	Secção do Editor de Texto

Os tipos não listados acima são tipos de dados variados utilizados por aplicações de software.

Fechar o ecrã VAR-LINK

Para fechar o ecrã **VAR-LINK** e voltar à aplicação actual, utilize [ENTER] ou [ESC] conforme descrito abaixo.

Prima:	Para:
[ENTER]	Colar o nome da variável ou da pasta realçada na posição do cursor na aplicação actual.
[ESC]	Voltar à aplicação actual sem colar o nome realçado.

Visualizar informações sobre variáveis no ecrã principal

No ecrã principal, é possível visualizar informações sobre variáveis sem abrir o ecrã VAR-LINK.

- Para determinar se uma variável com um determinado nome existe na tabela do sistema, introduza a função **IsVar()** no ecrã principal.

  IsVar (*var_nome*)

  IsVar é uma função, que requer que coloque o nome da variável entre

- Para determinar se uma variável está arquivada, utilize a função **IsArchiv()**.

  IsArchiv *(var_nome)*

- Para determinar se uma variável está bloqueada, utilize a função **IsLocked()**.

  IsLocked *(var_nome)*

Manipulação de variáveis e pastas com VAR-LINK

O ecrã **VAR-LINK** permite visualizar o conteúdo de uma variável. É também possível manipular um ou mais itens seleccionados com as operações descritas nesta secção.

Visualizar o conteúdo de uma variável

É possível visualizar todos os tipos de variáveis, excepto **ASM**, **DATA**, **GDB**, e as variáveis criadas pelas Apps Flash. Por exemplo, deve abrir uma variável **DATA** no **Editor de Dados/Matrizes**.

1. Na **VAR-LINK**, desloque o cursor para realçar a variável.

2. Prima:

   [2nd] [F6]

   [F6]

   Aparece o número de variáveis contidas na pasta realçada.

3. Para voltar à **VAR-LINK**, prima qualquer tecla.

Nota: não é possível editar o conteúdo deste ecrã.

Selecção de um item da lista

Para realizar outras operações, seleccione uma ou mais variáveis e/ou pastas.

Para seleccionar:	Operação:
Uma única variável ou pasta	Desloque o cursor para realçar o item e prima [F4].

Para seleccionar:	Operação:
Um grupo de variáveis ou pastas	Realce cada item e prima F4. À esquerda de cada item seleccionado, aparece o sinal ✓. (Se seleccionar uma pasta, todas as variáveis nela contidas serão seleccionadas automaticamente). Utilize F4 para seleccionar ou anular a selecção de um item.
Todas as pastas e variáveis 	Prima ⊕ para expandir a pasta e, em seguida, prima F5 **All** e seleccione **1:Select All**. Se seleccionar **3:Select Current**, selecciona o último conjunto de itens transmitidos para a unidade durante a sessão actual de **VAR-LINK**. Se seleccionar **4:Expand All** ou **5:Collapse All,** expande ou fecha as pastas ou as aplicações Flash.

Nota: prima ⊖ ou ⊕ para alternar entre expandir ou reduzir a visão quando tiver uma pasta realçada.

Pastas e variáveis

As pastas permitem gerir as variáveis, organizando-as em grupos relacionados entre si.

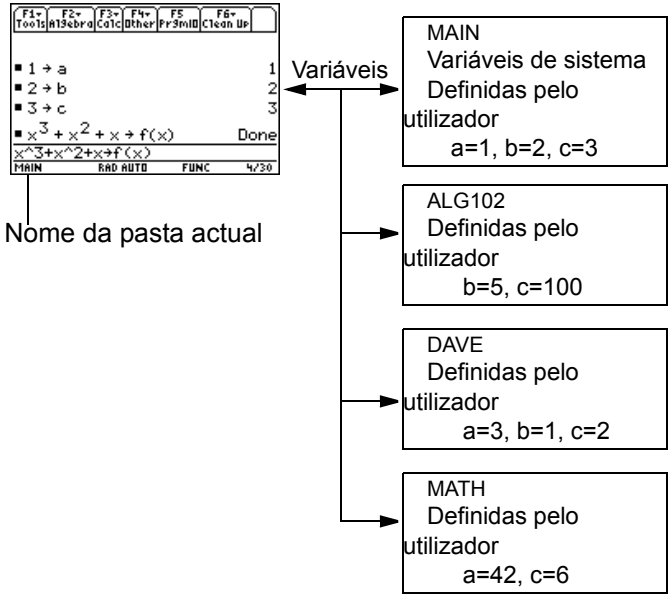
A TI-89 Titanium / Voyage™ 200 Graphing Calculator tem uma pasta **MAIN**. Por predefinição, todas as variáveis serão armazenadas na pasta **MAIN**, excepto se criar outras pastas e designar como actual uma pasta criada pelo utilizador. Pode armazenar

uma variável de sistema ou uma variável com um nome reservado apenas na pasta **MAIN**.

Exemplo de variáveis que podem ser armazenadas apenas na pasta MAIN
Variáveis Window (**xmin**, **xmax**, etc.) Variáveis de configuração das tabelas (**TblStart**, **ΔTbl**, etc.) Funções de Y= Editor (**y1(x)**, etc.)

Pela criação de outras pastas, é possível armazenar conjuntos independentes de variáveis (e funções) definidas pelo utilizador. Por exemplo: é possível criar pastas separadas para aplicações diferentes da TI-89 Titanium / Voyage™ 200 (Math, Text Editor, etc.) ou por classes. É possível armazenar uma variável definida pelo utilizador em qualquer pasta.

As variáveis definidas pelo utilizador de uma pasta são independentes das variáveis das outras pastas. Portanto, é possível armazenar conjuntos de variáveis com o mesmo nome, mas com valores diferentes.



Não é possível criar uma pasta dentro de uma outra.

É possível ter acesso directo às variáveis de sistema da pasta **MAIN**, independentemente da pasta actual.

Nota: as variáveis definidas pelo utilizador são armazenadas na “pasta actual”.

Criação de uma pasta no ecrã VAR-LINK

1. Prima [2nd] [VAR-LINK].

2. Prima [F1] **Manage** e seleccione **5:Create Folder**.

3. Digite um nome exclusivo da pasta e prima [ENTER] duas vezes.

Depois de criar uma pasta nova em **VAR-LINK**, a mesma não é configurada automaticamente como a pasta actual.

Criação de uma pasta no ecrã principal

Introduza o comando **NewFold** no ecrã principal da calculadora.

NewFold *Nome da pasta*

Nome da pasta que deseja criar. A nova pasta é automaticamente definida como a actual.

Configuração da pasta actual no ecrã principal

Introduza a função **setFold** no ecrã principal da calculadora.

setFold (*Nome da pasta*)

setFold é uma função; portanto o nome da pasta deve estar entre parênteses.

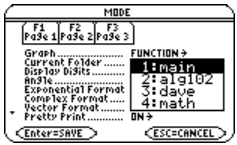
A execução de **setFold** restaura como pasta actual o nome da pasta anteriormente configurada.

Definição da pasta actual na caixa de diálogo MODE

1. Prima [MODE].

2. Realce a configuração **Current Folder**.
3. Prima [▶] para visualizar o menu com as pastas existentes.

   Nota: para cancelar o menu ou sair da caixa de diálogo sem armazenar as modificações, prima [ESC].



4. Seleccione a pasta desejada de uma das seguintes formas:
   - Realce o nome da pasta e realce [ENTER].

     – ou –
   - Prima o número ou a letra que corresponde à pasta.
5. Prima [ENTER] para armazenar as modificações e fechar a caixa de diálogo.

Atribuir nomes às variáveis ou pastas

Não se esqueça de que se [F4] for utilizada para seleccionar uma pasta, as variáveis nela contidas são seleccionadas automaticamente. Se necessário, utilize [F4] para desmarcar cada variável seleccionada.

1. Na **VAR-LINK**, seleccione as variáveis e/ou pastas.

2. Prima F1 **Manage** e seleccione **3:Rename**.

3. Digite um nome exclusivo e prima ENTER duas vezes.

   Se seleccionar vários itens, será pedido um nome novo para cada um deles.

Utilização de variáveis noutras pastas

É possível aceder a uma variável ou função definida pelo utilizador que não está contida na pasta actual. Especifique o caminho em vez de apenas o nome da variável.

Um percurso tem a seguinte forma:

Nome da pasta \ *Nome da variável*
– ou –
Nome da pasta \ *Nome da função*

Por exemplo:

Se a pasta actual = MAIN	Pastas e variáveis
■ 1 → a 1 ■ $x^3 + x^2 + x$ → f(x) Done ■ 42 → math\a 42 ■ $3 \cdot x^2 + 4 \cdot x + 25$ → math\f(x) Done 3*x^2+4*x+25→math\f(x) MAIN RAD AUTO FUNC 4/30	MAIN a=1 f(x)=x^3+x^2+x
■ 4·a 4 ■ 4·math\a 168 ■ f(5) 155 ■ math\f(5) 120 math\f(5) MAIN RAD AUTO FUNC 4/30	MATH a=42 f(x)=$3x^2+4x+25$

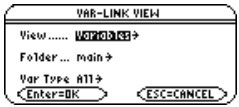
Para ver uma lista das pastas e variáveis existentes, prima [2nd] [VAR-LINK]. No ecrã **VAR-LINK** é possível realçar uma variável e premir [ENTER] para colar esse nome de variável na linha de entrada da aplicação aberta. Se colar um nome de uma variável que não está na pasta actual, o caminho é colado (*Nome da pasta**Nome da variável*).

Listar uma única pasta e/ou tipo de variável, ou aplicações Flash

Quando tiver muitas variáveis e/ou pastas, ou aplicações Flash pode ser difícil localizar uma variável específica. Se trocar a apresentação do ecrã **VAR-LINK**, é possível especificar a informação que deseja ver.

No ecrã **VAR-LINK**:

1. Prima [F2] **View**.
2. Realce a definição que deseja trocar e prima ⊙ para visualizar um menu com as opções válidas. (Para cancelar um menu, prima [ESC].)



View — Permite escolher variáveis, aplicações Flash ou variáveis de sistema a visualizar.

Nota: para visualizar as variáveis de sistema (variáveis Window, etc.), seleccione **3:System**.

Folder — Mostra sempre **1:All** e **2:main** e eventualmente as pastas que foram criadas.

Var Type — Mostra os tipos válidos de variáveis.

↓ — indica que é possível percorrer o ecrã para ver outras variáveis.

3. Seleccione a nova configuração.
4. Volte ao ecrã **VAR-LINK VIEW** e prima [ENTER].

O ecrã **VAR-LINK** é actualizado para mostrar apenas a pasta especificada, o tipo de variável ou a aplicação Flash.

Cópia ou transferência de variáveis de uma pasta para outra

É preciso ter pelo menos uma outra pasta além da pasta **MAIN**. Não é possível utilizar **VAR-LINK** para copiar variáveis dentro da mesma pasta.

1. Na **VAR-LINK**, seleccione as variáveis.
2. Prima [F1] **Manage** e seleccione **2:Copy** ou **4:Move**.
3. Seleccione a pasta de destino.

4. Prima [ENTER]. As variáveis copiadas ou transferidas mantêm seus nomes originais.

   Nota: para copiar uma variável com um nome diferente na mesma pasta (como a1→a2), utilize [STO▸] ou o comando **CopyVar** a partir do ecrã principal.

Bloqueio ou desbloqueio de pastas de variáveis ou aplicações Flash

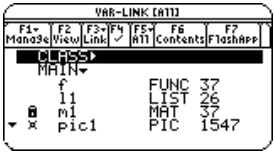
Quando uma variável está bloqueada, não é possível excluí-la, trocar o seu nome, ou armazená-la. Porém, é possível copiá-la, movê-la ou mostrar o conteúdo. Quando uma pasta está bloqueada, é possível manipular as variáveis nela contidas - desde que não estejam bloqueadas - mas não é possível excluí-la. Quando uma aplicação Flash está bloqueada, não é possível exclui-la.

1. Na **VAR-LINK**, seleccione as variáveis e/ou pastas ou aplicação Flash.

2. Prima [F1] **Manage** e seleccione **6:Lock** ou **7:UnLock** .

🔒 indica uma variável ou pasta bloqueada.

⌘ indica uma variável arquivada que é automaticamente bloqueada.



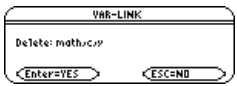
Cancelamento de uma pasta no ecrã VAR-LINK

Quando excluir uma pasta do ecrã **VAR-LINK**, todas as variáveis dessa pasta também são excluídas. Não pode excluir a pasta **MAIN**.

1. Prima [2nd] [VAR-LINK].
2. Prima [F4] para seleccionar a(s) pasta(s) a serem excluídas. (As variáveis da pasta são seleccionadas automaticamente.)

3. Prima [F1] **1:Delete** eller [←].

4. Prima [ENTER] para confirmar a exclusão da pasta e de todas as suas variáveis.



Exclusão de uma variável ou pasta do ecrã principal

Antes de cancelar uma pasta no ecrã principal, deve cancelar primeiro todas as variáveis armazenadas.

- Para cancelar uma variável, introduza o comando **DelVar** no ecrã principal da calculadora.

  DelVar *var1* [, *var2*] [, *var3*] ...

- Paa eliminar todas as variávies de um tipo específico, introduza o comando **DelType** no ecrã principal da calculadora.

  DelType *var_tipo* em que *var_tipo* é o tipo de variável.

  Nota: o comando **DelType** elimina todas as variáveis do tipo especificado em todas as pastas.

- Para cancelar uma pasta vazia, introduza o comando **DelFold** no ecrã principal da calculadora.

  DelFold *pasta1* [, *pasta2*] [, *pasta3*] ...

  Nota: não é possível cancelar a pasta **MAIN**.

Procedimento de colar o nome de uma variável numa aplicação

Suponha que está a digitar uma expressão no ecrã principal e não se consegue lembrar da variável que deseja utilizar. Visualize o ecrã **VAR-LINK**, seleccione uma variável de lista e cole o nome directamente na linha de entrada do ecrã principal.

Aplicações que podem ser utilizadas

É possível colar um nome de uma variável na posição do cursor de uma das seguintes aplicações.

- Ecrã principal, Y= Editor, Editor de Tabelas ou Editor de Dados/Matrizes — O cursor deve estar na linha de entrada.
- Editor de Texto, Editor de Janelas, Solucionador Numérico ou Editor de Programas — O cursor pode estar posicionado em qualquer lugar do ecrã.

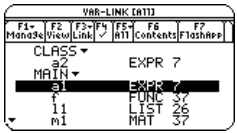
Pode também colar um nome de variável na localização actual do cursor em várias aplicações Flash.

Procedimento

Se iniciar uma das aplicações acima:

1. Posicione o cursor na posição em que deseja inserir o nome da variável.

2. Prima [2nd] [VAR-LINK].
3. Realce a variável correspondente.



Nota: também é possível realçar e colar os nomes das pastas.

4. Prima [ENTER] para colar o nome da variável.

   Nota: neste procedimento, cola o nome da variável, mas não o conteúdo. Para recuperar o conteúdo de uma variável, utilize [2nd] [RCL], em vez de [2nd] [VAR-LINK].

5. Termine de digitar a expressão.

```
sin(a1|
```

```
sin(a1)|
```

Se colar o nome de uma variável que não está contida na pasta actual, cola o caminho da variável.

```
sin(class\a2
```

Supondo que CLASS não é a pasta actual, é colada se realçar a variável a2 em CLASS.

Procedimento de arquivar e desarquivar uma variável

Para arquivar e desarquivar uma ou mais variáveis de forma interactiva, utilize o ecrã **VAR-LINK**. É possível também utilizar estas operações no ecrã principal ou de um programa.

Por que é necessário arquivar uma variável?

O arquivo de dados do utilizador permite:

- Armazenar dados, programas ou quaisquer outras variáveis num local seguro onde não possam ser inadvertidamente excluídos ou alterados.
- Criar memória RAM adicional, arquivando as variáveis.
  Por exemplo:
  - Pode arquivar as variáveis necessárias, mas que não precisa editar ou alterar, ou as variáveis que não está a utilizar actualmente, porém precisa guardar para utilização futura.

    Nota: não é possível arquivar variáveis com nomes reservados ou variáveis de sistema.
  - Se adquirir programas adicionais para a TI-89 Titanium / Voyage™ 200 Graphing Calculator, especialmente, se forem grandes, pode ser que seja necessário libertar mais memória RAM antes de poder instalar esses programas.

Mais memória RAM pode melhorar o desempenho em determinados tipos de cálculos.

No ecrã VAR-LINK

Para arquivar ou desarquivar:

1. Prima [2nd] [VAR-LINK] para visualizar o ecrã **VAR-LINK**.

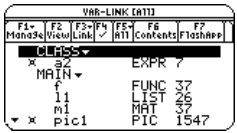
2. Seleccione uma ou mais variáveis, que podem estar em pastas diferentes. (Pode seleccionar uma pasta inteira através da selecção do nome da pasta.)

   Nota: Para seleccionar uma única variável, realce-a. Para seleccionar várias variáveis, realce cada variável e prima [F4] ✓.

3. Prima [F1] e seleccione:

   8:Archive Variable
   – ou –
   9:Unarchive Variable

Se seleccionar **8:Archive Variable**, as variáveis são movidas para o arquivo de dados do utilizador.



¤ = variáveis arquivadas

É possível aceder a uma variável arquivada exactamente como se faz com qualquer variável bloqueada. Para todos os fins, uma variável arquivada continua na pasta original; é simplesmente armazenada no arquivo de dados do utilizador em vez de na RAM.

Nota: uma variável arquivada é bloqueada automaticamente. Pode aceder à variável, mas não a pode editar ou cancelar.

No ecrã principal ou de um programa

Utilize os comandos **Archive** e **Unarchiv**.

Archive *variável1*, *variável2*, …

Unarchiv *variável1*, *variável2*, …

Se aparecer uma mensagem de recolha de lixo

Se utilizar o arquivo de dados do utilizador extensivamente, pode aparecer uma mensagem de recolha de lixo. Este procedimento ocorre se tentar arquivar uma variável quando não houver memória de arquivo livre suficiente. Entretanto, a TI-89 Titanium / Voyage™ 200 Graphing Calculator tenta reorganizar as variáveis arquivadas para criar espaço adicional.

Resposta à mensagem de recolha de lixo

Se aparecer a mensagem à direita:

- Para continuar o arquivo, prima [ENTER].
  – ou –
- Para cancelar, prima [ESC].

Nota: se as pilhas estiverem gastas, devem set substituídas antes de efectuar a reciclagem, porque a memória de arquivo pode ser perdida.

Depois da recolha do lixo, dependendo do espaço libertado, a variável pode ou não ser arquivada. Se ainda não for possível arquivar a variável, algumas variáveis podem ser desarquivadas e pode tentar novamente.

Por que não realizar a recolha de lixo automaticamente sem uma mensagem?

A mensagem:

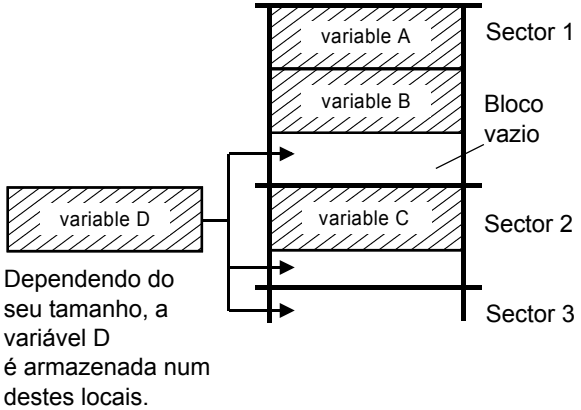
- Permite saber o motivo da demora do arquivo. Alerta também que o arquivo pode falhar se não houver memória suficiente.
- Pode alertar quando um programa entrar em loop, enchendo repetitivamente o arquivo de dados do utilizador. Cancele o arquivo e investigue o motivo.

Por que é necessária uma recolha de lixo?

O arquivo de dados do utilizador está dividido em sectores. Quando começar o arquivo, as variáveis são armazenadas a partir do início do sector 1. Quando não houver espaço suficiente no sector, a próxima variável é armazenada no início do próximo sector. Tipicamente, um bloco vazio fica no fim do sector anterior.

Cada variável arquivada é armazenada no primeiro bloco vazio, suficientemente grande para a armazenar.

Nota: Uma variável arquivada é armazenada num bloco contínuo dentro de um único sector; não pode cruzar um limite de sector.

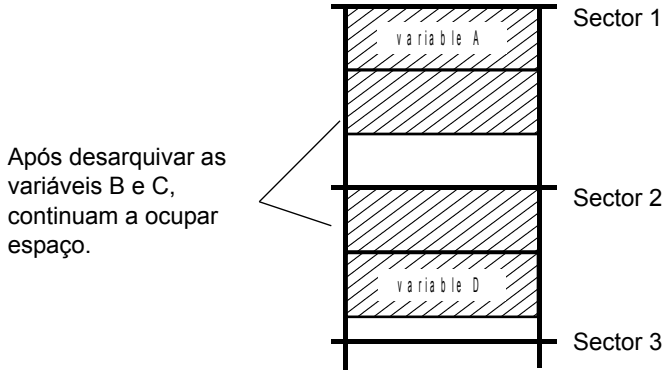


Este processo continua até ao fim do último sector. Dependendo do tamanho das variáveis individuais, os blocos vazios podem representar uma quantidade significativa de espaço.

Nota: a recolha de lixo ocorre quando a variável que se está a arquivar é maior do que qualquer bloco vazio.

Como o desarquivamento de uma variável afecta o processo

Quando uma variável é desarquivada, é copiada para a RAM, mas não é excluída da memória de arquivo de dados do utilizador.

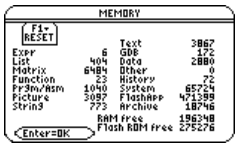


As variáveis desarquivadas são “marcadas para cancelamento”, significando que serão excluídas durante a próxima recolha de lixo.

Se o ecrã MEMORY apresentar espaço livre suficiente

Mesmo que o ecrã **MEMORY** apresente espaço livre suficiente para arquivar uma variável, pode receber ainda uma mensagem de recolha de lixo.

Este ecrã de memória da TI-89 Titanium mostra o espaço livre que ficará disponível depois da exclusão de todas as variáveis “marcadas para cancelamento”.



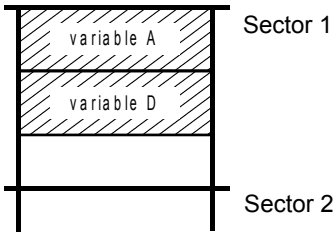
Quando uma variável é desarquivada, a quantidade de Flash ROM livre aumenta imediatamente, mas o espaço só fica disponível depois da próxima recolha de lixo.

A Voyage™ 200 tem 2,7 MB de memória Flash ROM disponível para o utilizador. Pode utilizar os 2,7 MB para as Apps Flash, mas apenas cerca de 1 MB desses 2,7 MB pode ser utilizado para arquivo de dados do utilizador.

Processo de recolha de lixo

O processo de recolha de lixo:

- Exclui variáveis desarquivadas a partir do arquivo de dados do utilizador.
- Reorganiza as variáveis restantes em blocos consecutivos.



Erro de memória no acesso a uma variável arquivada

Uma variável arquivada é tratada da mesma forma que uma variável bloqueada. Pode aceder à variável, mas não é possível editá-la ou cancelá-la. Entretanto, em alguns casos, pode receber uma mensagem de erro de memória (**Memory Error**), se tentar aceder a uma variável arquivada.

O que causa um erro de memória?

A mensagem de erro de memória (**Memory Error**) aparece se não houver RAM livre para aceder à variável arquivada. Talvez pergunte: “Se a variável estiver no arquivo de dados do utilizador, por que é importante a quantidade de RAM disponível?” A resposta é que as operações apresentadas em seguida podem ser realizadas apenas se uma variável estiver na RAM.

- Abertura de uma variável de texto no Editor de Texto.
- Abertura de uma variável de dados, lista, ou matriz no Editor de Dados e Matrizes.
- Abertura de um programa ou função no Editor de Programa.
- Execução de um programa ou referência a uma função.

Nota: Conforme descrito abaixo, uma cópia temporária permite abrir ou executar uma variável arquivada. Entretanto, não é possível guardar quaisquer mudanças na variável.

Para que não seja preciso desarquivar variáveis desnecessariamente, a TI-89 Titanium / Voyage™ 200 Graphing Calculator realiza uma cópia “invisível ao utilizador”. Por exemplo, se executar um programa do arquivo de dados do utilizador, a TI-89 Titanium / Voyage™ 200:

1. Copia o programa para a RAM.
2. Executa o programa.
3. Cancela a cópia da RAM quando o programa termina.

A mensagem de erro aparece se não houver espaço suficiente de RAM para cópia temporária.

Nota: excepto para programas e funções, a consulta a uma variável arquivada não a copia. Se a variável ab é arquivada, não é copiada se for feito **6∗ab**.

Correcção do erro

Para libertar RAM suficiente para aceder à variável:

1. Utilize o ecrã **VAR-LINK** (2nd [VAR-LINK]) para determinar o tamanho da variável arquivada que pretende aceder.
2. Utilize o ecrã **MEMORY** (2nd [MEM]) para verificar o tamanho de RAM livre.
3. Liberte a quantidade necessária de memória através do:
   - Cancelamento de variáveis desnecessárias da RAM.
   - Arquivo de variáveis ou programas grandes (movendo-os da RAM para o arquivo de dados do utilizador).

   Nota: o espaço RAM livre tem de ser maior do que a variável arquivada.

Conectividade

Ligar duas unidades

A TI-89 Titanium e a Voyage™ 200 Graphing Calculator têm um cabo que permite ligar duas unidades. Assim que estiverem ligadas, pode transmitir informações entre as duas unidades. Um cabo USB de unidade-a-unidade é incluído na TI-89 Titanium; utilize a porta USB do dispositivo com o cabo. Um cabo standard de unidade-a-unidade é incluído na Voyage™ 200; utilize a porta E/S do dispositivo com o cabo.

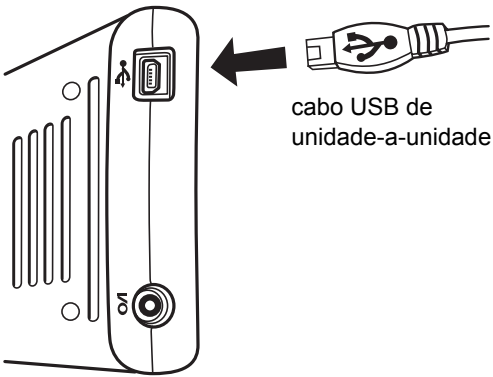
Nota: A TI-89 Titanium integra uma porta USB e uma porta E/S para que possa ligar às unidades portáteis gráficas TI com um dos tipos de porta de ligação. No entanto, a utilização da porta E/S requer o cabo standard de unidade-a-unidade (vendido separadamente) ou o TI Connectivity Cable USB (também vendido separadamente), que é utilizado para ligar a um computador.

Ligar antes de enviar ou receber

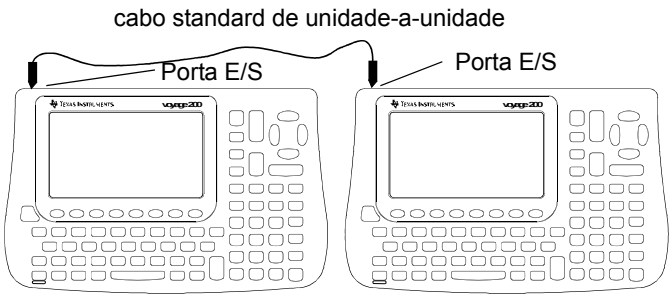
Pressione firmemente uma extremidade do cabo para a inserir na porta de ligação de cada unidade. A unidade pode enviar ou receber, dependendo de como configurar a unidade no ecrã **VAR-LINK**.

Pode ligar uma TI-89 Titanium ou Voyage™ 200 a outra TI-89 Titanium, Voyage™ 200, TI-89 ou TI-92 Plus.

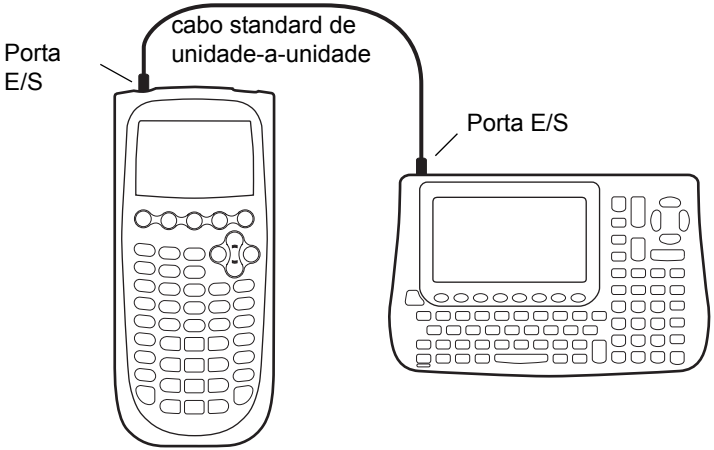
Duas unidades portáteis TI-89 Titanium ligadas em conjunto



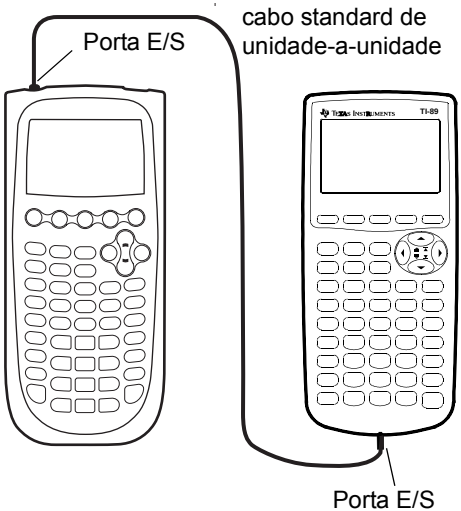
Posicione-as para que os símbolos USB fiquem virados um para o outro e, em seguida, insira o conector.



Duas unidades portáteis Voyage™ 200 ligadas em conjunto



Uma TI-89 Titanium e uma Voyage™ 200 ligadas em conjunto



Uma TI-89 Titanium e uma TI-89 ligadas em conjunto

Transmitir variáveis, aplicações Flash e pastas

A transmissão de variáveis é uma forma conveniente para partilhar qualquer variável listada no ecrã **VAR-LINK** — funções, programas, etc. Também pode transmitir aplicações Flash (Apps) e pastas.

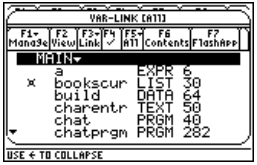
Configurar as unidades

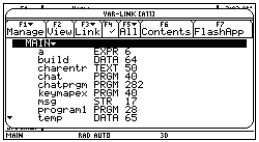
As aplicações Flash só se transferem entre determinadas unidades. Por exemplo, pode transferir uma App de uma TI-89 Titanium para outra TI-89 Titanium ou de uma TI-89

Titanium para uma TI-89. Pode transferir uma App de uma Voyage™ 200 para outra Voyage™ 200 ou de uma Voyage™ 200 para uma TI-92 Plus.

1. Ligue duas unidades portáteis gráficas com o cabo adequado.

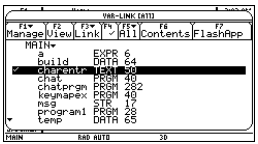
2. Na unidade *transmissora*, prima [2nd] [VAR-LINK] para ver o ecrã **VAR-LINK**.



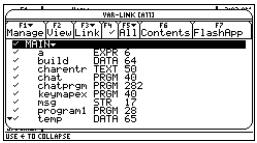


3. Na unidade *transmissora*, seleccione as variáveis, pastas ou aplicações Flash que pretende enviar.

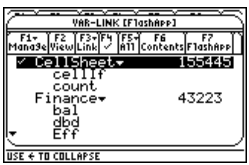
   - Para seleccionar uma variável, aplicação Flash ou pasta, mova o cursor para a realçar e prima [F4] para colocar uma marca de verificação (✓) numa delas.

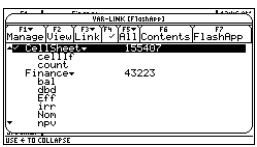


- Se estiver no ecrã **VAR-LINK** predefinido, este selecciona a pasta e o conteúdo. As pastas fechadas abrem-se quando seleccionadas.

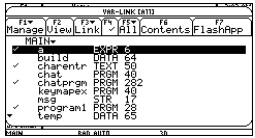


- Se seleccionar uma App Flash (a partir do separador F7), esta selecciona a pasta da App e o conteúdo. Aparece uma marca de verificação antes da pasta, mas não aparece antes do conteúdo. As pastas das Apps Flash fechadas não se abrem automaticamente.

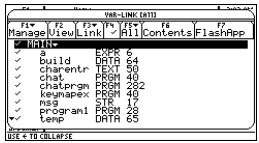




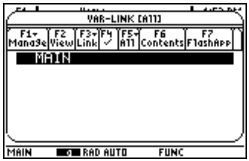
- Para seleccionar várias variáveis, aplicações Flash ou pastas, realce cada uma delas e prima F4 para colocar uma marca de verificação (✓) numa delas. Utilize novamente F4 para anular a selecção de qualquer coisa que não queira transmitir.

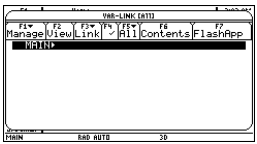


- Para seleccionar todas as variáveis, aplicações Flash ou pastas, utilize F5 **All 1:Select All**.

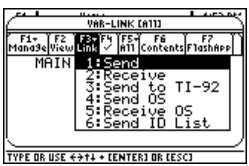


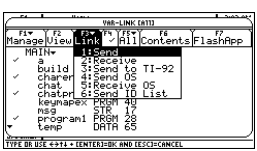
4. On the *receiving* unit, press [2nd] [VAR-LINK] to display the **VAR-LINK** screen. (A unidade transmissora permanece no ecrã **VAR-LINK**.)





5. Nas unidades *receptora* e transmissora, prima [F3] **Link** para ver as opções do menu.





6. Na unidade *receptora*, seleccione **2:Receive**.

   A mensagem **VAR-LINK: WAITING TO RECEIVE** e o indicador **BUSY** aparecem na linha de estado da unidade receptora.

7. Na unidade *transmissora*, seleccione **1:Send**

   Este procedimento inicia a transmissão.

Aparece uma barra de progresso na linha de estado da unidade receptora durante a transmissão. Quando a transmissão terminar, o ecrã **VAR-LINK** é actualizado na unidade receptora.

Nota: Antes de transferir uma App adquirida, a unidade receptora deve ter o certificado adequado, se necessário. Um certificado é um ficheiro gerado pela TI. As Apps gratuitas e de conceitos não requerem um certificado.

Regras para transmitir variáveis, aplicações Flash ou pastas

As variáveis desbloqueadas e desarquivadas que tenham o mesmo nome nas unidades receptoras e transmissoras serão substituídas a partir da unidade transmissora.

As variáveis bloqueadas que tenham o mesmo nome nas unidades transmissoras e receptoras devem ser desbloqueadas na unidade receptora antes de poderem ser substituídas a partir da unidade transmissora. Se as variáveis arquivadas tiverem os mesmos nomes nas unidades transmissoras e receptoras, aparece uma mensagem a pedir-lhe para confirmar se permite substituir as variáveis.

Se seleccionar:	O que acontece:
Variável desbloqueada	A variável é transmitida para a pasta actual e permanece desbloqueada na unidade receptora.
Variável bloqueada	A variável é transmitida para a pasta actual e permanece bloqueada na unidade receptora.
Variável arquivada	A variável é transmitida para a pasta actual e permanece arquivada na unidade receptora.
Aplicação Flash desbloqueada	Se a unidade receptora tiver a certificação correcta, a aplicação Flash é transmitida. Permanece desbloqueada na unidade receptora.

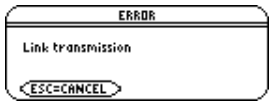
Se seleccionar:	O que acontece:
Aplicação Flash bloqueada	Se a unidade receptora tiver a certificação correcta, a aplicação Flash é transmitida. Permanece bloqueada na unidade receptora.
Pasta desbloqueada	A pasta e os conteúdos seleccionados são transmitidos. A pasta permanece desbloqueada na unidade receptora.
Pasta bloqueada	A pasta e os conteúdos seleccionados são transmitidos. A pasta desbloqueia-se na unidade receptora.

Cancelar uma transmissão

A partir da unidade receptora ou transmissora:

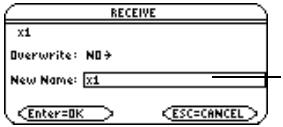
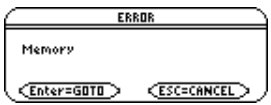
1. Prima [ON].

   Aparece uma mensagem de erro.



2. Prima [ESC] ou [ENTER].

Apresentado em:	Mensagem e descrição:
Unidade transmissora	ERROR Link transmission ESC=CANCEL Aparece passados alguns segundos se: • Não instalar um cabo na porta de ligação da unidade transmissora. – ou – • Uma unidade receptora não estiver instalada na outra extremidade do cabo. – ou – • A unidade receptora não estiver configurada para receber. Prima ESC ou ENTER para cancelar a transmissão. **Nota:** A unidade transmissora pode não mostrar sempre esta mensagem. Em vez desta mensagem, pode aparecer **BUSY** até cancelar a transmissão.
Unidade transmissora	ERROR Unlicensed OS or Flash application Enter=OK ESC=CANCEL A unidade receptora não tem o certificado correcto para o sistema operativo (SO) ou a aplicação Flash a enviar.

Apresentado em:	Mensagem e descrição:
Unidade receptora	 New Name fica activo apena se alterar Overwrite para NO A unidade receptora tem uma variável com o mesmo nome da variável a enviar. • Para substituir a variável existente, prima [ENTER]. (Por predefinição, **Overwrite = YES**.) • Para guardar a variável com um nome diferente, defina **Overwrite = NO**. Na caixa de entrada **New Name**, digite um nome de variável que não exista na unidade receptora. Prima duas vezes [ENTER]. • Para ignorar esta variável e continuar com a seguinte, defina **Overwrite = SKIP** e prima [ENTER]. • Para cancelar a transmissão, prima [ESC].
Unidade receptora	 A unidade receptora não tem memória suficiente para o material que está a ser enviado. Prima [ESC] ou [ENTER] para cancelar a transmissão.

Eliminar variáveis, aplicações Flash ou pastas

1. Prima [2nd] [VAR-LINK] para ver o ecrã **VAR-LINK**.

2. Seleccione as variáveis, as pastas ou as aplicações Flash a eliminar.
   - Para seleccionar uma variável, aplicação Flash ou pasta, mova o cursor para a realçar e prima [F4] para colocar uma marca de verificação (✓) numa delas.
     - Se estiver no ecrã **VAR-LINK** predefinido, este selecciona a pasta e o conteúdo. As pastas fechadas abrem-se quando seleccionadas.
     - Se seleccionar uma App Flash (a partir do separador F7), esta selecciona a pasta da App e o conteúdo. Aparece uma marca de verificação antes da pasta, mas não aparece antes do conteúdo. As pastas das Apps Flash fechadas não se abrem automaticamente.

     Nota: Não pode eliminar a pasta **Main**.
   - Para seleccionar várias variáveis, aplicações Flash ou pastas, realce-as e prima [F4] para colocar uma marca de verificação (✓) nas opções pretendidas. Utilize novamente [F4] para anular a selecção de qualquer coisa que não queira transmitir.
   - Para seleccionar todas as variáveis, aplicações Flash ou pastas, utilize [F5] **All 1:Select All**.
3. Prima [F1] e seleccione **1:Delete**.
   – or –
   Prima [←]. Aparece uma mensagem de confirmação.
4. Prima [ENTER] para confirmar a eliminação.

Onde obter aplicações Flash (Apps)

Para obter informações actualizadas sobre as aplicações Flash disponíveis, consulte o site Web da Texas Instruments em education.ti.com ou contacte a Texas Instruments em TI-Cares™.

Muitas das Apps já não requerem um certificado. Se tentar transferir uma App de uma unidade para a outra e receber a mensagem **Unlicensed OS or Flash application** , tente transferir a App novamente do site Web education.ti.com.

Pode transferir uma aplicação Flash e/ou certificado do site Web da Texas Instruments para um computador e utilizar um USB cable ou TI Connectivity Cable USB para instalar a aplicação ou o certificado na TI-89 Titanium / Voyage™ 200 Graphing Calculator.

Para obter instruções de instalação das Apps Flash, consulte education.ti.com/guides.

Transmitir variáveis com o controlo do programa

Pode utilizar um programa com **GetCalc** e **SendCalc** para transmitir uma variável de um dispositivo para outro.

SendCalc envia uma variável para a porta de ligação, onde um dispositivo ligado pode receber a variável. O dispositivo ligado tem de estar no ecrã inicial ou tem de executar **GetCalc** a partir de um programa.

Pode utilizar os parâmetros opcionais com o comando SendCalc ou GetCalc para especificar a porta USB ou E/S. (Consulte o Anexo A para mais informações.) Se não incluir estes parâmetros, a TI-89 Titanium comunica através da porta USB.

Programa “Chat”

O programa seguinte utiliza os comandos **GetCalc** e **SendCalc**. O programa configura dois circuitos que permite aos dispositivos ligados enviarem e receberem/mostrarem uma variável denominada **msg**. **InputStr** permite que cada utilizador introduza uma mensagem na variável **msg**

```
:Chat()
:Prgm
:ClrIO
:Disp "On first unit to send,","
 enter 1;","On first to receive,"
:InputStr " enter 0",msg
:If msg="0" Then
:   While true
❶   :      GetCalc msg
:      Disp msg
❷   :      InputStr msg
:      SendCalc msg
:   EndWhile
:Else
:   While true
❹   :     InputStr msg
:     SendCalc msg
❺   :     GetCalc msg
:     Disp msg
:  EndWhile
:EndIf
:EndPrgm
```

❸

❻

Notas:

❶ Configura esta unidade para receber e ver a variável msg.
❷ Permite a este utilizador introduzir uma mensagem em msg e enviála.
❸ Circuito executado pela unidade que recebe a primeira mensagem.
❹ Permite a este utilizador introduzir uma mensagem em msg e enviála.
❺ Configura esta unidade para receber e ver msg.
❻ Circuito executado pela unidade que envia a primeira mensagem.

Para sincronizar **GetCalc** e **SendCalc**, os circuitos são dispostos para que a unidade receptora execute o comando **GetCalc** enquanto a unidade transmissora espera que o utilizador introduza uma mensagem.

Executar o programa

Este procedimento assume que:

- Os dois dispositivos estão ligados com o cabo de ligação.
- O programa Chat está instalado em ambos os dispositivos.
  - Utilize o editor de programas dos dispositivos introduzir o programa.
    – ou –
  - Introduza o programa num dispositivo e, em seguida, utilize **VAR-LINK** para transmitir a variável do programa para o outro dispositivo.

Para executar o programa em ambos os dispositivos:

1. No ecrã inicial de cada dispositivo, introduza **chat( )**.
2. Quando cada dispositivo mostrar a linha de comandos inicial, responda como apresentado em seguida.

No:	Digite:
Dispositivo que enviará a primeira mensagem.	**1** e prima [ENTER].
Dispositivo que receberá a primeira mensagem.	**0** e prima [ENTER].

3. Digite uma mensagem e prima [ENTER] para enviar a variável **msg** para o outro dispositivo.

Parar o programa

Como o programa **Chat** configura um circuito infinito em ambos os dispositivos, prima [ON] (em ambos os dispositivos) para parar o programa. Se premir [ESC] para confirmar a mensagem de erro, o programa pára no ecrã Program I/O. Prima [F5] ou [ESC] para voltar ao ecrã inicial.

Actualizar o sistema operativo (SO)

Pode actualizar o SO da TI-89 Titanium ou da Voyage™ 200 com o computador. Pode também transferir o SO de uma unidade para outro modelo idêntico (por exemplo, de uma TI-89 Titanium para uma TI-89 Titanium ou de uma Voyage™ 200 para uma Voyage™ 200).

A instalação do SO reinicia a memória do dispositivo para as definições de fábrica originais. Isto significa que todas as variáveis definidas pelo utilizador (na RAM e no arquivo de dados do utilizador), funções, programas, listas e pastas (excepto pasta Main) são eliminadas. É possível que as aplicações possam ser também eliminadas. Consulte as informações importantes relacionadas com as pilhas antes de actualizar o SO.

Informações importantes sobre a transferência do sistema operativo

Deve instalar pilhas novas antes de iniciar a transferência do SO.

Se estiver a utilizar a TI-89 Titanium num idioma diferente do Inglês, deve certificar-se de que tem a aplicação de localização mais recente instalada quando actualizar o SO. Se não tiver a aplicação de localização mais recente instalada, linhas de comando, mensagens de erro e informações de estado relacionados com as novas funcionalidades do SO podem não aparecer correctamente.

Quando estiver no modo de transferência do SO, a função Automatic Power Down™ (APD™) não funciona. Se deixar o dispositivo no modo de transferência durante um longo período de tempo antes de iniciar o processo de transferência, as pilhas podem gastar-se. Necessitará de substituir as pilhas gastas por pilhas novas antes de iniciar a transferência.

Se interromper acidentalmente a transferência antes de terminar, necessita de reinstalar o SO. Não se esqueça de instalar pilhas novas antes de iniciar a transferência.

Fazer cópia de segurança da unidade antes de instalar o sistema operativo

Quando instalar uma actualização do SO, o processo de instalação:

- Elimina todas as variáveis definidas pelo utilizador (na RAM e no arquivo de dados do utilizador), funções, programas e pastas.
- Pode eliminar todas as aplicações Flash.
- Reinicia todas os modos e as variáveis do sistema para as definições de fábrica originais. Este procedimento é equivalente a utilizar o ecrã **MEMORY** para reiniciar a memória.

Para reter quaisquer variáveis ou aplicações Flash, efectue o seguinte procedimento antes de instalar a actualização:

- **Importante:** Instale pilhas novas.
- Transmita as variáveis ou as aplicações Flash para outro dispositivo.
  – ou –
- Utilize um USB cable ou um TI Connectivity Cable USB e o software TI Connect™ (education.ti.com/downloadticonnect) para enviar as variáveis e/ou as aplicações Flash para um computador.

Onde obter actualizações do sistema operativo

Para obter informações actualizadas sobre as actualizações do SO disponíveis, consulte o site Web da Texas Instruments em education.ti.com ou contacte a Texas Instruments em TI-Cares™.

Pode transferir uma actualização do SO ou da aplicação Flash a partir do site Web da Texas Instruments Web para um computador e utilizar um USB cable ou um TI Connectivity Cable USB para instalar o SO ou a aplicação na TI-89 Titanium / Voyage™ 200.

Para mais informações, consulte as instruções na Web.

Transferir o sistema operativo

O SO só é transferido de uma TI-89 Titanium para uma TI-89 Titanium, de uma TI-89 para uma TI-89, de uma Voyage™ 200 para uma Voyage™ 200 ou de uma TI-92 Plus para uma TI-92 Plus.

Para transferir o sistema operativo (SO) de unidade para unidade:

1. Ligue duas unidades similares, uma TI-89 Titanium a uma TI-89 Titanium; ou uma Voyage™ 200 a uma Voyage™ 200.
2. Nas unidades receptora e transmissora, prima [2nd] [VAR-LINK] para ver o ecrã **VAR-LINK**.
3. Nas unidades receptora e transmissora, prima [F3] **Link** para ver as opções do menu.
4. Na unidade receptora, seleccione **5:Receive OS**.

   Aparece uma mensagem de aviso. Prima [ESC] para parar o processo ou prima [ENTER] para avançar. Se premir [ENTER], **VAR-LINK: WAITING TO RECEIVE** e **BUSY** aparecem na linha de estado da unidade receptora.
5. Na unidade receptora, seleccione **4:Send OS**.

   Aparece uma mensagem de aviso. Prima [ESC] para parar o processo ou prima [ENTER] para iniciar a transmissão.

Importante:

- Não se esqueça de fazer uma cópia de segurança das informações necessárias e instalar pilhas novas na unidade receptora.
- Certifique-se de que as unidades receptora e transmissora estão no ecrã **VAR-LINK**.

A unidade receptora mostra o progresso da transferência. Quando a transferência terminar:

- A unidade transmissora volta ao ecrã **VAR-LINK**.
- A unidade receptora volta ao ambiente de trabalho das Apps ou ao ecrã inicial. Pode necessitar de utilizar [◆] [−] (clarear) ou [◆] [+] (escurecer) para ajustar o contraste.

Não tente cancelar uma transferência do sistema operativo

Depois de começar a transferência, o SO existente na unidade receptora é eliminado. Se interromper a transferência antes de terminar, a unidade receptora não funciona correctamente. Necessita de reinstalar a actualização do SO.

Se estiver a actualizar o sistema operativo em várias unidades

Para efectuar a actualização de um SO em várias unidades, transfira e instale o SO numa unidade e, em seguida, transfira a actualização do SO de uma unidade para outra. Este método é mais rápido do que o instalar em cada unidade através de um computador. As actualizações do SO são gratuitas e não necessitam de um certificado antes de as transferir ou instalar.

Mensagens de erro

A maioria das mensagens de erro aparecem na unidade transmissora. Dependendo de quando o erro ocorre no processo de transferência, pode ver uma mensagem de erro na unidade receptora.

Mensagem de erro	Descrição
ERROR Link transmission ESC=CANCEL	As unidades receptora e transmissora não estão ligadas correctamente ou a unidade receptora não está configurada para receber.

Mensagem de erro	Descrição
ERROR Unlicensed OS or Flash application Enter=OK ESC=CANCEL	O certificado da unidade receptora não é válido para o sistema operativo (SO) na unidade transmissora. Tem de obter e instalar um certificado válido. Se a App já não requerer um certificado, pode transferi-la novamente do site Web da Texas Instruments em education.ti.com e instalar a App novamente na unidade portátil.
ERROR Signature error ESC=CANCEL	Ocorreu um erro durante a transferência. O SO actual da unidade receptora está corrompido. Tem de reinstalar o software do produto a partir de um computador.
ERROR Batteries too low for sending/receiving OS ESC=CANCEL	Substitua as pilhas da unidade que apresenta esta mensagem.

Recolher e transmitir listas de ID

A opção do menu F3 **6:Send ID List** do ecrã **VAR-LINK** permite recolher números de ID electrónicos de dispositivos individuais TI-89 Titanium, TI-89, Voyage™ 200 Graphing Calculator ou TI-92 Plus.

Listas de ID e certificados de grupo

A função ID list oferece uma forma conveniente para recolher ID de dispositivos para compras de grupo de aplicações comerciais. Depois de recolher as ID, transmite-as à Texas Instruments para que possa ser emitido um certificado de grupo.

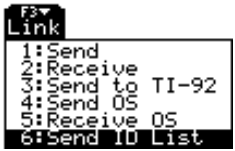
Um certificado de grupo permite distribuir software adquirido a várias unidades TI-89 Titanium, TI-89, Voyage™ 200 ou TI-92 Plus. Pode carregar, eliminar e recarregar o software as vezes que forem necessárias desde que os dispositivos permaneçam listados no certificado de grupo. Pode adicionar novos números de ID e/ou novas aplicações comerciais a um certificado de grupo.

Recolher listas de ID

Pode recolher todas as ID ou utilizar várias unidades de recolha e, em seguida, consolidar as listas de ID num dispositivo.

Para enviar um número de ID de um dispositivo para outro, ligue as duas unidades com um cabo USB de unidade-a-unidade ou cabo standard de unidade-a-unidade.

Passo:	Na(s):	Efectue este procedimento:
1.	Unidade de recolha (Unidade receptora)	Visualize o ecrã inicial. Prima: [HOME] [◆] [CALC HOME]
2.	Unidade transmissora	a. Prima [2nd] [VAR-LINK] para ver o ecrã **VAR-LINK** .

Passo:	Na(s):	Efectue este procedimento:
		b. Prima F3 **Link** e seleccione **6:Send ID List**. 
		A unidade transmissora adiciona uma cópia do número de ID único à lista de ID da unidade de recolha. A unidade transmissora retém sempre o seu próprio número de ID, que não é possível eliminar do dispositivo.
3.	Unidades adicionais	Repita os passos 1 e 2 até recolher todas as ID de um dispositivo. Dependendo da memória disponível no dispositivo de recolha, pode recolher mais de 4000 ID.

Notas:

- Não pode ver a lista de ID nas unidades de recolha ou transmissora.
- Sempre que uma lista de ID é enviada de um dispositivo para outro, a lista de ID é eliminada automaticamente da unidade transmissora.
- Se recolher duas vezes a ID de um dispositivo, a ID duplicada é eliminada automaticamente da lista.

Limpar a lista de ID

A lista de ID permanece no dispositivo de recolha até ser enviada para o computador. Pode utilizar o dispositivo de recolha para enviar a lista para outros computadores.

Para limpar a lista de ID da unidade de recolha:

1. Prima [2nd] [VAR-LINK] para ver o ecrã **VAR-LINK**.

2. Prima [F1] **Manage** e seleccione **A:Clear ID List**.

Compatibilidade entre a TI-89 Titanium, a Voyage™ 200, a TI-89 e a TI-92 Plus

Geralmente, os programas e os dados da TI-89 Titanium, da TI-89, da Voyage™ 200 e da TI-92 Plus são compatíveis uns com os outros, com algumas excepções.

A maioria das funções da TI-89 Titanium são compatíveis com a TI-89, a Voyage™ 200 e a TI-92 Plus. A TI-89 Titanium e a TI-89 são similares, excepto se a TI-89 Titanium tiver mais memória (mais espaço para Apps e o arquivo do utilizador) e a TI-89 Titanium tiver uma porta USB. A Voyage™ 200 é igual à TI-92 Plus, excepto que tem mais memória e, por conseguinte, mais espaço para aplicações (Apps).

Todos os dados são compatíveis entre a TI-89 Titanium, a TI-89, a Voyage™ 200 e a TI-92 Plus, mas alguns programas escritos para um dispositivo podem não ser executados da mesma forma noutro dispositivo devido às diferenças dos teclados e dos tamanhos dos ecrãs dos dispositivos, e a porta USB da TI-89 Titanium.

Podem ocorrer outras incompatibilidades devido às diferenças nas versões dos sistemas operativos. Para transferir a versão mais recente do sistema operativo, visite o site Web da Texas Instruments em education.ti.com/downloadticonnect.

Tabela de transmissão e ligação

Para → **De ↓**	**TI-89 Titanium**	**TI-89**	**Voyage™ 200**	**TI-92 Plus**
TI-89 Titanium	SO Apps Variáveis	Apps Variáveis	Variáveis	Variáveis
TI-89	Apps Variáveis	SO Apps Variáveis	Variáveis	Variáveis
Voyage™ 200	Variáveis	Variáveis	SO Apps Variáveis	Apps Variáveis
TI-92 Plus	Variáveis	Variáveis	Apps Variáveis	SO Apps Variáveis

Actividades

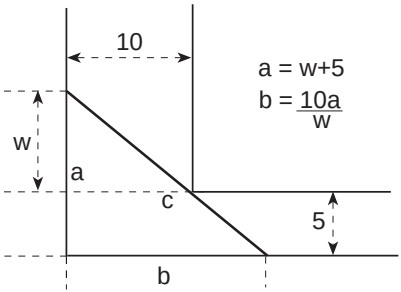
Análise do problema do poste e do canto

Dois corredores, de 10 e 5 pés de largura, respectivamente, encontram-se no canto de um edifício. Tendo de passar um poste neste canto sem o inclinar, calcule o comprimento máximo do poste.

Comprimento máximo do poste no corredor

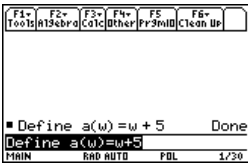
O comprimento máximo do poste *c* é o menor segmento de recta que toca o canto interno e os lados opostos dos dois corredores, conforme o diagrama abaixo.

Calcule o comprimento *c* em relação a *w* com proporcionalidade entre lados e o teorema de Pitágoras. Em seguida, encontre a raiz da primeira derivada de **c(w)**. O valor mínimo de **c(w)** é o comprimento máximo do poste.

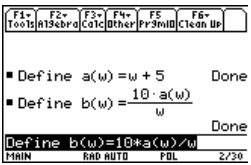


1. Defina a expressão para o lado *a* em função de *w* e armazene-a em **a(w)**.

   Nota: para definir uma função, utilize nomes com vários caracteres.

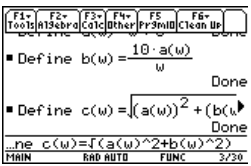


2. Defina a expressão para o lado *b* em função de *w* e armazene-a em **b(w)**.



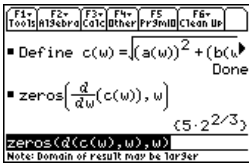
3. Defina a expressão para o lado *c* em função de *w* e armazene-a em **c(w)**.

   Introduza: **Define c(w)= √(a(w)^2+b(w)^2)**



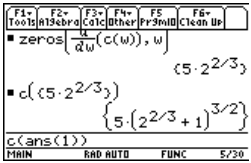
4. Utilize o comando **zeros( )** para calcular a raiz da primeira derivada de **c(w)** para encontrar o valor mínimo de **c(w)**.

   Nota: o comprimento máximo do poste é o valor mínimo de **c(w)**.



5. Calcule o comprimento máximo exacto do poste.

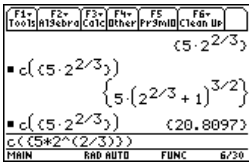
   Introduza: **c** (2nd [ANS])



6. Calcule o comprimento máximo aproximado do poste.

   Resultado: Aproximadamente 20.8097 pés.

   Nota: copie e cole o resultado da etapa 4 na linha de entrada entre os parêntesis de **c( )** e prima [◆] [ENTER].



Dedução de uma solução de uma equação do segundo grau

Esta aplicação descreve como deduzir a solução de uma equação de segundo grau:

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

Para mais informações sobre o procedimento utilizado neste exemplo, consulte no módulo *Manipulação simbólica*.

Operações para deduzir a solução de uma equação do segundo grau

Para encontrar a solução de uma equação de segundo grau completando os quadrados, realize as seguintes etapas:

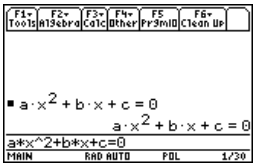
1. Apague todas as variáveis de um carácter da pasta actual.

   [2nd] [F6]

   [F6]

   Escolha **1:Clear a-z** e prima [ENTER] para confirmar.

2. No ecrã principal, introduza a equação genérica de segundo grau: $\mathbf{ax^2+bx+c=0}$.

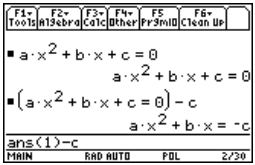


3. Subtraia c de ambos os lados da equação.

   [2nd] [ANS] [−] [alpha] **C**

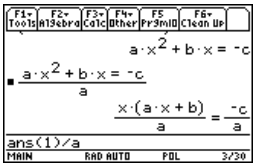
   [2nd] [ANS] [−] **C**

   Nota: este exemplo utiliza o resultado da última resposta para realizar operações na TI-89 Titanium / Voyage™ 200. Esta função permite reduzir o número de teclas premidas e, desde logo, a possibilidade de cometer erros.

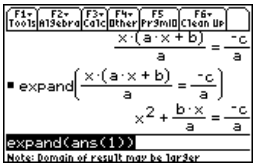


4. Divida ambos os lados da equação pelo primeiro coeficiente principal *a*.

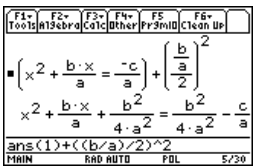
   Nota: continue com a última resposta ([2nd] [ANS]) da mesma forma que na etapa 3, nas etapas de 4 a 9.



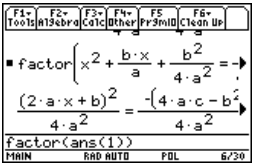
5. Utilize a função **expand( )** para expandir o resultado da última resposta.



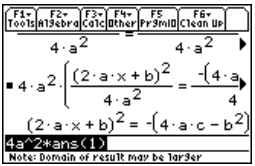
6. Complete os quadrados e acrescente **((b/a)/2)**^{2} em ambos os lados da equação.



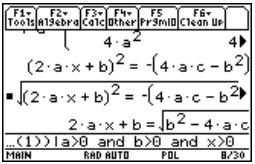
7. Calcule o resultado com a função **factor( )**.



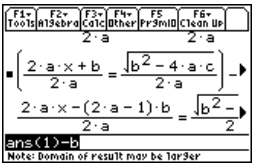
8. Multiplique ambos os lados da equação por **4a**^{2}.



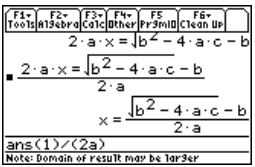
9. Determine a raiz quadrada de ambos os lados da equação, considerando **a>0** e **b>0** e **x>0**.



10. Calcule *x* subtraindo *b* de ambos os lados e então dividindo por **2a**.



Nota: devido às condições impostas na etapa 9, esta é apenas uma das duas soluções de uma equação do segundo grau.



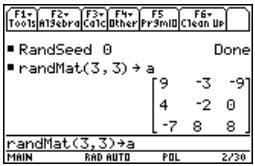
Explorar uma matriz

Esta actividade mostra como realizar várias operações com matrizes.

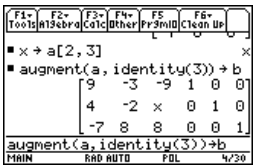
Explorar uma matriz 3x3

Proceda da seguinte maneira para gerar uma matriz aleatória, ampliá-la com a matriz identidade e obter um valor inválido para a matriz inversa.

1. No ecrã principal, utilize o comando **RandSeed** para estabelecer o início do gerador de números aleatórios com o valor predefinido; em seguida, utilize **randMat( )** para criar uma matriz aleatória 3x3 e armazená-la em *a*.



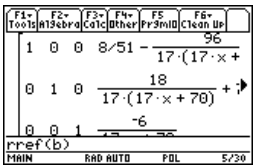
2. Substitua o elemento *[2,3]* da matriz pela variável *x* e utilize a função **augment( )** para ampliar a matriz a com *a* identidade 3x3 e armazenar o resultado em *b*.



3. Utilize **rref( )** para “reduzir as linhas” da matriz *b*:

   No resultado, a matriz identidade aparece nas três primeiras colunas e **a^-1** nas três últimas.

   Nota: para chegar ao resultado no ecrã, utilize o cursor na área do histórico.

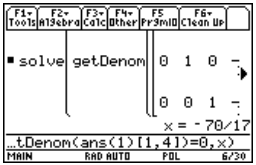


4. Localize o valor de *x* que fará com que a matriz inversa seja inválida.

   Introduza: **solve(getDenom(** [2nd] [ANS] **[1,4])=0,x)**

   Resultado: **x=⁻70/17**

   Nota: para chegar ao resultado no ecrã, utilize o cursor na área do histórico.



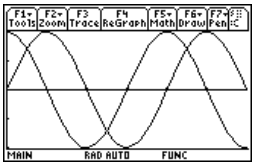
Explorar cos(x) = sin(x)

Esta actividade utiliza dois métodos para determinar onde **cos(x) = sin(x)** para valores de *x* entre 0 e 3π.

Método 1: representação gráfica

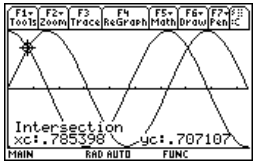
Proceda da seguinte maneira para observar a intersecção entre os gráficos das funções **y1(x)=cos(x)** e **y2(x)=sin(x)**:

1. No **Y= Editor**, defina **y1(x)=cos(x)** e **y2(x)=sin(x)**.
2. No **Window Editor**, defina **xmin=0** e **xmax=3π**.
3. Prima [F2] e seleccione **A:ZoomFit**.



4. Localize os pontos de intersecção entre as duas funções.

   Nota: prima F5 e seleccione **5:Intersection**. Responda às solicitações do ecrã para seleccionar as duas curvas e os limites inferior e superior da intersecção *A*, introduza a informação adequada.



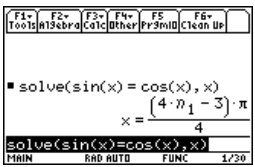
5. Anote as coordenadas de *x* e *y*. (Repita as etapas 4 e 5 para encontrar as outras intersecções.)

Método 2: manipulação simbólica

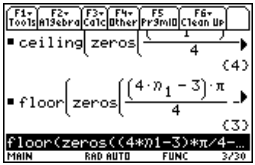
Proceda da seguinte maneira para resolver a equação **sin(x)=cos(x)** em relação a *x*.

1. No ecrã principal, introduza **solve(sin(x)= cos(x),x)**.

   A solução para *x* são os pontos em que *@n1* é um número inteiro.

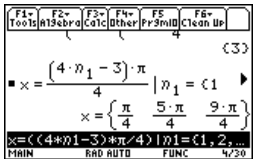


2. Utilize os comandos **ceiling ( )** e **floor ( )** para localizar os valores inteiros superiores e inferiores aos pontos de intersecção como indicado.



   Nota: posicione o cursor na área do histórico para realçar a última resposta. Prima [ENTER] para copiar o resultado da solução geral.

3. Introduza a solução geral para *x* e aplique as condições *@n1* como indicado.



   Compare o resultado com o método 1.

   Nota: Para obter o operador "*with*":

   [|]

   [2nd] [{][QUIT][✓]

Cálculo da área mínima de um paralelepípedo

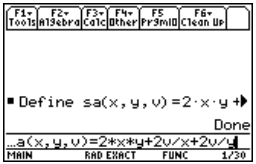
Esta actividade mostra como determinar a área da superfície mínima de um paralelepípedo que tenha um volume *V* constante. Informações detalhadas sobre os procedimentos utilizados neste exemplo estão descritas no módulo *Manipulação simbólica* e no módulo *Representação gráfica em 3D*.

Explorar um gráfico em 3D da área de um paralelepípedo

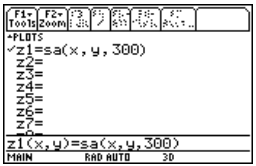
Proceda da seguinte maneira para definir uma função para a área de um paralelepípedo, traçar um gráfico em 3D e utilizar a ferramenta **Trace** para localizar um ponto próximo à área mínima da superfície.

1. No ecrã principal, defina a função **sa(x,y,v)** para a área do paralelepípedo.

   Introduza:
   define sa(x,y,v)=2∗x∗y+2v/x+2v/y

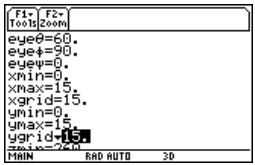


2. Seleccione o modo 3D Graph e, em seguida, introduza a função para **z1(x,y)** conforme indicado neste exemplo com o volume *v=300*.

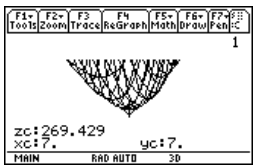


3. Defina as variáveis Window da seguinte maneira:

   eye= **[60,90,0]**
   x= **[0,15,15]**
   y= **[0,15,15]**
   z= **[260,300]**
   ncontour= **[5]**



4. Represente graficamente a função e utilize **Trace** para posicionar no ponto mais próximo do valor mínimo da função da superfície.

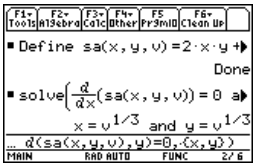


Cálculo analítico da área mínima da superfície

Proceda da seguinte maneira para resolver o problema de forma analítica no ecrã principal.

1. Calcule o valor de *x* e *y* em função de *v*.

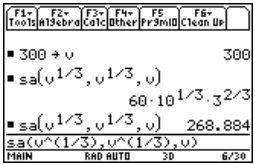
   Introduza: **solve(d(sa(x,y,v),1x)=0** e **(d(sa(x,y,v),y)=0, {x,y})**.



2. Calcule a área mínima da superfície quando o valor de *v* for igual a 300.

   Introduza: **300→v**
   Introduza: **sa(v^(1/3), v^(1/3),v)**

   Nota: prima [ENTER] para obter o resultado exacto na forma simbólica. Prima [◆] [ENTER] para obter o resultado aproximado na forma decimal.



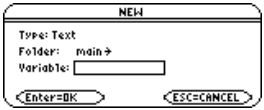
Execução de um script com o Editor de Texto

Esta actividade mostra como utilizar o Editor de Texto para executar um script de tutorial. Para mais informações sobre as operações de texto, consulte no módulo *Editor de Texto*.

Execução de um script

Proceda da seguinte maneira para escrever um script com o **Editor de Texto**, testar cada linha e verificar os resultados na área do histórico do ecrã principal.**Nota:** para

1. Abra o **Editor de Texto** e crie uma nova variável denominada **demo1**.

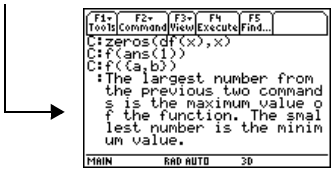


aceder ao símbolo de comando "*C*", utilize o menu F2 **1:Command** da barra de ferramentas.

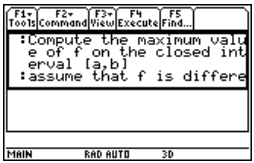
2. Digite as seguintes linhas no **Editor de Texto**.

```
  : Compute the maximum value of f on the closed interval [a,b]
  : assume that f is differentiable on [a,b]
C : define f(x)=x^3−2x^2+x−7
C : 1→a:3.22→b
C : d(f(x),x)→df(x)
C : zeros(df(x),x)
C : f(ans(1))
C : f({a,b})
```

: The largest number from the previous two commands is the maximum value of the function. The smallest number is the minimum value.

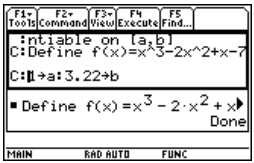


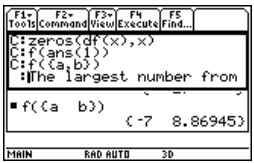
3. Prima F3 e seleccione **1:Script view** para visualizar o **Editor de Texto** e o ecrã principal com o ecrã dividido. Posicione o cursor na primeira linha do **Editor de Texto**.



4. Prima F4 repetidamente para executar as linhas do script, uma de cada vez.

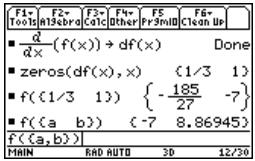
   Nota: prima F3 e seleccione **2:Clear split** para que o **Editor de Texto** volte a aparecer com o ecrã completo.





5. Para ver os resultados do script com o ecrã completo, passe para o ecrã principal.

   Nota: prima [2nd] [QUIT] duas vezes para visualizar o ecrã principal.



Decomposição de uma função racional

Esta actividade verifica o que acontece quando uma função racional é decomposta em quociente e resto. Para mais informações sobre o procedimento utilizado neste exemplo, consulte no módulo *Representação gráfica de funções básicas* e no módulo *Manipulação simbólica*.

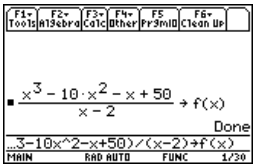
Decomposição de uma função racional

Para examinar a decomposição da função racional **f(x)=(x^3−10x^2−x+50)/(x−2)** num gráfico:

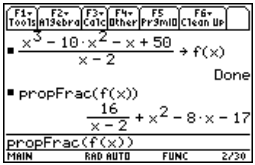
1. No ecrã principal, introduza a função racional conforme descrito abaixo e armazene-a em uma função **f(x)**.

   Introduza: **(x^3−10x^2−x+50)/(x−2)→f(x)**

   Nota: os dados introduzidos aparecem com caracteres invertidos nos ecrãs de exemplos.

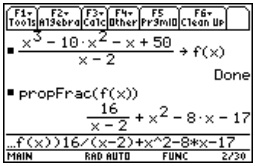


2. Utilize o comando fracção própria (**propFrac**) para decompor a função em quociente e resto.



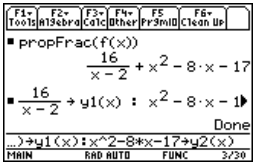
3. Copie a última resposta na linha de entrada.
—ou—
Introduza: **16/(x−2)+x^2−8∗x−17**

Nota: posicione o cursor na área do histórico para realçar a última resposta. Prima [ENTER] para a copiar para a linha de entrada.

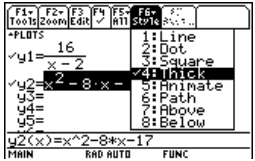


4. Edite a última resposta na linha de entrada. Armazene o resto em **y1(x)** e o quociente em **y2(x)** como indicado.

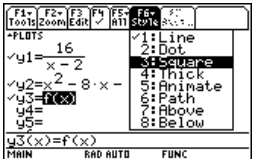
Introduza:
16/(x−2)→y1(x): x^2−8∗x−17→y2(x)



5. No **Y= Editor**, seleccione o estilo de representação “Thick” (contínua) para **y2(x)**.

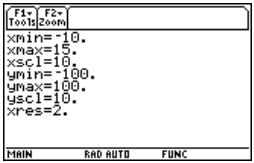


6. Adicione a função original **f(x)** a **y3(x)** e seleccione o estilo de representação “Square” (em quadrados).



7. No **Window Editor**, defina as variáveis Window da seguinte maneira:

   x= [⁻10,15,10]
   y= [⁻100,100,10]



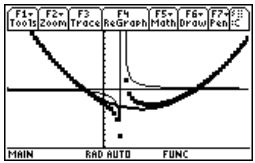
8. Desenhe o gráfico.

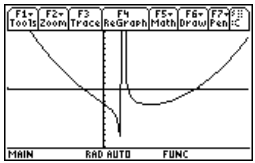
   Nota: verifique se o modo Graph está definido em Function.

Observe que o comportamento geral da função **f(x)** é basicamente representado no quociente de segundo grau **y2(x)**. A expressão racional é basicamente uma função de

segundo grau porque *x* torna-se muito grande tanto na direcção positiva como na negativa.

O gráfico inferior **y3(x)=f(x)** é representado separadamente com o estilo “Line” (de linha).





Estudos estatísticos: filtrar dados por categorias

Esta actividade fornece um estudo estatístico dos pesos dos estudantes da escola secundária, onde utiliza categorias para filtrar os dados.

Filtrar dados por categorias

Os estudantes foram divididos em oito categorias, conforme o sexo e o ano (primeiro, segundo, terceiro e quarto ano). Os dados (peso expresso em libras) e as categorias correspondentes são introduzidos no **Editor de Dados/Matrizes**.

Tabela 1: Categoria X Descrição

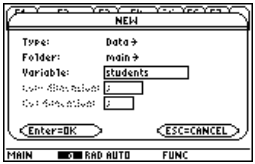
Categoria (C2)	Ano do curso e sexo
1	Alunos do primeiro ano
2	Alunas do primeiro ano
3	Alunos do segundo ano
4	Alunas do segundo ano
5	Alunos do terceiro ano
6	Alunas do terceiro ano
7	Alunos do quarto ano
8	Alunas do quarto ano

Tabela 2: Comparação C1 (peso de cada aluno em libras) X C2 (categoria)

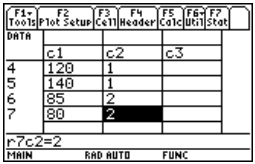
C1	C2	C1	C2	C1	C2	C1	C2
110	1	115	3	130	5	145	7
125	1	135	3	145	5	160	7
105	1	110	3	140	5	165	7
120	1	130	3	145	5	170	7
140	1	150	3	165	5	190	7
85	2	90	4	100	6	110	8
80	2	95	4	105	6	115	8
90	2	85	4	115	6	125	8
80	2	100	4	110	6	120	8
95	2	95	4	120	6	125	8

Proceda da seguinte maneira para comparar o peso dos estudantes segundo o ano do curso.

1. Inicie o **Editor de Dados/Matrizes** e crie uma nova variável de dados denominada **students**.

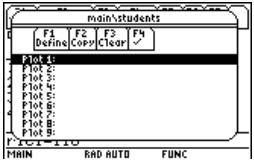


2. Introduza os dados e as categorias da tabela 2 nas colunas **c1** e **c2**, respectivamente.

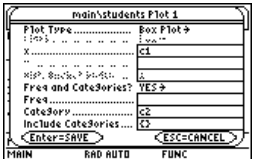


3. Abra o menu F2 **Plot Setup** da barra de ferramentas.

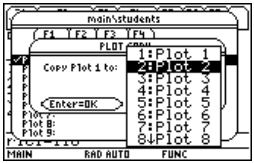
   Nota: crie vários gráficos para comparar os diferentes subconjuntos de dados.



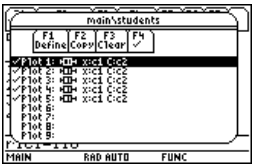
4. Defina o gráfico e os parâmetros do filtro para **Plot 1** conforme apresentado neste ecrã.



5. Copie **Plot 1** em **Plot 2**.

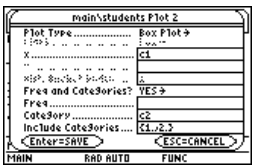


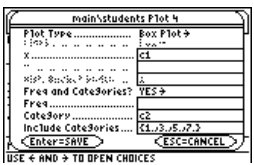
6. Repita a etapa 5 e copie **Plot 1** em **Plot 3**, **Plot 4** e **Plot 5**.



7. Prima F1 e modifique o item **Include Categories** de **Plot 2** até **Plot 5**, da seguinte maneira:

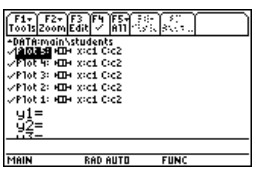
   Plot 2: {1,2}
   (alunos e alunas do primeiro ano)
   Plot 3: {7,8}
   (alunos e alunas do quarto ano)
   Plot 4: {1,3,5,7}
   (todos os rapazes)
   Plot 5: {2,4,6,8}
   (todas as raparigas)



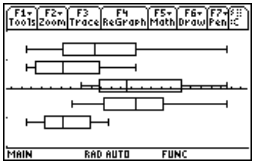


8. No **Y= Editor**, anule a selecção de qualquer função que possa ter sido seleccionada a partir de uma actividade anterior.

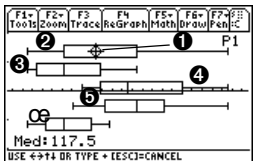
   Nota: seleccione apenas **Plot 1** a **Plot 5**.



9. Visualize os gráficos, pressionando F2 e seleccionando **9:Zoomdata**.



10. Utilize a ferramenta **Trace** para comparar a mediana dos pesos dos estudantes nos vários subconjuntos.



❶ mediana, todos os estudantes
❷ Todos os estudantes
❸ Todos os alunos do primeiro ano
❹ Todos os alunos do quarto ano
❺ Todos os rapazes
❻ Todas as raparigas

Programa CBL 2 ™ para a TI-89 Titanium / Voyage™ 200

Estea actividade fornece um programa que pode ser utilizado quando a TI-89 Titanium / Voyage™ 200 estiver ligada a uma unidade Calculator-Based Laboratory™ (CBL2™). Este programa funciona com a “Newton’s Law of Cooling”. Pode utilizar o teclado do computador para digitar texto e utilizar o software TI Connect™ para o enviar para a calculadora. Estão disponíveis mais programas CBL2™ no sítio Web da TI education.ti.com.

Instrução do programa	Descrição
:cooltemp()	Nome do programa

Instrução do programa	Descrição
:Prgm	
:Local i	Estabelece uma variável local que existe apenas durante o tempo de execução.
:setMode("Graph","FUNCTION")	Configura a TI-89 Titanium / Voyage™ 200 para a representação gráfica de uma função.
:PlotsOff	Desactiva eventuais gráficos precedentes.
:FnOff	Desactiva eventuais funções precedentes.
:ClrDraw	Limpa eventuais objectos previamente traçados nos ecrãs gráficos.
:ClrGraph	Limpa eventuais gráficos precedentes.
:ClrIO	Limpa o ecrã Program IO (input/output) da TI-89 Titanium / Voyage™ 200.
:-10→xmin:99→xmax:10→xscl	Define as variáveis Window.
:-20→ymin:100→ymax:10→yscl	
:{0}→data	Cria e/ou limpa uma lista denominada data.
:{0}→time	Cria e/ou limpa uma lista denominada time.
:Send{1,0}	Envia um comando para limpar a unidade CBL 2/CBL.

Instrução do programa	Descrição
:Send{1,2,1}	Define Chan. 2 da unidade CBL 2/CBL em AutoID para registar a temperatura.
:Disp "Press ENTER to start"	Solicita ao utilizador que prime [ENTER].
:Disp "graphingTemperature."	
:Pause	Espera que o utilizador esteja pronto para começar.
:PtText "TEMP(C)",2,99	Rotula o eixo y do gráfico.
:PtText "T(S)",80,⁻5	Rotula o eixo x do gráfico.
:Send{3,1,⁻1,0}	Envia o comando Trigger à unidade CBL 2/CBL; recolhe dados em tempo real.
:For i,1,99	Repete as duas instruções seguintes para 99 leituras de temperatura.
:Get data[i]	Recebe uma temperatura da unidade CBL 2/CBL e a armazena numa lista.
:PtOn i,data[i]	Representa graficamente os dados de temperatura.
:EndFor	
:seq(i,i,1,99,1)→time	Cria uma lista para representar os números da amostra time ou data.
:NewPlot 1,1,time,data,,,,4	Traça time e data com NewPlot e a ferramenta Trace.
:DispG	Mostra o gráfico.
:PtText "TEMP(C)",2,99	Torna a rotular os eixos.

Instrução do programa	Descrição
:PtText "T(S)",80,-5	
:EndPrgm	Interrompe o programa.

É também possível utilizar a Calculator-Based Ranger™ (CBR™) para explorar as relações matemáticas e científicas entre distância, velocidade, aceleração e tempo, utilizando os dados recolhidos nas actividades realizadas.

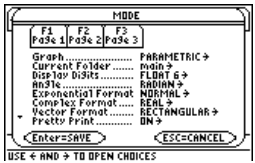
Estudo da trajectória de uma bola de beisebol

Esta actividade utiliza a configuração de ecrã dividido para mostrar um gráfico paramétrico e uma tabela e, em simultâneo, estudar a trajectória de uma bola de beisebol após uma tacada.

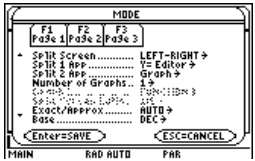
Definição de um gráfico paramétrico e de uma tabela

Proceda da seguinte maneira para estudar a trajectória de uma bola de beisebol, que parte com uma velocidade inicial de 95 pés por segundo e um ângulo inicial de 32 graus.

1. Defina os modos para **Page 1** conforme apresentado neste ecrã.



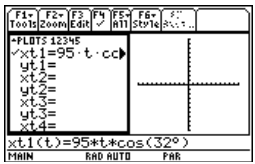
2. Defina os modos para **Page 2** conforme apresentado neste ecrã.



3. No **Y= Editor** à esquerda do ecrã, introduza a equação da distância da bola no tempo **t** para **xt1(t)**.

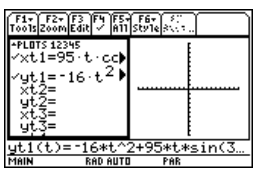
   xt1(t)=95∗t∗cos(32°)

   Nota: prima [2nd] [°] para obter o símbolo de graus.



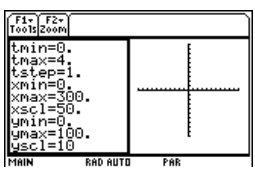
4. No **Y= Editor**, introduza a equação para a altura da bola no tempo **t** para **yt1(t)**.

   yt1(t)=⁻16∗t^2+95∗t∗sin(32°)



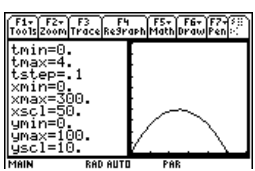
5. Defina as variáveis Window da seguinte maneira:

   t values= [0,4,.1]
   x values= [0,300,50]
   y values= [0,100,10]



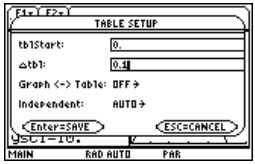
6. Vá para o lado direito do ecrã e visualize o gráfico.

   Nota: prima [2nd] [↔].



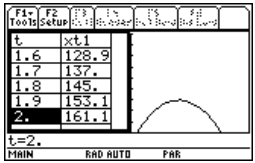
7. Visualize a caixa de diálogo **TABLE SETUP** e modifique **tblStart** para **0** e **Δtbl** para **0.1**.

   Nota: prima [◆] [TBLSET].



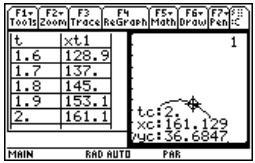
8. Visualize a tabela no lado esquerdo do ecrã e prima ⊖ para realçar *t=2*.

   Nota: prima [◆] [TABLE].



9. Vá para o lado direito do ecrã. Prima [F3] e trace o gráfico para visualizar os valores de **xc** e **yc** quando *tc=2*.

   Nota: se deslocar o cursor de *tc=0.0* para *tc=3.1*, é possível ver a posição da bola no tempo *tc*.



Exercício opcional

Supondo que a velocidade inicial é 95 pés por segundo, calcule o ângulo com que a bola deve ser lançada para alcançar a distância máxima.

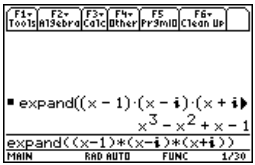
Visualização de raízes complexas de um polinómio de terceiro grau

Esta actividade descreve a representação gráfica das raízes complexas de um polinómio cúbico.

Visualizar raízes complexas

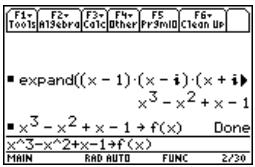
Proceda da seguinte maneira para expandir o polinómio de terceiro grau **(x−1)(x−*i*)(x+*i*)**, encontrar o valor absoluto da função, representar graficamente a superfície do módulo e utilizar a ferramenta **Trace** para explorar a superfície.

1. No ecrã principal, utilize a função **expand( )** para expandir a expressão de terceiro grau **(x−1)(x−*i*)(x+*i*)** e ver o primeiro polinómio.



2. Copie e cole a última resposta na linha de entrada e armazene-a na função **f(x)**.

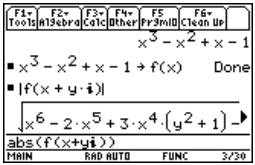
   Nota: mova o cursor para a área do histórico para realçar a última resposta e prima [ENTER], para a copiar para a linha de entrada.



3. Utilize a função **abs( )** para encontrar o valor absoluto de **f(x+y*i*)**.

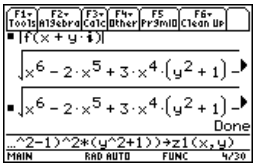
   (Este cálculo pode durar cerca de dois minutos).

   Nota: o valor absoluto da função produz um efeito visual segundo o qual as raízes apenas tocam no eixo *x* em vez de o cortar. Da mesma forma, o valor absoluto de uma função de duas variáveis apenas tocará no plano *xy*.



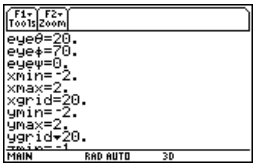
4. Copie e cole a última resposta na linha de entrada e armazene-a na função **z1(x,y)**.

   Nota: o gráfico de **z1(x,y)** é a superfície do módulo.



5. Defina o modo de representação gráfica em 3D, active os eixos para o formato gráfico e defina as variáveis Window da seguinte maneira:

   eye= [20,70,0]
   x= [⁻2,2,20]
   y= [⁻2,2,20]
   z= [⁻1,2]
   ncontour= [5]

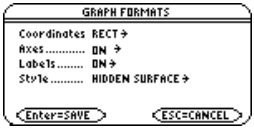


6. No **Y= Editor**, prima:

   [ ] [◆] [|]

   [ ] [◆] **F**

   e configure as variáveis em Graph Format para:

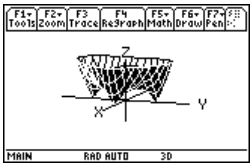


   Axes= ON
   Labels= ON
   Style= HIDDEN SURFACE

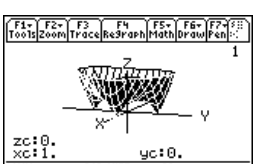
   Nota: o cálculo e o desenho do gráfico duram aproximadamente três minutos.

7. Represente graficamente a superfície do módulo.

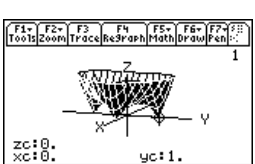
   O gráfico em 3D permite visualizar uma imagem das raízes nos pontos em que a superfície toca no plano **xy**.



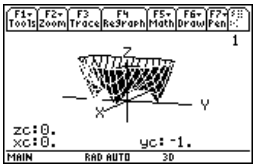
8. Utilize a ferramenta Trace para explorar os valores da função em que **x=1** e **y=0**.



9. Utilize a ferramenta Trace para explorar os valores da função em que **x=0** e **y=1**.



10. Utilize a ferramenta Trace para explorar os valores da função em que **x=0** e **y=⁻1**.



Resumo

Observe que o valor de **zc** é zero para cada valor da função nas etapas 7–9. Portanto, é possível visualizar as raízes complexas 1, ^{-}i, i do polinómio $\mathbf{x^3-x^2+x-1}$ com os três pontos em que o gráfico da superfície toca o plano **xy**.

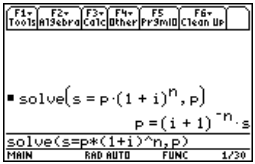
Solução de um problema de anuidade simples

Esta actividade pode ser utilizada para calcular a taxa de juros, o valor do principal inicial, o número de períodos compostos e o valor de uma anuidade no futuro.

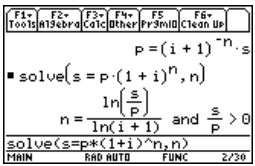
Cálculo da taxa de juros de uma anuidade

Proceda da seguinte maneira para calcular a taxa de juros (**i**) de uma anuidade em que o capital inicial (**p**) é igual a 1.000, o número de períodos de capitalização (**n**) é 6 e o valor futuro (**s**) é igual a 2.000.

1. Na tela principal, introduza a equação para calcular **p**.



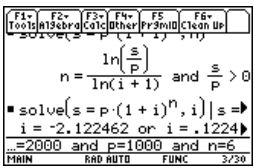
2. Introduza a equação para calcular **n**.



3. Introduza a equação para calcular i com o operador "*with*".

   solve(s=p∗(1+i)^n,i) | s=2000 e **p=1000** e **n=6**

   Resultado: a taxa de juros é de 12,246%.



   Nota:

   - Para introduzir o operador "*with*" (|):
     [|];
     [2nd] ['']
   - Prima [◆] [ENTER] para obter um resultado com ponto flutuante.

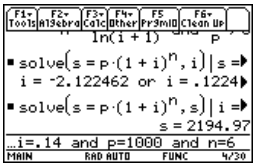
Cálculo do valor futuro da anuidade

Calcule o valor futuro de uma anuidade, utilizando os valores do exemplo anterior, com uma taxa de juros de 14%.

Introduza a equação para calcular **s**.

solve(s=p∗(1+i)^n,s) | i=.14 e **p=1000** e **n=6**

Resultado: o valor futuro com uma taxa de juros de 14% é igual a 2.194,97.



Cálculo de rendas

Esta actividade cria uma função que permite calcular o custo de um carro comprado a prazo. Para obter informações detalhadas sobre os passos utilizados neste exemplo, consulte o capítulo electrónico Programação disponível no site Web da TI em education.ti.com e no CD deste produto.

Função Time-Value-of- Money

No Editor de Programa, defina a função Time-Value-of-Money (**tvm**), onde **temp1** = número das prestações, **temp2** = taxa de juros anual, **temp3** = valor presente, **temp4** =

prestação mensal, **temp5** = valor futuro e **temp6** = início ou final do período de pagamento (1 = início do mês, 0 = final do mês).

```
:tvm(temp1,temp2,temp3,temp4,temp5,temp6)
:Func
:Local tempi,tempfunc,tempstr1
:⁻temp3+(1+temp2/1200∗temp6)∗temp4∗((1−(1+temp2/1200)^
    (⁻temp1))/(temp2/1200))−temp5∗(1+temp2/1200)^(⁻temp1)
    →tempfunc
:For tempi,1,5,1
:"temp"&exact(string(tempi))→tempstr1
:If when(#tempstr1=0,false,false,true) Then
:If tempi=2
:Return approx(nsolve(tempfunc=0,#tempstr1) | #tempstr1>0
    and #tempstr1<100)
:Return approx(nsolve(tempfunc=0,#tempstr1))
:EndIf
:EndFor
:Return "parameter error"
:EndFunc
```

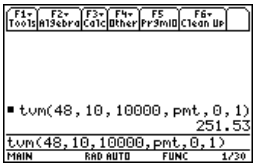
Nota: é possível utilizar o teclado do computador para digitar textos longos e, em seguida, utilizar o software TI Connect™ para os enviar para a TI-89 Titanium / Voyage™ 200.

Cálculo das prestações mensais

Calcule a prestação mensal para comprar um carro que custa 10.000 euros, pagando em 48 vezes com 10% de juros anuais.

No ecrã principal, introduza os valores **tvm** para calcular *pmt*.

Resultado: a prestação mensal é igual a 251,53.

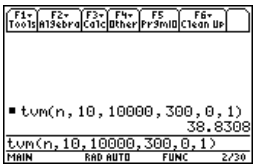


Cálculo do número de prestações

Calcule o número de prestações necessárias para saldar o empréstimo do carro, pagando 300 por mês.

No ecrã principal, introduza os valores **tvm** para calcular *n*.

Resultado: o número de prestações é igual a 38,8308.



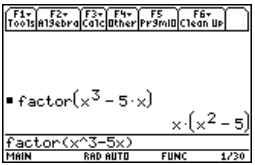
Cálculo com números racionais, reais e complexos

Esta actividade permite calcular expressões e obter números racionais, reais e complexos. Para mais informações sobre o procedimento utilizado neste exemplo, consulte no módulo *Manipulação simbólica*.

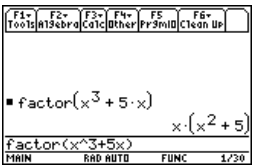
Cálculo

Introduza as expressões seguintes no ecrã principal.

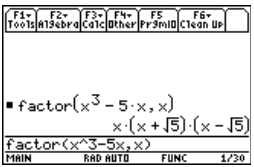
1. **factor(x^3−5x)** [ENTER] mostra um resultado racional.



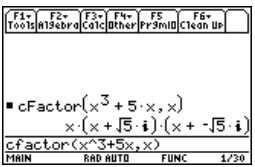
2. **factor(x^3+5x)** [ENTER] mostra um resultado racional.



3. **factor(x^3−5x,x)** [ENTER] mostra um resultado real.



4. **cfactor(x^3+5x,x)** [ENTER] mostra um resultado complexo.



Simulação de uma extracção de amostra sem reposição

Esta actividade simula o resultado obtido com a extracção de bolas de diferentes cores de uma urna sem reposição. Para mais informações sobre o procedimento utilizado neste exemplo, consulte no módulo *Programação*.

Função de extracção de amostra sem reposição

No Editor de Programa, defina **drawball( )** como uma função que pode ser chamada com dois parâmetros. O primeiro parâmetro é uma lista onde cada elemento corresponde ao número de bolas de uma determinada cor. O segundo parâmetro corresponde ao número de bolas que serão seleccionadas. A função devolve uma lista na qual cada elemento corresponde ao número de bolas seleccionadas de cada cor.

```
:drawball(urnlist,drawnum)
:Func
:Local templist,drawlist,colordim,
    numballs,i,pick,urncum,j
:If drawnum>sum(urnlist)
:Return “too few balls”
:dim(urnlist)→colordim
:urnlist→templist
:newlist(colordim)→drawlist
:For i,1,drawnum,1
:sum(templist)→numballs
:rand(numballs)→pick
```

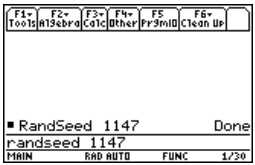
(*continua na próxima coluna*)

```
:For j,1,colordim,1
:cumSum(templist)→urncum
:If pick ≤ urncum[j] Then
:drawlist[j]+1→drawlist[j]
:templist[j]−1→templist[j]
:Exit
:EndIf
:EndFor
:EndFor
:Return drawlist
:EndFunc
```

Extracção de amostra sem reposição

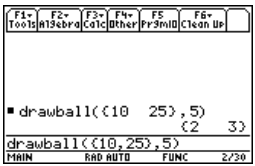
Supondo que cada urna contém *n1* bolas de uma cor, *n2* bolas de uma segunda cor, *n3* bolas de uma terceira cor, etc., simule efectuar a extracção sem reposição de bolas.

1. Introduza um gerador de números aleatórios com o comando **RandSeed**.



2. Supondo que a urna contém 10 bolas vermelhas e 25 brancas, simule a extracção de 5 bolas ao acaso, sem reposição. Introduza **drawball({10,25},5)**.

   Resultado: 2 bolas vermelhas e 3 brancas.



Appendix A: Funções e instruções

Listas de operações catégoricas 834
Listas de operações alfabéticas 838

Esta secção descreve a sintaxe e a acção de cada função e instrução da TI-89 Titanium/Voyage™200 incluída no sistema operativo (OS). Consulte os módulos relacionados com as aplicações de software da calculador (Apps) para obter informações das funções e instruções específicas para essas Apps.

Nome da função ou instrução.

Tecla ou menu para introduzir o nome. Pode também digitar o nome.

Exemplo

Circle **CATALOG**

Circle *x, y, r* [, *drawMode*]

Desenha uma circunferência com o centro nas coordenadas da janela (*x, y*) e com um raio de *r*.

x, y, e *r* têm de ser valores reais..

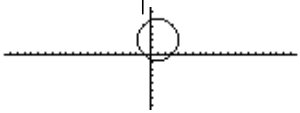
Se o *ModoDraw* = 1, desenha a circunferência (predefinição).
Se o *ModoDraw* = 0, desliga a circunferência.
Se o *ModoDraw* = -1, inverte os pixéis ao longo da circunferência.

Nota: A nova representação gráfica apaga todos os itens desenhados.

Os argumentos aparecem em itálico. Os argumentos entre parêntesis [ ] são opcionais. Não escreva os parêntesis.

Numa janela de visualização ZoomSqr:

ZoomSqr:Circle 1,2,3 ENTER



Explicação da função ou instrução.

Listas de operações catégoricas

Esta secção lista as funções e as instruções da TI-89 Titanium / Voyage™ 200 em grupos funcionais juntamente com os números de página em que são descritas.

Álgebra

("with")	968	cFactor()	843	comDenom()	846
cSolve()	851	cZeros	855	expand()	869
factor()	870	**getDenom()**	876	**getNum()**	877
nSolve()	900	propFrac()	908	**randPoly()**	914
solve()	932	tCollect()	943	**tExpand()**	944
zeros()	950				

Cálculo

integrar,	883	product()	907	(soma)	964
arcLen()	840	avgRC()	841	*d()*	857
deSolve()	859	fMax()	872	fMin()	872
impDif()	**882**	**limit()**	885	nDeriv()	896
nInt()	898	**' (primo)**	966	**seq()**	922
taylor()	943				

Gráficos

AndPic	839	BldData	**842**	Circle	844
ClrDraw	844	ClrGraph	845	CyclePic	854
DrawFunc	864	DrawInv	864	DrawParm	864
DrawPol	864	DrawSlp	864	DrwCtour	865
FnOff	872	FnOn	873	**Graph**	880
Line	886	LineHorz	886	**LineTan**	887
LineVert	887	**NewPic**	897	**PtChg**	908
PtOff	908	**PtOn**	908	**ptTest()**	908
PtText	909	**PxlChg**	909	**PxlCrcl**	909
PxlHorz	909	**PxlLine**	909	**PxlOff**	910
PxlOn	910	**pxlTest()**	910	**PxlText**	910
PxlVert	910	**RclGDB**	914	RclPic	914
RplcPic	919	**Shade**	927	**StoGDB**	938
StoPic	938	**Style**	939	**Trace**	946
XorPic	950	**ZoomBox**	952	**ZoomData**	952
ZoomDec	953	**ZoomFit**	953	**ZoomIn**	953
ZoomInt	954	**ZoomOut**	954	**ZoomPrev**	954
ZoomRcl	954	**ZoomSqr**	955	**ZoomStd**	955
ZoomSto	955	**ZoomTrig**	956		

Listas

+ (adicionar)	956	−(subtrair)	956	**∗(multiplicar)**	957
/ (dividir)	958	(anular)	965	(potência)	958
aumentar()	841	**crossP()**	850	cumSum()	853
dim()	862	dotP()	863	exp▸list()	868
left()	885	**Δlist()**	888	**list▸mat()**	888
listamat()	893	**max()**	893	**mid()**	894
min()	895	**newList()**	897	**polyEval()**	906
product()	907	**right()**	917	**rotate()**	917
shift()	927	**SortA**	935	**SortD**	935
sum()	939				

Math

+ (adicionar)	956	−(subtrair)	956	∗**(multiplicar)**	957
∕ (dividir)	958	(anular)	960	(percentagem)	960
(factorial)	962	(raiz quadrada)	963	(potência)	958
G(gradiano)	964	(grau)	965	(ângulo)	965
°, ', "	**966**	_ **(carácter sublinhado)**	966	▸ **(converter)**	967
10^()	**967**	0b, 0h	**969**	▸**Bin**	841
▸**Cylind**	854	▸**DD**	858	▸**Dec**	858
▸**DMS**	863	▸Grad	**838**	▸**Hex**	880
▸**Polar**	905	▸**Rect**	915	▸**Sphere**	936
abs()	838	and	838	∠ **(ângulo)**	965
approx()	840	ceiling()	842	conj()	846
cos()	847	cosh()**()**	848	cosh()	848
cosh^{-1}()	848	**cot()**	849	**cot()**	849
coth()	849	**coth^{-1}()**	849	**csc()**	850
csc^{-1}()	850	**csch()**	850	cosh^{-1}()	851
E	865	e^*()*	865	exact()	868
▸ln	**888**	▸logbase	**890**	floor()	872
fPart()	874	**gcd()**	875	**imag()**	881
impDif()		**int()**	883	**intDiv()**	883
iPart()	883	**isPrime()**	884	**lcm()**	885
ln()	888	**log()**	890	**max()**	893
min()	895	**mod()**	895	**nCr()**	896
nPr()	900	**PRx()**	902	**P▸Ry()**	902
r(radiano)		**R▸Pθ()**	913	**R▸Pr()**	913
real()	915	**remain()**	916	**rotate()**	917
round()	918	**sec()**	920	root()	**917**
sec()	920	**sech()**	920	**sech^{-1}()**	920
shift()	927	sign()	929	**sinh()**	930
sin^{-1}()	930	**sinh()**	931	**sinh^{-1}()**	931
tan()	941	**tan^{-1}()**	942	**tanh()**	942
tanh^{-1}()	943	**tmpCnv()**	945	**ΔtmpCnv()**	946
x^{-1}	**967**				

Matrices

+ (adicionar)	956	−(subtrair)	956	∗**(multiplicar)**	957
∕ **(dividir)**	958	⁻(negar)	960	.+ **(ponto adição)**	959
.− **(ponto subtracção)**	959	.∗(ponto **multiplicação)**	959	./ **(ponto divisão)**	959
.^ **(ponto potência)**	959	^ **(potência)**	958	aumentar()	**841**
colDim()	845	colNorm()	845	**crossP()**	850
cumSum()	853	**data▸mat**	857	det()	861
diag()	861	dim()	862	dotP()	863
eigVc()	866	eigVl()	866	Fill	871
identity()	881	**list▸mat()**	888	**LU**	892
mat▸data	892	listamat()	893	**max()**	893
mean()	893	**median()**	893	**min()**	895
mRow()	895	**mRowAdd()**	895	**newMat()**	897
norm()	899	**product()**	907	**QR**	911
randMat()	914	**ref()**	916	**rowAdd()**	919
rowDim()	919	**rowNorm()**	919	**rowSwap()**	919
rref()	919	**simult()**	929	**stdDev()**	937
stdDevPop()	937	**subMat()**	939	**sum()**	939
T(transposição)	940	**unitV()**	947	**variance()**	948
x^{-1}	**967**				

Programação

= (igual)	**960**	**≠**	**961**	**<**	**961**
≤	**961**	**>**	**961**	**≥**	**962**
# (acção indirecta)	**964**	**Store**	**938**	**◉ (comentário)**	**968**
and	838	ans()	840	Archive	840
checkTmr()	843	ClockOff	844	ClockOn	844
ClrErr	844	ClrGraph	845	ClrHome	845
ClrIO	845	ClrTable	845	CopyVar	846
CustmOff	853	CustmOn	853	Custom	854
Cycle	854	dayOfWk()	857	**Define**	858
DelFold	859	**DelType**	**859**	DelVar	859
:Dialog	862				
:Disp	862	**DispG**	863	DispHome	863
DispTbl	863	DropDown	865	: Else	866
ElseIf	867	EndCustm	867	EndDlog	867
EndFor	867	EndFunc	867	EndIf	867
EndLoop	867	EndPrgm	867	EndTBar	867
EndTry	867	EndWhile	867	entry()	867
Exec	868	Exit	868	For	873
format()	874	Func	874	**Get**	875
GetCalc	875	**getConfg()**	875	getDate()	876
getDtFmt()	876	getDtStr()	876	**getKey()**	877
getKey()	877	**getMode()**	877	getTime()	877
getTmFmt()	878	getTmStr()	878	getTmZn()	878
getType()	878	**getUnits()**	879	**Goto**	879
If	881	**Input**	882	**InputStr**	882
isClkOn()	883	**Item**	884	**Lbl**	884
isArchiv()	**883**	isLocked **()**	**884**	isVar()	**884**
left()	885	**Local**	889	**Lock**	889
:Loop	891	MoveVar	895	**NewFold**	897
NewProb	898	**not**	899	**ou**	901
Output	902	**part()**	902	**PassErr**	904
Pause	905	**PopUp**	906	**Prgm**	907
Prompt	907	**Rename**	916	**Request**	916
Return	917	**right()**	917	**Send**	921
SendCalc	921	SendChat	921	setDate()	922
setDtFmt()	922	**setFold()**	922	**setGraph()**	922
setMode()	923	**setTable()**	925	setTime()	925
setTmFmt()	925	setTmZn()	926	startTmr()	936
setUnits()	926	**Stop**	938	**Style**	939
switch()	940	**Table**	941	**Text**	944
Then	944	timeCnv()	944	**Title**	945
Toolbar	946	**Try**	947	**Unarchiv**	947
Unlock	948	**when()**	948	**While**	949
xor	949				

Estatística

! **(factorial)**	962	BldData	842	**CubicReg**	853
cumSum()	853	**ExpReg**	870	**LinReg**	887
LnReg	889	**Logistic**	891	**mean()**	893
median()	893	**MedMed**	894	**nCr()**	896
NewData	896	**NewPlot**	898	nPr()	900
OneVar	901	**PlotsOff**	905	**PlotsOn**	905
PowerReg	907	**QuadReg**	911	**QuartReg**	912
rand()	913	**randNorm()**	914	**RandSeed**	914
ShowStat	928	**SinReg**	932	**SortA**	935
SortD	935	**stdDev()**	937	stdDevPop()	**937**
TwoVar	947	**variance()**	948		

Cadeias

& **(acrescentar)**	962	# **(acção indirecta)**	964	char()	843
dim()	862	expr()	870	format()	874
inString()	883	**left()**	885	**mid()**	894
ord()	901	**right()**	917	**rotate()**	917
shift()	927	**string()**	938		

Listas de operações alfabéticas

As operações cujos nomes não sejam alfabéticos (como +, !, e >) são listadas no fim deste anexo que começa na página 956. Excepto se especificado em contrário, todos os exemplos desta secção foram efectuados no modo de reinicialização predefinido, e todas as variáveis são assumidas como sendo indefinidas. Além disso, devido às restrições de formatação, os resultados aproximados são truncados para três casas decimais (3.14159265359 aparece como 3.141...).

abs() — Menu MATH/Number

abs(*expressão 1***)** ⇒ *expressão*
abs(*lista1***)** ⇒ *lista*
abs(*matriz1***)** ⇒ *matriz*

Devolve o valor absoluto do argumento.

Se o argumento for um número complexo, devolve o módulo do número.

Nota: Todas as variáveis indefinidas são tratadas como variáveis reais.

abs({π/2, ⁻π/3}) ENTER $\{\frac{\pi}{2}\ \frac{\pi}{3}\}$

abs(2−3*i*) ENTER $\sqrt{13}$

abs(z) ENTER $|z|$

abs(x+y*i*) ENTER $\sqrt{x^2+y^2}$

and — Menus MATH/Test e MATH/Base

Expressão booleana 1 **e** *expressão 2* ⇒ *Expressão booleana*
Lista booleana 1 **e** *lista 2* ⇒ *Lista booleana*
Matriz booleana 1 **e** *matriz 2* ⇒ *Booleana matriz*

Devolve verdadeiro ou falso, ou uma forma simplificada da entrada original.

x≥3 and x≥4 ENTER x≥4

{x≥3,x≤0} and {x≥4,x≤⁻2} ENTER $\{x \geq 4\ \ x \leq {}^{-}2\}$

---

número inteiro 1 **e** *número inteiro 2* ⇒ *número inteiro*

Compara dois números inteiros reais bit a bit com uma operação **e**. Internamente, ambos os números inteiros são convertidos para números binários de 32 bits. Quando os bits correspondentes forem comparados, o resultado é 1 se ambas as partes forem 1; caso contrário, o resultado é 0.

Pode introduzir os números inteiros em qualquer base do número. Para uma introdução binária ou hexadecimal, tem de utilizar o prefixo 0b ou 0h, respectivamente. Sem um prefixo, os números inteiros são tratados como decimal (base10).

Se introduzir um número inteiro decimal muito grande para uma forma binária de 32 bits, é utilizada uma operação do módulo simétrico para colocar o valor no intervalo adequado.

No modo base Hex:

0h7AC36 and 0h3D5F ENTER 0h2C16

Importante: Zero, não a letra O.

No modo base Hex:

0b100101 and 0b100 ENTER 0b100

No modo base Hex:

37 and 0b100 ENTER 4

Nota: Uma entrada binária pode ter até 32 dígitos (não contar o prefixo 0b). Uma entrada hexadecimal pode ter até 8 dígitos.

AndPic CATALOG

AndPic *picVar*[, *linha, coluna*]

Mostra o ecrã Graph e logicamente "ANDS" a imagem guardada em picVar e o ecrã do gráfico actual nas coordenadas de pixéis (*linha, coluna*).

picVar tem de ser um tipo de imagem.

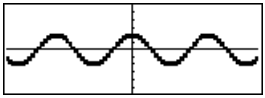
As coordenadas predefinidas são (0,0), que estão no canto superior esquerdo do ecrã.

No modo de representação gráfica da função e no editor Y=:

y1(x) = cos(x) ⊕
[2nd][F6] Style = 3:Square

[F6] Style = 3:Square

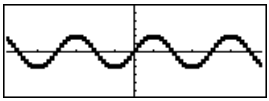
[F2] Zoom = 7:ZoomTrig
[F1] = 2:Save Copy As...
Type = Picture, Variable = PIC1



y2(x) = sin(x)
[2nd][F6] Style = 3:Square

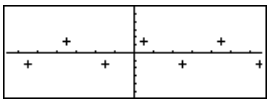
[F6] Style = 3:Square

y1 = no checkmark (F4 to deselect)
[F2] Zoom = 7:ZoomTrig



[HOME]
[◆][CALC HOME]

AndPic PIC1 [ENTER] Done



angle() Menu MATH/Complex

angle(*expressão 1*) ⇒ *expressão*

Devolve o ângulo da *expressão 1*, interpretando a *expressão 1* como um número complexo.

Nota: Todas as variáveis indefinidas são tratadas como variáveis reais.

No modo de ângulo Degree:

angle(0+2*i*) [ENTER] 90

No modo de ângulo Gradian:

angle(0+3*i*) [ENTER] 100

No modo de ângulo Radian::

angle(1+*i*) [ENTER] $\frac{\pi}{4}$

angle(z) [ENTER]
angle(x+ *i*y) [ENTER]

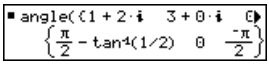
■ angle(z) $\frac{-\pi\cdot(\text{sign}(z)-1)}{2}$
■ angle(x + i · y)
$\frac{\pi\cdot\text{sign}(y)}{2} - \tan^{-1}\left(\frac{x}{y}\right)$

angle(*lista1***)** ⇒ *lista*
angle(*matriz1***)** ⇒ *matriz*

Devolve uma *lista* ou *matriz* de ângulos dos elementos na *lista 1* ou *matriz 1*, interpretando cada elemento como um número complexo que representa um ponto da coordenada rectangular bidimensional.

No modo de ângulo Radian:

angle({1+2i,3+0i,0-4i}) [ENTER]



ans() **Teclas** [2nd] [ANS]

ans( ) ⇒ *valor*
ans(*número inteiro***)** ⇒ *valor*

Devolve uma resposta anterior da área do histórico do ecrã inicial.

número inteiro, se incluído, especifica a resposta anterior a visualizar. Intervalo inteiro para o número inteiro é de 1 a 99 e não pode ser uma expressão. A predefinição é 1, a resposta mais recente.

Para utilizar **ans()** para gerar a sequência de Fibonacci no ecrã inicial, prima:

1 [ENTER]	1
1 [ENTER]	1
[2nd] [ANS] [+] [2nd] [ANS] [◀] [←] 2 [ENTER]	2
[ENTER]	3
[ENTER]	5

approx() **Menu MATH/Algebra**

approx(*expressão***)** ⇒ *valor*

Devolve a avaliação da *expressão* como um valor decimal, quando possível, independentemente do modo Exact/Approx actual.

É equivalente a introduzir *expressão* e a premir [◆] [ENTER] no ecrã inicial.

approx(π) [ENTER] 3.141...

approx(*lista1***)** ⇒ *lista*
approx(*matriz1***)** ⇒ *matriz*

Devolve uma lista ou matriz em que cada elemento foi avaliado para um valor decimal, quando possível.

approx({sin(π),cos(π)}) [ENTER]
{0. -1.}

aprox([(2),(3)]) 1.732...]

Archive **CATALOG**

Archive *var1* [, *var2*] [, *var3*] …

Move as variáveis especificadas da RAM para a memória de arquivo de dados do utilizador.

Pode aceder a uma variável arquivada da mesma forma que acederia a uma variável da RAM. No entanto, não pode eliminar, mudar o nome ou guardar numa variável arquivada porque é bloqueada automaticamente.

Para desarquivar variáveis, utilize **Unarchiv**.

10→arctest [ENTER]	10
Archive arctest [ENTER]	Done
5*arctest [ENTER]	50
15→arctest [ENTER]	

[ESC]	
Unarchiv arctest [ENTER]	Done
15→arctest [ENTER]	15

arcLen() **Menu MATH/Calculus**

arcLen(*expressão 1***,***var***,***início***,***fim***)** ⇒ *expressão*

Devolve o comprimento do arco da *expressão 1* do *início* ao *fim* no que respeita à variável *var*.

Independentemente do modo de gráfico, o comprimento do arco é calculado como um integral assumindo uma definição do modo de função.

arcLen(cos(x),x,0,π) [ENTER] 3.820...

arcLen(f(x),x,a,b) [ENTER]

$$\int_a^b \sqrt{\left(\frac{d}{dx}(f(x))\right)^2+1}\ dx$$

arcLen(*lista1,var,início,fim***)** ⇒ *lista*

Devolve uma lista de comprimentos de arcos de cada elemento da *lista 1* do *início* ao *fim* no que respeita a *var*.

```
arcLen(sin(x),cos(x),x,0,π,)
                    {3.820...  3.820...}
```

aumentar() — Menu MATH/Matrix

aumentar(*lista1, lista2***)** ⇒ *lista*

Devolve uma lista nova que é a *lista 2* acrescentada ao fim da *lista 1*.

```
augment({1,-3,2},{5,4}) ENTER
                    {1 -3 2 5 4}
```

aumentar(*matriz 1, matriz 2***)** ⇒ *matriz*
aumentar(*matriz 1; matriz 2***)** ⇒ *matriz*

Devolve uma matriz nova que é a *matriz 2* acrescentada à *matriz 1*. Quando utilizar o carácter ",", as matrizes têm de ter colunas com as dimensões iguais, e a *matriz 2* é acrescentada à *matriz 1* como novas linhas. Quando utilizar o carácter ";", as matrizes têm de ter colunas com as dimensões iguais, e a *matriz 2* é acrescentada à *matriz 1* como novas linhas. Não altera a *matriz 1* ou a *matriz 2*.

```
[1,2;3,4]→M1 ENTER          [1 2; 3 4]
[5;6]→M2 ENTER              [5; 6]
augment(M1,M2) ENTER        [1 2 5; 3 4 6]
[5,6]→M2 ENTER              [5 6]
augment(M1;M2) ENTER        [1 2; 3 4; 5 6]
```

avgRC() — CATALOG

avgRC(*expressão 1, var* [, *h*]**)** ⇒ *expressão*

Devolve o quociente de diferença progressiva (taxa de câmbio média).

*expressão 1*pode ser o nome da função definida pelo utilizador (ver **Func**).

h é o valor do passo. Se *h* for omitido, é predefinido para 0.001.

Não se esqueça de que a função similar **nDeriv()** utiliza o quociente de diferença central.

```
avgRC(f(x),x,h) ENTER
avgRC(sin(x),x,h)x=2 ENTER
avgRC(x^2-x+2,x) ENTER
                    2.•(x-.4995)
avgRC(x^2-x+2,x,.1) ENTER
                    2.•(x-.45)
avgRC(x^2-x+2,x,3) ENTER    2•(x+1)
```

▸Bin — Menu MATH/Base

número inteiro 1 **▸Bin** ⇒ *número inteiro*

Converte o *número inteiro 1* para um número binário. Os números binários ou hexadecimais têm sempre um prefixo 0b ou 0h, respectivamente.

Zero, não a letra O, seguido por b ou h.
0b *Número binário*
0h *Número hexadecimal*
Um número hexadecimal pode ter até 32. Um número hexadecimal pode ter até 8.

Sem um prefixo, o número *inteiro 1* é tratado como decimal (base10). O resultado aparece em binário, independentemente do modo Base.

Se introduzir um número inteiro decimal muito grande para uma forma binária de 32 bits, é utilizada uma operação do módulo simétrico para colocar o valor no intervalo adequado.

```
256▸Bin ENTER               0b100000000
0h1F▸Bin ENTER              0b11111
```

BldData CATALOG

BldData [*dataVar*]

Cria a variável de dados *dataVar* baseada nas informações utilizadas para desenhar o gráfico actual. **BldData** é válido em todos os modos de gráficos.

Se dataVar for omitido, os dados são guardados na variável do sistema sysData.

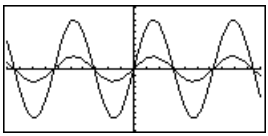
Nota: A primeira vez que iniciar o Editor de Dados/Matriz depois de utilizar **BldData**, dataVar ou sysData (dependendo do argumento utilizado com o **BldData**) é definido como a variável de dados actual.

Os valores incrementais utilizados para quaisquer variáveis independentes (x no exemplo à direita) são calculados de acordo com os valores da variável da janela.

Para mais informações sobre os incrementos utilizados para avaliar um gráfico, consulte o módulo que descreve esse modo de gráfico.

No modo de representação gráfica da função e no modo de ângulo Radian:

8*sin(x)→y1(x) [ENTER] Done
2*sin(x)→y2(x) [ENTER] Done
ZoomStd [ENTER]



[HOME]
[◆] [CALC HOME]

BldData [ENTER] Done
[APPS] 6 [ENTER]

DATA	x	y1	y2
	c1	c2	c3
1	-10.	4.3522	1.088
2	-9.832	3.168	.792
3	-9.664	1.8945	.47363
4	-9.496	.56769	.14192

O modo de representação gráfica 3D tem duas variáveis independentes. Nos dados da amostra à direita, não se esqueça de que x permanece constante à medida que y incrementa através do intervalo de valores.

Em seguida, x aumenta para o próximo valor e y incrementa novamente através do intervalo. Este padrão continua até x ter incrementado através do intervalo.

Nota: Os dados da amostra seguinte pertencem a um gráfico 3D.

DATA	x	y	z1
	c1	c2	c3
1	-10.	-10.	0.
2	-10.	-8.571	5.8309
3	-10.	-7.143	8.9706
4	-10.	-5.714	9.8677

ceiling() Menu MATH/Number

ceiling(*expressão 1*) ⇒ *número inteiro*

Devolve o número inteiro mais próximo que é o argumento.

O argumento pode ser um número complexo ou real.

Nota: Consulte também **floor()**.

ceiling(0.456) [ENTER] 1.

ceiling(*lista1*) ⇒ *lista*
ceiling(*matriz1*) ⇒ *matriz*

Devolve uma lista ou matriz do ponto máximo de cada elemento.

ceiling({-3.1,1,2.5}) [ENTER]
{-3. 1 3.}

ceiling([0,-3.2i;1.3,4]) [ENTER]

$$\begin{bmatrix} 0 & -3.\cdot i \\ 2. & 4 \end{bmatrix}$$

cFactor() Menu MATH/Algebra/Complex

cFactor(*expression1*[**,** *var*]**)** ⇒ *expressão*
cFactor(*lista 1*[*,var*]**)** ⇒ *lista*
cFactor(*matriz 1*[*,var*]**)** ⇒ *matriz*

cfactor(*expressão 1***)** devolve *expressão 1* decomposta em factores no que respeita a todas as variáveis sobre um denominador comum.

expressão 1 é decomposta no maior número possível de factores em relação a factores racionais lineares mesmo que introduza os números não reais novos. Esta alternativa é adequada se quiser a factorização no que respeita a mais de uma variável.

```
cFactor(a^3*x^2+a*x^2+a^3+a) ENTER
        a•(a + ⁻i)•(a + i)•(x + ⁻i)•(x + i)

cFactor(x^2+4/9) ENTER
        (3•x + ⁻2•i)•(3•x + 2• i)
        ---------------------------
                    9

cFactor(x^2+3) ENTER                x² + 3

cFactor(x^2+a) ENTER                x² + a
```

---

cFactor(*expression 1,var***)** devolve a *expressão 1* decomposta em factores no que respeita à variável *var*.

expressão 1 é decomposta no maior número possível de factores em relação a factores que são lineares em *var*, talvez com constantes não reais, mesmo que introduzam constantes irracionais ou subexpressões que sejam irracionais noutras variáveis.

Os factores e os termos são ordenados com var como a variável principal. As potências similares de var são recolhidas em cada factor. Inclua var se a factorização for necessária apenas para essa variável e estiver disposto a aceitar expressões irracionais em quaisquer outras variáveis para aumentar a factorização no que respeita a var. Podem existir alguns factores imprevistos no que respeita a outras variáveis.

Para a definição AUTO do modo Exact/Approx, incluindo var também permite a aproximação de coeficientes de ponto flutuante em que os coeficientes irracionais não podem ser explicitamente expressos em termos de funções integradas. Mesmo quando existir apenas uma variável, incluindo var pode produzir a factorização mais completa.

Nota: Consulte também **cfactor()**.

```
cFactor(a^3*x^2+a*x^2+a^3+a,x) ENTER
        a•(a² + 1)•(x + ⁻ i)•(x + i)

cFactor(x^2+3,x) ENTER
        (x + √3• i)•(x + ⁻√3• i)

cFactor(x^2+a,x) ENTER
        (x + √a• ⁻i)•(x + √a• i)

cFactor(x^5+4x^4+5x^3-6x-3) ENTER
        x⁵ + 4•x⁴ + 5•x³ -6•x-3

cFactor(ans(1),x) ENTER

        (x -.965)•(x +.612)•(x + 2.13)•
                (x + 1.11 - 1.07•i)•
                (x + 1.11 + 1.07•i)
```

char() Menu MATH/String

char(*número inteiro***)** ⇒ *carácter*

Devolve uma cadeia de caracteres com o carácter de *número inteiro* do conjunto de caracteres da TI-89Titanium/Voyage™200. Consulte o Anexo B para obter uma lista de códigos de caracteres completa. O intervalo válido para *número inteiro* é 0–255.

```
char(38) ENTER                      "&"

char(65) ENTER                      "A"
```

checkTmr() CATALOG

checkTmr(*starttime***)** ⇒ *número inteiro*

Devolve um número inteiro que representa o número de segundos decorridos desde o início do temporizador. *starttime* é um número inteiro devolvido pela função **startTmr()**.

Pode também utilizar uma lista ou matriz de números inteiros starttime. Os números inteiros starttime válidos têm de ficar entre 0 e a hora actual do relógio. Pode executar vários temporizadores em simultâneo.

```
startTmr() ENTER                    148083315

checkTmr(148083315)34                    34

startTmr()→Timer1
⋮
startTmr()→Timer2
⋮
checkTmr(Timer1)→Timer1Value
⋮
checkTmr(Timer2)→Timer2Value
```

Nota: Consulte também **startTmr()** e **timeCnv()**.

Circle CATALOG

Circle *x, y, r* [**,** *ModoDraw*]

Desenha uma circunferência com o centro nas coordenadas da janela (x,y) e com um raio de r.

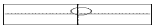
x, y, e *r* têm de ser valores reais.

Se o ModoDraw = 1, desenha a circunferência (predefinição).
Se o ModoDraw = 0, desliga a circunferência.
Se o ModoDraw = -1, inverte os pixéis ao longo da circunferência.

Nota: A nova representação gráfica apaga todos os itens desenhados. Consulte também **PxlCrcl**.

Numa janela de visualização **ZoomSqr**:

```
ZoomSqr:Circle 1,2,3
```



ClockOff CATALOG

ClockOff

Desliga o relógio.

ClockOn CATALOG

ClockOn

Liga o relógio.

ClrDraw CATALOG

ClrDraw

Apaga o ecrã Graph e reinicia a função Smart Graph para que da próxima vez que o ecrã Graph apareça, o gráfico seja redesenhado.

Durante a visualização do ecrã Graph, pode apagar todos os itens desenhados (como linhas e pontos), premindo [F4] (ReGraph) ou premindo [2nd] [F6] e seleccionando 1:ClrDraw.

ClrErr CATALOG

ClrErr

Apaga o estado de erro. Define errornum para zero e apaga todas as variáveis de contexto de erro internas.

A condição **Else** de **Try...EndTry** no programa deve utilizar **ClrErr** ou **PassErr**. Se tiver de processar ou ignorar o erro, utilize **ClrErr**. Se não souber o que fazer com o erro, utilize **PassErr** para o enviar para o processador de erros seguinte. Se não existirem mais processadores de erros **Try...EndTry** pendentes, a caixa de diálogo de erro aparece normalmente.

Nota: Consulte também **PassErr** e **Try**.

Lista de programas:

```
:clearerr()
:Prgm
:PlotsOff:FnOff:ZoomStd
:For i,0,238
:Δx*i+xmin→xcord
: Try
:  : PtOn xcord,ln(xcord) Else
:  : If errornum=800 or
                    errornum=260 Then
:   ClrErr ◉clear the error
:  Else
:   PassErr ◉pass on any other
                                error
:  EndIf
: EndTry
:EndFor
:EndPrgm
```

ClrGraph CATALOG

ClrGraph

Apaga quaisquer funções ou expressões com gráficos desenhados com o comando **Graph** ou criados com o comando **Table**. (Consulte **Graph** ou **Table**.)

Quaisquer funções Y= seleccionadas previamente serão representadas graficamente da próxima vez que o gráfico aparecer.

ClrHome CATALOG

ClrHome

Apaga todos os itens guardados na área do histórico do ecrã inicial **entry()** e **ans()**. Não apaga a alinha de entrada actual.

Durante a visualização do ecrã inicial, pode apagar a área do histórico, premindo F1 e seleccionando 8:Clear Home.

Para funções como **solve()** que devolvem constantes ou números inteiros arbitrários (*@*1, *@* 2, etc.), **ClrHome** reinicia o sufixo para 1.

ClrIO CATALOG

ClrIO

Apaga o ecrã Program I/O.

ClrTable CATALOG

ClrTable

Apaga todos os valores da tabela. Aplica-se apenas à definição ASK na caixa de diálogo Table Setup.

Durante a visualização do ecrã Table no modo Ask , pode apagar os valores, premindo F1 e seleccionando 8:Clear Table.

colDim() Menu MATH/Matrix/Dimensions

colDim(*matriz***)** ⇒ *expressão*

Devolve o número de colunas contidas na matriz.

Nota: Consulte também **rowDim()**.

colNorm() Menu MATH/Matrix/Norms

colNorm(*matriz***)** ⇒ *expressão*

Devolve o máximo de somas dos valores absolutos dos elementos nas colunas na matriz.

Nota: Os elementos da matriz indefinidos não são permitidos. Consulte também **rowNorm()**.

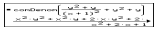
```
colNorm()

[1,-2,3;4,5,-6]→mat ENTER
                    [1 -2 3]
                    [4  5 -6]
colNorm(mat) ENTER        9
```

comDenom() Menu MATH/Algebra

comDenom(*expressão 1*[,*var*]**)** ⇒ *expressão*
comDenom(*lista1*[,*var*]**)** ⇒ *lista*
comDenom(*matriz1*[,*var*]**)** ⇒ *matriz*

```
comDenom((y^2+y)/(x+1)^2+y^2+y)
```



comDenom(*expressão 1***)** devolve um rácio reduzido de um numerador alargado sobre um denominador alargado.

comDenom(*expressão 1,var***)** devolve um rácio reduzido do numerador e do denominador alargados no que respeita a *var*. Os factores e os termos são ordenados com *var* como a variável principal. As potências similares de var são recolhidas. Podem existir alguns factores imprevistos nos coeficientes recolhidos. Quando comparado com a omissão var, este procedimento poupa frequentemente tempo, memória e espaço no ecrã, enquanto torna a expressão mais compreensível. Torna também as operações subsequentes no resultado mais rápidas e faz com que consumam menos memória.

```
comDenom((y^2+y)/(x+1)^2+y^2+y)
```

```
comDenom((y^2+y)/(x+1)^2+y^2+y)
```

Se *var* não ocorrer na *expressão 1*, **comDenom(***expressão 1,var***)** devolve um rácio reduzido de um numerador não alargado sobre um denominador não alargado. Estes resultados poupam ainda mais tempo, memória e espaço no ecrã. Estes resultados parcialmente decompostos em factores tornam também as operações subsequentes mais rápidas e fazem com que consumam menos memória.

```
comDenom(exprn,abc)→comden
(exprn) ENTER                    Done
comDenom((y^2+y)/(x+1)^2+y^2+y)
```

Mesmo quando não haja denominador, a função **comden** é muitas vezes uma forma rápida para alcançar a factorização parcial se **factor()** for muito lento ou consumir a memória.

```
comden(1234x^2*(y^3-y)+2468x
*(y^2-1)) ENTER
        1234·x·(x·y+2)·(y²-1)
```

Sugestão: Introduza esta definição da função **comden()** e tente-a habitualmente como uma alternativa a **comDenom()** e **factor()**.

conj() Menu MATH/Complex

conj(*expressão 1***)** ⇒ *expressão*
conj(*lista1***)** ⇒ *lista*
conj(*matriz1***)** ⇒ *matriz*

```
conj(1+2i) ENTER                 1-2·i
conj([2,1-3i;-i,-7]) ENTER
```

Devolve o número conjugado complexo do argumento.

Nota: Todas as variáveis indefinidas são tratadas como variáveis reais.

$$\begin{bmatrix} 2 & 1+3\cdot i \\ i & -7 \end{bmatrix}$$

```
conj(z)                          z
conj(x+iy)                 x+-i·y
```

CopyVar CATALOG

CopyVar *var1, var2*

Copia o conteúdo da variável *var1* para *var2*. Se *var2* não existir, **CopyVar** cria-a.

Nota: CopyVar é similar à instrução de armazenamento (→) quando estiver a copiar uma expressão, lista, matriz ou cadeia de caracteres, excepto quando não houver nenhuma simplificação quando utilizar **CopyVar**. Tem de utilizar **CopyVar** com tipos de variáveis não algébricas, como variáveis Pic e GDB.

```
x+y→a ENTER                      x+y
10→x ENTER                        10
CopyVar a,b ENTER               Done
a→c ENTER                       y+10
DelVar x ENTER                  Done
b ENTER                          x+y
c ENTER                         y+10
```

cos(*expressão 1***)** ⇒ *expressão*
cos(*lista 1***)** ⇒ *lista*

cos(*expressão 1***)** devolve o co-seno do argumento como uma expressão.

cos(*lista 1***)** devolve uma lista dos co-senos de todos os elementos da *lista 1*.

Nota: O argumento é interpretado como um ângulo em graus, grados ou radianos, de acordo com a definição do modo de ângulo actual. Pode utilizar ° , ^{G} o u ^{r} para substituir o modo de ângulo temporariamente.

No modo de ângulo Degree:

cos((π/4)^{r}) ENTER $\frac{\sqrt{2}}{2}$

cos(45) ENTER $\frac{\sqrt{2}}{2}$

cos({0,60,90}) ENTER {1 1/2 0}

No modo de ângulo Gradian:

cos({0,50,100}) ENTER $\{1\ \frac{\sqrt{2}}{2}\ 0\}$

No modo de ângulo Radian:

cos(π/4) ENTER $\frac{\sqrt{2}}{2}$

cos(45°) ENTER $\frac{\sqrt{2}}{2}$

---

cos(*matriz quadrada 1***)** ⇒ *matriz quadrada*

Devolve o co-seno da matriz da matriz quadrada 1. Este procedimento não é a mesma coisa que calcular o co-seno de cada elemento.

Quando uma função escalar f(A) operar na matriz quadrada 1 (A), o resultado é calculado pelo algoritmo:

1. Calcule os valores próprios (λ_i) e os valores próprios (V_i) de A.

matriz quadrada 1 tem de estar na diagonal. Também não pode ter variáveis simbólicas que não tenham um valor atribuído.

2. Formato das matrizes:

$$B = \begin{bmatrix} \lambda_1 & 0 & \dots & 0 \\ 0 & \lambda_2 & \dots & 0 \\ 0 & 0 & \dots & 0 \\ 0 & 0 & \dots & \lambda_n \end{bmatrix} \text{ e } X = [V_1, V_2, \dots, V_n]$$

3. Em seguida A = X B X^{-1} e f(A) = X f(B) X^{-1}. Por exemplo, cos(A) = X cos(B) X^{-1} em que:

$$\cos(B) = \begin{bmatrix} \cos(\lambda_1) & 0 & \dots & 0 \\ 0 & \cos(\lambda_2) & \dots & 0 \\ 0 & 0 & \dots & 0 \\ 0 & 0 & \dots & \cos(\lambda_n) \end{bmatrix}$$

Todos os cálculos são efectuados com a aritmética de ponto flutuante.

No modo de ângulo Degree:

cos

$$\begin{bmatrix} .212\dots & .205\dots & .121\dots \\ .160\dots & .259\dots & .037\dots \\ .248\dots & -.090\dots & .218\dots \end{bmatrix}$$

cosh() Teclas ◆ [COS-1]

cos-1 (*expressão 1***)** ⇒ *expressão*
cos-1 (*lista1***)** ⇒ *lista*

cos-1 (*expressão 1***)** devolve o ângulo cujo co-seno é *expressão 1* como uma expressão.

cos-1 (*lista 1*) devolve uma lista dos co-senos hiperbólicos de cada elemento da *lista 1*.

Nota: O resultado é devolvido como um ângulo em graus, grados ou radianos, de acordo com a definição do modo de ângulo actual.

No modo de ângulo Degree:

cos-1(1) ENTER 0

No modo de ângulo Gradian:

cos-1(0) ENTER 100

No modo de ângulo Radian:

cos-1({0,.2,.5}) ENTER

$\{\frac{\pi}{2}$ 1.369... 1.047...}

---

cos-1(*matriz quadrada 1***)** ⇒ *matriz quadrada*

Devolve o co-seno hiperbólico da matriz de *matriz quadrada 1*. Este procedimento não éa mesma coisa que calcular o co-seno inverso de cada elemento. Para mais informações sobre o método de cálculo, consulte **cos()**.

matriz quadrada 1 tem de estar na diagonal. O resultado contém sempre números de pontos flutuantes.

No modo de ângulo Radian e no modo do formato complexo Rectangular:

cos-1([1,5,3;4,2,1;6,-2,1]) ENTER

$$\begin{bmatrix} 1.734...+.064...\cdot i & -1.490...+2.105...\cdot i & ... \\ -.725...+1.515...\cdot i & .623...+.778...\cdot i & ... \\ -2.083...+2.632...\cdot i & 1.790...-1.271...\cdot i & ... \end{bmatrix}$$

cosh() Menu MATH/Hyperbolic

cosh(*expressão 1***)** ⇒ *expressão*
cosh(*lista1***)** ⇒ *lista*

cosh (*expressão 1***)** devolve o co-seno hiperbólico do argumento como uma expressão.

cosh (*lista 1***)** devolve uma lista dos co-senos hiperbólicos de cada elemento da lista 1.

cosh(1.2) ENTER 1.810...

cosh({0,1.2}) ENTER {1 1.810...}

---

cosh(*matriz quadrada 1***)** ⇒ *matriz quadrada*

Devolve o co-seno hiperbólico da matriz de *matriz quadrada 1*. Este procedimento não é a mesma coisa que calcular o co-seno hiperbólico de cada elemento. Para mais informações sobre o método de cálculo, consulte **cos()**.

*matriz quadrada 1*tem de estar na diagonal. O resultado contém sempre números de pontos flutuantes.

No modo de ângulo Radian:

cosh([1,5,3;4,2,1;6,-2,1]) ENTER

$$\begin{bmatrix} 421.255 & 253.909 & 216.905 \\ 327.635 & 255.301 & 202.958 \\ 226.297 & 216.623 & 167.628 \end{bmatrix}$$

cosh-1() Menu MATH/Hyperbolic

cosh-1 (*expressão 1***)** ⇒ *expressão*
cosh-1 (*lista1***)** ⇒ *lista*

cosh-1(*expressão 1*) devolve o co-seno hiperbólico do argumento como uma expressão.

cosh-1(*lista 1*) devolve uma lista dos co-senos hiperbólicos de cada elemento da lista 1.

cosh-1(1) ENTER 0

cosh-1({1,2.1,3}) ENTER

{0 1.372... cosh-1(3)}

cosh⁻¹(*matriz quadrada 1***)** ⇒ *matriz quadrada*

Devolve o co-seno hiperbólico da matriz de *matriz quadrada 1*. Este procedimento *não* é a mesma coisa que calcular o co-seno hiperbólico inverso de cada elemento. Para mais informações sobre o método de cálculo, consulte **cos()**.

*matriz quadrada 1*tem de estar na diagonal. O resultado contém sempre números de pontos flutuantes.

No modo de ângulo Radian e no modo do formato complexo Rectangular:

cosh⁻¹([1,5,3;4,2,1;6,⁻2,1]) ENTER

$$\begin{bmatrix} 2.525...+1.734...\cdot i & -.009...-1.490...\cdot i & ... \\ .486...-.725...\cdot i & 1.662...+.623...\cdot i & ... \\ -.322...-2.083...\cdot i & 1.267...+1.790...\cdot i & ... \end{bmatrix}$$

cot() Menu MATH/String

cot(*expressão 1***)** ⇒ *expressão*
cot(*lista 1***)** ⇒ *lista*

Devolve a cotangente de expressão 1 ou devolve uma lista de cotangentes de todos os elementos de *lista 1*.

Nota: O resultado é devolvido como um ângulo em graus, grados ou radianos, de acordo com a definição do modo de ângulo actual.

No modo de ângulo Degree:

cot(45) ENTER 1

No modo de ângulo Gradian:

cot(50) ENTER 1

No modo de ângulo Radian:

cot({1,2.1,3}) ENTER

$$\left\{\frac{1}{\tan(1)}\ -.584...\ \frac{1}{\tan(3)}\right\}$$

cot⁻¹() Menu MATH/String

cot⁻¹(*expressão 1***)** ⇒ *expressão*
cot⁻¹(*lista 1***)** ⇒ *lista*

Devolve o ângulo cuja cotangente é *expressão 1* ou devolve uma lista que contém as cotangentes inversas de cada elemento de *lista 1*.

Nota: O resultado é devolvido como um ângulo em graus, grados ou radianos, de acordo com a definição do modo de ângulo actual.

No modo de ângulo Degree:

cot⁻¹(1) ENTER 45

No modo de ângulo Degree:

cot⁻¹(1) ENTER 50

No modo de ângulo Degree:

cot⁻¹(1) ENTER $\frac{\pi}{4}$

coth() Menu MATH/Hyperbolic

coth(*expressão 1***)** ⇒ *expressão*
cot(*lista 1***)** ⇒ *lista*

Devolve a cotangente hiperbólica de *expressão 1* ou devolve uma lista de cotangentes hiperbólicas de todos os elementos de *lista 1*.

coth(1.2) ENTER 1.199...

coth({1,3.2}) ENTER

$$\left\{\frac{1}{\tanh(1)}\ 1.003...\right\}$$

coth1() Menu MATH/Hyperbolic

coth⁻¹(*expressão 1***)** ⇒ *expressão*
coth⁻¹(*lista 1***)** ⇒ *lista*

Devolve a cotangente hiperbólica inversa de *expressão 1* ou devolve uma lista que contém as cotangentes de cada elemento de *lista 1*.

coth⁻¹(3.5) ENTER .293...

coth⁻¹({-2,2.1,6}) ENTER

$$\left\{\frac{-\ln(3)}{2}\ .518...\ \frac{\ln(7/5)}{2}\right\}$$

crossP() Menu MATH/Matrix/Vector

crossP(*lista 1***,** *lista 2***)** ⇒ *lista*

Devolve o produto cruzado de *lista 1* e *lista 2* como uma lista.

lista 1 e *lista 2* têm de ter igual dimensão e esta tem de ser 2 ou 3.

```
crossP({a1,b1},{a2,b2}) ENTER
                    {0 0 a1•b2−a2•b1}
crossP({0.1,2.2,⁻5},{1,⁻.5,0}) ENTER
                    {⁻2.5 ⁻5. ⁻2.25}
```

crossP(*vector1***,** *vector2***)** ⇒ *vector*

Devolve o vector de uma coluna ou linha (dependendo dos argumentos) que é um produto cruzado de *vector 1* e *vector 2*.

O *vector 1* e o *vector 2* têm de ser vectores de linhas ou ambos têm de ser vectores de colunas. Ambos os vectores têm de ter igual dimensão e esta tem de ser 2 ou 3.

```
crossP([1,2,3],[4,5,6]) ENTER
                    [⁻3 6 ⁻3]
crossP([1,2],[3,4]) ENTER
                    [0 0 ⁻2]
```

csc() Menu MATH/String

csc(*expressão 1***)** ⇒ *expressão*
csc(*lista 1***)** ⇒ *lista*

Devolve a cosecante de *expressão 1* ou devolve uma lista que contém as cosecantes de todos os elementos de *lista 1*.

No modo de ângulo Degree:

csc(45) ENTER $\sqrt{2}$

No modo de ângulo Gradian:

csc(50) ENTER $\sqrt{2}$

No modo de ângulo Radian:

csc({1,π/2,π/3}) ENTER

$$\left\{ \frac{1}{\sin(1)} \quad 1 \quad \frac{2\cdot\sqrt{3}}{3} \right\}$$

csc1() Menu MATH/String

csc⁻¹(*expressão 1***)** ⇒ *expressão*
csc⁻¹(*lista 1***)** ⇒ *lista*

Devolve o ângulo cuja cosecante é *expressão 1 ou devolve uma lista que contém as* cosecantes inversas de cada elemento de *lista 1*.

Nota: O resultado é devolvido como um ângulo em graus, grados ou radianos, de acordo com a definição do modo de ângulo actual.

No modo de ângulo Degree:

csc⁻¹(1) ENTER 90

No modo de ângulo Gradian:

csc⁻¹(1) ENTER 100

No modo de ângulo Radian:

csc⁻¹({1,4,6}) ENTER

$$\left\{ \frac{\pi}{2} \quad \sin^{-1}(1/4) \quad \sin^{-1}(1/6) \right\}$$

csch() Menu MATH/Hyperbolic

csch(*expressão 1***)** ⇒ *expressão*
csch(*lista 1***)** ⇒ *lista*

Devolve a cosecante hiperbólica de *expressão 1* ou devolve uma lista de cosecantes hiperbólicas de todos os elementos de *lista 1*.

csch(3) ENTER $\frac{1}{\sinh(3)}$

csch({1,2.1,4}) ENTER

$$\left\{ \frac{1}{\sinh(1)} \quad .248\ldots \quad \frac{1}{\sinh(4)} \right\}$$

csch^{-1}() Menu MATH/Hyperbolic

csch^{-1}(*expressão 1***)** ⇒ *expressão*
csch^{-1}(*lista 1***)** ⇒ *lista*

Devolve a cosecante hiperbólica inversa de *expressão 1* ou devolve uma lista que contém as cosecantes hiperbólicas inversas de cada elemento de *lista 1*.

csch^{-1}(1) ENTER sinh^{-1}(1)

csch^{-1}({1,2.1,3}) ENTER

sinh1(1) .459… sinh1(1/3)

cSolve() Menu MATH/Algebra/Complex

cSolve(*equação***,** *var***)** ⇒ *Expressão booleana*

Devolve soluções complexas candidatas de uma equação para *var*. O objectivo é produzir candidatos para todas as soluções reais e não reais. Mesmo que *equação* seja real, **cSolve(**) permite resultados não reais no modo real.

Apesar de a TI-89 Titanium/Voyage™ 200 processar todas as variáveis indefinidas que não terminem com um carácter sublinhado (_) como se fosse real, **cSolve()** pode resolver equações polinomiais para soluções complexas.

cSolve(x^3=⁻1,x) ENTER
solve(x^3=⁻1,x) ENTER

cSolve() define temporariamente o domínio para o complexo durante a solução mesmo que o domínio actual seja real. No domínio complexo, as potências fraccionárias com denominadores ímpares utilizam o principal em vez da secção real. Consequentemente, as soluções de **solve()** para equações que envolvam essas potências fraccionárias não são necessariamente um subconjunto dessas do **cSolve()**.

cSolve(x^(1/3)=⁻1,x) ENTER false

solve(x^(1/3)=⁻1,x) ENTER x = ⁻1

cSolve() começa com métodos simbólicos exactos. Excepto no modo EXACT, **cSolve()** utiliza também factores polinomiais complexos aproximados repetitivos, se for necessário.

Modo Display Digits em Fix 2:

exact(cSolve(x^5+4x^4+5x ^3-6x-3=0,x)) ENTER

Nota: Consulte também **cZeros()**, **solve()**, e **zeros()**.

cSolve(ans(1),x) ENTER

Nota: Se a *equação* for não polinomial com funções como **abs()**, **angle()**, **conj()**, **real()**, ou **imag()**, deve colocar um carácter sublinhado (▼ [_]) no fim de *var*. Por predefinição, uma variável é tratada como um valor real.
Se utilizar *var_* , a variável é tratada como complexa.

z é tratado como real:

cSolve(conj(z)=1+ *i*,z) ENTER z=1+ *i*

Deve também utilizar *var_* para quaisquer outras variáveis em *equação* que possam ter valores não reais. Caso contrário, pode receber resultados imprevistos.

z_ é tratado como complexo:

cSolve(conj(z_)=1+ *i*,z_) ENTER z_=1− *i*

---

cSolve(*equação 1* **and** *equação 2* [**and** …],
{*varOrGuess1*, *varOrGuess2* [, …]})
⇒ *Expressão booleana*

Devolve as soluções complexas candidatas para equações algébricas simultâneas, em que cada *varOrGuess* especifica uma variável que pretenda resolver.

Optionalmente, pode especificar uma hipótese inicial para uma variável. Cada *varOrGuess* tem de

ter o formato:

variável
– ou –
variável = número real ou não real

Por exemplo, x é válido e, por isso, é x=3+*i*.

Se todas as equações forem polinomiais e não especificar quaisquer hipóteses iniciais, **cSolve()** utilize o método de eliminação Gröbner/Buchberger lexical para tentar determinar **todas** as soluções complexas.

Nota: Os exemplos seguintes utilizam um carácter sublinhado ([◆] [_]) para que todas as variáveis sejam tratadas como complexas.

As soluções podem incluir soluções reais e não reais, como no exemplo à direita.

```
cSolve(u_*v_-u_=v_ and
v_^2=-u_,{u_,v_}) [ENTER]
```

$$u\_=1/2+\frac{\sqrt{3}}{2}\cdot i \text{ and } v\_=1/2-\frac{\sqrt{3}}{2}\cdot i$$
$$\text{or } u\_=1/2-\frac{\sqrt{3}}{2}\cdot i \text{ and } v\_=1/2+\frac{\sqrt{3}}{2}\cdot i$$
$$\text{or } u\_=0 \text{ and } v\_=0$$

As equações *polinomiais* simultâneas podem ter variáveis adicionais sem valores, mas representam valores numéricos determinados que podem ser substituídos posteriormente.

```
cSolve(u_*v_-u_=c_*v_ and
v_^2=-u_,{u_,v_}) [ENTER]
```

$$u\_=\frac{-(\sqrt{1-4\cdot c\_}+1)^2}{4} \text{ and } v\_=\frac{\sqrt{1-4\cdot c\_}+1}{2}$$

or

$$u\_= \quad \text{and } v\_=\frac{-(\sqrt{1-4\cdot c\_}-1)}{2}$$
$$\text{or } u\_=0 \text{ and } v\_=0$$

Pode também incluir variáveis de solução que não aparecem nas equações. Estas soluções mostram como as famílias de soluções podem conter constantes arbitrárias da forma @*k*, em que *k* é um sufixo do número inteiro de 1 a 255. O sufixo reinicia-se para 1 quando utilizar **ClrHome** ou [F1] 8:Clear Home.

```
cSolve(u_*v_-u_=v_ and
v_^2=-u_,{u_,v_,w_}) [ENTER]
```

$$u\_=1/2+\frac{\sqrt{3}}{2}\cdot i \text{ and } v\_=1/2-\frac{\sqrt{3}}{2}\cdot i$$
$$\text{and } w\_=@1$$

or

$$u\_=1/2-\frac{\sqrt{3}}{2}\cdot i \text{ and } v\_=1/2+\frac{\sqrt{3}}{2}\cdot i$$
$$\text{and } w\_=@1$$
$$\text{or } u\_=0 \text{ and } v\_=0 \text{ and } w\_=@1$$

Para sistemas polinomiais, o tempo de cálculo e o consumo da memória podem depender fortemente da ordem em que lista as variáveis de solução. Se a escolha inicial consumir a memória ou a sua paciência, tente redispor as variáveis nas equações e/ou na lista *varOrGuess*.

Se não incluir quaisquer hipóteses e se qualquer equação for não polinomial em qualquer variável, mas todas as equações forem lineares em todas as variáveis de solução, **cSolve()** utiliza a eliminação Gaussian para tentar determinar todas as soluções.

```
cSolve(u_+v_=e^(w_) and u_-v_=i,
{u_,v_}) [ENTER]
```

$$u\_=\frac{e^{w\_}}{2}+1/2\cdot i \text{ and } v\_=\frac{e^{w\_}-i}{2}$$

Se um sistema não for polinomial em todas variávies nem linear nas variáveis de solução, **cSolve()** determina a solução mais pretendida com um método de repetição aproximada. Para o fazer, o número de variáveis de solução tem de ser igual ao múmero de equações, e todas as outras variáveis das equações têm de ser simplificadas para números.

```
cSolve(e^(z_)=w_ and w_=z_^2,
{w_,z_}) [ENTER]
```

$$w\_=.494\ldots \text{ and } z\_=-.703\ldots$$

Uma hipótese não real é muitas vezes necessária para determinar uma solução não real. Para convergência, uma hipótese pode ter de ser próxima de uma solução.

```
cSolve(e^(z_)=w_ and w_=z_^2,
{w_,z_=1+i}) [ENTER]
```

$$w\_=.149\ldots+4.891\ldots\cdot i \text{ and}$$
$$z\_=1.588\ldots+1.540\ldots\cdot i$$

CubicReg — Menu MATH/Statistics/Regressions

CubicReg *lista 1*, *lista 2*[, [*lista 3*] [, *lista 4*, *lista 5*]]

Calcula a regressão polinomial cúbica e actualiza todas as variáveis estatísticas.

Todas as listas têm de ter dimensões iguais, excepto a lista 4.

lista 1 representa xlist.
lista 2 representa ylist.
lista 3 representa a frequência.
lista 4 representa os códigos de categorias.
lista 5 representa representa a lista de inclusão de categorias.

Nota: De *lista 1* têm de ser o nome de uma variável ou c1–c99 colunas na última variável de dados apresentada no Editor de Dados/Matriz). *Lista 5* não tem de ser o nome de uma variável e não pode ser c1–c99.

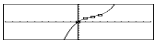
No modo de representação gráfica da função.

```
{0,1,2,3}→L1 [ENTER]          {0 1 2 3}
{0,2,3,4}→L2 [ENTER]          {0 2 3 4}
CubicReg L1,L2 [ENTER]            Done
ShowStat [ENTER]
```

[ENTER]

```
regeq(x)→y1(x) [ENTER]            Done
NewPlot 1,1,L1,L2 [ENTER]         Done
```

[◆][GRAPH]



cumSum() — Menu MATH/List

cumSum(*lista 1***)** ⇒ *lista*

Devolve uma lista das somas acumulativas dos elementos em *lista 1*, começando no elemento1.

```
cumSum({1,2,3,4}) [ENTER]
                    {1 3 6 10}
```

cumSum(*matriz 1***)** ⇒ *matriz*

Devolve uma matriz das somas acumulativas dos elementos em *matriz 1*. Cada elemento é a soma acumulativa da coluna completa.

```
[1,2;3,4;5,6]→m1 [ENTER]
```

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix}$$

```
cumSum(m1) [ENTER]
```

$$\begin{bmatrix} 1 & 2 \\ 4 & 6 \\ 9 & 12 \end{bmatrix}$$

CustmOff — CATALOG

CustmOff

Remove uma barra de ferramentas personalizada.

CustmOn e **CustmOff** activam um programa para controlar uma barra de ferramentas personalizada. Manualmente, pode premir [2nd] [CUSTOM] para activar e desactivar a barra de ferramentas personalizada. A barra de ferramentas personalizada também é removida automaticamente quando mudar as aplicações.

Consulte o exemplo da lista de programas **Custom**.

CustmOn — CATALOG

CustmOn

Activa uma barra de ferramentas personalizada que já tenha sido configurada num bloco **Custom...EndCustm**.

CustmOn e **CustmOff** activam um programa para controlar uma barra de ferramentas personalizada. Manualmente, pode premir [2nd] [CUSTOM] para activar e desactivar a barra de ferramentas personalizada.

Consulte o exemplo da lista de programas **Custom**.

Custom — Teclas 2nd [CUSTOM]

Custom
bloco
EndCustm

Configura uma barra de ferramentas que é activada quando premir 2nd [CUSTOM]. É muito similar à instrução **ToolBar** excepto que as instruções Title e Item não podem ter etiquetas.

bloco pode ser uma instrução individual ou uma série de instruções separadas com o carácter ":" .

Nota: 2nd [CUSTOM] acta como uma comutação. A primeira acção invoca o menu e a segunda acção remove o menu. O menu também é removido quando mudar as aplicações.

Lista de programas:

```
:Test()
:Prgm
:Custom
:Title                    "Lists"
:Item                     "List1"
:Item                    "Scores"
:Item                        "L3"
:Title                "Fractions"
:Item                      "f(x)"
:Item                      "h(x)"
:Title                    "Graph"
:EndCustm
:EndPrgm
```

Cycle — CATALOG

Cycle

Transfere o controlo do programa imediatamente para a próxima repetição da espiral actual (**For, While** ou **Loop**).

Cycle não é permitido fora das três estruturas de espiral (**For**, **While**, ou **Loop**).

Lista de programas:

```
:◉ Sum the integers from 1 to
                    100 skipping 50.
:0→temp
:For i,1,100,1
:If i=50
:Cycle
:temp+i→temp
:EndFor
:Disp temp
```

Contents of **temp** after execution: **5000**

CyclePic — CATALOG

CyclePic *picNameString*, *n* [, [*aguardar*] , [*ciclos*], [direcção]]

Mostra todas as variáveis PIC especificadas e no intervalo especificado. O utilizador tem o controlo opcional sobre o tempo entre imagens, o número de vezes para ver as imagens e a direcção , circular ou progressiva e regressiva.

direcção é 1 para circular ou ⁻1 para progressiva e regressiva. Predefinição = 1.

1. Guarde as três imagens chamadas pic1, pic2, e pic3.
2. Introduza: CyclePic "pic",3,.5,4,⁻1
3. As três imagens (3) aparecerão — automaticamente com meio segundo (.5) entre as imagens durante quatro ciclos (4), e progressiva e regressiva (⁻1).

▸Cylind — Menu MATH/Matrix/Vector

vector▸**Cylind**

Mostra o vector da linha ou coluna em forma cilíndrica [r, z].

vector tem de ter exactamente três elementos. Pode ser uma coluna ou linha.

[2,2,3] ▸Cylind ENTER

$[2\cdot\sqrt{2}\ \angle\frac{\pi}{4}\ \ 3]$

cZeros(*expressão***,** *var***)** ⇒ *lista*

Devolve uma lista de valores reais e não reais candidatos de *var* que tornam a *expressão*=0. **cZeros()** efectua isto através do cálculo **exp▶list(cSolve(***expressão*=0,*var***),** *var***)**. Caso contrário, **cZeros()** é similar a **zeros()**.

Nota: Consulte também **cSolve()**, **solve()**, e **zeros()**.

Nota: Se a *expressão* for não polinomial com funções como **abs()**, **angle()**, **conj()**, **real()**, ou **imag()**, deve colocar um carácter sublinhado (◆ [_]) no fim da *var*. Por predefinição, uma variável é tratada como um valor real. Se utilizar *var_* , a variável é tratada como complexa.

Deve também utilizar *var_* para quaisquer outras variáveis na *expressão* que possam ter valores não reais. Caso contrário, pode receber resultados imprevistos.

Modo Display Digits em **Fix 3**:

```
cZeros(x^5+4x^4+5x^3-6x-3,x) ENTER
{-2.125  -.612  .965
 -1.114-1.073·i
 -1.114+1.073·i}
```

z é tratado como real:

```
cZeros(conj(z)-1-i,z) ENTER
{1+i}
```

z_ é tratado como complexo:

```
cZeros(conj(z_)-1-i,z_) ENTER
{1-i}
```

---

cZeros({expressão 1, expressão 2 [**,** …]**},**
{*varOrGuess1***,** *varOrGuess2* [**,** …]**})** ⇒ *matrix*

Devolve as posições candidatas em que as expressões sejam zero simultaneamente. Cada *varOrGuess* especifica uma desconhecida cujo valor procura.

Optionalmente, pode especificar uma hipótese inicial para uma variável. Cada *varOrGuess* tem de ter o formato:

variável
– ou –
variável= número real ou não real

Por exemplo, x é válido e, por isso, é x=3+*i*.

Se todas as expressões forem polinomiais e não especificar quaisquer hipóteses iniciais, **cZeros()** utilize o método de eliminação Gröbner/Buchberger lexical para tentar determinar **todos** os zeros complexos.

Os zeros complexos podem incluir zeros reais ou não reais, como no exemplo à direita.

Cada linha da matriz resultante representa um zero alternativo, com os componentes ordenados como na lista *varOrGuess*. Para extrair uma linha, indexe a matriz por [*linha*].

Nota: Os exemplos seguintes utilizam um carácter sublinhado _ (▦ ◆ [_], ⌨ 2nd [_]) para que todas as variáveis sejam tratadas como complexas.

```
cZeros({u_*v_-u_-v_.v_^2+u_},
{u_.v_}) ENTER
```

$$\begin{bmatrix} 1/2-\frac{\sqrt{3}}{2}\cdot i & 1/2+\frac{\sqrt{3}}{2}\cdot i \\ 1/2+\frac{\sqrt{3}}{2}\cdot i & 1/2-\frac{\sqrt{3}}{2}\cdot i \\ 0 & 0 \end{bmatrix}$$

Extraia a linha 2:

```
ans(1)z=52
```

Os polinómios simultâneos podem ter variáveis adicionais que não tenham valores, mas representem valores numéricos dados que podem ser substituídos posteriormente.

```
cZeros({u_*v_-u_-(c_*v_),
v_^2+u_},{u_,v_}) ENTER
```

$$\begin{bmatrix} \frac{-(\sqrt{1-4\cdot c\_}+1)^2}{4} & \frac{\sqrt{1-4\cdot c\_}+1}{2} \\ \frac{-(\sqrt{1-4\cdot c\_}-1)^2}{4} & \frac{-(\sqrt{1-4\cdot c\_}-1)}{2} \\ 0 & 0 \end{bmatrix}$$

Pode também incluir variáveis desconhecidas que não aparecem nas expressões. Estes zeros mostram como as famílias de zeros podem conter constantes arbitrárias da forma @*k*, em que *K é* um sufixo do número inteiro de 1 a 255. O sufixo reinicia-se para 1 quando utilizar **ClrHome** ou F1 8:Clear Home.

```
cZeros({u_*v_-u_-v_,v_^2+u_},
{u_,v_,w_}) ENTER
```

$$\begin{bmatrix} 1/2-\frac{\sqrt{3}}{2}\cdot i & 1/2+\frac{\sqrt{3}}{2}\cdot i & @1 \\ 1/2+\frac{\sqrt{3}}{2}\cdot i & 1/2-\frac{\sqrt{3}}{2}\cdot i & @1 \\ 0 & 0 & @1 \end{bmatrix}$$

Para sistemas polinomiais, o tempo de cálculo ou o consumo da memória podem depender muito da ordem em que lista os desconhecidos. Se a escolha inicial consumir a memória ou a sua paciência, tente redispor as variáveis nas expressões e/ou na lista *varOrGuess*.

Se não incluir quaisquer hipóteses e se qualquer expressão for não polinomial em qualquer variável, mas todas as expressões forem lineares em todos os desconhecidos, **cZeros()** utiliza a eliminação Gaussian para tentar determinar todos os zeros.

```
cZeros({u_+v_-e^(w_),u_-v_-i},
{u_,v_}) ENTER
```

$$\begin{bmatrix} \frac{e^{w\_}}{2}+1/2\cdot i & \frac{e^{w\_}-i}{2} \end{bmatrix}$$

Se um sistema não for polinomial em todas variávies nem linear nos deconhecidos, **cZeros()** determina o zero mais pretendido com um método de repetição aproximada. Para o fazer, o número de desconhecidos tem de ser igual ao número de expressões e todas as outras variáveis nas expressões têm de ser simplificadas para números.

```
cZeros({e^(z_)-w_,w_-z_^2}, {w_,z_})
ENTER
```

$$[.494\ldots \quad -.703\ldots]$$

Uma hipótese não real é muitas vezes necessária para determinar um zero não real. Para convergência, uma hipótese pode ter de ser próxima de um zero.

```
cZeros({e^(z_)-w_,w_-z_^2},
{w_,z_=1+ i}) ENTER
```

d() Teclas 2nd [*d*] ou menu MATH/Calculus

d(*expressão 1*, *var* [,*ordem*]) ⇒ *expressão*
d(*lista 1*,*var* [,*ordem*]) ⇒ *lista*
d(*matriz 1*,*var* [,*ordem*]) ⇒ *matriz*

Devolve a primeira derivada de *expressão 1* no que respeita à var. *expressão 1* pode ser uma lista ou matriz.

ordem, se incluída, tem de ser um número inteiro. Se a ordem for inferior a zero, o resultado será uma antiderivada.

***d*()** não segue o mecanismo de avaliação normal de simplificação completa dos argumentos e de aplicação da definição da função para estes argumentos simplificados. Em vez disso, ***d*()** efectua os seguintes passos:

1. Simplifique apenas o segundo argumento até ao ponto em que não conduza para uma não variável.
2. Simplifique o primeiro argumento até ao ponto em que rechame qualquer valor guardado para a variável determinada pelo passo 1.
3. Determine a derivada simbólica do resultado do passo 2 no que respeita à variável do passo 1.
4. Se a variável do passo 1 tiver um valor guardado ou especificado por um operador "with" (|),substitua esse valor no resultado a partir do passo 3.

d(f(x)∗g(x),x) ENTER

$$\frac{d}{dx}(f(x))\cdot g(x) + \frac{d}{dx}(g(x))\cdot f(x)$$

d(*d*(x^2∗y^3,x),y) ENTER $6\cdot y^2\cdot x$

d(x^2,x,⁻1) ENTER $\frac{x^3}{3}$

d({x^2,x^3,x^4},x) ENTER

$\{2\cdot x \quad 3\cdot x^2 \quad 4\cdot x^3\}$

data▸mat Menu CATALOG/MATH/List

data▸mat *data,mat*[,*linha1*] [,*col1*] [,*linha2*] [,*col2*]

Converte os dados para uma matriz.

Cada argumento *[,linha 1][,col 1][,linha 2][,col 2]* pode ser omitido individualmente. Se a *linha 1* for omitida, a predefinição é 1.Se a *col1* for omitida, a predefinição é. Se a *linha 2* for omitida, a predefinição é "linha máxima." Se a *col2* for omitida, a predefinição é "coluna máxima."

A estrutura DATA permite células vazias. As linhas não têm de ter tamanho igual. Quando os dados forem guardados como uma matriz, as células vazias serão preenchidas com "undef."

data▸mat d1,m1,1, , ,1 ENTER

Done

dayOfWk() CATALOG

dayOfWk(*ano,mês,dia***)** ⇒ *número inteiro*

Devolve um número inteiro entre 1 e 7. Cada número inteiro representa um dia da semana. Utilize **dayOfWk()** para determinar o dia da semana no qual uma data específica ocorre.
Nota: Pode não obter resultados exactos para os anos anteriores a 1583 (calendário pré-gregoriano).

Introduza o ano como um número inteiro com quatro dígitos. O mês e o dia podem ser inteiros

dayOfWk(1948,9,6)2 2

Valores inteiros:

1 = Sunday

2 = Monday

3 = Tuesday

4 = Wednesday

5 = Thursday

com um ou dois dígitos.

```
6 = Friday
7 = Saturday
```

▸DD — Menu MATH/Angle

número ▸**DD** ⇒ *valor*
lista 1 ▸**DD** ⇒ *lista*
matriz 1 ▸**DD** ⇒ *matriz*

Devolve o equivalente decimal do argumento expresso em graus. O argumento é um número, lista ou matriz que é interpretado pela definição de modo em grados, radianos ou graus.

No modo de ângulo Degree:

```
1.5° ▸DD [ENTER]                          1.5°
45° 22'14.3" ▸DD [ENTER]                  45.370...°
{45° 22'14.3",60° 0'0"} ▸DD [ENTER]
                                  {45.370...  60}°
```

No modo de ângulo Gradian:

```
1▸DD [ENTER]                              (9/10)°
```

No modo de ângulo Radian:

```
1.5 ▸DD [ENTER]                           85.9°
```

▸Dec — Menu MATH/Base

número inteiro 1 ▸**Dec** ⇒ *número inteiro*

Converte o *número inteiro 1* para um número decimal (base 10). Uma entrada binária ou hexadecimal tem de ter sempre um prefixo 0b ou 0h, respectivamente.

Zero, não a letra O, seguido por b ou h.
0b *número binário*
0h *número hexadecimal*
Um número hexadecimal pode ter até 32. Um número hexadecimal pode ter até 8.

Sem um prefixo, o *número inteiro 1* é tratado como decimal. O resultado aparece em decimais, independentemente do modo Base.

```
0b10011 ▸Dec [ENTER]                      19
0h1F ▸Dec [ENTER]                         31
```

Define — CATALOG

Define *funcName*(*arg1Name, arg2Name, ...***) = =**
expressão

Cria *funcName* como uma função definida pelo utilizador. Pode utilizar *funcName*(**)**, tal como utiliza as funções integradas**.** A função avalia a *expressão* com os argumentos fornecidos e devolve o resultado.

funcName não pode ser o nome de uma variável do sistema ou uma função integrada.

Os nomes dos argumentos são marcadores de posição; não deve utilizar esses mesmo nomes como argumentos quando utilizar a função.

Nota: Esta forma de **Define** é equivalente à execução da expressão: *expressão*→*funcName*(*arg1Name*,*arg2Name*). Este comando pode também ser utilizado para definir variáveis simples; por exemplo, **Define a=3**.

```
Define g(x,y)=2x- 3y [ENTER]              Done
g(1,2) [ENTER]                            -4
1→a:2→b:g(a,b) [ENTER]                    -4
Define h(x)=when(x<2,2x-3, -2x+3)
[ENTER]                                   Done
h(-3) [ENTER]                             -9
h(4) [ENTER]                              -5
Define própriosvl(a)=
cZeros(det(identity(dim(a)
[1])-x*a),x) [ENTER]                      Done
própriosvl([-1,2;4,3]) [ENTER]
```

$$\left\{\frac{2\cdot\sqrt{3}-1}{11}\quad\frac{-(2\cdot\sqrt{3}+1)}{11}\right\}$$

Define *funcName(arg1Name, arg2Name, ...)* **= :Func**
bloco
EndFunc

É idêntica à forma anterior de **Define**, excepto nesta forma, a função definida pelo utilizador **funcName()** pode executar um bloco de várias instruções.

bloco pode ser uma instrução individual ou uma série de instruções separadas com o carácter ":". *bloco* pode também incluir expressões e instruções (como **If**, **Then**, **Else**, e **For**). Permite que a função *funcName()* utilize a instrução **Return** para devolver um resultado específico.

Nota: É geralmente mais fácil fazer e editar esta forma de função no editor de programas em vez de na linha de entrada.

```
Define g(x,y)=Func:If x>y Then
:Return x:Else:Return y:EndIf
:EndFunc ENTER                    Done

g(3,-7) ENTER                     3
```

Define *progName(arg1Name, arg2Name, ...)* **:Prgm**
bloco
EndPrgm

Cria *progName* como um programa ou subprograma, mas não podem devolver um resultado com **Return**. Pode executar um bloco de várias instruções.

bloco pode ser uma instrução individual ou uma série de instruções separadas com o carácter ":". *bloco* pode também incluir expressões e instruções (como **If**, **Then**, **Else**, e **For**).

Nota: É geralmente mais fácil fazer e editar um bloco de programas no editor de programas em vez de na linha de entrada.

```
Define listinpt()=prgm:Local
n,i,str1,num:InputStr "Enter name of
list",str1:Input "No. of
elements",n:For i,1,n,1:Input
"element "&string(i),num:
num→#str1[i]:EndFor:EndPrgm ENTER
                                  Done

listinpt() ENTER    Enter name of list
```

DelFold CATALOG

DelFold *folderName1*[, *folderName2*] [, *folderName3*] **...**

Elimina as pastas definidas pelo utilizador com os nomes *folderName1, folderName2*, etc. Uma mensagem de erro aparece se as pastas contiverem quaisquer variáveis.

Nota: Não pode eliminar a pasta main.

```
NewFold games ENTER               Done
(creates the folder games)

DelFold games ENTER               Done
(deletes the folder games)
```

DelType

DelTypevar_type

Elimina todas as variáveis desbloqueadas do tipo especificado pelo *var_type*.

Nota: Os valores possíveis para *var_type* são:

ASM, DATA, EXPR, FUNC, GDB, LIST, MAT, PIC, PRGM, STR, TEXT, AppVar_type_name, All.

```
Deltype "LIST" ENTER              Done
```

DelVar CATALOG

DelVar *var1*[, *var2*] [, *var3*] **...**

Elimina as variáveis especificadas da memória.

```
2→a ENTER                         2
(a+2)^2 ENTER                     16
DelVar a ENTER                    Done
(a+2)^2 ENTER                     (a + 2)²
```

deSolve() Menu MATH/Calculus

deSolve(*1stOr2ndOrderOde***,** *independentVar***,**

Nota: Para digitar o símbolo de primo (′),

dependentVar) ⇒ *uma solução geral*

Devolve uma equação que especifica explicita ou implicitamente uma solução geral para a equação diferencial ordinária de 1ª ou 2ª ordem (ODE). Na ODE:

- Utilize o símbolo de primo (', prima 2nd [′]) tpara indicar a 1ª derivada da variável dependente no que respeita à variável independente.
- Utilize os dois símbolos de primo para indicar a segunda derivada correspondente.

O símbolo só é utilizado para derivadas de **deSolve()**. Noutros casos, utilize ***d*()**.

TA solução geral de uma equação de 1ª ordem contém uma constante arbitrária da forma *ck*, em que *k* é um sufixo do número inteiro de 1 a 255. O sufixo reinicia-se para 1 quando utilizar **ClrHome** ou F1 8: Clear Home. A solução de uma equação de 2ª ordem contém duas constantes.

prima 2nd [′].

```
deSolve(y''+2y'+y=x^2,x,y) ENTER
```
$$y=(@1\cdot x+@2)\cdot e^{-x}+x^2-4\cdot x+6$$

```
right(ans(1))→temp ENTER
```
$$(@1\cdot x+@2)\cdot e^{-x}+x^2-4\cdot x+6$$

```
d(temp,x,2)+2*d(temp,x)+temp-x^2
ENTER
```
0

```
DelVar temp ENTER
```
Done

Aplique **solve()** a uma solução implícita se a quiser converter para uma ou mais soluções explícitas equivalentes.

```
deSolve(y'=(cos(y))^2*x,x,y) ENTER
```
$$\tan(y)=\frac{x^2}{2}+@3$$

```
solve(ans(1),y) ENTER
```
$$y=\tan^{-1}\left(\frac{x^2+2\cdot @3}{2}\right)+@n1\cdot\pi$$

```
ans(1)|@3=c-1 and @n1=0 ENTER
```
$$y=\tan^{-1}\left(\frac{x^2+2\cdot(c-1)}{2}\right)$$

Quando comparar os resultados com as soluções dos manuais ou livros de exercícios, não se esqueça de que diferentes métodos introduzem constantes arbitrárias em diferentes pontos do cálculo, que podem produzir diferentes soluções gerais.

---

deSolve(*1stOrderOde* **and** *initialCondition*, **independentVar, dependentVar)**
⇒ *uma solução específica*

Devolve uma solução específica que satisfaça 1stOrderOde e initialCondition. É geralmente mais fácil do que determinar uma solução geral, substituindo valores iniciais, resolvendo a constante arbitrária e substituindo esse valor na solução geral.

initialCondition é uma equação da forma:

dependentVar (*initialIndependentValue*) = *initialDependentValue*

initialIndependentValue and *initialDependentValue* podem ser variáveis como x0 e y0 que não têm valores guardados. A diferenciação implícita podem ajudar a verificar as soluções implícitas.

```
sin(y)=(y*e^(x)+cos(y))y'→ode ENTER
```
$$\sin(y)=(e^x\cdot y+\cos(y))\cdot y'$$

```
deSolve(ode and y(0)=0,x,y)→soln
ENTER
```
$$\frac{-(2\cdot\sin(y)+y^2)}{2}=-(e^x-1)\cdot e^x\cdot\sin(y)$$

```
soln|x=0 and y=0 ENTER
```
true

```
d(right(eq)-left(eq),x)/
(d(left(eq)-right(eq),y))
→impdif(eq,x,y) ENTER
```
Done

```
ode|y'=impdif(soln,x,y) ENTER
```
true

```
DelVar ode,soln ENTER
```
Done

---

deSolve(*2ndOrderOde* **and** *initialCondition1* **and** *initialCondition2*, *independentVar*, *dependentVar*) ⇒ *uma solução específica*

Devolve uma solução específica que satisfaça *2ndOrderOde* e tenha um valor especificado da variável dependente e a primeira derivada num ponto.

```
deSolve(y''=y^(-1/2) and y(0)=0 and
y'(0)=0,t,y) ENTER
```
$$\frac{2\cdot y^{3/4}}{3}=t$$

```
solve(ans(1),y)
```

$$y=\frac{2^{2/3}\cdot(3\cdot t)^{4/3}}{4} \text{ and } t\geq 0$$

Para *initialCondition1*, utilize a forma:

dependentVar (*initialIndependentValue*) = *initialDependentValue*

Para *initialCondition2*, utilize a forma:

dependentVar' (*initialIndependentValue*) = *initial1stDerivativeValue*

deSolve(*2ndOrderOde* **and** *boundaryCondition1* **and** *boundaryCondition2*, *independentVar*, *dependentVar***)** ⇒ *uma solução específica*

Devolve uma solução específica que satisfaça *2ndOrderOde* e tenha valores especificados em dois pontos diferentes.

deSolve(w''−2w'/x+(9+2/x^2)w= x∗e^(x) and w(π/6)=0 and w(π/3)=0,x,w) ENTER

$$w=\frac{e^{\frac{\pi}{3}}\cdot x\cdot\cos(3\cdot x)}{10}-\frac{e^{\frac{\pi}{6}}\cdot x\cdot\sin(3\cdot x)}{10}+\frac{x\cdot e^{x}}{10}$$

det() Menu MATH/Matrix

det(*matriz quadrada***[,** *tol***])** ⇒ *expressão*

Devolve o determinante de *matriz quadrada*.

Opcionalmente, qualquer elemento da matriz é tratado como zero se o valor absoluto for menor do que *tol*. Esta tolerância só é utilizada se a matriz tiver entradas de ponto flutuante e não contiver quaisquer variáveis simbólicas que não tenham um valor. Caso contrário, *tol* é ignorado.

- Se utilizar ◆ ENTER ou definir o modo para Exact/Approx=APPROXIMATE, os cálculos são efectuados com a aritmética de ponto flutuante.
- ·Se *tol* for omitido ou não utilizado, a tolerância predefinida é calculada como:

  5E⁻14 ∗ **max(dim(***matriz quadrada***))** ∗ **rowNorm(***matriz quadrada***)**

det([a,b;c,d]) ENTER a•d−b•c

det([1,2;3,4]) ENTER -2

det(identity(3)−x∗[1,-2,3;-2,4,1;-6,-2,7]) ENTER

$$-(98\cdot x^3-55\cdot x^2+12\cdot x-1)$$

[1E20,1;0,1]→mat1 $\begin{bmatrix}1.E20 & 1\\ 0 & 1\end{bmatrix}$

det(mat1) ENTER 0

det(mat1,.1) ENTER 1.E20

diag() Menu MATH/Matrix

diag(*lista***)** ⇒ *matriz*
diag(*matriz da linha***)** ⇒ *matriz*
diag(*matriz da coluna***)** ⇒ *matriz*

Devolve uma matriz com os valores da matriz ou lista de argumentos na diagonal principal.

diag({2,4,6}) ENTER $\begin{bmatrix}2&0&0\\0&4&0\\0&0&6\end{bmatrix}$

diag(*matriz quadrada***)** ⇒ *matriz da linha*

Devolve uma matriz de linhas com os elementos da diagonal principal de *matriz quadrada*.

matriz quadrada tem de ser quadrada.

[4,6,8;1,2,3;5,7,9] ENTER $\begin{bmatrix}4&6&8\\1&2&3\\5&7&9\end{bmatrix}$

diag(ans(1)) ENTER [4 2 9]

:Dialog CATALOG

:Dialog
bloco
EndDlog

Gera uma caixa de diálogo quando o programa é executado.

bloco pode ser uma instrução individual ou uma série de instruções separadas com o carácter ":" . As opções de *bloco* válidas no F3 I/O, 1:Dialog item de menu do Editor de programas são 1:Text, 2:Request, 4:DropDown, and 7:Title.

Pode dar valores às variáveis numa caixa de diálogo que aparecerão como o valor predefinido (ou inicial). Se premir ENTER as variáveis são actualizadas a partir da caixa de diálogo e a variável 1. Se premir ESC as variáveis não são actualizadas e a variável do sistema ok é definida para zero.

```
Lista de programas:

:Dlogtest()
:Prgm
:Dialog
:Title          "This is a dialog box"
:Request              "Your name",Str1
:Dropdown       "Month you were born",
            seq(string(i),i,1,12),Var1
:EndDlog
:EndPrgm
```

dim() Menu MATH/Matrix/Dimensions

dim(*list***)** ⇒ *número inteiro*

Devolve a dimensão de *lista*.

```
dim({0,1,2}) ENTER                    3
```

dim(*matriz***)** ⇒ *lista*

Devolve as dimensões de matriz como {colunas, linhas}, lista de dois elementos.

```
dim([1,-1,2;-2,3,5]) ENTER        {2 3}
```

dim(*cadeia***)** ⇒ *número inteiro*

Devolve o número de caracteres contidos na *cadeia* de caracteres cadeia.

```
dim("Hello") ENTER                    5
dim("Hello"&" there") ENTER          11
```

:Disp CATALOG

Disp [*exprOrString1*] [, *exprOrString2*] **...**

Mostra o conteúdo actual do ecrã Program I/O. Se especificar uma ou mais *exprOrString*, cada cadeia de caracteres ou expressão aparece numa linha separada do ecrã Program I/O.

Uma expressão pode incluir operações de conversão como ▸**DD** e ▸**Rect**. Pode também utilizar o operador ▸ para efectuar conversões da base do número e da unidade.

Se Pretty Print = ON, as expressões aparecem em pretty print.

No ecrã Program I/O, pode premir F5 para ver o ecrã inicial ou **DispHome** num programa.

```
Disp "Hello" ENTER                Hello
Disp cos(2.3) ENTER               -.666…
{1,2,3,4}→L1 ENTER
Disp L1 ENTER                 {1  2  3  4}
Disp 180_min▸_hr ENTER          3.•_hr
```

Nota: Para digitar um carácter sublinhado (_), prima [•] [_]

Para digitar ▸, prima [2nd] [▸].

DispG CATALOG

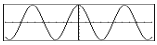
DispG

Mostra o conteúdo actual do ecrã Graph.

No modo de representação gráfica da função.

Segmento do programa:

```
⋮
:5*cos(x)→y1(x)
:⁻10→xmin
:10→xmax
:⁻5→ymin
:5→ymax
:DispG
⋮
```



DispHome CATALOG

DispHome

Mostra o conteúdo actual do ecrã inicial.

Segmento do programa:

```
  ⋮
:Disp "The result is: ",xx
:Pause "Press Enter to quit"
:DispHome
:EndPrgm
```

DispTbl CATALOG

DispTbl

Mostra o conteúdo actual do ecrã Table.

Nota: O cursor está activo para deslocação. Prima [ESC] ou [ENTER] para retomar a execução se estiver num programa.

5*cos(x)→y1(x) [ENTER]
DispTbl [ENTER]

▸DMS Menu MATH/Angle

expressão **▸DMS**
lista **▸DMS**
matriz **▸DMS**

Interpreta o argumento como um ângulo e mostra o número equivalente DMS (DDDDDD MM'SS.ss"). Consulte °, ', " na página 966 para o formato DMS (graus, minutos, segundos)

Nota: **▸DMS** converterá de radianos para graus quando utilizado no modo de radianos. Se a entrada for seguida por um símbolo de graus (°), nenhuma conversão ocorrerá. Só pode utilizar **▸DMS** no fim de uma linha de entrada.

No modo de ângulo Degree:

45.371 ▸DMS [ENTER] 45° 22'15.6"

{45.371,60} ▸DMS [ENTER]
{45° 22'15.6" 60° }

dotP() Menu MATH/Matrix/Vector

dotP(*lista 1***,** *lista 2***)** ⇒ *expressão*

Devolve o produto "dot" de duas listas.

dotP({a,b,c},{d,e,f}) [ENTER]
a•d+b•e+c•f

dotP({1,2},{5,6}) [ENTER] 17

dotP(*vector1***,** *vector2***)** ⇒ *expressão*

Devolve o produto "dot" de dois vectores.

Ambos têm de ser vectores de linhas ou vectores de colunas.

dotP([a,b,c],[d,e,f]) [ENTER]
a•d+b•e+c•f

crossP([1,2,3],[4,5,6])

DrawFunc CATALOG

Expressão **DrawFunc**

Desenha a *expressão* como uma função, utilizando x como a variável independente.

Nota: A nova representação gráfica apaga todos os itens desenhados.

No modo de desenho de gráfico da função e na janela **ZoomStd**:

`DrawFunc 1.25x*cos(x)` ENTER

DrawInv CATALOG

Expressão **DrawInv**

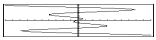
Desenha o inverso da *expressão*, desenhando os valores x no eixo y e os valores y no eixo x.

x é a variável independente.

Nota: A nova representação gráfica apaga todos os itens desenhados.

No modo de desenho de gráfico da função e na janela **ZoomStd**:

`DrawInv 1.25x*cos(x)` ENTER



DrawParm CATALOG

DrawParm *expressão 1, expressão 2* [, *tmin*] [, *tmax*] [, *tstep*]

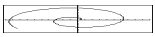
Desenha as equações paramétricas *expressão 1* e *expressão 2*, utilizando t como a variável independente.

As predefinições para *tmin*, *tmax*, e *tstep* são definições actuais para as variáveis da janela tmin, tmax, e tstep. A especificação de valores não altera as definições da janela. Se o modo de representação gráfica actual não for paramétrico, estes três argumentos são necessários.

Nota: A nova representação gráfica apaga todos os itens desenhados.

No modo de desenho de gráfico da função e na janela **ZoomStd**:

`DrawParm t*cos(t),t*sin(t),0,10,.1` ENTER



DrawPol CATALOG

DrawPol *expressão*[, θ*min*] [, θ*max*] [, θ*step*]

Desenha o gráfico polar de *expressão*, utilizando θ como a variável independente.

As predefinições para θ*min*, θ*max*, e θ*passo* são as definições actuais para as variáveis da janela θmin, θmax, e θstep. A especificação de valores não altera as definições da janela. Se o modo de representação gráfica actual não for polar, estes três argumentos são necessários.

Nota: A nova representação gráfica apaga todos os itens desenhados.

No modo de desenho de gráfico da função e na janela **ZoomStd**:

`DrawPol 5*cos(3*θ),0,3.5,.1` ENTER

DrawSlp CATALOG

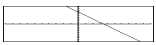
DrawSlp *x1*, *y1*, *declive*

Mostra o gráfico e desenha uma linha com a fórmula y−y1=slope·(x−x1).

Nota: A nova representação gráfica apaga todos os itens desenhados.

No modo de desenho de gráfico da função e na janela **ZoomStd**:

`DrawSlp 2,3,-2` ENTER



DropDown CATALOG

DropDown *titleString*, **{***item1String*, *item2String*, **...},** *varName*

Mostra um menu pendente com o nome *titleString* e com os itens **1:** *item1String*, **2:** *item2String*, etc. **DropDown** tem de estar num bloco **Dialog...EndDlog**.

Se *varName* já existir e tiver um valor no intervalo de itens, o item referenciado aparece como a selecção predefinida. Caso contrário, o primeiro item do menu é a selecção predefinida.

Quando seleccionar um item a partir do menu, o número do item correspondente é guardado na variável *varName*. (Se for necessário, **DropDown** cria *varName*.)

Consulte o exemplo da lista de programas **Dialog**.

DrwCtour CATALOG

DrwCtour *expressão*
DrwCtour *lista*

Desenha os contornos do gráfico 3D actual com os valores z especificados pela *expressão* ou lista. O modo de representação gráfica 3D já tem de estar definido. **DrwCtour** define automaticamente o estilo do formato do gráfico para CONTOUR LEVELS.

Por predefinição, o gráfico contém automaticamente os contornos com espaços iguais especificados pela variável da janela ncontour. **DrwCtour** desenha os contornos além das predefinições.

Para desligar os contornos predefinidos, defina ncontour para zero, utilizando o ecrã Window ou guardando 0 para a variável do sistema ncontour.

No modo de representação gráfica 3D:

```
(1/5)x^2+(1/5)y^2-10→z1(x,y) ENTER
                                Done
-10→xmin:10→xmax ENTER          10
-10→ymin:10→ymax ENTER          10
-10→zmin:10→zmax ENTER          10
0→ncontour ENTER                 0
DrwCtour {-9,-4.5,-3,0,4.5,9} ENTER
```

Utilize o cursor para alterar o ângulo de visualização. Prima 0 (zero) para voltar à vista original.

Para alternar entre diferentes estilo de formatos de gráficos, prima I

Prima X, Y, ou Z para ver o eixo correspondente.

E Tecla EE

*mantissa***E***exponente*

Introduz um número em notação científica. O número é interpretado como *mantissa* × 10*exponente*.

Sugestão: Se quiser introduzir uma potência de 10 sem provocar um resultado de valor decimal, utilize o *número inteiro* 10^.

```
2.3E 4 ENTER                    23000.
2.3E 9+4.1E 15 ENTER            4.1E 15
3* 10^4 ENTER                   30000
```

e^() Teclas ◆ [e^x]

e*^(expressão 1***)** ⇒ *expressão*

Devolve e elevado à potência de *expressão* 1.

Nota: Na TI-89 Titanium, premir ◆ [e^x] para ver e^(é diferente de premir alpha [E]. Na Voyage 200, premir 2nd [e^x] para ver e^ é diferente de aceder ao carácter e a partir do teclado QWERTY.

Pode introduzir um número complexo em formato polar $re^{i\theta}$. No entanto, utilize apenas este formato no modo de ângulo Radian; provoca um erro do domínio no modo de ângulo Degree ou Gradian.

```
e^(1) ENTER                     e
e^(1.) ENTER                    2.718...
e^(3)^2 ENTER                   e9
```

e*^(*lista 1*) ⇒ *lista

Devolve *e* elevado à potência de cada elemento na *lista 1*.

```
e^({1.1.,0,.5}) ENTER
                {e  2.718...  1  1.648...}
```

e*^(*matriz quadrada 1*) ⇒ *matriz quadrada

Devolve a matriz exponencial de *matriz quadrada* 1. Este procedimento *não* é a mesma coisa que calcular e elevado à potência de cada elemento. Para mais informações sobre o método de cálculo, consulte **cos()**.

matriz quadrada 1 tem de estar na diagonal. O resultado contém sempre números de pontos flutuantes.

```
eigVc()
```

$$\begin{bmatrix} 782.209 & 559.617 & 456.509 \\ 680.546 & 488.795 & 396.521 \\ 524.929 & 371.222 & 307.879 \end{bmatrix}$$

eigVc() Menu MATH/Matrix

eigVc(*matriz quadrada***)** ⇒ *matriz*

Devolve uma matriz com os vectores para uma *matriz quadrada* real ou complexa, em que cada coluna no resultado corresponde a um valor próprios. Não se esqueça de que um vector próprios não é única; pode ser escalada por qualquer factor constante. Os vectores próprios são normalizados, significando que se $V = [x_1, x_2, \ldots, x_n]$, então:

$$\sqrt{x_1{}^2 + x_2{}^2 + \ldots + x_n{}^2} = 1$$

matriz quadrada é equilibrada primeiro com transformações de similaridade até os tipos de colunas e linhas estarem o mais perto possível do mesmo valor. A *matriz quadrada* é reduzida para a forma Hessenberg superior e os vectores próprios são calculados através de uma factorização Schur.

No modo de formato complexo Rectangular:

```
[-1,2,5;3,-6,9;2,-5,7]→m1 ENTER
```

$$\begin{bmatrix} -1 & 2 & 5 \\ 3 & -6 & 9 \\ 2 & -5 & 7 \end{bmatrix}$$

```
eigVc(m1) ENTER
```

$$\begin{bmatrix} -.800\ldots & .767\ldots & .767\ldots \\ .484\ldots & .573\ldots+.052\ldots\cdot i & .573\ldots-.052\ldots\cdot i \\ .352\ldots & .262\ldots+.096\ldots\cdot i & .262\ldots-.096\ldots\cdot i \end{bmatrix}$$

eigVl() Menu MATH/Matrix

eigVl(*matriz quadrada***)** ⇒ *list*

Devolve uma lista dos valores próprios de uma *matriz quadrada* real ou complexa.

matriz quadrada é equilibrada primeiro com transformações de similaridade até os tipos de colunas e linhas estarem o mais perto possível do mesmo valor. A *matriz quadrada* é reduzida para a forma Hessenberg superior e os valores próprios são calculados a partir da matriz Hessenberg superior.

No modo de formato complexo Rectangular:

```
[-1,2,5;3,-6,9;2,-5,7]→m1 ENTER
```

$$\begin{bmatrix} -1 & 2 & 5 \\ 3 & -6 & 9 \\ 2 & -5 & 7 \end{bmatrix}$$

```
eigVl(m1)
```

{-4.409… 2.204…+.763…·*i*
2.204…−.763…·*i*}

: Else

Consulte também **If**, na página 881.

ElseIf

CATALOG Consulte também **If** na página 881.

If *Expressão booleana 1* **Then**
bloco1
ElseIf *Expressão booleana 2* **Then**
bloco2
⋮
ElseIf *Expressão booleana N* **Then**
bloco N
EndIf
⋮

ElseIf pode ser utilizada como uma instrução do programa para divisão do programa.

Segmento do programa:

```
⋮
:If choice=1 Then  Goto option1
:  : ElseIf choice=2 Then  Goto
option2
:  : ElseIf choice=3 Then  Goto
option3
:  : ElseIf choice=4 Then  Disp
"Exiting Program"
:  Return
:EndIf
⋮
```

EndCustm

Consulte **Custom** na página 854

EndDlog

Consulte **Dialog** na página 862

EndFor

Consulte **For** na página873.

EndFunc

Consulte **Func** na página 874

EndIf

Consulte **If** na página 881.

EndLoop

Consulte **Loop** na página 891

EndPrgm

Consulte **Prgm** na página 907

EndTBar

Consulte **ToolBar** na página 946

EndTry

Consulte **Try** na página 947.

EndWhile

Consulte **While** na página 949

entry()

CATALOG

entry() ⇒ *expressão*
entry(*número inteiro***)** ⇒ *expressão*

Devolve uma entrada da linha de entrada anterior a partir da área do histórico do ecrã inicial.

número inteiro, se incluído, especifica a expressão de entrada na área do histórico. A predefinição é 1, a entrada avaliada mais recentemente. O intervalo válido é de 1 a 99 e não pode ser uma expressão.

Nota: Se a última entrada ainda estiver realçada no ecrã inicial, premir [ENTER] é equivalente a executar **entry(1)**.

No ecrã inicial:

1+1/x [ENTER]	$\frac{1}{x}+1$
1+1/entry(1)	
[ENTER]	$\frac{1}{2\cdot(2\cdot x+1)}+3/2$
[ENTER]	$5/3-\frac{1}{3\cdot(3\cdot x+2)}$
entry(4) [ENTER]	$\frac{1}{x}+1$

exact() Menu MATH/Number

exact(*expressão 1* [, *tol*]**)** ⇒ *expressão*
exact(*lista 1* [, *tol*]**)** ⇒ *list*
exact(*matriz1* [, *tol*]**)** ⇒ *matriz*

Utiliza a aritmética do modo Exact independentemente da definição do modo Exact/Approx para devolver, quando possível, o número racional equivalente do argumento.

tol especifica a tolerância para a conversão; a predefinição é 0 (zero).

```
exact(.25) ENTER                    1/4
exact(.333333) ENTER                333333/1000000
exact(.33333,.001)                  1/3
exacto (3.5x+y)
exact({.2,.33,4.125}) ENTER
                                    {1/5 33/100 33/8}
```

Exec CATALOG

Exec *cadeia* [, *expressão 1*] [, *expressão 2*] **...**

Executa uma *cadeia* composta por uma série de códigos op do Motorola 68000. Estes códigos actuam como um programa de linguagem assembly. Se necessário, as *expressões* opcionais permitem passar um ou mais argumentos para o programa.

Para mais informações, visite o site Web da TI: **http://www.ti.com/calc**

Aviso: Exec dá-lhe acesso à potência total do microprocessor. Não se esqueça de que pode cometer um erro facilmente que bloqueie a calculadora e provoque a perda de dados. Sugerimos que efectue uma cópia de segurança do conteúdo da calculadora antes de tentar utilizar o comando **Exec**.

Exit CATALOG

Exit

Sai do bloco **For**, **While**, ou **Loop** actual.

Exit não é permitido fora das três estruturas de espiral (**For**, **While**, ou **Loop**).

```
Lista de programas:

:0→temp
:For i,1,100,1
:  temp+i→temp
:  If temp>20
:  Exit
:EndFor
:Disp temp

Conteúdo de temp depois da execução: 2
```

exp▸list() CATALOG

exp▸list(*expressão*,*var***)** ⇒ *lista*

Examina a *expressão* para equações separadas pela palavra "or" e devolve uma lista com os lados direitos das equações da forma *var=expressão*. Proporciona uma forma simples para extrair alguns valores de solução embebidos nos resultados das funções **solve(, cSolve(, fMin(** e **fMax(**).

Nota: exp▸list(não é necessário com as funções **zeros** e **cZeros(** porque devolvem uma lista de valores de selecção directamente.

```
solve(x^2-x-2=0,x) ENTER        x=2 or
                                x=-1
exp▸list(solve(x^2-x-2=0,x),x) ENTER
                                {-1 2}
```

expand(*expressão 1* **[,** *var***])** ⇒ *expressão*
expand(*lista 1* **[,***var***])** ⇒ *lista*
expand(*matriz 1* **[,***var***])** ⇒ *matriz*

expand(*expressão* 1) devolve expressão 1 expandida no que respeita a todas as variáveis. A expansão é a expansão polinomial para polinómios e a expansão parcial das fracções para expressões racionais.

O objectivo de **expand()** é transformar a *expressão 1* numa soma e/ou diferença de termos simples. Pelo contrário, o objectivo de **factor()** é transformar *expressão 1* num produto e/ou quociente de factores simples.

```
expand((x+y+1)^2) ENTER
```
$$x^2+2\cdot x\cdot y+2\cdot x+y^2+2\cdot y+1$$

```
expand((x^2-x+y^2-y)/(x^2*y^2-x^2*
y-x*y^2+x*y)) ENTER
```

---

expand(*expressão* 1,*var***)** devolve a *expressão* expandida no que respeita a *var*. As potências similares de var são recolhidas. Os factores e os termos são ordenados com var como a variável principal. Podem existir alguns factores ou expansão imprevistos nos coeficientes recolhidos. Quando comparado com a omissão var, este procedimento poupa frequentemente tempo, memória e espaço no ecrã, enquanto torna a expressão mais compreensível.

```
expand((x+y+1)^2,y) ENTER
```
$$y^2+2\cdot y\cdot(x+1)+(x+1)^2$$

```
expand((x+y+1)^2,x) ENTER
```
$$x^2+2\cdot x\cdot(y+1)+(y+1)^2$$

```
expand((x^2-x+y^2-y)/(x^2*y^2-x^2*
y-x*y^2+x*y),y) ENTER
```

```
expand(ans(1),x) ENTER
```


Mesmo quando existir apenas uma variável, a utilização de var pode tornar a factorização de denominadores utilizada para a expansão parcial das fracções mais completa.

```
expand((x^3+x^2-2)/(x^2-2)) ENTER
```
$$\frac{2\cdot x}{x^2-2}+x+1$$

Sugestão: Para expressões racionais, **propFrac()** é uma alternativa mais rápida, mas menos extrema para **expand()**.

Nota: Consulte também **comDenom()** para um numerador expandido sobre um denominador expandido.

```
expand(ans(1),x) ENTER
```
$$\frac{1}{x-\sqrt{2}}+\frac{1}{x+\sqrt{2}}+x+1$$

expand(*expressão 1,*[*var*]**)** distribui também logaritmos e potências fraccionárias independentemente de *var*. Para uma maior distribuição de logaritmos e potências fraccionárias, as restrições das desigualdades podem ser necessárias para garantir que alguns factores são não negativos.

expand(*expressão 1,* [*var*]**)** também distribui valores absolutos, **sign()**, e exponenciais, independentemente de *var*.

Nota: Consulte também **tExpand()** para a soma de ângulo trigonométrica e a expansão de vários ângulos.

```
ln(2x*y)+√(2x*y) ENTER
```
$$\ln(2\cdot x\cdot y)+\sqrt{(2\cdot x\cdot y)}$$

```
expand(ans(1)) ENTER
```
$$\ln(x\cdot y)+\sqrt{2}\cdot\sqrt{(x\cdot y)}+\ln(2)$$

```
expand(ans(1))|y>=0 ENTER
```
$$\ln(x)+\sqrt{2}\cdot\sqrt{x}\cdot\sqrt{y}+\ln(y)+\ln(2)$$

```
sign(x*y)+abs(x*y)+e^(2x+y) ENTER
```
$$e^{2\cdot x+y}+\mathrm{sign}(x\cdot y)+|x\cdot y|$$

```
expand(ans(1)) ENTER
```
$$\mathrm{sign}(x)\cdot\mathrm{sign}(y)+|x|\cdot|y|+(e^x)^2\cdot e^y$$

expr() — Menu MATH/String

expr(*cadeia***)** ⇒ *expressão*

Devolve a cadeia de caracteres contida na *cadeia* como uma expressão e executa-a imediatamente.

```
expr("1+2+x^2+x") [ENTER]                x² + x + 3

expr("expand((1+x)^2)") [ENTER]
                                        x² + 2• x + 1

"Define cube(x)=x^3"→ funcstr [ENTER]
                    "Define cube(x)=x^3"

expr(funcstr) [ENTER]                    Done

cube(2) [ENTER]                           8
```

ExpReg — Menu MATH/Statistics/Regressions

ExpReg *lista 1, lista 2* [, [*lista 3*] [, *lista 4, lista 5*]]

Calcula a regressão exponencial e actualiza todas as variáveis estatísticas do sistema.

Todas as listas têm de ter dimensões iguais, excepto a *lista 5*.

lista 1 representa xlist.
lista 2 representa ylist.
lista 3 representa a frequência.
lista 4 representa códigos de categorias.
lista 5 representa a lista de inclusão de categorias.

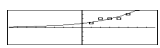
Nota: De *lista 1* a *lista 4* têm de ser o nome de uma variável ou c1–c99 (colunas na última variável de dados apresentada no Editor de Dados/Matriz). *Lista 5* não tem de ser o nome de uma variável e não pode ser c1–c99.

No modo de representação gráfica da função.

```
{1,2,3,4,5,6,7,8}→L1 [ENTER]
                                        {1 2 ...}
{1,2,2,2,3,4,5,7}→L2 [ENTER]
                                        {1 2 ...}
ExpReg L1,L2 [ENTER]                     Done
ShowStat [ENTER]
```

```
[ENTER]
Regeq(x)→y1(x) [ENTER]                   Done
NewPlot 1,1,L1,L2 [ENTER]                Done
```

◆[GRAPH]



factor() — Menu MATH/Algebra

factor(*expression1***[,** *var***])** ⇒ *expressão*
factor(*lista 1***[,***var***])** ⇒ *lista*
factor(*matriz1***[,***var***])** ⇒ *matriz*

factor(*expressão 1***)** devolve expressão 1 decomposta em factores no que respeita a todas as variáveis sobre um denominador comum.

expressão 1 é decomposta no maior número possível de factores em relação a factores racionais lineares sem introduzir novas subexpressões não reais. Esta alternativa é adequada se quiser a factorização no que respeita a mais de uma variável.

```
factor(a^3* x^2- a* x^2- a^3+a) [ENTER]
                    a•(a -1)•(a + 1)•(x -1)•(x + 1)

factor(x^2+1) [ENTER]                    x² + 1

factor(x^2- 4) [ENTER]                  (x - 2)• (x + 2)

factor(x^2- 3) [ENTER]                   x² - 3

factor(x^2- a) [ENTER]                   x² - a
```

factor(*expressão 1,var***)** devolve *expressão 1* decomposta em factores no que respeita à variável *var*.

expressão 1 é decomposta no maior número possível de factores em relação a factores reais que são lineares em *var*, mesmo que introduza constantes irracionais ou subexpressões que sejam irracionais noutras variáveis.

Os factores e os termos são ordenados com *var* como a variável principal. As potências similares de *var* são recolhidas em cada factor. Inclua *var* se a factorização for necessária apenas para essa variável e estiver disposto a aceitar expressões irracionais em quaisquer outras variáveis para aumentar a factorização no que respeita a *var*. Podem existir alguns factores imprevistos no que respeita a outras variáveis.

Para a definição AUTO do modo Exact/Approx incluindo *var* permite a aproximação de coeficientes de ponto flutuante em que os coeficientes irracionais não podem ser explicitamente expressos em termos de funções integradas. Mesmo quando existir apenas uma variável, incluindo *var* pode produzir a factorização mais completa.

Nota: Consulte também **comDenom()** para uma forma mais rápida para atingir a decomposição de factores parcial quando **factor()** não for suficientemente rápido ou se consumir a memória.

Nota: Consulte também **cFactor()** para decompor factores para coeficientes complexos em busca de factores lineares.

```
factor(a^3*x^2-a*x^2-a^3+a,x) ENTER
                a•(a²-1)•(x-1)•(x+1)

factor(x^2-3,x) ENTER
                (x+√3)•(x-√3)

factor(x^2-a,x) ENTER
                (x+√a)•(x-√a)

factor(x^5+4x^4+5x^3-6x-3) ENTER
                x⁵+4•x⁴+5•x³-6•x-3

factor(ans(1),x) ENTER
                (x-.964…)•(x+.611…)•
                (x+2.125…)•(x²+2.227…•
                x+2.392…)
```

factor(*número racional***)** devolve o número racional decomposto em primos. Para números compostos, o tempo de cálculo cresce exponencialmente com o número de dígitos no segundo maior factor. Por exemplo, decompor em factores um número inteiro de 30 dígitos pode demorar mais de um dia e decompor em factores um número de 100 dígitos pode demorar mais de um século.

Nota: Para parar um cálculo, prima ON.

Se quiser apenas determinar se um número é primo, utilize **isPrime()**. É muito mais rápido, em especial, se o *número racional* não for primo e se o segundo maior factor tiver mais de cinco dígitos.

```
factor(152417172689) ENTER
                123457•1234577

isPrime(152417172689) ENTER      false
```

Fill — Menu MATH/Matrix

Fill *expressão , matrizVar* ⇒ *matriz*

Substitui cada elemento da variável *variável da matriz* por *expressão*.

variável da matriz já tem de existir.

```
[1,2;3,4]→amatrx ENTER          [1 2; 3 4]
Fill 1.01,amatrx ENTER          Done
amatrx ENTER                    [1.01 1.01; 1.01 1.01]
```

Fill *expressão, listaVar* ⇒ *lista*

Substitui cada elemento na variável variável da lista por expressão.

variável da lista já tem de existir.

```
{1,2,3,4,5}→alist ENTER
                {1 2 3 4 5}
Fill 1.01,alist ENTER           Done
alist ENTER
        {1.01 1.01 1.01 1.01 1.01}
```

floor() Menu MATH/Number

floor(*expressão***)** ⇒ *número inteiro*

Devolve o maior número inteiro que é ≤ o argumento. Esta função é idêntica a **int()**.

O argumento pode ser um número complexo ou real.

```
floor(⁻2.14) ENTER                    -3.
```

floor(*lista 1***)** ⇒ *lista*
floor(*matriz 1***)** ⇒ *matriz*

Devolve uma lista ou matriz do ponto mínimo de cada elemento.

Nota: Consulte também **ceiling()** e **int()**.

```
floor({3/2,0,⁻5.3}) ENTER
                          {1 0 ⁻6.}
floor([1.2,3.4;2.5,4.8]) ENTER
                          [1. 3.
                           2. 4.]
```

fMax() Menu MATH/Calculus

fMax(*expressão, var***)** ⇒ *Expressão booleana*

Devolve uma expressão booleana que especifica valores candidatos de *var* que maximizam a expressão ou localizam o limite superior menor.

Utilize o operador "|" para restringir o intervalo de solução e/ou especificar o sinal de outras variáveis indefinidas.

Para a definição APPROX do modo Exact/Approx, **fMax()** procura repetidamente um local máximo aproximado. É muitas vezes mais rápido, em especial, se utilizar o operador "|" para restringir a procura a uma intervalo relativamente mais pequeno que contenha exactamente um local máximo.

Nota: Consulte também **fMin()** e **max()**.

```
fMax(1-(x-a)^2-(x-b)^2,x) ENTER
                          x = (a+b)/2
fMax(.5x^3-x-2,x) ENTER   x = ∞
fMax(.5x^3-x-2,x)|x≤1 ENTER
                          x = ⁻.816...
fMax(a*x^2,x) ENTER
      x = ∞ or x = ⁻∞ or x = 0 or a = 0
fMax(a*x^2,x)|a<0 ENTER   x = 0
```

fMin() Menu MATH/Calculus

fMin(*expressão, var***)** ⇒ *Expressão booleana*

Devolve uma expressão booleana que especifica valores candidatos devar que minimizam a expressão ou localizam o limite inferior maior.

Utilize o operador "|" para restringir o intervalo de solução e/ou especificar o sinal de outras variáveis indefinidas.

Para a definição APPROX do modo Exact/Approx, **fMin()** procura repetidamente um local mínimo aproximado. É muitas vezes mais rápido, em especial, se utilizar o operador "|" para restringir a procura a uma intervalo relativamente mais pequeno que contenha exactamente um local mínimo.

Nota: Consulte também **fMax()** e **min()**.

```
fMin(1-(x-a)^2-(x-b)^2,x) ENTER
                  x = ∞ or x = ⁻∞
fMin(.5x^3-x-2,x)|x≥1 ENTER   x = 1
fMin(a*x^2,x) ENTER
      x = ∞ or x = ⁻∞ or x = 0 or a = 0
fMin(a*x^2,x)|a>0 and x>1 ENTER
                          x = 1.
fMin(a*x^2,x)|a>0 ENTER   x = 0
```

FnOff CATALOG

FnOff

Deselecciona todas as funções Y= para o modo de representação gráfica actual.

No ecrã dividido, o modo de dois gráficos, **FnOff** só se aplica ao gráfico activo.

FnOff [1] [, 2] ... [,99] [,99]

Deseleccione as funções Y= especificadas para o modo de representação gráfica actual.

No modo de representação gráfica da função.

FnOff 1,3 [ENTER] deselects y1(x) and y3(x).

No modo de representação gráfica paramétrica:

FnOff 1,3 [ENTER] deselects xt1(t), yt1(t), xt3(t), and yt3(t).

FnOn CATALOG

FnOn

Selecciona todas as funções Y= definidas para o modo de representação gráfica actual.

No ecrã dividido, o modo de dois gráficos, **FnOn** só se aplica ao gráfico activo.

FnOn [1] [, 2] ... [,99] [,99]

Selecciona as funções Y= especificadas para o modo de representação gráfica actual.

Nota: No modo de representação gráfica 3D, só pode seleccionar uma função de cada vez. FnOn 2 selecciona z2(x,y) e deselecciona qualquer função seleccionada previamente. Noutros modos de gráficos, as funções seleccionadas previamente não são afectadas.

For CATALOG

For *var*, *baixo*, *alto* [, *passo*]
bloco
EndFor

Executa as instruções no *bloco* repetidamente para cada valor de *var*, de *baixo* a *alto*, em incrementos de *passo*.

var não tem de ser uma variável do sistema.

passo pode ser positivo ou negativo. O valor predefinido é 1.

bloco pode ser uma instrução individual ou uma série de instruções separadas com o carácter ":" .

Segmento do programa:

```
⋮
:0→tempsum : 1→step
:For i,1,100,step
:  tempsum+i→tempsum
:EndFor
:Disp tempsum
⋮
```

Conteúdo de tempsum depois da execução: 5050

Conteúdo de tempsum quando o passo for alterado para 2: 2500

format() Menu MATH/String

format(*expressão* [, *cadeia de formato*]**)** ⇒ *cadeia*

Devolve a *expressão* como uma cadeia de caracteres baseada no modelo de formatação.

A *expressão* tem de ser simplificada para um número. *Cadeia de formato* é uma cadeia e tem de estar na forma: "F[*n*]", "S[*n*]", "E[*n*]", "G[*n*][*c*]", em que [ ] indica partes opcionais.

F[*n*]: Formato fixo. *n* é o número de dígitos a mostrar após o ponto decimal.

S[*n*]: Formato científico. *n* é o número de dígitos a mostrar após o ponto decimal.

E[*n*]: Formato de engenharia *n* é o número de dígitos após o primeiro dígito significante. O expoente é ajustado para um múltiplo de três, e o ponto decimal é movido para a direita zero, um ou dois dígitos.

G[*n*][*c*]: Igual ao formato fixo, mas também separa dígitos à esquerda da raiz em grupos de três. especifica o carácter do separador de grupo e é predefinido para uma vírgula. Se c for um ponto, a raiz aparece como uma vírgula,

[R*c*]: Qualquer um dos especificadores acima pode ter um sufixo com o marcador de raiz R*c* em que *c* é um carácter individual que especifica o substituto do ponto da raiz.

```
format(1.234567,"f3") ENTER
                                "1.235"
format(1.234567,"s2") ENTER
                              "1.23E 0"
format(1.234567,"e3") ENTER
                             "1.235E 0"
format(1.234567,"g3") ENTER
                                "1.235"
format(1234.567,"g3") ENTER
                            "1,234.567"
format (1.234567,""g3,r:"" ENTER
                              "1:235"
```

fPart() Menu MATH/Number

fPart(*expressão 1***)** ⇒ *expressão*
fPart(*lista 1***)** ⇒ *lista*
fPart(*matriz1***)** ⇒ *matriz*

Devolve a parte fraccionário do argumento.

Para uma lista ou matriz, devolve as partes fraccionárias dos elementos.

O argumento pode ser um número complexo ou real.

```
fPart(-1.234) ENTER                 -.234
fPart({1, -2.3, 7.003}) ENTER
                          {0  -.3  .003}
```

Func CATALOG

Func
bloco
EndFunc

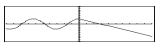
Requerido como a primeira instrução numa definição de função multi-instruções.

bloco pode ser uma instrução individual ou uma série de instruções separadas com o carácter ":" .

Nota: when() pode também ser utilizado para definir e fazer o gráfico das funções definidas por troços.

No modo de representação gráfica da função, defina uma função por troços:

```
Define g(x)=Func:If x<0 Then
:Return 3*cos(x):Else:Return
3-x:EndIf:EndFunc ENTER             Done
Graph g(x) ENTER
```



gcd() Menu MATH/Number

gcd(*número 1, número 2***)** ⇒ *expressão*

Devolve o máximo divisor comum dos argumentos. O **gcd** das duas fracções é o gcd dos numeradores dividido pelo **lcm** dos denominadores.

No modo Auto ou Approximate, o **gcd** dos números fracionários de ponto flutuante é 1.0.

```
gcd(18,33) ENTER                    3
```

gcd(*lista 1, lista 2***)** ⇒ *lista*

Devolve os máximos divisores comuns dos elementos correspondentes na *lista 1* e na *lista 2*.

```
gcd({12,14,16},{9,7,5}) ENTER
                                {3 7 1}
```

gcd(*matriz 1, matriz 2***)** ⇒ *matriz*

Devolve os máximos divisores comuns dos elementos correspondentes na *matriz 1* e na *matriz 2*.

```
gcd([2,4;6,8],[4,8;12,16])
```

Get CATALOG

Get var

Recupera um valor CBL 2/CBL™ (Calculator-Based Laboratory) ou CBR (Calculator-Based Ranger) a partir da porta de ligação e guarda-o na variável var.

Segmento do programa:

```
:Send {3,1,-1,0}
:For i,1,99
:  : Get data[i]  PtOn i,data[i]
:EndFor
```

GetCalc CATALOG

GetCalc *var*

Recupera um valor da porta de ligação e guarde-o na variável *var*. Destina-se à ligação de unidades.

Nota: Para obter uma variável para a porta de ligação a partir de outra unidade, utilize 2nd [VAR-LINK] na outra unidade para seleccionar e enviar uma variável, ou fazer um **SendCalc** na outra unidade.

Segmento do programa:

```
:Disp "Press Enter when ready"
:Pause
:GetCalc L1
:Disp "List L1 received"
```

GetCalc *var[,porta]*

Recupera um valor da porta de ligação e guarda-o na variável var na TI-89 Titanium de recepção.

Se a porta não for especificada ou a *porta = 0* for especificada, a TI-89 Titanium aguarda pelos dados a partir de outra porta.

Se a *porta = 1*, a TI-89 Titanium aguarda pelos dados a partir da porta USB.

Se a *porta = 2*, a TI-89 Titanium aguarda pelos dados a partir da porta I/O.

getConfg() CATALOG

getConfg() ⇒ *Pares da lista*

Devolve uma lista de atributos da calculadora. O nome do atributo é listado primeiro seguido pelo seu valor.

```
getConfg() ENTER
          {"Product Name" "Advanced
              Mathematics Software"
        "Version" "2.00, 09/25/1999"
           "Product ID" "03-1-4-68"
          "ID #" "01012 34567 ABCD"
                  "Cert. Rev. #" 0
                 "Screen Width" 160
                "Screen Height" 100
```

```
"Window Width" 160
"Window Height" 67
"RAM Size" 262132
"Free RAM" 197178
"Archive Size" 655360
"Free Archive" 655340}
```

Nota: O ecrã pode mostrar diferentes valores de atributos. **Rev. #** só aparece se adquirir e instalar software adicional na calculadora.

getDate() CATALOG

getDate() ⇒ *lista*

Devolve uma lista de datas associadas ao valor actual do relógio. A lista tem o formato *ano,mês,dia*.

```
getDate() ENTER          {2002  2  22}
```

getDenom() Menu MATH/Algebra/Extract

getDenom(*expressão 1***)** ⇒ *expressão*

Transforma a expressão 1 noutra com um denominador comum reduzido e, em seguida, devolve o denominador.

```
getDenom((x+2)/(y-3)) ENTER        y-3
getDenom(2/7) ENTER                  7
getDenom(1/x+(y^2+y)/y^2) ENTER
                                   x•y
```

getDtFmt() CATALOG

getDtFmt() ⇒ *número inteiro*

Devolve um número inteiro que representa o formato de data definido actualmente no dispositivo.

Valores inteiros:

1 = MM/DD/AA

2 = DD/MM/AA

3 = MM.DD.AA

4 = DD.MM.AA

5 = AA.MM.DD

6 = MM-DD-AA

7 = DD-MM-AA

8 = AA-MM-DD

getDtStr() CATALOG

getDtStr([*número inteiro*]**)** ⇒ *cadeia*

Devolve uma cadeia de caracteres da data actual no formato de data actual. Por exemplo, uma cadeia de caracteres devolvida com *28/09/02* representa o dia 28 de Setembro de 2002 (quando o formato de data está definido como DD/MM/AA).

Se introduzir o inteiro opcional que corresponde a um formato de data, a cadeia de caracteres devolve a data actual no formato especificado.

Valores inteiros opcionais:

1 = MM/DD/AA

2 = DD/MM/AA

3 = MM.DD.AA

4 = DD.MM.AA

5 = AA.MM.DD

6 = MM-DD-AA

7 = DD-MM-AA

8 = AA-MM-DD

getKey() CATALOG

getFold() ⇒ *cadeia do nome*

Devolve o nome da pasta actual como uma cadeia.

```
getFold() ENTER                "main"
getFold()→oldfoldr ENTER       "main"
oldfoldr ENTER                 "main"
```

getKey() CATALOG

getKey() ⇒ *número inteiro*

Devolve o código da tecla premida. Devolve 0 se não premir nenhuma tecla.

As teclas de prefixo (deslocar [↑], segunda função [2nd], opção [◆], alpha [alpha], e arrastar [⊕]) não são reconhecidas; no entanto, modificam os códigos de teclas da tecla que os segue. Por exemplo: [◆] [X] ≠ [X] ≠ [2nd] [X].

Para obter uma lista de códigos de teclas, consulte o Anexo B.

```
Lista de programas:

:Disp
:Loop
:  getKey()→key
:  while key=0
:    getKey()→key
:  EndWhile
:  Disp key
:  If key = ord("a")
:  Stop
:EndLoop
```

getMode() CATALOG

getMode(*cadeia do nome do modo***)** ⇒ *cadeia*
getMode("ALL") ⇒ *Pares de cadeias da lista*

Se o argumento for o nome de um modo específico, devolve uma cadeia com a definição actual para esse modo.

Se o argumento for "**ALL**", devolve uma lista de pares de cadeias com as definições de todos os modos. Se quiser recuperar as definições de modo posteriormente, tem de guardar o resultado **ModoGet** ("**ALL**") numa variável e, em seguida, utilize **ModoSet ()** para restaurar os modos.

Para obter uma lista de nomes de modos e definições possíveis, consulte **ModoSet ()**.

Nota: Para definir ou devolver informações sobre o modo Unit System utilize **setUnits()** ou **getUnits()** em vez de **setMode()** ou **getMode()**.

```
getMode("angle") ENTER          "RADIAN"
getMode("graph") ENTER        "FUNCTION"
getMode("all") ENTER
                  {"Graph" "FUNCTION"
     "Display Digits" "FLOAT 6"
                 "Angle" "RADIAN"
"Exponential Format" "NORMAL"
       "Complex Format" "REAL"
"Vector Format" "RECTANGULAR"
            "Pretty Print" "ON"
         "Split Screen" "FULL"
          "Split 1 App" "Home"
         "Split 2 App" "Graph"
       "Number of Graphs" "1"
          "Graph 2" "FUNCTION"
  "Split Screen Ratio"  "1.1"
       "Exact/Approx" "AUTO"
               "Base" "DEC"}
```

Nota: O ecrã pode mostrar diferentes definições de modo.

getNum() Menu MATH/Algebra/Extract

getNum(*expressão 1***)** ⇒ *expressão*

Transforma a *expressão 1* numa com um denominador comum reduzido e, em seguida, devolve o numerador.

```
getNum((x+2)/(y-3)) ENTER      x + 2
getNum(2/7) ENTER                  2
getNum(1/x+1/y) ENTER          x + y
```

getTime() CATALOG

getTime() ⇒ *lista*

Devolve uma lista com a hora de acordo com o valor actual do relógio. A lista tem o formato {*hora,minuto,segundo*}. A hora é devolvida no formato de 24 horas.

getTmFmt() CATALOG

getTmFmt() ⇒ *número inteiro*

Devolve um número inteiro que representa o formato de hora do relógio definido actualmente no dispositivo.

Valores inteiros:

12 = relógio de 12 horas

24 = relógio de 24 horas

getTmStr() CATALOG

getTmStr([*número inteiro*]**)** ⇒ *cadeia*

Devolve uma cadeia de caracteres da hora actual do relógio no formato de hora actual.

Se introduzir o inteiro opcional correspondente ao formato de hora do relógio, a cadeia de caracteres devolve a hora actual no formato especificado.

Valores inteiros opcionais:

12 = relógio de 12 horas

24 = relógio de 24 horas

getTmZn() CATALOG

getTmZn() ⇒ *número inteiro*

Devolve um número inteiro que representa o fuso horário definido actualmente no dispositivo.

O número inteiro devolvido representa o número de minutos que o fuso horário se desvia da hora média de Greenwich (GMT), como estabelecido em Greenwich, Inglaterra. Por exemplo, se o fuso horário se desviar de GMT cerca de duas horas, o dispositivo devolve 120 (minutos).

Os números inteiros para fusos horários a oeste de GMT são negativos.

Os números inteiros para fusos horários a este de GMT são positivos.

Se GMT for 14:07:07, serão:

8:07:07 a.m. em Denver, Colorado (Hora Diurna da Montanha)
(–360 minutos de GMT)

16:07:07 p.m. em Bruxelas, Bélgica (Hora padrão da Europa central)
(+120 minutos de GMT)

getType() CATALOG

getType(*var***)** ⇒ *cadeia*

Devolve uma cadeia que indica o tipo de dados da variável var.

Se var não estiver definida, devolve a cadeia "NONE"

```
{1,2,3}→temp ENTER          {1 2 3}
getType(temp) ENTER         "LIST"

2+3i→temp ENTER             2 + 3i
getType(temp) ENTER         "EXPR"

DelVar temp ENTER           Done
getType(temp) ENTER         "NONE"
```

Tipo de dados	Conteúdo da variável
"ASM"	Programa da linguagem assembly
"DATA"	Tipo de dados
"EXPR"	Expressão (inclui complexa/arbitrária/indefinida, ∞, ⁻∞, TRUE, FALSE, pi, *e*)
"FUNC"	Função
"GDB"	Base de dados de gráficos
"LIST"	Lista
"MAT"	Matriz
"NONE"	Variável não existe
"NUM"	Número real

"OTHER"	Tipo de dados diversos para futura utilização pelas aplicações de software
"PIC"	Imagem
"PRGM"	Programa
"STR"	Cadeia
"TEXT"	Tipo de texto
"VAR"	Nome de outra variável

getUnits() CATALOG

getUnits() ⇒ *lista*

Devolve uma *lista* de cadeias que contêm as unidades predefinidas actuais para todas as categorias, excepto constantes, temperatura, quantidade de substância, intensidade luminosa e aceleração. A lista tem a forma:

{"*sistema*" "*cat1*" "*unidade 1*" "*cat2*" "*unidade 2*" …}

A primeira cadeia dá o sistema (SI,ENG/US ou CUSTOM). Os pares de cadeias subsequentes dão uma categoria (como comprimento) e a unidade predefinida (como m para metros).

Para definir as unidades predefinidas, utilize **setUnits()**.

```
getUnits() ENTER
                {"SI"  "Area"  "NONE"
                  "Capacitance"  "_F"
                    "Charge"  "_coul"
                                  … }
```

Nota: O ecrã pode mostrar diferentes unidades predefinidas.

Goto CATALOG

Goto *Nome da etiqueta*

Transfere o controlo do programa para a etiqueta *Nome da etiqueta*.

nome da etiqueta tem de ser definida no mesmo programa com uma instrução **Lbl**.

Segmento do programa:

```
⋮
:0→temp
:1→i
:Lbl TOP
:  temp+i→temp
:  If i<10 Then
:   i+1→i
:   Goto TOP
:  EndIf
:Disp temp
⋮
```

▸Grad Menu CATALOG/MATH/Angle

▸Grad *expressão*

Converte uma expressão para uma medida de ângulo em grados.

No modo de ângulo Degree:

1.5 ▸Grad ENTER 1.66667^{G}

No modo de ângulo Radian:

1.5 ▸Grad ENTER 95.493^{G}

Graph CATALOG

Graph *expressão 1*[, *expressão 2*] [, *var1*] [, *var2*]

A função Smart Graph faz o gráfico das funções/expressões requeridas com o modo de desenho do gráfico actual.

As expressões introduzidas com os comandos **Graph** ou **Table** são atribuídas, aumentando os números das funções começando por 1. Podem ser modificados ou eliminados individualmente com as funções de edição quando a tabela aparecer, premindo F4 Header. As funções Y= seleccionadas actualmente são ignoradas.

Se omitir um argumento var opcional, **Graph** utiliza a variável independente do modo de desenho do gráfico actual.

Nota: Nem todos os argumentos opcionais são válidos em todos os modos porque pode nunca ter os quatro argumentos em simultâneo.

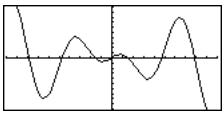
Algumas variações válidas desta instrução são:

Representação gráfica da função:	**Graph** *expr*, *x*
Gráfico paramétrico	**Graph** *xExpr*, *yExpr*, *t*
Gráfico polar	**Graph** *expr*, θ
Gráfico da sequência	Não permitido.
Gráfico 3D	**Graph** *expr*, *x*, *y*
Gráfico de equações diferenciais	Não permitido.

Nota: Utilize **ClrGraph** para apagar estas funções ou ir para o editor Y= para reactivar as funções Y= do sistema.

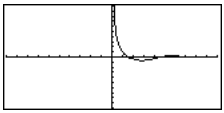
No modo de desenho de gráfico da função e na janela **ZoomStd**:

```
Graph 1.25a*cos(a),a ENTER
```



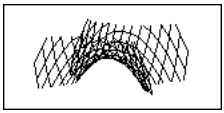
No modo de desenho de gráfico da função e na janela **ZoomStd**:

```
Graph time,2cos(time)/time,time
```



No modo de representação gráfica 3D:

```
Graph (v^2-w^2)/4,v,w ENTER
```



▸Hex Menu MATH/Base

número inteiro 1 **▸Hex** ⇒ *número inteiro*

Converte o *número inteiro 1* para um número hexadecimal. Os números binários ou hexadecimais têm sempre um prefixo 0b ou 0h, respectivamente.

Zero, não a letra O, seguido por b ou h.

0b *Número binário*
0h *Número hexadecimal*

Um número hexadecimal pode ter até 32. Um número hexadecimal pode ter até 8.

Sem um prefixo, o *número inteiro 1* é tratado como decimal (base10). O resultado aparece em binário, independentemente do modo Base.

Se introduzir um número inteiro decimal muito grande para uma forma binária de 32 bits, é utilizada uma operação do módulo simétrico para colocar o valor no intervalo adequado.

```
256▸Hex ENTER                    0h100
0b111100001111▸Hex ENTER         0hF0F
```

identity() Menu MATH/Matrix

identity(*expressão***)** ⇒ *matriz*

Devolve a matriz de identidade com uma dimensão de expressão.

A *expressão* tem de ser avaliada para um número inteiro positivo.

```
identity(4) ENTER
```

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

If CATALOG

If *Expressão booleana*
instrução

If *Expressão booleana* **Then**
bloco
EndIf

Se a *Expressão booleana* for avaliada para verdadeira, executa a *instrução* individual instrução ou o bloco de instruções bloco antes de continuar a execução.

Se a *Expressão booleana* for avaliada para falsa, continua a execução sem executar a instrução ou o bloco de instruções.

bloco pode ser uma instrução individual ou uma sequência de instruções separadas com o carácter ":".

Segmento do programa:

```
⋮
:If x<0
:Disp "x is negative"
⋮
 — ou —
⋮
:If x<0 Then
:  Disp "x is negative"
:  abs(x)→x
:EndIf
⋮
```

If *Expressão booleana* **Then**
bloco 1
Else
bloco 2
EndIf

Se a *Expressão booleana* for avaliada para verdadeira, executa *bloco 1* e ignora o *bloco 2*.

Se a *Expressão booleana* for avaliada para falsa, ignora *bloco 1*, mas executa *bloco 2*.

bloco 1 e *bloco 2* podem ser uma instrução individual.

Segmento do programa:

```
⋮
:If x<0 Then
:  Disp "x is negative"
:    Else
:  Disp "x is positive or zero"
:EndIf
⋮
```

If *Expressão booleana 1* **Then**
bloco 1
ElseIf *Expressão booleana 2* **Then**
bloco 2
⋮
ElseIf *Expressão booleana N* **Then**
bloco N
EndIf

Permite dividir o programa. Se a *Expressão booleana 1* for avaliada para verdadeira, executa o *bloco 1*. Se a *Expressão booleana* 1 for avaliada para falsa, avalia a *Expressão booleana 2*, etc.

Segmento do programa:

```
⋮
:If choice=1 Then  Goto option1
:  : ElseIf choice=2 Then    Goto
option2
:  : ElseIf choice=3 Then    Goto
option3
:  : ElseIf choice=4 Then    Disp
"Exiting Program"
:    Return
:EndIf
⋮
```

imag() Menu MATH/Complex

imag(*expressão 1***)** ⇒ *expressão*

imag(*expressão 1*) devolve a parte imaginária do argumento.

Nota: Todas as variáveis indefinidas são tratadas como variáveis reais. Consulte também **real()**.

```
imag(1+2i) ENTER        2
imag(z) ENTER           0
imag(x+iy) ENTER        y
```

imag(*lista 1***)** ⇒ *lista*

Devolve uma lista de partes imaginárias dos elementos.

```
imag({-3,4-i,i}) ENTER        {0 -1 1}
```

imag(*matriz 1***)** ⇒ *matriz*

Devolve uma matriz das partes imaginárias dos elementos.

```
imag([a,b;ic,id]) ENTER        [0 0; c d]
```

impDif() Menu MATH/Calculus, CATALOG

ImpDif(*equação, independentVar, dependentVar[,ordem]***)** ⇒ *expressão*

em que a ordem é predefinida para 1.

Calcula a derivada implícita para equações em que uma variável é definida implicitamente em termos de outra.

```
impDif(x^2+y^2=100,x,y)ENTER
                                -x/y
```

Acção indirecta

Consulte **#()** na página 964

Input CATALOG

Input

Interrompe o programa, mostra o ecrã Graph actual e permite actualizar variáveis *xc* e *yc* (também *rc* e θ*c* para o modo de coordenadas polares) através do posicionamento do cursor do gráfico.

Quando premir ENTER, o programa é retomado.

Segmento do programa:

```
⋮
:◉ Get 10 points from the Graph
                                Screen
:For i,1,10
:  Input
:  xc→XLIST[i]
:  yc→YLIST[i]
:EndFor
⋮
```

Input [*Cadeia da linha de comando***,**] *var*

Input [*Cadeia da linha de comando*], var interrompe o programa, mostra a *Cadeia da linha de comando* no ecrã Program I/O, aguarda que introduza uma expressão e guarda a expressão na variável var.

Se omitir a *Cadeia da linha de comando*, "?" aparece como uma linha de comando.

Segmento do programa:

```
⋮
:For i,1,9,1  "Enter x" &
string(i)→str1
:  Input str1,#(right(str1,2))
:EndFor
⋮
```

InputStr CATALOG

InputStr [*Cadeia da linha de comando***,] var**

Interrompe o programa, mostra a *Cadeia da linha de comando* no ecrã Program I/O, aguarda que introduza uma resposta e guarda a resposta como uma cadeia na variável var.

Se omitir a *Cadeia da linha de comando*, "?" aparece como uma linha de comando.

Nota: A diferença entre a **Input** e a **InputStr** é que a **InputStr** guarda sempre o resultado como uma cadeia para que " " não seja necessário.

Segmento do programa:

```
⋮
:InputStr "Enter Your Name",str1
⋮
```

inString() Menu MATH/String

inString(*Cadeia SRC, Subcadeia***[,** *início***])** ⇒ *número inteiro*

Devolve a posição do carácter na cadeia *Cadeia SRC* em que começa a primeira ocorrência da cadeia *Subcadeia*.

início, se incluído, especifica a posição do carácter na *Cadeia SRC* em que começa a procura. Predefinição = 1 (o primeiro carácter da *Cadeia SRC*).

Se a *Cadeia SRC* não contiver *Subcadeia* ou *início* for > o comprimento da *Cadeia SRC*, devolve zero.

```
inString("Hello there","the")
ENTER                                  7
"ABCEFG"→s1:If inString(s1,
"D")=0:Disp "D not found." ENTER
                              D not found.
```

int() CATALOG

int(*expressão***)** ⇒ *número inteiro*
int(*lista 1***)** ⇒ *lista*
int(*matriz 1***)** ⇒ *matriz*

Devolve o maior número inteiro que é menor ou igual ao argumento. Esta função é idêntica ao **floor()**.

O argumento pode ser um número complexo ou real.

Para uma lista ou matriz, devolve o maior número inteiro de cada um dos elementos.

```
int(-2.5) ENTER                       -3.
int([-1.234,0,0.37]) ENTER
                             [-2. 0 0.]
```

intDiv() CATALOG

intDiv(*número 1, número 2***)** ⇒ *número inteiro*
intDiv(*lista 1***,** *lista 2***)** ⇒ *lista*
intDiv(*matriz 1***,** *matriz 2***)** ⇒ *matriz*

Devolve a parte do número inteiro do argumento1 dividida pelo argumento2.

Para listas e matrizes, devolve a parte do número inteiro do argumento1 dividida pelo argumento2 para cada par de elementos.

```
intDiv(-7,2) ENTER                    -3
intDiv(4,5) ENTER                      0
intDiv({12,-14,-16},{5,4,-3}) ENTER
                              {2 -3 5}
```

integrar, Consulte **()** na página **883**

iPart() Menu MATH/Number

iPart(*número***)** ⇒ *número inteiro*
iPart(*lista 1***)** ⇒ *lista*
iPart(*matriz 1***)** ⇒ *matriz*

Devolve a parte do número inteiro do argumento.

Para listas e matrizes, devolve a parte do número inteiro de cada elemento.

O argumento pode ser um número complexo ou real.

```
iPart(-1.234) ENTER                   -1.
iPart({3/2,-2.3,7.003}) ENTER
                           {1 -2. 7.}
```

isArchiv() CATALOG

isArchiv(*var_name***)** ⇒ *verdadeiro, falso*

Determina se *var_name* é arquivada ou não.
Devolve verdadeiro se *var_name* for arquivada.
Devolve falso se *var_name* não for arquivada.

```
isArchiv(PROG1) ENTER                True
```

isClkOn() CATALOG

isClkOn() ⇒ *verdadeiro, falso*

Determina se o relógio está ligado ou desligado. Devolve verdadeiro se o relógio estiver ligado. Devolve falso se o relógio estiver desligado.

isLocked CATALOG

isLocked(*var_name***)** ⇒ *verdadeiro, falso*

Determina se *var_name* está bloqueada ou não. Devolve verdadeiro se *var_name* estiver bloqueada ou arquivada. Devolve falso se *var_name* não estiver bloqueada ou arquivada.

```
isLocked(PROG1) ENTER          False
```

isPrime() Menu MATH/Test

isPrime(*número***)** ⇒ *Expressão constante booleana*

Devolve verdadeiro ou falso para indicar se o *número* é um número inteiro ≥ 2 que só é divisível por si próprio e 1.

Se o *número* exceder cerca de 306 dígitos e não tiver factores 1021, **isPrime(***número***)** mostra uma mensagem de erro.

Se quiser determinar apenas se o *número* é primo, utilize **isPrime()** em vez de **factor()**. É muito mais rápido, em especial, se o *número* não for primo e tiver o segundo maior factor que exceda cerca de cinco dígitos.

```
IsPrime(5) ENTER                true
IsPrime(6) ENTER               false
```

Função de localização do primo seguinte após um número especificado:

```
Define nextPrim(n)=Func:Loop:
n+1→n:if isPrime(n):return n:
EndLoop:EndFunc ENTER           Done

nextPrim(7) ENTER                11
```

isVar() CATALOG

isVar(*var_name***)** ⇒ *verdadeiro, falso*

Determina se *var_name* está a ser utilizado. Devolve verdadeiro se *var_name* existir. Devolve falso se *var_name* não existir.

```
isArchiv(PROG1) ENTER          True
```

Item CATALOG

Item *Cadeia do nome do item*
Item *Cadeia do nome do item*, *etiqueta*

Válido apenas num bloco**Custom...EndCustm** ou **ToolBar...EndTBar**. Configura um elemento do menu pendente para lhe permitir colar texto na posição do cursor (**Custom**) ou derivar para uma etiqueta (**ToolBar**).

Nota: A derivação para uma etiqueta não é permitida num bloco **Custom**.

Consulte o exemplo **Custom**.

Lbl CATALOG

Lbl *Nome da etiqueta*

Define uma etiqueta com o nome *Nome da etiqueta* no programa.

Pode utilizar uma instrução **Goto** *labelName* para transferir o controlo do programa para a instrução imediatamente a seguir à etiqueta.

nome da etiqueta tem de cumprir os mesmos requisitos de atribuição de nomes como no nome de uma variável.

Segmento do programa:

```
⋮
:Lbl lbl1
:InputStr "Enter password", str1
:If str1≠password
:  Goto lbl1
:Disp "Welcome to ..."
⋮
```

lcm() Menu MATH/Number

lcm(*número 1***,** *número 2***)** ⇒ *expressão*
lcm(*lista 1***,** *lista 2***)** ⇒ *lista*
lcm(*matriz 1***,** *matriz 2***)** ⇒ *matriz*

Devolve o menor múltiplo comum dos dois argumentos. O lcm das duas fracções é o **lcm** dos numeradores dividido pelo **gcd** dos denominadores. O **lcm** dos pontos fraccionários de ponto flutuante é o produto.

Para as duas listas ou matrizes, devolve os menores múltiplos comuns dos elementos correspondentes.

```
lcm(6,9) ENTER                                  18
lcm({1/3,-14,16},{2/15,7,5}) ENTER
                                   {2/3 14 80}
```

left() Menu MATH/String

left(*Cadeia de origem***[,** *num***])** ⇒ *cadeia*

Devolve os caracteres *num* mais à esquerda contidos na cadeia de caracteres *Cadeia de origem*.

Se omitir *num*, devolve tudo da *Cadeia de origem*.

```
left("Hello",2) ENTER                          "He"
```

left(*lista 1***[,** *num***])** ⇒ *lista*

Devolve os elementos *num* mais à esquerda contidos na *lista 1*.

Se omitir *num*, devolve tudo de *lista 1*.

```
left({1,3,-2,4},3) ENTER
                                     {1 3 -2}
```

left(*comparação***)** ⇒ *expressão*

Devolve o lado esquerdo de uma equação ou desigualdade.

```
left(x<3) ENTER                                 x
```

limit() Menu MATH/Calculus

limit(*expressão 1***,** *var***,** *ponto***[,** *direcção***])** ⇒ *expressão*
limit(*lista 1***,** *var***,** *ponto***[,** *direcção***])** ⇒ *lista*
limit(*matriz 1***,** *var***,** *ponto***[,** *direcção***])** ⇒ *matriz*

Devolve o limite pedido.

direcção negativa=da esquerda, positiva=da direita, caso contrário=ambas. (Se omitida, a, *direcção* é predefinida para ambas.)

Os limites ao positivo ∞ e ao negativo ∞ são sempre convertidos para limites de um lado a partir do lado finito.

Dependendo das circunstâncias, **limit()** devolve-se a si próprio ou undef quando não conseguir determinar um limite único. Isto não significa necessariamente que o limite único não existe. undef significa que o resultado é um número desconhecido com magnitude finita ou infinita, ou o conjunto completo desses números.

```
limit(2x+3,x,5) ENTER                           13
limit(1/x,x,0,1) ENTER                          ∞
limit(sin(x)/x,x,0) ENTER                       1
limit((sin(x+h)-sin(x))/h,h,0) ENTER
                                        cos(x)
limit((1+1/n)^n,n,∞) ENTER                      e
```

limit() utiliza métodos como a regra L'Hopital, por isso, existem limites únicos que não podem ser determinados. Se a *expressão 1* contiver variáveis indefinidas diferentes de *var*, pode ter de os restringir para obter um resultado mais conciso.

Os limites podem ser muito sensíveis para o erro de arredondamento. Quando possível, evite a definição de APPROX do modo Exact/Approx e os números aproximados quando calcular os limites. Caso contrário, os limites que deverão ser zero ou ter magnitude infinita, provavelmente não têm, e os limites que deverão ter magnitude não zero finita, podem não ter.

```
limit(a^x,x,∞) ENTER                    undef
limit(a^x,x,∞)|a>1 ENTER                  ∞
limit(a^x,x,∞)|a>0 and a<1
ENTER                                     0
```

Line CATALOG

Line *InícioX, InícioY, FimX, FimY*[**,** *ModoDraw*]

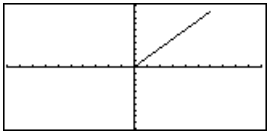
Mostra o ecrã Graph e desenha, apaga ou inverte um segmento de linha entre as coordenadas da janela (*InícioX, InícioY*) e (*FimX, FimY*), incluindo ambos os pontos finais.

Se o *ModoDraw* = 1, desenha a linha (predefinição).
Se o *ModoDraw* = 0, desliga a linha.
Se o *ModoDraw* = 1, desliga ou liga uma linha (inverte os pixéis ao longo da linha).

Nota: A nova representação gráfica apaga todos os itens desenhados. Consulte também **PxlLine**.

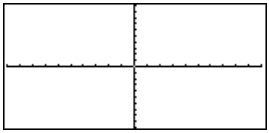
Na janela **ZoomStd**, desenhe uma linha e, em seguida, apague-a.

```
Line 0,0,6,9 ENTER
```



HOME

```
Line 0,0,6,9,0 ENTER
```



LineHorz CATALOG

LineHorz *y* [**,** *ModoDraw*]

Mostra o ecrã Graph e desenha, apaga ou inverte uma linha horizontal na posição da janelay.

Se o *ModoDraw* = 1, desenha a linha (predefinição).
Se o *ModoDraw* = 0, desliga a linha.
Se o *ModoDraw* = 1, desliga ou liga uma linha (inverte os pixéis ao longo da linha).

Nota: A nova representação gráfica apaga todos os itens desenhados. Consulte também **PxlHorz**.

Numa janela **ZoomStd**:

```
LineHorz 2.5 ENTER
```

LineTan CATALOG

LineTan *expressão 1, expressão 2*

Mostra o ecrã Graph e desenha uma tangente de linha para *expressão 1* no ponto especificado.

expressão 1 é uma expressão ou o nome de uma função, em que x é assumido ser a variável independente e *expressão 2* é o valor x do ponto que é a tangente.

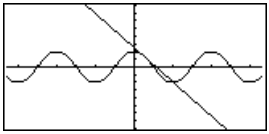
Nota: No exemplo mostrado, o gráfico de, *expressão 1* é desenhado separadamente. **LineTan** não desenha o gráfico de *expressão 1.*

No modo de representação gráfica da função e na janela **ZoomTrig**:

`Graph cos(x)`

[HOME]
[◆][CALC HOME]

`LineTan cos(x),/4`



LineVert CATALOG

LineVert *x* [, *ModoDraw*]

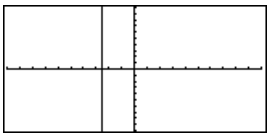
Mostra o ecrã Graph e desenha, apaga ou inverte uma linha vertical na posição da janela *x*.

Se o *ModoDraw* = 1, desenha a linha (predefinição).
Se o *ModoDraw* = 0, desliga a linha.
Se o *ModoDraw* = 1, desliga ou liga uma linha (inverte os pixéis ao longo da linha).

Nota: A nova representação gráfica apaga todos os itens desenhados. Consulte também **PxlVert**.

Numa janela **ZoomStd**:

`LineVert ⁻2.5` [ENTER]



LinReg Menu MATH/Statistics/Regressions

LinReg *lista 1*, *lista 2*[, [*lista 3*] [, *lista 4*, *lista 5*]]

Calcula a regressão linear e actualiza todas as variáveis estatísticas do sistema.

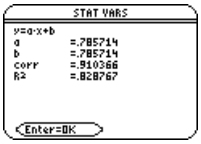
Todas as listas têm de ter dimensões iguais, excepto a *lista 5*.

lista 1 representa xlist.
lista 2 representa ylist.
lista 3 representa a frequência.
lista 4 representa os códigos de categorias.
lista 5 representa a lista de inclusão de categorias.

Nota: De *lista 1* a *lista 4* têm de ser um nome da variável ou c1–c99 (colunas da última variável de dados apresentada no Editor de Dados/Matriz). *lista 5* não tem de ser o nome de uma variável e não pode ser c1–c99.

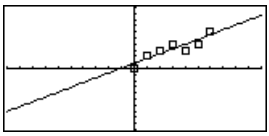
No modo de representação gráfica da função.

`{0,1,2,3,4,5,6}→L1` [ENTER]
`{0,2,3,4,3,4,6}→L2` [ENTER]
`LinReg L1,L2` [ENTER] Done
`ShowStat` [ENTER]



[ENTER]
`Regeq(x)→y1(x)` [ENTER] Done
`NewPlot 1,1,L1,L2` [ENTER] Done

[◆][GRAPH]



Δlist() Menu MATH/List

list(*lista 1*) ⇒ *lista*

Devolve uma lista com as diferenças entre os elementos consecutivos na *lista 1*. Cada elemento de *lista 1* é subtraído do próximo elemento de *lista 1*. A lista resultante é sempre um elemento mais pequeno do que a *lista 1* original.

```
Δlist({20,30,45,70}) ENTER
                    {10,15,25}
```

list▸mat() Menu MATH/List

list▸mat(*lista* [**,** *elementos por linha***])** ⇒ *matriz*

Devolve uma matriz preenchida linha a linha com os elementos da *lista*.

elementos por linha, se incluído, especifica o número de elementos por linha. A predefinição é o número de elementos de *lista* (uma linha).

Se a *lista* não preencher a matriz resultante, são adicionados zeros.

```
list▸mat({1,2,3}) ENTER        [1 2 3]
list▸mat({1,2,3,4,5},2) ENTER
```

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 0 \end{bmatrix}$$

▸ln Menu MATH/String

▸**ln** *expressão* ⇒ *expressão*

Faz com que a expressão de entrada seja convertida para uma expressão apenas com os logaritmos naturais (ln).

```
Log(x)ln
```

$$\frac{\ln(x)}{\ln(10)}$$

ln() Teclas 2nd [LN]

ln(*expressão 1***)** ⇒ *expressão*
ln(*lista 1***)** ⇒ *lista*

Devolve o logaritmo natural do argumento.

Para uma lista, devolve os logaritmos naturais dos elementos.

```
ln(2.0) ENTER                  .693...
```

Se o modo de formato complexo for **REAL**:

```
ln({-3,1.2,5}) ENTER
Error: Non-real result (Resultado não
```

Se o modo de formato complexo for RECTANGULAR:

```
ln({-3,1.2,5}) ENTER
        {ln(3) + π• i   .182...  ln(5)}
```

---

ln(*matriz quadrada 1***)** ⇒ *matriz quadrada*

Devolve o logaritmo natural da matriz de *Matriz quadrada 1*. Este procedimento *não* é a mesma coisa que calcular o logaritmo natural de cada elemento. Para mais informações sobre o método de cálculo, consulte **cos()**.

*matriz quadrada 1*tem de estar na diagonal. O resultado contém sempre números de pontos flutuantes.

No modo de ângulo Radian e no modo do formato complexo Rectangular:

```
ln([1,5,3;4,2,1;6,-2,1]) ENTER
```

$$\begin{bmatrix} 1.831...+1.734...\cdot i & .009...-1.490...\cdot i & ... \\ .448...-.725...\cdot i & 1.064...+.623\cdot i & ... \\ -.266...-2.083...\cdot i & 1.124...+1.790...\cdot i & ... \end{bmatrix}$$

LnReg — Menu MATH/Statistics/Regressions

LnReg *lista 1*, *lista 2*[, [*lista 3*] [, *lista 4*, *lista 5*]]

Calcula a regressão logarítmica e actualiza todas as variáveis estatísticas do sistema.

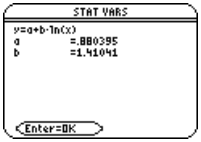
Todas as listas têm de ter dimensões iguais, excepto a *lista 5*.

lista 1 representa xlist.
lista 2 representa ylist.
lista 3 representa a frequência.
lista 4 representa os códigos de categorias.
lista 5 representa a lista de inclusão de categorias.

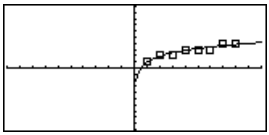
Nota: De *lista 1* a *lista 4* têm de ser um nome da variável ou c1–c99 (colunas da última variável de dados apresentada no Editor de Dados/Matriz). *lista 5* não tem de ser o nome de uma variável e não pode ser c1–c99.

No modo de representação gráfica da função.

```
{1,2,3,4,5,6,7,8}→L1 ENTER
                                {1 2 3 ...}
{1,2,2,3,3,3,4,4}→L2 ENTER
                                {1 2 2 ...}
LnReg L1,L2 ENTER                    Done
ShowStat ENTER
```




```
ENTER
Regeq(x)→y1(x) ENTER                 Done
NewPlot 1,1,L1,L2 ENTER              Done
◆[GRAPH]
```



Local — CATALOG

Local *var1*[, *var2*] [, *var3*] ...

Declara as *vars* especificadas como variáveis locais. Essas variáveis só existem durante a avaliação de um programa ou função e são eliminadas quando o programa ou a função terminar a execução.

Nota: As variáveis locais poupam memória porque só existem temporariamente. Também não perturbam quaisquer valores de variáveis globais existentes. As variáveis locais têm de ser utilizadas para espirais **For** e para guardar temporariamente os valores de uma função multilinhas visto que as modificações em variáveis globais não são permitidas numa função.

```
Lista de programas:

:prgmname()
:Prgm
:Local x,y
:Input "Enter x",x
:Input "Enter y",y
:Disp x*y
:EndPrgm
```

Nota: *x* e *y* não existem depois da execução do programa.

Lock — CATALOG

Bloquear *var1[, var2]* ...

Bloqueia as variáveis especificadas. Impede que elimine ou modifique acidentalmente a variável sem primeiro utilizar a instrução Bloquear nessa variável.

No exemplo à direita, a variável L1 está bloqueada e não pode ser eliminada nem modificada.

Nota: As variáveis podem ser desbloqueadas com o comando **Unlock**.

```
{1,2,3,4}→L1 ENTER                {1,2,3,4}
Lock L1 ENTER                        Done
DelVar L1 ENTER
  Error: Variable is locked or protected
```

log() CATALOG/ Teclas ◆ 7

log(*expressão 1[,expressão 2]***)** ⇒ *expressão*
log(*lista 1[,expressão 2]***)** ⇒ *lista*

Devolve o logaritmo *expressão 2* base do argumento.

Para uma lista, devolve o logaritmo *expressão 2* base dos elementos.

Se a expression 2 for omitida, 10 é utilizado como base.

```
log(2.0) ENTER                    .301...
```

Se o modo de formato complexo for **REAL**:

```
log({-3,1.2,5}) ENTER
                Error: Non-real result
```

Se o modo de formato complexo for **RECTANGULAR**:

```
log({-3,1.2,5}) ENTER
      {log(3)+ 1.364...·i  .079... log(5)}
```

---

log(*matriz quadrada 1***)** ⇒ *matriz quadrada*

Devolve o logaritmo *expressão 2* base da matriz de *Matriz quadrada 1*. Este procedimento não é a mesma coisa que calcular o logaritmo *expressão 2* base de cada elemento. Para mais informações sobre o método de cálculo, consulte **cos()**.

matriz quadrada 1 tem de estar na diagonal. O resultado contém sempre números de pontos flutuantes.

No modo de ângulo Radian e no modo do formato complexo Rectangular:

```
log([1,5,3;4,2,1;6,-2,1]) ENTER
```

$$\begin{bmatrix} .795...+.753...\cdot i & .003...-.647...\cdot i & ... \\ .194...-.315...\cdot i & .462...+.270\cdot i & ... \\ -.115...-.904...\cdot i & .488...+.777...\cdot i & ... \end{bmatrix}$$

---

log(*x,b***)** ⇒ *expressão*
log(*matriz quadrada 1***)** ⇒ *matriz quadrada*

Numa lista, devolve o logaritmo da expressão 2 base dos elementos.

```
Log(10,3) - log(5,3) ENTER    Log3(2)
Log(2.0,4)ENTER                  .5
```

▸logbase Menu MATH/String

expressão **▸logbase(***expressão 1***)** ⇒ *expressão*

Faz com que a expressão de entrada seja simplificada para uma expressão com a *expressão 1* base.

```
Log(10,3) - log(5,5)logbase(5)
ENTER
```

$$\frac{\log_5(30)}{\log_5(3)}$$

Logistic — Menu MATH/Statistics/Regressions

Logistic *lista 1*, *lista 2* [, [*repetições*], [*lista 3*] [, *lista 4*, *lista 5*]]

Calcula a regressão logística e actualiza todas as variáveis estatísticas do sistema.

Todas as listas têm de ter dimensões iguais, excepto a *lista 5*.

lista 1 representa xlist.
lista 2 representa ylist.
lista 3 representa a frequência.
lista 4 representa os códigos de categorias.
lista 5 representa a lista de inclusão de categorias.

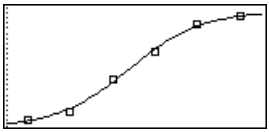
repetições especificam o número máximo de vezes que uma solução é tentada. Se omitido, o 64 é utilizado. Geralmente, os valores maiores resultam em melhor precisão, mas em tempos de execução mais longos, e vice-versa.

Nota: *lista 1* a *lista 4* têm de ser o nome de uma variável ou c1–c99 (colunas na última variável de dados apresentada no Editor de Dados/Matriz). *Lista 5* não tem de ser o nome de uma variável e não pode ser c1–c99.

No modo de representação gráfica da função.

```
{1,2,3,4,5,6}→L1 [ENTER]                {1 2 3 …}
{1,1.3,2.5,3.5,4.5,4.8}→L2
[ENTER]
                                   {1 1.3 2.5 …}
Logistic L1,L2 [ENTER]                      Done
ShowStat [ENTER]
```

```
[ENTER]
regeq(x)→y1(x) [ENTER]                      Done
NewPlot 1,1,L1,L2 [ENTER]                   Done
[◆][GRAPH]
[F2]9
```



:Loop — CATALOG

Loop
bloco
EndLoop

Executa repetidamente as instruções no *bloco*. Não se esqueça de que a espiral é executada continuamente, excepto se uma instrução **Goto** ou **Exit** for executada no *bloco*.

bloco é uma sequência de instruções separadas com o carácter ":" .

Segmento do programa:

```
                                                ⋮
:1→i
:Loop
:  Rand(6)→die1
:  Rand(6)→die2
:  If die1=6 and die2=6
:    Goto End
:  i+1→i
:EndLoop
:Lbl End
:Disp "The number of rolls is", i
                                                ⋮
```

LU — Menu MATH/Matrix

LU *matriz*, *NomeMatl*, *NomeMatU*, *NomeMatP*[, *tol*]

Calcula a decomposição Doolittle LU (inferior-superior) de uma *matriz* complexa ou real. A matriz triangular inferior é guardada em *NomeMatl*, a matriz triangular superior em *NomeMatU*, e a matriz de permutação (que descreve as permutas de linhas durante o cálculo) em *NomeMatP*.

NomeMatl ∗ *NomeMatU* = *NomeMatP* ∗ *matriz*

Opcionalmente, qualquer elemento da matriz é tratado como zero se o valor absoluto for menor do que *tol*. Esta tolerância só é utilizada se a matriz tiver entradas de ponto flutuante e não contiver quaisquer variáveis simbólicas que não tenham um valor. Caso contrário, *tol* é ignorado.

- Se utilizar ◆ ENTER ou definir o modo para Exact/Approx=APPROXIMATE, os cálculos são efectuados com a aritmética de ponto flutuante.
- ·Se *tol* for omitido ou não utilizado, a tolerância predefinida é calculada como:

  5ᴇ⁻14 ∗ **max(dim(***matriz***))** ∗ **rowNorm(***matriz***)**

O algortimo de factorização **LU** utiliza a rotação parcial com intercâmbios de linhas.

`[6,12,18;5,14,31;3,8,18]→m1` ENTER

$$\begin{bmatrix} 6 & 12 & 18 \\ 5 & 14 & 31 \\ 3 & 8 & 18 \end{bmatrix}$$

`LU m1,lower,upper,perm` ENTER Done

`lower` ENTER

$$\begin{bmatrix} 1 & 0 & 0 \\ 5/6 & 1 & 0 \\ 1/2 & 1/2 & 1 \end{bmatrix}$$

`upper` ENTER

$$\begin{bmatrix} 6 & 12 & 18 \\ 0 & 4 & 16 \\ 0 & 0 & 1 \end{bmatrix}$$

`perm` ENTER

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

`[m,n;o,p]→m1` ENTER

$$\begin{bmatrix} m & n \\ o & p \end{bmatrix}$$

`LU m1,lower,upper,perm` ENTER Done

`lower` ENTER

$$\begin{bmatrix} 1 & 0 \\ \frac{m}{o} & 1 \end{bmatrix}$$

`upper` ENTER

$$\begin{bmatrix} o & p \\ 0 & n - \frac{m \cdot p}{o} \end{bmatrix}$$

`perm` ENTER

$$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

mat▸data — Menu MATH/List

mat▸data mat▸data[,*linha 1*][,*col 1*][,*linha 2*][,*col 2*]

Converte uma matriz para dados.

Cada argumento [,*linha 1*][,*col 1*][,*linha 2*][,*col 2*] pode ser omitido individualmente. Se a *linha 1* for omitida, a predefinição é 1. Se a *col1* for omitida, a predefinição é 1. Se a *linha 2* a predefinição é "linha máxima." Se a *col2* for omitida, a predefinição é "coluna máxima."

`mat▸data,m1,d1,1,,,1` ENTER
Done

listamat() Menu MATH/List

mat▸list(*matriz***)** ⇒ *lista*

Devolve uma lista preenchida com os elementos da matriz. Os elementos são copiados da *matriz* linha a linha.

```
mat▸list([1,2,3]) ENTER          {1 2 3}
[1,2,3;4,5,6]→M1 ENTER
                                 [1 2 3
                                  4 5 6]
mat▸list(M1) ENTER          {1 2 3 4 5 6}
```

max() Menu MATH/List

max(*expressão 1***, *** expressão 2***)** ⇒ *expressão*
max(*lista 1***, *** lista 2***)** ⇒ *lista*
max(*matriz1***, *** matriz 2***)** ⇒ *matriz*

Devolve o máximo de dois argumentos. Se os argumentos forem duas listas ou matrizes, devolve uma lista ou matriz com o valor máximo de cada par de elementos correspondentes.

```
max(2.3,1.4) ENTER                2.3
max({1,2},{-4,3}) ENTER          {1 3}
```

max(*lista***)** ⇒ *expressão*

Devolve o elemento máximo da *lista*.

```
max({0,1,-7,1.3,.5}) ENTER        1.3
```

max(*matrix1***)** ⇒ *matrix*

Devolve um vector de linha com o elemento máximo de cada coluna em *matriz 1*.

Nota: Consulte também **fMax()** e **min()**.

```
max([1,-3,7;-4,0,.3]) ENTER
                                 [1 0 7]
```

mean() Menu MATH/Statistics

mean(*lista***[***, lista de frequência***])** ⇒ *expressão*

Devolve a média dos elementos da lista.

Cada elemento da *lista de frequência* conta o número de ocorrências consecutivas do elemento correspondente na lista.

```
mean({.2,0,1,-.3,.4}) ENTER       .26
mean({1,2,3},{3,2,1}) ENTER       5/3
```

mean(*matriz1***[***, matriz de frequência***])** ⇒ *matriz*

Devolve um vector de linha das médias de todas as colunas da *matriz 1*.

Cada elemento da *matriz de frequência* conta o número de ocorrências consecutivas do elemento correspondente na *matriz 1*.

```
No modo rectangular de formato do
vector:
média ([.2,0;1,3;.4,.5]) .833...]
mean([1/5,0;-1,3;2/5,-1/2]) ENTER
                          [-2/15  5/6]
mean([1,2;3,4;5,6],[5,3;4,1;
6,2]) ENTER             [47/15, 11/3]
```

median() Menu MATH/Statistics

median(*lista***)** ⇒ *expressão*

Devolve a mediana dos elementos da *lista 1*.

```
median({.2,0,1,-.3,.4}) ENTER     .2
```

median(*matriz 1***)** ⇒ *matriz*

Devolve um vector de linha com as medianas das colunas da *matriz 1*.

Nota: Todas as entradas da lista ou matriz têm de ser simplificadas para números.

```
median([.2,0;1,-.3;.4,-.5]) ENTER
                            [.4 -.3]
```

MedMed — Menu MATH/Statistics/Regressions

MedMed *lista 1*, *lista 2*[, [*lista 3*] [, *lista 4*, *lista 5*]]

Calcula a linha de mediana-mediana e actualiza todas as variáveis estatísticas do sistema.

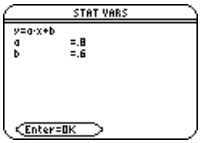
Todas as listas têm de ter dimensões iguais, excepto a *lista 5*.

lista 1 representa xlist.
lista 2 representa ylist.
lista 3 representa a frequência.
lista 4 representa os códigos de categorias.
lista 5 representa a lista de inclusão de categorias.

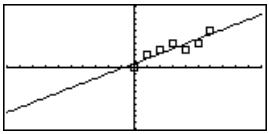
Nota: De *lista 1* a *lista 4* têm de ser um nome da variável ou c1–c99 (colunas da última variável de dados apresentada no Editor de Dados/Matriz). *lista 5* não tem de ser o nome de uma variável e não pode ser c1–c99.

No modo de representação gráfica da função.

```
{0,1,2,3,4,5,6}→L1 ENTER        {0 1 2 ...}
{0,2,3,4,3,4,6}→L2 ENTER        {0 2 3 ...}
MedMed L1,L2 ENTER                    Done
ShowStat ENTER
```




```
ENTER
Regeq(x)→y1(x) ENTER                  Done
NewPlot 1,1,L1,L2 ENTER               Done
◆[GRAPH]
```



mid() — Menu MATH/String

mid(*Cadeia de origem*, *início*[, *contagem*]**)** ⇒ *cadeia*

Devolve os caracteres *contagem* a partir da cadeia de caracteres *Cadeia de origem*, começando pelo número de caracteres início.

Se a *contagem* for omitida ou maior do que a dimensão da *Cadeia de origem*, devolve todos os caracteres da *Cadeia de origem*, começando pelo número de caracteres início.

contagem tem de ser ≥ 0. Se a *contagem* = 0, devolve uma cadeia vazia.

```
mid("Hello there",2) ENTER
                    "ello there"
mid("Hello there",7,3) ENTER
                            "the"
mid("Hello there",1,5) ENTER
mid("Hello there",1,0) ENTER
                              ""
```

mid(*Lista de origem*, *início* [, *contagem*]**)** ⇒ *lista*

Devolve os elementos da *contagem* da *Lista de origem*, começando pelo número de elementos *início*.

Se a *contagem* for omitida ou maior do que a dimensão da *Lista de origem*, devolve todos os elementos da *Lista de origem*, começando pelo número de elementos *início*.

contagem tem de ser ≥ 0. Se a *contagem* = 0, devolve uma cadeia vazia.

```
mid({9,8,7,6},3) ENTER          {7 6}
mid({9,8,7,6},2,2) ENTER        {8 7}
mid({9,8,7,6},1,2) ENTER        {9 8}
mid({9,8,7,6},1,0) ENTER            {}
```

mid(*Lista da cadeia de origem*, *início* [, *contagem*]**)** ⇒ *lista*

Devolve as cadeias *contagem* da lista de cadeias *Lista de cadeias de origem*, começando pelo número de elementos início.

```
mid({"A","B","C","D"},2,2) ENTER
                      {"B" "C"}
```

min() — Menu MATH/List

min(*expressão 1***,** *expressão 2***)** ⇒ *expressão*
min(*lista 1, lista 2***)** ⇒ *lista*
min(*matriz 1, matriz 2***)** ⇒ *matriz*

Devolve o mínimo dos dois argumentos. Se os argumentos forem duas listas ou matrizes, devolve uma lista ou matriz com o valor mínimo de cada par de elementos correspondentes.

```
min(2.3,1.4) ENTER                    1.4
min({1,2},{-4,3}) ENTER            {-4 2}
```

---

min(*lista***)** ⇒ *expressão*

Devolve o elemento mínimo da lista.

```
min({0,1,-7,1.3,.5}) ENTER            -7
```

---

min(*matriz 1***)** ⇒ *matriz*

Devolve um vector de linha com o elemento mínimo de cada coluna em *matriz 1*.

Nota: Consulte também **fMin()** e **max()**.

```
min([1,-3,7;-4,0,.3]) ENTER
                          [-4 -3 .3]
```

mod() — Menu MATH/Number

mod(*expressão 1***,** *expressão 2***)** ⇒ *expressão*
mod(*lista 1***,** *lista 2***)** ⇒ *lista*
mod(*matriz 1***,** *matriz 2***)** ⇒ *matriz*

Devolve o primeiro argumento e o segundo argumento como definido pelas identidades:

mod(x,0) x
mod(x,y) ≡ x− y floor(x/y)

Quando o segundo argumento for não zero, o resultado é periódico nesse argumento. O resultado é zero ou tem o mesmo sinal d segundo argumento.

Se os argumentos forem duas listas ou matrizes, devolve uma lista ou matriz com o módulo de cada par de elementos correspondentes.

Nota: Consulte também **remain()**.

```
mod(7,0) ENTER                        7
mod(7,3) ENTER                        1
mod(-7,3) ENTER                       2
mod(7,-3) ENTER                      -2
mod(-7,-3) ENTER                     -1
mod({12,-14,16},{9,7,-5}) ENTER
                          {3 0 -4}
```

MoveVar — CATALOG

MoveVar *var***,** *oldFolder***,** *newFolder*

Move a variável *var* de *oldFolder* para *newFolder*. Se a *newFolder* não existir, **MoveVar** cria-a.

```
{1,2,3,4}→L1 ENTER                {1 2 3 4}
MoveVar L1,Main,Games ENTER            Done
```

mRow() — Menu MATH/Matrix/Row

mRow(*expressão***,** *matriz 1***,** *índice***)** ⇒ *matrix*

Devolve uma cópia de *matriz 1* com cada elemento na linha *índice* de *matriz 1* multiplicada por *expressão*.

```
mRow(-1/3,[1,2;3,4],2) ENTER
                          [1  2
                          -1  -4/3]
```

mRowAdd() — Menu MATH/Matrix/Row

mRowAdd(*expressão***,** *matriz 1***,** *índice 1***,** *índice 2***)** ⇒ *matriz*

Devolve uma cópia de *matriz 1* com cada elemento na linha *índice 2* de *matriz 1* substituída por:

expressão× linha *índice 1* + linha *índice 2*

```
mRowAdd(-3,[1,2;3,4],1,2) ENTER
                          [1  2
                          0  -2]
mRowAdd(n,[a,b;c,d],1,2) ENTER
                          [a        b
                          a•n+c    b•n+d]
```

nCr() Menu MATH/Probability

nCr(*expressão 1***,** *expressão 2***)** ⇒ *expressão*

Para os números inteiros de *expressão 1* e *expressão 2* com *expressão 1* ≥ *expressão 2* ≥ 0, **nCr()** é o número de combinações de itens da *expressão 1* removidos da *expressão 2* em simultâneo. (Isto é também conhecido como um coeficiente binomial.) Ambos os argumentos podem ser números inteiros ou expressões simbólicas.

```
nCr(z,3)
```

$$\frac{z\cdot(z-2)\cdot(z-1)}{6}$$

```
ans(1)[2]
```

10

```
nCr(z,c)
```

$$\frac{z!}{c!(z-c)!}$$

```
ans(1)/nPr(z,c)
```

$$\frac{1}{c!}$$

nCr(*expressão,* 0**)** ⇒ 1

nCr(*expressã , negInteger***)** ⇒ 0

nCr(*expressão , posInteger***)** ⇒
expressão • (*expressão*– 1)... (*expressão*– *posInteger*+1)/ *posInteger*!

nCr(*expressão, nonInteger***)** ⇒ *expressão*!/ ((*expressão*– *nonInteger*)! • *nonInteger*!)

---

nCr(*lista 1***,** *lista 2***)** ⇒ *list*

Devolve uma lista de combinações baseada nos pares de elementos correspondentes das duas listas. Os argumentos têm de ter a mesma lista de tamanhos.

```
nCr({5,4,3},{2,4,2}) ENTER
{10  1  3}
```

---

nCr(*matriz 1***,** *matriz 2***)** ⇒ *matriz*

Devolve uma matriz de combinações baseada nos pares de elementos correspondentes das duas matrizes. Os argumentos têm de ter a mesma matriz de tamanhos.

```
nCr([6,5;4,3],[2,2;2,2])
```

nDeriv() Menu MATH/Calculus

nDeriv(*expressão 1***,** *var***[,** *h***])** ⇒ *expressão*
nDeriv(*expressão 1***,** *var, lista***)** ⇒ *lista*
nDeriv(*lista***,** *var***[,** *h***])** ⇒ *lista*
nDeriv(*matriz***,** *var***[,** *h***])** ⇒ *matriz*

Devolve a derivada numérica como uma expressão. Utiliza a fórmula do quociente de diferença central.

hé o valor do passo. Se h for omitido, é predefinido para 0.001.

Quando utilizar *lista* ou *matriz*, a operação mapeia através dos valores na lista ou dos elementos da matriz.

Nota: Consulte também **avgRC()** e ***d*()**.

```
nDeriv(cos(x),x,h)
limit(nDeriv(cos(x),x,h),h,0)
nDeriv(x^3,x,0.01) ENTER
```

$3.\cdot(x^2+.000033)$

```
nDeriv(cos(x),x)|x=π/2 ENTER
-1.
nDeriv(x^2,x,{.01,.1}) ENTER
{2.•x  2.•x}
```

NewData CATALOG

NewData *dataVar***,** *lista 1***[,** *lista 2***]** **[,** *lista 3***]...**

Cria a variável de dados *dataVar*, em que as colunas são as listas por ordem.

Tem de ter pelo menos uma lista.

lista 1, lista 2, ..., lista n podem ser listas como mostradas, expressões que resolvam para listas, ou os nomes das variáveis da lista.

NewData torna a nova variável actual no Editor de Dados/Matriz.

```
NewData mydata,{1,2,3},{4,5,6} ENTER
Done
```

(Vá para o Editor de Dados/Matriz e abra a *var* mydata para ver a variável de dados abaixo.)

DATA			
	c1	c2	c3
1	1	4	
2	2	5	
3	3	6	
4			

NewData *dataVar, matriz*

Cria a variável de dados *dataVar* baseada na matriz.

NewData sysData, *matriz*

Carrega o conteúdo da *matriz* nas variáveis de dados do sistema sysData.

NewFold CATALOG

NewFold *Nome da pasta*

Cria uma pasta definida pelo utilizador com o nome *Nome da pasta* e, em seguida, define a pasta actual para essa pasta. Depois de executar esta instrução, está na pasta nova.

```
NewFold games ENTER        Done
```

newList() CATALOG

newList(*numElements***)** ⇒ *lista*

Devolve uma lista com uma dimensão de *numElements*. Cada elemento é zero.

```
newList(4) ENTER           {0 0 0 0}
```

newMat() CATALOG também menu Math/Matrix

newMat(*numRows***,** *numColumns***)** ⇒ *matrix*

Devolve uma matriz de zeros com a dimensão *numRows* por *numColumns*.

```
newMat(2,3) ENTER          [0 0 0
                           0 0 0]
```

NewPic CATALOG

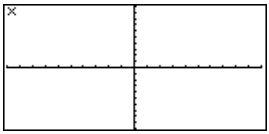
NewPic *matriz***,** *picVar* [, *Linha máxima*][, *Coluna máxima*]

Cria uma variável pic *picVar* baseada na *matriz*. *matriz* tem de ser uma matriz *n×2* em que cada linha representa um pixel. As coordenadas dos pixéis começam em 0,0. Se *picVar* já existir, **NewPic** substitui-a.

A predefinição para *picVar* é a área mínima necessária para os valores da matriz. Os argumentos opcionais *Linha máxima* e *Coluna máxima* determinam os limites máximos para *picVar*.

```
NewPic [1,1;2,2;3,3;4,4;5,5;
5,1;4,2;2,4;1,5],xpic ENTER   Done

RclPic xpic ENTER
```



NewPlot CATALOG

NewPlot *n*, *tipo*, *ListaX* [,[*ListaY*], [*Lista de frequências*], [*Lista de categorias*],
[*Incluir lista de categorias*], [*marca*] [, *Tamanho do balde*]]

Cria uma nova definição de desenho para o número de desenho *n*.

tipo especifica o tipo de desenho do gráfico.
1 = desenho de dispersão
2 = desenho xyline
3 = desenho de caixa
4 = histograma
5 = desenho de caixa modificada

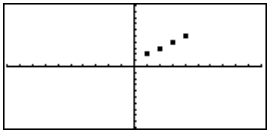
marca especifica o tipo de visualização da marca.
1 = ▫ (caixa)
2 = × (cruz)
3 = + (mais)
4 = ▪ (quadrado)
5 = • (ponto)

tamanho do balde é a largura de cada histograma "bucket" (*tipo* = 4), e varia baseado nas variáveis da janela xmin e xmax. *tamanho do balde* do tem de ser >0. Predefinição = 1.

Nota: As listas têm de ser nomes de variáveis ou c1–c99 (colunas na última variável de dados apresentada no Editor de Dados/Matriz), excepto para *Incluir lista de categorias*, que não tem de ser o nome de uma variável e não pode ser

```
FnOff ENTER                          Done
PlotsOff ENTER                       Done
{1,2,3,4}→L1 ENTER              {1 2 3 4}
{2,3,4,5}→L2 ENTER              {2 3 4 5}
NewPlot 1,1,L1,L2,,,,4 ENTER         Done
```

Prima ◆ [GRAPH] para ver:



NewProb CATALOG

NewProb

```
NewProb ENTER                        Done
```

Efectua uma variedade de operações que lhe permitem iniciar um novo problema a partir de um estado livre sem reiniciar a memória.

- Apaga todos os nomes de variáveis de carácter individual (Apagar a–z) na pasta actual, excepto se as variáveis estiverem bloqueadas ou arquivadas.
- Desliga todas as funções e desenhos estatísticos (**FnOff** e **PlotsOff**) no modo de representação gráfica actual.
- Efectua **ClrDraw**, **ClrErr**, **ClrGraph**, **ClrHome**, **ClrIO**, e **ClrTable**.

nInt() Menu MATH/Calculus

nInt(*expressão 1, var, inferior, superior)* ⇒ *expressão*

```
nInt(e^(-x^2),x,-1,1) ENTER
                                 1.493...
```

Se a expressão a integrar *expressão 1* só contiver a variável *var*, e se *inferior* e *superior* forem constantes, positivo ∞, ou negativo ∞, **nInt()** devolve uma aproximação de ∫**(***expressão 1*, *var*, *inferior* , *superior***)**. Esta aproximação é uma média ponderada de alguns valores amostrais da expressão a integrar no intervalo *inferior<var<superior*.

O objectivo é 6 dígitos significantes. O algoritmo adaptável termina quando paracer que provavelmente o objectivo foi atingido, ou quando parecer que improvavelmente as amostras adicionais produzem uma melhoria compensadora.

```
nInt(cos(x),x,-π,π+1E-12) ENTER
                                  -1.041...E-12
∫(cos(x),x,-π,π+10^(-12)) ENTER
                                  -sin(1/1000000000000)
```

Aparece um aviso ("Precisão questionável") quando parecer que o objectivo não foi atingido.

```
ans(1)◆ ENTER                     -1.E-12
```

Aninhe **nInt()** para fazer a integração numérica múltipla. Os limites de integração podem depender das variáveis de integração fora delas.

```
nInt(nInt(e^(-x*y)/√(x^2-y^2),
y,-x,x),x,0,1) ENTER              3.304...
```

Nota: Consulte também ∫**()**.

norm() Menu MATH/Matrix/Norms

norm(*matriz***)** ⇒ *expressão*

Devolve o tipo Frobenius.

```
tipo ([a,b;c,d])
```

$$\sqrt{a^2+b^2+c^2+d^2}$$

```
tipo ([1,2;3,4])
```

not Menu MATH/Test

not *Expressão booleana 1* ⇒ *Expressão booleana*

Devolve verdadeiro, falso ou uma *Expressão booleana 1* simplificada.

```
not 2>=3 ENTER                    true
not x<2 ENTER                     x ≥ 2
not not innocent ENTER            innocent
```

not *número inteiro 1* ⇒ *número inteiro*

Devolve um complemento de um número inteiro real. Internamente, o *número inteiro 1* é convertido para um número binário de 32 bits. O valor de cada bit é aumentado (0 torna-se 1 e vice-versa) para um complemento. Os resultados aparecem de acordo com o modo Base.

Pode introduzir o número em qualquer base do número. Para uma introdução binária ou hexadecimal, tem de utilizar o prefixo 0b ou 0h, respectivamente. Sem um prefixo, o número inteiro é tratado como decimal (base 10).

Se introduzir um número inteiro decimal muito grande para uma forma binária de 32 bits, é utilizada uma operação do módulo simétrico para colocar o valor no intervalo adequado.

No modo base Hex:

```
not 0h7AC36 ENTER                 0hFFF853C9
```

Importante: Zero, não a letra O.

No modo base Bin:

```
0b100101▸dec ENTER                37
not 0b100101 ENTER
0b11111111111111111111111111011010
ans(1)▸dec ENTER                  -38
```

Nota: Uma entrada binária pode ter até 32 dígitos (não contar o prefixo 0b). Uma entrada hexadecimal pode ter até 8 dígitos.

Nota: Para digitar o operador de conversão ▸ prima [2nd] [▸]. Pode também seleccionar as conversões da base a partir do menu MATH/Base.

nPr() Menu MATH/Probability

nPr(*expressão 1***,** *expressão 2***)** ⇒ *expressão*

Para os números inteiros de *expressão 1* e *expressão 2* com expressão 1 ≥ *expressão 2* ≥ 0, **nPr()** é o número de permutações de itens de *expressão 1* removidos da expressão 2 de uma vez. Ambos os argumentos podem ser números inteiros ou expressões simbólicas.

```
nPr(z,3) ENTER                    z•(z-2)•(z-1)
ans(1)z=560
nPr(z,⁻3) ENTER                   1/((z+1)•(z+2)•(z+3))
nPr(z,c) ENTER                    z!/(z-c)!
ans(1)*nPr(z-c,⁻c) ENTER          1
```

nPr(*expressão,* 0**)** ⇒ 1

nPr(*expressão, negInteger***)** ⇒
1/((*expressão*+1) • (*expressão*+2)... (*expressão* – *negInteger*))

nPr(*expressão, posInteger***)** ⇒
expressão • (*expressão*– 1)... (*expressão* – *posInteger*+1)

nPr(*expressão, nonInteger***)** ⇒ *expressão*!/
(*expressão* – *nonInteger*)!

---

nPr(*lista 1***,** *lista 2***)** ⇒ *lista*

Devolve uma lista de permutações baseada nos pares de elementos correspondentes nas duas listas. Os argumentos têm de ter a mesma lista de tamanhos.

```
nPr({5,4,3},{2,4,2}) ENTER
                                  {20 24 6}
```

---

nPr(*matriz 1***,** *matriz 2***)** ⇒ *matriz*

Devolve uma matriz de permutações baseada nos pares de elementos correspondentes das duas matrizes. Os argumentos têm de ter a mesma matriz de tamanhos.

```
nPr([6,5;4,3],[2,2;2,2])
```

nSolve() Menu MATH/Algebra

nSolve(*expressão***,** *varOrGuess***)** ⇒ *número da cadeia de erro*

Procura repetidamente uma solução numérica real aproximada para a *equação* para uma variável. Especifique *varOrGuess* como:

variável
– ou –
variável = *número real*

Por exemplo, **x** é válido e, por isso, é **x=3**.

nSolve() é muitas vezes mais rápida do que **solve()** ou **zeros()**, em especial, se utilizar o operador "|" para restringir a procura a um pequeno intervalo com exactamente uma solução simples.

nSolve() tenta determinar um ponto em que o residual é zero ou dois pontos relativamente fechados em que o residual tem sinais opostos e a magnitude do residual não é excessiva. Se não conseguir atingir este com um número modesto de pontos amostrais, devolve a cadeia "no solution found."

Se utilizar **nSolve()** num programa, pode utilizar **getType()** para procurar um resultado numérico antes de o utilizar numa expressão algébrica.

Nota: Consulte também **cSolve()**, **cZeros()**, **solve()**, e **zeros()**.

```
nSolve(x^2+5x-25=9,x) ENTER
                                  3.844...
nSolve(x^2=4,x=⁻1) ENTER          ⁻2.
nSolve(x^2=4,x=1) ENTER           2.
```

Nota: Se existirem várias soluções, pode utilizar uma hipótese para ajudar a localizar uma solução específica.

```
nSolve(x^2+5x-25=9,x)|x<0 ENTER
                                  ⁻8.844...
nSolve(((1+r)^24-1)/r=26,r)|r>0 and
r<.25 ENTER                       .0068...
nSolve(x^2=⁻1,x) ENTER
                                  "no solution found"
```

OneVar — Menu MATH/Statistics

OneVar *lista 1* [[, *lista 2*] [, *lista 3*] [, *lista 4*]]

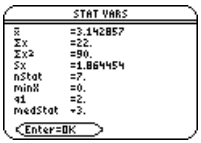
Calcula estatísticas de 1 variável e actualiza todas as variáveis estatísticas do sistema.

Todas as listas têm de ter dimensões iguais, excepto a *lista 4*.

lista 1 representa xlist.
lista 2 representa a frequência.
lista 3 representa os códigos de categorias.
lista 4 representa a lista de inclusão de categorias.

Nota: De *lista 1* a *lista 3* têm de ser um nome da variável ou c1–c99 (colunas da última variável de dados apresentada no Editor de Dados/Matriz). *lista 4* não tem de ser o nome de uma variável e não pode ser c1–c99.

```
{0,2,3,4,3,4,6}→L1 ENTER
OneVar L1 ENTER                    Done
ShowStat ENTER
```



ou — Menu MATH/Test

Expressão booleana 1 **or** *Expressão booleana 2* ⇒ *Expressão booleana*

Devolve verdadeiro ou falso, ou uma forma simplificada da entrada original.

Devolve verdadeiro se uma ou ambas as expressões simplificarem para verdadeiro. Devolve falso apenas se ambas as expressões avaliarem para falso.

Nota: Consulte **xor**.

```
x≥3 or x≥4 ENTER                   x ≥ 3
```

Segmento do programa:

```
                                   ⋮
If x<0 or x≥5
  Goto END
                                   ⋮
If choice=1 or choice=2
  Disp "Wrong choice"
                                   ⋮
```

---

número inteiro 1 **or** *número inteiro 2* ⇒ *número inteiro*

Compara dois números inteiros reais bit a bit com uma operação **or**. Internalmente, ambos os números inteiros são convertidos para números binários de 32 bits. Quando bits correspondentes forem comparados, o resultado é 1 se um dos bits for 1; o resultado é 0 apenas se ambos os bits forem 0. O valor devolvido representa os resultados das partes e aparece de acordo com o modo Base.

Pode introduzir os números inteiros em qualquer base do número. Para uma introdução binária ou hexadecimal, tem de utilizar o prefixo 0b ou 0h, respectivamente. Sem um prefixo, os números inteiros são tratados como decimal (base 10).

Se introduzir um número inteiro decimal muito grande para uma forma binária de 32 bits, é utilizada uma operação do módulo simétrico para colocar o valor no intervalo adequado.

Nota: Consulte **xor**.

No modo base Hex:

```
0h7AC36 or 0h3D5F ENTER            0h7BD7F
```

Importante: Zero, not the letter O.

No modo base Bin:

```
0b100101 or 0b100 ENTER            0b100101
```

Nota: Uma entrada binária pode ter até 32 dígitos (não contar o prefixo 0b). Uma entrada hexadecimal pode ter até 8 dígitos.

ord() — Menu MATH/String

ord(*cadeia***)** ⇒ *número inteiro*
ord(*lista 1***)** ⇒ *lista*

Devolve o código numérico do primeiro carácter na *cadeia* de caracteres cadeia ou uma lista dos primeiros caracteres de cada elemento da lista.

Consulte o Anexo B para obter uma lista de códigos de caracteres completa.

```
ord("hello") ENTER                 104
char(104) ENTER                    "h"
ord(char(24)) ENTER                24
ord({"alpha","beta"}) ENTER
                                   {97  98}
```

Output CATALOG

Output *linha,coluna*, *Expressão ou Cadeia*

Mostra *Expressão ou Cadeia* (uma expressão ou cadeia de caracteres) no ecrã Program I/O nas coordenadas do texto (*linha,coluna*).

Uma expressão pode incluir operações de conversão como ▸**DD** e ▸**Rect**. Pode também utilizar o operador ▸ para efectuar conversões da base do número e da unidade.

Se Pretty Print = ON, *Expressão ou Cadeia* é "pretty printed."

No ecrã Program I/O, pode premir [F5] para ver o ecrã inicial ou **DispHome** num programa.

Segmento do programa:

```
   ⋮
:RandSeed 1147
:ClrIO
:For i,1,90,10
:  Output i, rand(100),"Hello"
:EndFor
   ⋮
```

Resultado depois da execução:

```
        Hello
   Hello
          Hello
      Hello
              Hello
     Hello
                Hello
```

PRx() Menu MATH/Angle

P▸Rx(*expressão*, θ*expressão***)** ⇒ *expressão*
P▸Rx(*rLista*, θ*Lista***)** ⇒ *lista*
P▸Rx(*rMatriz*, θ*Matriz***)** ⇒ *matriz*

Devolve a coordenada x equivalente do par (r, θ).

Nota: O argumento θ é interpretado como um ângulo de graus, grados ou radianos de acordo com o modo de ângulo actual. Se o argumento for uma expressão, pode utilizar °, ᴳ ou ʳ para substituir a definição do modo de ângulo temporariamente.

No modo de ângulo Degree:

P▸Rx(r,θ) [ENTER] cos(θ)•r

P▸Rx(4,60°) [ENTER] 2

P▸Rx({⁻3,10,1.3},{π/3,⁻π/4,0}) [ENTER]

$$\left\{-3/2 \quad 5\cdot\sqrt{2} \quad 1.3\right\}$$

P▸Ry() Menu MATH/Angle

P▸Ry(*expressão*, θ*expressão***)** ⇒ *expexpressãoression*
P▸Ry(*rLista*, θ*Lista***)** ⇒ *lista*
P▸Ry(*rMatriz*, θ*Matriz***)** ⇒ *matriz*

Devolve a coordenada x equivalente do par (r, θ).

Nota: O argumento θ é interpretado como um ângulo de graus, grados ou radianos de acordo com o modo de ângulo actual. Se o argumento for uma expressão, pode utilizar °, ᴳ ou ʳ para substituir a definição do modo de ângulo temporariamente.

No modo de ângulo Radian:

P▸Ry(r,θ) [ENTER] sin(θ)•r

P▸Ry(4,60°) [ENTER] 2•√3

P▸Ry({⁻3,10,1.3},{π/3,⁻π/4,0}) [ENTER]

$$\left\{\frac{-3\cdot\sqrt{3}}{2} \quad -5\cdot\sqrt{2} \quad 0.\right\}$$

part() CATALOG

part(*expressão 1***[** ,*Número inteiro não negativo***])**

Esta função de programação avançada permite identificar e extrair todas as subexpressões no resultado simplificado da *expressão 1*.

Por exemplo, se a *expressão 1* for simplificada para cos(π∗ x+3):

- A função **cos()** tem um argumento: (π∗ x+3).
- A soma de (π∗ x+3) tem dois operandos: π∗ x e 3.
- O número 3 não tem argumentos nem operandos.
- O produto π∗ x tem dois operandos: π e x.
- A variável x e a constante simbólica π não têm argumentos nem operandos.

Se x tiver um valor numérico e premir ◆ ENTER, o valor numérico de π∗ x é calculado, o resultado é adicionado a 3 e, em seguida, o co-seno é calculado. **cos()** é o operador de **top-level** porque é aplicado **last**.

---

part(*expressão 1***)** ⇒ ***número***

Simplifica a *expressão 1* e devolve o número de operandos e argumentos de nível superior. Este devolve 0 se a expressão 1 for um número, variável ou constante simbólica como π, *e*, ***i***, ou ∞.

```
part(cos(π∗x+3)) ENTER                    1
```

Nota: cos(π∗ x+3) tem um argumento.

part(*expressão 1***, 0)** ⇒ *cadeia*

Simplifica a *expressão 1* e devolve uma cadeia que contém o operador ou nome da função de nível superior. Este devolve a **string(***expressão 1***)** se *expressão 1* for um número, variável ou constante simbólica como π, *e*, ***i***, ou ∞.

```
part(cos(π∗x+3),0) ENTER                "cos"
```

part(*expressão 1***, *n*)** ⇒ *expressão*

Simplifica *expressão 1* e devolve o operando ou o argumento n^{th} em que *n* é > 0 e ≤ o número de operandos ou argumentos de nível superior devolvido pela **part(***expressão 1***)**. Caso contrário, é devolvido um erro.

```
part(cos(π∗x+3),1) ENTER                3+π•x
```

Nota: A simplificação alterou a ordem do argumento.

Ao combinar as variações de **part()**, pode extrair todas as subexpressões no resultado simplificado de *expressão 1*. Como mostrado no exemplo da direita, pode guardar um argumento ou operando e, em seguida, utilizar **part()** para extrair mais subexpressões.

Nota: Quando utilizar **part()**, não confie em nenhuma ordem específicas nas somas e produtos.

```
part(cos(π∗x+3)) ENTER                    1
part(cos(π∗x+3),0) ENTER                "cos"
part(cos(π∗x+3),1)→temp ENTER
                                      3+π•x
temp ENTER                            π•x+3
part(temp,0) ENTER                      "+"
part(temp) ENTER                          2
part(temp,2) ENTER                        3
part(temp,1)→temp ENTER                 π•x
part(temp,0) ENTER                      "∗"
part(temp) ENTER                          2
part(temp,1) ENTER                        π
part(temp,2) ENTER                        x
```

Expressões como (x+y+z) e (x− y− z) são representadas internamente como (x+y)+z e (x− y)− z. Isto afecta os valores devolvidos para o primeiro e o segundo argumento. Existem razões técnicas para a **part**(x+y+z,1) devolver y+x em vez de x+y.

```
part(x+y+z) ENTER                         2
part(x+y+z,2) ENTER                       z
part(x+y+z,1) ENTER                     y+x
```

Similarmente, x∗ y∗ z é representado internamente como (x∗ y)∗ z. Existem também razões técnicas para o primeiro argumento ser devolvido como y•x x em vez de x•y.

```
part(x∗y∗z) ENTER                         2
part(x∗y∗z,2) ENTER                       z
part(x∗y∗z,1) ENTER                     y•x
```

Quando extrair subexpressões de uma matriz, não se esqueça de que as matrizes são guardadas como listas de listas, conforme ilustrado no exemplo da direita.

```
part([a,b,c;x,y,z],0) ENTER          "{"
part([a,b,c;x,y,z]) ENTER              2
part([a,b,c;x,y,z],2)→temp ENTER
                                 {x  y  z}
part(temp,0) ENTER                   "{"
part(temp) ENTER                       3
part(temp,3) ENTER                     z
delVar temp ENTER                    Done
```

TO exemplo da função do Editor de Programas utiliza **getType()** e **part()** para implementar parcialmente a diferenciação simbólica. O estudo e a finalização desta função pode ensiná-lo a diferenciar manualmente. Pode mesmo incluir funções que não podem diferenciar, como funções Bessel.

```
:d(y,x)
:Func
:Local f
:If getType(y)="VAR"
:  Return when(y=x,1,0,0)
:If part(y)=0
:  Return 0 ◉ y=π,∞,i,numbers
:part(y,0)→f
:If f="-" ◉ if negate
:  Return ⁻d(part(y,1),x)
:If f="–" ◉ if minus
:  Return d(part(y,1),x)
     -d(part(y,2),x)
:If f="+"
:  Return d(part(y,1),x)
     +d(part(y,2),x)
:If f="*"
:  Return part(y,1)*d(part(y,2),x)
     +part(y,2)*d(part(y,1),x)
:If f="{"
:  Return seq(d(part(y,k),x),
     k,1,part(y))
:Return undef
:EndFunc
```

PassErr CATALOG

PassErr

Passa um erro para o nível seguinte.

Se "errornum" for zero, **PassErr** não faz nada.

A condição **Else** do programa deve utilizar **ClrErr** ou **PassErr**. Se tiver de processar ou ignorar o erro, utilize **ClrErr**. Se não souber o que fazer com o erro, utilize **PassErr** para o enviar para o processador de erros seguinte. (Consulte também **ClrErr**.)

Consulte o exemplo da lista de programas **ClrErr**.

Pause CATALOG

Pause [*expressão*]

Suspende a execução do programa. Se incluir a *expressão*, mostra a *expressão* no ecrã Program I/O.

expressão pode incluir operações de conversão como **▸DD** e **▸Rect**. Pode também utilizar o operador ▸ para efectuar conversões da base do número e da unidade.

Se o resultado da *expressão* for muito grande para caber num ecrã, pode utilizar o cursor para percorrer o ecrã.

A execução do programa é executada quando premir [ENTER].

Segmento do programa:

```
  ⋮
:ClrIO
:DelVar temp
:1→temp[1]
:1→temp[2]
:Disp temp[2]
:◉ Guess the Pattern
:For i,3,20
:  temp[i-2]+temp[i-1]→temp[i]
:  Disp temp[i]
:  Disp temp,"Can you guess the
     next","number?"
:  Pause
:EndFor
  ⋮
```

PlotsOff CATALOG

PlotsOff [**1**] [**, 2**] [**, 3**] **...** [**, 9**]

Desliga os desenhos especificados para gráficos. Quando estiver no modo de 2 gráficos, afecta apenas o gráfico activo.

Se não existirem parâmetros, desliga todos os desenhos.

PlotsOff 1,2,5 [ENTER] Done

PlotsOff [ENTER] Done

PlotsOn CATALOG

PlotsOn [**1**] [**, 2**] [**, 3**] **...** [**, 9**]

Liga os desenhos especificados para gráficos. Quando estiver no modo de 2 gráficos, afecta apenas o gráfico activo.

Se não incluir quaisquer argumentos, liga todos os desenhos.

PlotsOn 2,4,5 [ENTER] Done

PlotsOn [ENTER] Done

▸Polar Menu MATH/Matrix/Vector

vectorPolar▸**Polar**

Mostra o vector em formato polar [r ∠θ]. O vector tem de ser de dimensão 2 e pode ser uma linha ou coluna.

Nota: **▸Polar** é uma instrução de formato de visualização, não uma função de conversão. Pode utilizá-la apenas no fim de uma linha de entrada e não actualiza ans.

Nota: Consulte também **▸Rect**.

[1,3.]▸Polar [ENTER]
[x,y]▸Polar [ENTER]

```
■[1  3.]▸Polar
          [3.16228 ∠ 1.24905]
■[x  y]▸Polar
[√(x²+y²) ∠ π·sign(y)/2 - tar▸
```

Valor complexo▸**Polar**

Mostra o *Vector complexo* em formato polar.

- O modo de ângulo Degree devolve (r∠θ).
- O modo de ângulo Radian devolve $re^{i\theta}$.

Valor complexo pode ter qualquer formato complexo. No entanto, uma entrada $re^{i\theta}$ provoca um erro no modo de ângulo Degree.

Nota: Tem de utilizar o parêntesis para uma entrada polar (r∠θ).

No modo de ângulo Radian:

3+4*i*▸Polar [ENTER] $e^{i\cdot\left(\frac{\pi}{2}-\tan^{-1}(3/4)\right)}\cdot 5$

(4∠π/3)▸Polar [ENTER] $e^{\frac{i\cdot\pi}{3}}\cdot 4$

No modo de ângulo Gradian:

4*i*▸Polar [ENTER] (4∠100)

No modo de ângulo Degree:

3+4*i*▸Polar [ENTER] $(5\angle 90-\tan^{-1}(3/4))$

polyEval() Menu MATH/List

polyEval(*lista 1***,** *expressão1***)** ⇒ *expressão*
polyEval(*lista 1***,** *lista 2***)** ⇒ *expressão*

Interpreta o primeiro argumento como o coeficiente de um polinómio de grau descendente, e devolve o polinómio avaliado para o valor do segundo argumento.

polyEval({a,b,c},x) [ENTER] $a\cdot x^2+b\cdot x+c$

polyEval({1,2,3,4},2) [ENTER] 26

polyEval({1,2,3,4},{2,⁻7}) [ENTER] {26 ⁻262}

PopUp CATALOG

PopUp *Lista de itens, var*

Mostra um menu pendente com as cadeias de caracteres a partir de *Lista de itens*, aguarda pelo utilizador para seleccionar um item e guarda o número da selecção em *var*.

Os elementos de *Lista de itens* têm de ser cadeias de caracteres: {*Cadeia de itens 1, Cadeia de itens 2, Cadeia de itens 3*, ...}

Se a *var* já existir e tiver um número de item válido, esse item aparece como a escolha predefinida.

Lista de itens tem de conter pelo menos uma escolha.

PopUp {"1990","1991","1992"},var1 [ENTER]

PowerReg — Menu MATH/Statistics/Regressions

PowerReg *lista 1*, *lista 2*[, [*lista 3*] [, *lista 4*, *lista 5*]]

Calcula a regressão da potência e actualiza todas as variáveis estatísticas do sistema.

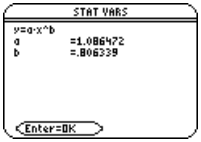
Todas as listas têm de ter dimensões iguais, excepto a *lista 5*.

lista 1 representa xlist.
lista 2 representa ylist.
lista 3 representa a frequência.
lista 4 representa os códigos de categorias.
lista 5 representa a lista de inclusão de categorias.

Nota: De *lista 1* a *lista 4* têm de ser um nome da variável ou c1–c99 (colunas da última variável de dados apresentada no Editor de Dados/Matriz). *lista 5* não tem de ser o nome de uma variável e não pode ser c1–c99.

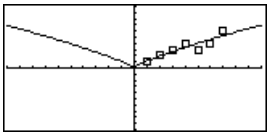
No modo de representação gráfica da função.

```
{1,2,3,4,5,6,7}→L1 ENTER
                                {1 2 3 ...}
{1,2,3,4,3,4,6}→L2 ENTER
                                {1 2 3 ...}
PowerReg L1,L2 ENTER                Done
ShowStat ENTER
```




```
ENTER
Regeq(x)→y1(x) ENTER                Done
NewPlot 1,1,L1,L2 ENTER             Done
```

◆[GRAPH]



Prgm — CATALOG

:Prgm
⋮
EndPrgm

Instrução necessária que identifica o início de um programa. A última linha do programa tem de ser EndPrgm.

Segmento do programa:

```
:prgmname()
:Prgm
:
:EndPrgm
```

Product (PI) — Consulte Π() na página

product() — Menu MATH/List

product(*lista***[**, *início***[**, *fim***]])** ⇒ *expressão*

Devolve a média dos elementos da *lista*. *início* e *fim* são opcionais. Especificam um intervalo de elementos.

```
product({1,2,3,4}) ENTER            24
product({2,x,y}) ENTER            2•x•y
product({4,5,8,9},2,3) ENTER        40
```

product(*matriz1***[**, *início***[**, *fim***]])** ⇒ *matriz*

Devolve um vector de linha com as somas dos elementos nas colunas em *matriz 1*. *início* e *fim* são opcionais. Especificam um intervalo de linhas.

```
sum([1,2,3;4,5,6;7,8,9]2880.162)
product([1,2,3;4,5,6;7,8,9],
1,2) ENTER                    [4,10,18]
```

Prompt — CATALOG

Prompt *var1*[, *var2*] [, *var3*] **...**

Mostra uma linha de comando no ecrã Program I/O para cada variável na lista de argumentos com a linha de comando var1?. Guarda a expressão introduzida na variável correspondente.

Linha de comando tem de ter pelo menos um argumento.

Segmento do programa:

```
                                    ⋮
Prompt A,B,C
                                    ⋮
EndPrgm
```

propFrac() Menu MATH/Algebra

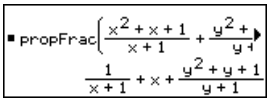
propFrac(*expressão1*[, *var*]**)** ⇒ *expressão*

propFrac(*número racional***)** devolve o número racional como a soma de um número inteiro e uma fracção com o mesmo sinal e uma maior magnitude de denominador do que magnitude de numerador.

```
propFrac(4/3) ENTER          1 + 1/3
propFrac(⁻4/3) ENTER         ⁻1– 1/3
```

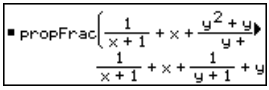
propFrac(*expressão racional*,*var***)** devolve a soma dos rácios adequados e um polinómio no que respeita a *var*. O grau de *var* no denominador excede o grau de *var* no numerador em cada rácio adequado. As potências similares de *var* são recolhidas. Os factores e os termos são ordenados com *var* como a variável principal.

```
propFrac((x^2+x+1)/(x+1)+
(y^2+y+1)/(y+1),x)
```



Se a *var* for omitida, uma expansão da fracção adequada é efectuada no que respeita à variável principal. Os coeficientes da parte polinomial são adequados à primeira variável principal, etc.

```
propFrac(ans(1))
```



Para expressões racionais, **propFrac()** é uma alternativa mais rápida, mas menos extrema para **expand()**.

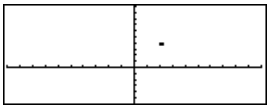
PtChg CATALOG

PtChg *x*, *y*
PtChg *xLista*, *yLista*

Mostra o ecrã Graph e inverte o pixel do ecrã mais próximo para as coordenadas da janela (*x*, *y*)

Nota: **PtChg** a **PtText** mostram exemplos similares contínuos.

```
PtChg 2,4 ENTER
```

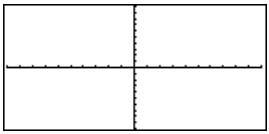


PtOff CATALOG

PtOff *x*, *y*
PtOff *xLista*, *yLista*

Mostra o ecrã Graph e desliga o pixel do ecrã mais próximo para as coordenadas da janela (*x*, *y*)

```
PtOff 2,4 ENTER
```

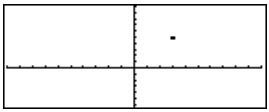


PtOn CATALOG

PtOn *x*, *y*
PtOn *xLista*, *yLista*

Mostra o ecrã Graph e liga o pixel do ecrã mais próximo para as coordenadas da janela (*x*, *y*)

```
PtOn 3,5 ENTER
```



ptTest() CATALOG

ptTest (*x*, *y***)** ⇒ *Expressão constante booleana*
ptTest (*xLista*, *yLista***)** ⇒ *Expressão constante booleana*

Devolve verdadeiro ou falso. Devolve verdadeiro apenas se o pixel do ecrã mais perto das coordenadas da janela (*x*,*y*) estiver ligado.

```
ptTest(3,5) ENTER            true
```

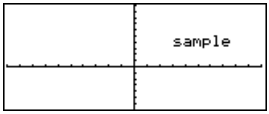
PtText CATALOG

PtText *cadeia, x, y*

Mostra o ecrã Graph e coloca a *cadeia* de caracteres string no ecrã no pixel mais próximo das coordenadas da janela especificadas (*x, y*).

cadeia é posicionada com o canto superior esquerdo do primeiro carácter nas coordenadas.

PtText "sample",3,5 ENTER



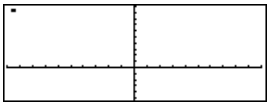
PxlChg CATALOG

PxlChg *row, col*
PxlChg *Lista de linhas, Lista de colunas*

Mostra o ecrã Graph e inverte o pixel nas coordenadas dos pixéis (*linha, col*).

Nota: A nova representação gráfica apaga todos os itens desenhados.

PxlChg 2,4 ENTER



PxlCrcl CATALOG

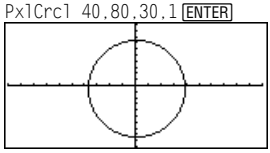
PxlCrcl linha, col, *r* [, *ModoDraw*]

Mostra o ecrã Graph e desenha uma circunferência centrada nas coordenadas dos pixéis (*linha, col*) com um raio de *r* pixéis.

Se o *ModoDraw* = 1, desenha a circunferência (predefinição).
Se o *ModoDraw* = 0, desliga a circunferência.
Se o *ModoDraw* = -1, inverte os pixéis ao longo da circunferência.

Nota: A nova representação gráfica apaga todos os itens desenhados. Consulte também **Circle**.

PxlCrcl 40,80,30,1 ENTER



PxlHorz CATALOG

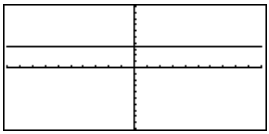
PxlHorz *linha* [, *ModoDraw*]

Mostra o ecrã Graph e desenha uma linha horizontal na posição do pixel *linha*.

Se o *ModoDraw* = 1, desenha a linha (predefinição).
Se o *ModoDraw* = 0, desliga a linha.
Se o *ModoDraw* = -1, desliga ou liga uma linha (inverte os pixéis ao longo da linha).

Nota: A nova representação gráfica apaga todos os itens desenhados. Consulte também **LinhaHorz**.

PxlHorz 25,1 ENTER



PxlLine CATALOG

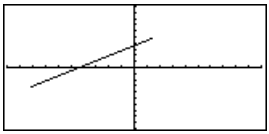
PxlLine *Início da linha, Início da coluna<, Fim da linha, Fim da coluna*[, *ModoDraw*]

Mostra o ecrã Graph e desenha uma linha entre as coordenadas de pixéis (*Início da linha, Início da coluna*) e (*Fim da linha, Fim da coluna*), incluindo ambos os pontos finais.

Se o *ModoDraw* = 1, desenha a linha (predefinição).
Se o *ModoDraw* = 0, desliga a linha.
Se o *ModoDraw* = -1, desliga ou liga uma linha (inverte os pixéis ao longo da linha).

Nota: A nova representação gráfica apaga todos os itens desenhados. Consulte também **Line**.

PxlLine 50,15,20,90,1 ENTER



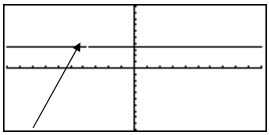
PxlOff CATALOG

PxlOff *linha*, *col*
PxlOff *Lista de linhas, Lista de colunas*

Mostra o ecrã Graph e desliga o pixel nas coordenadas dos pixéis (*linha, col*).

Nota: A nova representação gráfica apaga todos os itens desenhados.

```
PxlHorz 25,1 ENTER
PxlOff 25,50 ENTER
```



```
25,50
```

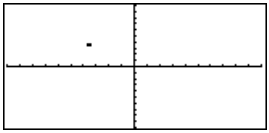
PxlOn CATALOG

PxlOn *linha*, *col*
PxlOn *Lista de linhas, Lista de colunas*

Mostra o ecrã Graph e liga o pixel às coordenadas dos pixéis (*linhas, col*).

Nota: A nova representação gráfica apaga todos os itens desenhados.

```
PxlOn 25,50 ENTER
```



pxlTest() CATALOG

pxlTest (*linha*, *col***)** ⇒ *Expressão booleana*
pxlTest (*Lista de linhas, Lista de colunas***)** ⇒ *Expressão booleana*

Devolve verdadeiro se o pixel nas coordenadas dos pixéis (*linha, col*) estiver ligado. Devolve falso se o pixel estiver desligado.

Nota: A nova representação gráfica apaga todos os itens desenhados.

```
PxlOn 25,50 ENTER
HOME
PxlTest(25,50) ENTER                    true
PxlOff 25,50 ENTER
HOME
PxlTest(25,50) ENTER                    false
```

PxlText CATALOG

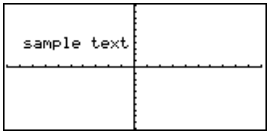
PxlText *cadeia, linha, col*

Mostra o ecrã Graph e coloca a cadeia de caracteres cadeia no ecrã, começando pelas coordenadas dos pixéis (*linha, col*).

cadeia é posicionada com o canto superior esquerdo do primeiro carácter nas coordenadas.

Nota: A nova representação gráfica apaga todos os itens desenhados.

```
PxlText "sample text",20,10 ENTER
```



PxlVert CATALOG

PxlVert *col* [, *ModoDraw*]

Desenha uma linha vertical no ecrã na posição dos pixéis *col*.

Se o *ModoDraw* = 1, desenha a linha (predefinição).
Se o *ModoDraw* = 0, desliga a linha.
Se o *ModoDraw* = -1, desliga ou liga uma linha (inverte os pixéis ao longo da linha).

Nota: A nova representação gráfica apaga todos os itens desenhados. Consulte também **LineVert**.

```
PxlVert 50,1 ENTER
```

QR — Menu MATH/Matrix

QR *matriz*, *NomeMatQ*, *NomeMatR*[, *tol*]

Calcula a factorização Householder QR numa matriz real ou complexa. As matrizes Q e R resultantes são guardadas nos NomesMat especificadas. A matriz Q é unitária. A matriz R é triangular superior.

Opcionalmente, qualquer elemento da matriz é tratado como zero se o valor absoluto for menor do que tol. Esta tolerância só é utilizada se a matriz tiver entradas de ponto flutuante e não contiver quaisquer variáveis simbólicas que não tenham um valor. Caso contrário, tol é ignorado.

- If you use [◆] [ENTER] ou definir o modo para Exact/Approx=APPROXIMATE, os cálculos são efectuados com a aritmética de ponto flutuante.
- ·Se *tol* for omitido ou não utilizado, a tolerância predefinida é calculada como:

  5E⁻14 ∗ **max(dim(***matriz***))** ∗ **rowNorm(***matriz***)**

O número de ponto flutuante (9.) em **m1** faz com que os resultados sejam calculados no formato de ponto flutuante.

[1,2,3;4,5,6;7,8,9.]→m1 [ENTER]

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9. \end{bmatrix}$$

QR m1,qm,rm [ENTER] Done

qm [ENTER]

$$\begin{bmatrix} .123\ldots & .904\ldots & .408\ldots \\ .492\ldots & .301\ldots & -.816\ldots \\ .861\ldots & -.301\ldots & .408\ldots \end{bmatrix}$$

rm [ENTER]

$$\begin{bmatrix} 8.124\ldots & 9.601\ldots & 11.078\ldots \\ 0. & .904\ldots & 1.809\ldots \\ 0. & 0. & 0. \end{bmatrix}$$

[m,n;o,p]→m1 [ENTER]

$$\begin{bmatrix} m & n \\ o & p \end{bmatrix}$$

A factorização QR é calculada numericamente com as transformações Householder. A solução simbólica é calculada com Gram-Schmidt. As colunas em *NomeMatQ* são vectores básicos ortonormais que abrangem o espaço definido pela matriz.

QR m1,qm,rm [ENTER] Done

qm [ENTER]

$$\begin{bmatrix} \frac{m}{\sqrt{m^2+o^2}} & \frac{-\text{sign}(m\cdot p-n\cdot o)\cdot o}{\sqrt{m^2+o^2}} \\ \frac{o}{\sqrt{m^2+o^2}} & \frac{m\cdot\text{sign}(m\cdot p-n\cdot o)}{\sqrt{m^2+o^2}} \end{bmatrix}$$

rm [ENTER]

$$\begin{bmatrix} \sqrt{m^2+o^2} & \frac{m\cdot n+o\cdot p}{\sqrt{m^2+o^2}} \\ 0 & \frac{|m\cdot p-n\cdot o|}{\sqrt{m^2+o^2}} \end{bmatrix}$$

QuadReg — Menu MATH/Statistics/Regressions

QuadReg , *lista 2*[, [*lista 3*] [, *lista 4*, *lista 5*]]

Calcula a regressão polinomial quadrática e actualiza as variáveis estatísticas do sistema.

Todas as listas têm de ter dimensões iguais, excepto a *lista 4*.

lista 1 representa xlist.
lista 2 representa ylist.
lista 3 representa a frequência.
lista 4 representa os códigos de categorias.
lista 5 representa a lista de inclusão de categorias.

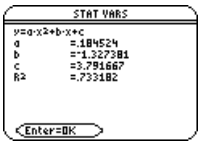
No modo de representação gráfica da função.

{0,1,2,3,4,5,6,7}→L1 [ENTER]
{1 2 3 ...}

{4,3,1,1,2,2,3,3}→L2 [ENTER]
{4 3 1 ...}

QuadReg L1,L2 [ENTER] Done

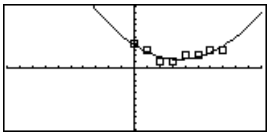
ShowStat [ENTER]



Nota: De *lista 1* a *lista 4* têm de ser o nome de uma variável ou c1–c99. (colunas na última variável de dados apresentada no Editor de Dados/Matriz). *lista 5* não tem de ser o nome de uma variável e não pode ser c1–c99

ENTER
Regeq(x)→y1(x) ENTER Done
NewPlot 1,1,L1,L2 ENTER Done

◆[GRAPH]



QuartReg — Menu MATH/Statistics/Regressions

QuartReg , *lista 2*[, [*lista 3*] [, *lista 4*, *lista 5*]]

Calcula a regressão polinomial quártica e actualiza as variáveis estatísticas do sistema.

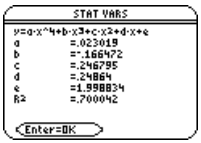
Todas as listas têm de ter dimensões iguais, excepto a *lista 5*.

lista 1 representa xlist.
lista 2 representa ylist.
lista 3 representa a frequência.
lista 4 representa os códigos de categorias.
lista 5 representa a lista de inclusão de categorias.

Nota: De *lista 1* a *lista 4* têm de ser um nome da variável ou c1–c99 (colunas da última variável de dados apresentada no Editor de Dados/Matriz). *lista 5* não tem de ser o nome de uma variável e não pode ser c1–c99.

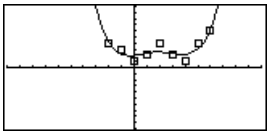
No modo de representação gráfica da função.

{⁻2,⁻1,0,1,2,3,4,5,6}→L1 ENTER
{⁻2 ⁻1 0 ...}
{4,3,1,2,4,2,1,4,6}→L2 ENTER
{4 3 1 ...}
QuartReg L1,L2 ENTER Done
ShowStat ENTER



ENTER
Regeq(x)→y1(x) ENTER Done
NewPlot 1,1,L1,L2 ENTER Done

◆[GRAPH]



R▸Pθ() — Menu MATH/Angle

R▸Pθ (*ExpressãoX***,** *ExpressãoY***)** ⇒ *expressão*
R▸Pθ (*ListaX***,** *ListaY***)** ⇒ *lista*
R▸Pθ (*MatrizX***,** *MatrizY***)** ⇒ *matriz*

Devolve a coordenada equivalente θ-dos argumentos dos pares (*x,y*).

Nota: O resultado é devolvido como um ângulo em graus, grados ou radianos, de acordo com a definição do modo de ângulo actual.

No modo de ângulo Degree:

RP(x,y)

■ R▸Pθ(x, y)
$90 \cdot \text{sign}(y) - \tan^{-1}\left(\frac{x}{y}\right)$

No modo de ângulo Gradian:

R▸Pθ(x,y) [ENTER]

■ R▸Pθ(x, y)
$100 \cdot \text{sign}(y) - \tan^{-1}\left(\frac{x}{y}\right)$

No modo de ângulo Radian:

R▸Pθ(3,2) [ENTER]
R▸Pθ([3,-4,2],[0,π/4,1.5]) [ENTER]

■ R▸Pθ(3, 2) $\tan^{-1}(2/3)$
■ R▸Pθ([3 -4 2],[0 $\frac{\pi}{4}$ 1.5▸
[0 $\tan^{-1}\left(\frac{16}{\pi}\right) + \frac{\pi}{2}$ .643501]

R▸Pr() — Menu MATH/Angle

R▸Pr (*xExpressão***,** *yExpressão***)** ⇒ *expressão*
R▸Pr (*ListaX***,** *ListaY***)** ⇒ *lista*
R▸Pr (*MatrizX***,** *MatrizY***)** ⇒ *matriz*

Devolve a coordenada equivalente r-dos argumentos dos pares (*x,y*).

No modo de ângulo Radian:

R▸Pr(3,2) [ENTER]
R▸Pr(x,y) [ENTER]
R▸Pr([3,-4,2],[0,π/4,1.5]) [ENTER]

■ R▸Pr(3, 2) $\sqrt{13}$
■ R▸Pr(x, y) $\sqrt{x^2 + y^2}$
■ R▸Pr([3 -4 2],[0 $\frac{\pi}{4}$ 1.5▸
[3 $\frac{\sqrt{\pi^2 + 256}}{4}$ 2.5]

▸Rad — Menu CATALOG/MATH/Angle

▸ Rad *expressão*

Converte uma expressão para uma medida de ângulo em radianos.

No modo de ângulo Degree:

1.5 ▸Rad [ENTER] .02618^R

No modo de ângulo Gradian:

1.5 ▸Rad [ENTER] .023562^R

rand() — Menu MATH/Probability

rand([*n***])** ⇒ *expressão*

n é um número inteiro ≠ zero.

Sem parâmetro, devolve o próximo número aleatório entre 0 e 1 na sequência. Quando um argumento for positivo, devolve um número inteiro aleatório no intervalo [1, *n*]
Quando um argumento for negativo, devolve um número inteiro aleatório no intervalo [⁻*n*, ⁻1]

RandSeed 1147 [ENTER] Done

(Define o grão do número aleatório.)

rand() [ENTER] .158...
rand(6) [ENTER] 5
rand(⁻100) [ENTER] ⁻49

randMat() Menu MATH/Probability

randMat(*Linhas de números, Colunas de números***)** ⇒ *matriz*

Devolve uma matriz de números inteiros entre -9 e 9 da dimensão especificada.

Ambos os argumentos têm de ser simplificados para números inteiros.

```
RandSeed 1147 [ENTER]                Done
randMat(3,3) [ENTER]
```

$$\begin{bmatrix} 8 & -3 & 6 \\ -2 & 3 & -6 \\ 0 & 4 & -6 \end{bmatrix}$$

Nota: Os valores desta matriz alteram-se sempre que premir [ENTER]

randNorm() Menu MATH/Probability

randNorm(*mèdia, sd***)** ⇒ *expressão*

Devolve um número decimal a partir da distribuição normal específica. Pode ser qualquer número real, mas estará fortemente concentrado no intervalo [*média*-3∗ *sd*, *média*+3∗ *sd*].

```
RandSeed 1147 [ENTER]                Done
randNorm(0,1) [ENTER]                .492...
randNorm(3,4.5) [ENTER]              -3.543...
```

randPoly() Menu MATH/Probability

randPoly(*var, ordem***)** ⇒ *expressão*

Devolve um polinómio em *var* da ordem especificada. Os coeficientes são números inteiros aleatórios no intervalo ⁻9 a 9. O coeficiente principal não é zero.

ordem tem de ser 0–99.

```
RandSeed 1147 [ENTER]                Done
randPoly(x,5) [ENTER]
```

$$-2 \cdot x^5+3 \cdot x^4-6 \cdot x^3+4 \cdot x-6$$

RandSeed Menu MATH/Probability

RandSeed *número*

Se o *número* = 0, define os grãos para as predefinições de fábrica para o gerador de números aleatórios. Se o *número*0, é utilizado para gerar dois grãos, que estão guardados nas variáveis do sistema seed1 e seed2.

```
RandSeed 1147 [ENTER]                Done
rand() [ENTER]                       .158...
```

RclGDB CATALOG

RclGDB *GDBvar*

Restaura todas as definições guardadas na variável da base de dados Graph GDBvar.

Para obter uma lista das definições, consulte **StoGDB**.

Nota: É necessário ter algo guardados em GDBvar antes de o poder restaurar.

```
RclGDB GDBvar [ENTER]                Done
```

RclPic CATALOG

RclPic *picVar* [**,** *linha, coluna*]

Mostra o ecrã Graph e adiciona a imagem guardada em picVar nas coordenadas dos pixéis do canto superior esquerdo (*linha, coluna*) com OR lógico.

picVar tem de ser um tipo de dados de imagem.

As coordenadas predefinidas são (0, 0).

real() — Menu MATH/Complex

real(*expressão1***)** ⇒ *expressão*

Devolve a parte real do argumento.

Nota: Todas as variáveis indefinidas são tratadas como variáveis reais. Consulte também **imag()**.

```
real(2+3i) ENTER          2
real(z) ENTER             z
real(x+iy) ENTER          x
```

real() ⇒ *lista*

Devolve as partes reais de todos os elementos.

```
real({a+i*b,3,i}) ENTER   {a  3  0}
```

real(*matriz1***)** ⇒ *matriz*

Devolve as partes reais de todos os elementos.

```
real([a+i*b,3;c,i]) ENTER
```

$$\begin{bmatrix} a & 3 \\ c & 0 \end{bmatrix}$$

▸Rect — Menu MATH/Matrix/Vector

vector **▸Rect**

Mostra o *vector* em formato rectangular [x, y, z]. O vector tem de ser de dimensão 2 ou 3 e pode ser uma linha ou coluna.

Nota: **▸Rect** é uma instrução de visualização, não uma função de conversão. Pode utilizá-la apenas no fim de uma linha de entrada e não actualiza ans.

Nota: Consulte também **▸Polar**.

```
[3,∠π/4,∠π/6]▸Rect ENTER
```

$$\left[\frac{3\cdot\sqrt{2}}{4}\quad\frac{3\cdot\sqrt{2}}{4}\quad\frac{3\cdot\sqrt{3}}{2}\right]$$

```
[a,∠b,∠c] ENTER     [a•cos(b)•sin(c)
                    a•sin(b)•sin(c)  a•cos(c)]
```

Valor complexo **▸Rect**

Mostra o *Valor complexo* em formato rectangular a+b. O *Valor complexo* pode ter qualquer formato complexo. No entanto, uma entrada $re^{i\theta}$ provoca um erro no modo de ângulo Degree.

Nota: Tem de utilizar o parêntesis para uma entrada polar (r∠θ).

No modo de ângulo Radian:

```
4e^(π/3)▸Rect ENTER
```

$$4\cdot e^{\frac{\pi}{3}}$$

```
(4∠π/3)▸Rect ENTER
```

$$2+2\cdot\sqrt{3}\cdot i$$

No modo de ângulo Gradian:

```
(1∠100)▸Rect ENTER
```

$$i$$

No modo de ângulo Degree:

```
(4∠60)▸Rect ENTER
```

$$2+2\cdot\sqrt{3}\cdot i$$

Nota: Para digitar **▸Rect** no teclado, prima [2nd] [▸] para o operador ▸. Para digitar ∠, prima [2nd] [∠].

ref() — Menu MATH/Matrix

ref(*matriz1***[,** *tol***])** ⇒ *matriz*

Devolve o tipo de escalão de linhas de *matriz 1*.

Opcionalmente, qualquer elemento da matriz é tratado como zero se o valor absoluto for menor do que *tol*. Esta tolerância só é utilizada se a matriz tiver entradas de ponto flutuante e não contiver quaisquer variáveis simbólicas que não tenham um valor. Caso contrário, *tol* é ignorado.

- Se utilizar ◆ ENTER ou definir o modo para Exact/Approx=APPROXIMATE, os cálculos são efectuados com a aritmética de ponto flutuante.
- Se *tol* for omitido ou não utilizado, a tolerância predefinida é calculada como:

  5E⁻14 ∗ **max(dim(***matriz1***))** ∗ **rowNorm(***matriz1***)**

Nota: Consulte também **rref().**

```
ref([-2,-2,0,-6;1,-1,9,-9;-5,
2,4,-4]) ENTER
```

$$\begin{bmatrix} 1 & -2/5 & -4/5 & 4/5 \\ 0 & 1 & 4/7 & 11/7 \\ 0 & 0 & 1 & -62/71 \end{bmatrix}$$

```
[a,b,c;e,f,g]→m1 ENTER
```

$$\begin{bmatrix} a & b & c \\ e & f & g \end{bmatrix}$$

```
ref(m1) ENTER
```

$$\begin{bmatrix} 1 & \frac{f}{e} & \frac{g}{e} \\ 0 & 1 & \frac{a\cdot g-c\cdot e}{a\cdot f-b\cdot e} \end{bmatrix}$$

Menu MATH/Number

remain(*expressão1***,** *expressão2***)** ⇒ *expressão*
remain(, *lista 2***)** ⇒ *lista*
remain(*matriz1***,** *matriz2***)** ⇒ *matriz*

Devolve o resto do primeiro argumento no que respeita ao segundo argumento como definido pelas identidades:

remain (x,0) x
remain(x,y) ≡ x− y∗iPart(x/y)

Por consequência, não se esqueça de que **remain(**− x,y**)** ≡ − **remain(**x,y**)**. O resultado é zero ou tem o mesmo sinal como o primeiro argumento.

Nota: Consulte também **mod()**.

```
remain(7,0) ENTER                    7
remain(7,3) ENTER                    1
remain(-7,3) ENTER                  -1
remain(7,-3) ENTER                   1
remain(-7,-3) ENTER                 -1
remain({12,-14,16},{9,7,-5}) ENTER
                              {3  0 1}
remain([9,-7;6,4],[4,3;4,-3]) ENTER
```

$$\begin{bmatrix} 1 & -1 \\ 2 & 1 \end{bmatrix}$$

Rename — CATALOG

Rename *oldVarName*, *newVarName*

Muda o nome da variável *oldVarName* como *newVarName*.

```
{1,2,3,4}→L1 ENTER          {1,2,3,4}
Rename L1, ENTER                 Done
ENTER                       {1,2,3,4}
```

Request — CATALOG

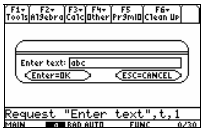
Request *promptString*, *var* [,*alphaOn/Off*]

Se Request estiver incluído numa construção Dialog...EndDlog, cria uma caixa de entrada na qual o utilizador pode escrever dados. Se for uma instrução autónoma, cria uma caixa de diálogo para esta entrada. Em qualquer um dos casos, se var contém uma cadeia de caracteres, será visualizada e realçada na caixa de entrada como uma opção predefinida. A Cadeia de comando deve ter ≤ 20 caracteres.

Esta instrução pode ser autónoma ou estar incluída numa construção de caixa de diálogo.

O argumento alphaOn/Off opcional podem ser qualquer expressão. Se for avaliado para zero, o

```
Request "Enter text",t,1 ENTER
```

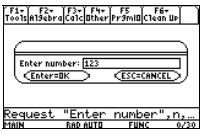


O argumento activou o bloqueio alfabético no exemplo acima.

```
Request "Enter number",n,0
```

bloqueio alfabético é definido como OFF. Se for avaliado para um valor diferente de zero, o bloqueio alfabético é definido como ON. Se o argumento opcional não for utilizado, o bloqueio alfabético é predefinido como ON.

Se aparecer mais do que um comando Request na construção Dialog...EndDlog, a primeira definição alfabética é utilizada e as restantes são ignoradas.



O argumento desactivou o bloqueio alfabético no exemplo acima.

Return — CATALOG

Devolve [*expressão*]

Devolve a *expressão* como resultado da função. Utilize um bloco **Func**...**EndFunc** ou um bloco **Prgm**...**EndPrgm**.

Nota: Utilize **Return** sem um argumento para sair de um programa.

Nota: Introduza o texto de uma linha no ecrã inicial (sem quebras de linha).

```
Define factoral(nn)=Func
:local answer,count:1→answer
:For count,1,nn
:answer*count→answer:EndFor
:Return answer:EndFunc ENTER          Done

factoral(3) ENTER                       6
```

right() — Menu MATH/List

right([, *num*]**)** ⇒ *lista*

Devolve os elementos *num* mais à direita contidos na lista 1.

Se omitir *num*, devolve tudo de lista 1.

```
right({1,3,-2,4},3) ENTER
                          {3 -2 4}
```

right(*Cadeia de origem*[, *num*]**)** ⇒ *cadeia*

Devolve os caracteres *num* mais à direita contidos na cadeia de caracteres Cadeia de origem.

Se omitir *num*, devolve tudo da Cadeia de origem.

```
right("Hello",2) ENTER                "lo"
```

right(*comparação***)** ⇒ *expressão*

Devolve o lado direito de uma equação ou desigualdade.

```
right(x<3) ENTER                        3
```

root() — Menu CATALOG/MATH/Number

root(*expressão***)** ⇒ *raiz*

Calcula uma raiz de índice n de x em que x pode ser uma constante de ponto flutuante complexa ou real, um número inteiro ou uma constante racional complexa ou uma expressão simbólica geral.

```
root(8,3) ENTER                         2

root(3,3) ENTER                         3^1/3

root(3,0,3) ENTER                 1.442249570
```

rotate() — Menu MATH/Base

rotate(*número inteiro 1*[,*# de rotações*]**)** ⇒ *número inteiro*

Roda os bits num número inteiro binário. Pode introduzir o número inteiro 1 em qualquer base do número; é convertido para um formato binário de 32 bits. Se a magnitude do *número inteiro 1* é também muito grande para este formato, uma operação do módulo simétrico trá-lo para o intervalo.

No modo base Bin:

```
rotate(0b1111010110000110101) ENTER
    0b10000000000000111101011000011010

rotate(256,1) ENTER          0b1000000000
```

Se *# de rotações* for positivo, a rotação vai para a esquerda. Se # de rotações for negativo, a rotação vai para a direita. A predefinição é ⁻1 (roda um bit para a direita).

Por exemplo, numa rotação para a direita:

Cada bit roda para a direita.

0b00000000000000111101011000011010 1

O bit mais à direita roda para mais à esquerda.

produz:

0b10000000000000011110101100001101 0

O resultado aparece de acordo com o modo Base.

No modo base Hex:

```
rotate(0h78E) ENTER                 0h3C7
rotate(0h78E,⁻2) ENTER        0h800001E3
rotate(0h78E,2) ENTER             0h1E38
```

Importante: Para introduzir um número binário ou hexadecimal, utilize sempre o prefixo 0b ou 0h (zero, não a letra O).

---

rotate([,*# de rotações*]**)** ⇒ *lista*

Devolve uma cópia de lista 1 rodada para a direita ou para a esquerda pelos elementos *# de rotações*. Não altere *lista 1*.

Se *# de rotações* for positivo, a rotação vai para a esquerda. Se *# de rotações* for negativo, a rotação vai para a direita. A predefinição é ⁻1 (roda um elemento para a direita).

No modo base Dec:

```
rotate({1,2,3,4}) ENTER
                          {4 1 2 3}
rotate({1,2,3,4},⁻2) ENTER
                          {3 4 1 2}
rotate({1,2,3,4},1) ENTER
                          {2 3 4 1}
```

---

rotate(*cadeia 1*[,*# de rotações*]**)** ⇒ *cadeia*

Devolve uma cópia de cadeia 1 rodada para a direita ou esquerda pelos caracteres *# de rotações*. Não altere a *cadeia 1*.

Se *# de rotações* for positivo, a rotação vai para a esquerda. Se *# de rotações* for negativo, a rotação vai para a direita. A predefinição é ⁻1 (rode um carácter para a direita).

```
rotate("abcd") ENTER              "dabc"
rotate("abcd",⁻2) ENTER           "cdab"
rotate("abcd",1) ENTER            "bcda"
```

round() Menu MATH/Number

round(*expressão 1*[, *dígitos*]**)** ⇒ *expressão*

Devolve o argumento arredondado para o número de dígitos especificado depois do ponto decimal.

dígitos tem de ser um número inteiro no intervalo 0–12. Se os *dígitos* não estiverem incluídos, devolve o argumento arredondado para 12 dígitos significantes.

Nota: O modo Display digits pode afectar a forma de visualização.

```
round(1.234567,3) ENTER           1.235
```

---

round([, *dígitos*]**)** ⇒ *lista*

Devolve uma lista dos elementos arredondados para o número de dígitos especificados.

```
round({π,√(2),ln(2)},4) ENTER
                 {3.1416 1.4142 .6931}
```

---

round(*matriz1*[, *dígitos*]**)** ⇒ *matriz*

Devolve uma matriz dos elementos arredondados para o número de dígitos especificados.

```
round([ln(5),ln(3);,e^(1)],1)
                        [1.6  1.1]
                        [3.1  2.7]
```

Menu MATH/Matrix/Row

rowAdd(*matriz1***,** *ÍndiceR 1***,** *ÍndiceR 2***)** ⇒ *matriz*

Devolve uma cópia de *matriz1* com a linha *ÍndiceR* substituída pela soma de linhas *ÍndiceR 1* e *ÍndiceR 2*.

```
rowAdd([3,4;-3,-2],1,2) ENTER
```

$$\begin{bmatrix} 3 & 4 \\ 0 & 2 \end{bmatrix}$$

```
rowAdd([a,b;c,d],1,2) ENTER
```

$$\begin{bmatrix} a & b \\ a+c & b+d \end{bmatrix}$$

rowDim()

Menu MATH/Matrix/Dimensions

rowDim(*matriz***)** ⇒ *expressão*

Devolve o número de linhas na *matriz*.

Nota: Consulte também **colDim()**.

```
[1,2;3,4;5,6]→M1 ENTER
```

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix}$$

```
rowdim(M1) ENTER
```

3

rowNorm()

Menu MATH/Matrix/Norms

rowNorm(*matriz***)** ⇒ *expressão*

Devolve o máximo das somas dos valores absolutos dos elementos nas linhas na *matriz*.

Nota: Todos os elementos da matriz têm de ser simplificados para números. Consulte também **colNorm()**.

```
rowNorm([-5,6,-7;3,4,9;9,-9,-7])
```

rowSwap()

Menu MATH/Matrix/Row

rowSwap(*matriz1***,** *ÍndiceR 1***,** *ÍndiceR 2***)** ⇒ *matriz*

Devolve a *matriz 1* com as linhas *ÍndiceR 1* e *ÍndiceR 2* trocadas.

```
[1,2;3,4;5,6]→Mat ENTER
```

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix}$$

```
rowSwap(Mat,1,3) ENTER
```

$$\begin{bmatrix} 5 & 6 \\ 3 & 4 \\ 1 & 2 \end{bmatrix}$$

RplcPic

CATALOG

RplcPic *picVar*[**,** *linha*][**,** *coluna*]

Apaga o ecrã Graph e coloque a imagem picVar nas coordenadas dos pixéis (*linha, coluna*). Se não quiser apagar o ecrã, utilize **RclPic**.

picVar tem de ser uma variável de tipo de dados da imagem. *linha* e *coluna*, se incluídas, especificam as coordenadas dos pixéis do canto superior esquerdo da imagem. As coordenadas predefinidas são (0, 0).

Nota: Para imagens inferiores ao ecrã completo, apenas a área afectada pela nova imagem é apagada.

rref()

Menu MATH/Matrix

rref(*matriz1***[,** *tol***])** ⇒ *matriz*

Devolve o tipo de escalão de linhas de *matriz 1*.

```
rref([-2,-2,0,-6;1,-1,9,-9;
-5,2,4,-4]) ENTER
```

$$\begin{bmatrix} 1 & 0 & 0 & 66/71 \\ 0 & 1 & 0 & \frac{147}{71} \\ 0 & 0 & 1 & -62/71 \end{bmatrix}$$

Opcionalmente, qualquer elemento da matriz é tratado como zero se o valor absoluto for menor do que *tol*. Esta tolerância só é utilizada se a matriz tiver entradas de ponto flutuante e não contiver quaisquer variáveis simbólicas que não tenham um valor. Caso contrário, *tol* é ignorado.

- Se utilizar ◆ ENTER ou definir o modo para Exact/Approx=APPROXIMATE,, os cálculos são efectuados com a aritmética de ponto flutuante.
- ·Se *tol* for omitido ou não utilizado, a tolerância predefinida é calculada como:

  5E⁻14 ∗ **max(dim(***matriz1***))** ∗ **rowNorm(***matriz1***)**

Nota: Consulte também **ref()**.

rref([a,b,x;c,d,y])

$$\begin{bmatrix} 1 & 0 & \frac{d\cdot x-b\cdot y}{a\cdot d-b\cdot c} \\ 0 & 1 & \frac{-(c\cdot x-a\cdot y)}{a\cdot d-b\cdot c} \end{bmatrix}$$

sec() Menu MATH/String

sec(*expressão1***)** ⇒ *expressão*
sec() ⇒ *lista*

Devolve a secante de *expressão 1* ou devolve uma lista que contém as secantes de todos os elementos de *lista 1*.

Nota: O resultado é devolvido como um ângulo em graus, grados ou radianos, de acordo com a definição do modo de ângulo actual.

No modo de ângulo Degree:

sec(45) ENTER $\sqrt{(2)}$

sec({1,2.3,4}) ENTER

$$\left\{\frac{1}{\cos(1)}\ 1.000\ldots\ \frac{1}{\cos(4)}\right\}$$

sec⁻¹() Menu MATH/String

sec⁻¹(*expressão1***)** ⇒ *expressão*
sec⁻¹() ⇒ *lista*

Devolve o ângulo cuja secante é expressão 1ou devolve uma lista que contém as secantes inversas de cada elemento de *lista 1*.

Nota: O resultado é devolvido como um ângulo em graus, grados ou radianos, de acordo com a definição do modo de ângulo actual.

No modo de ângulo Degree:

sec1(

No modo de ângulo Gradian:

$\sec^{-1}(\sqrt{2})$ ENTER 50

No modo de ângulo Radian:

$\sec^{-1}$({1,2,5}) ENTER

$$\left\{0\ \frac{\pi}{3}\ \cos^{-1}(1/5)\right\}$$

sech() Menu MATH/Hyperbolic

sech(*expressão1***)** ⇒ *expressão*
sech() ⇒ *lista*

Devolve a secante hiperbólica de *expressão 1* ou devolve uma lista que contém as secantes hiperbólicas dos elementos de *lista 1*.

sech(3) ENTER $\frac{1}{\cosh(3)}$

sech({1,2.3,4}) ENTER

$$\left\{\frac{1}{\cosh(1)}\ .198\ldots\ \frac{1}{\cosh(4)}\right\}$$

sech1() Menu MATH/Hyperbolic

sech⁻¹(*expressão1***)** ⇒ *expressão*
sech⁻¹ () ⇒ *lista*

Devolve a secante hiperbólica inversa de *expressão 1* ou devolve uma lista que contém as secantes hiperbólicas inversas de cada elemento de *lista 1*.

No modo complexo e ângulo radiano e rectangular:

$\text{sech}^{-1}(1)$ ENTER 0

sech^{-1}({1,-2,2.1}) ENTER

$$\left\{0\ \left(\frac{2\cdot\pi}{3}\right)\cdot i\ 1.074\ldots\cdot i\right\}$$

Send CATALOG

Send *lista*

Instrução CBL2/CBL (Calculator-Based Laboratory) ou CBR (Calculator-Based Ranger). Envia a *lista* para a porta de ligação.

Segmento do programa:

```
⋮
:Send {1,0}
:Send {1,2,1}
⋮
```

SendCalc CATALOG

SendCalc *var*

Envia a variável var para a porta de ligação, em que a outra unidade ligada a essa porta pode receber o valor da variável. A unidade de recepção tem de estar no ecrã inicial ou executar **GetCalc** a partir de um programa.

Se enviar a partir de uma TI-89, TI-92 Plus, ou Voyage™ 200 para uma TI-92, ocorre um erro se o TI-92 executar **GetCalc** a partir de um programa. Neste caso, a unidade de envio tem de utilizar **SendChat**.

Segmento do programa:

```
⋮
:a+b→x
:SendCalc x
⋮
```

SendCalc *var[,porta]*

Envia o conteúdo da *var* de uma TI-89 Titanium para outra TI-89 Titanium.

Se a porta não for especificada ou a *porta* = 0 for especificada, a TI-89 Titanium envia os dados através da porta USB se ligada, se não estiver ligada, envia através da porta E/S.

Se a *porta* = 1, a TI-89 Titanium só envia dados através da porta USB.

Se a *porta* = 2, a TI-89 Titanium só envia dados através da porta E/S.

SendChat CATALOG

SendChat *var*

Uma alternativa geral a **SendCalc**, este é muito útil se a unidade de recepção for um TI-92 (ou para um programa genérico de "chat" que permita a uma TI-92, Voyage™ 200, ou TI-92 Plus ser utilizada). Consulte **SendCalc** para obter mais informações.

SendChat só envia uma variável se essa variável for compatível com a TI-92, que é tipicamente verdadeiro nos programas de "chat". No entanto, **SendChat** não enviará uma variável arquivada, uma TI-89 base de dados de gráficos, etc.

Segmento do programa:

```
⋮
:a+b→x
:SendChat x
⋮
```

seq() Menu MATH/List

seq(*expressão***,** *var***,** *baixo***,** *alto***[,** passo**])** ⇒ *lista*

Aumenta a *var* de baixo para alto através de um incremento de passo,avalia a *expressão*, e devolve os resultados como uma lista. O conteúdo original de *var* ainda aqui estão depois da conclusão de **seq()**.

var não pode ser uma variável do sistema.

O valor predefinido para *passo* = 1.

```
seq(n^2,n,1,6) ENTER
                    {1 4 9 16 25 36}
seq(1/n,n,1,10,2) ENTER
                 {1 1/3 1/5 1/7 1/9}
sum(seq(1n^2,n,1,10,1))
                              196...
                              127...
or press ◆ ENTER to get:     1.549...
```

setDate() CATALOG

setDate(*ano,mês,dia***)** ⇒ *listold*

Define o relógio para uma hora especificada no argumento e devolve uma lista. **Nota:** O *ano* tem de estar no intervalo 1997 - 2132.) A lista devolvida está no formato {*yearold,monthold,dayold*} A data devolvida é o valor anterior do relógio.

Introduza o ano como um número inteiro com quatro dígitos. O mês e o dia podem ser inteiros com um ou dois dígitos.

```
setDate(2001,10,31) ENTER
                 {2001  11  1}
```

setDtFmt() CATALOG

setDtFmt(*número inteiro***)** ⇒ *integerold*

Define o formato de data para o ambiente de trabalho de acordo com o argumento e devolve o valor do formato de data anterior.

Valores inteiros:

1 = MM/DD/AA 5 = AA.MM.DD

2 = DD/MM/AA 6 = MM-DD-AA

3 = MM.DD.AA 7 = DD-MM-AA

4 = DD.MM.AA 8 = AA-MM-DD

setFold() CATALOG

setFold(*Nome da pasta nova***)** ⇒ *Cadeia da pasta antiga*

Devolve o nome da pasta actual como uma cadeia e define *Nome da pasta nova* como a pasta actual.

A pasta *Nome da pasta nova* tem de existir.

```
newFold chris ENTER                Done
setFold(main) ENTER             "chris"
setFold(chris)→oldfoldr ENTER
                                 "main"
1→a ENTER                            1
setFold(#oldfoldr) ENTER        "chris"
a ENTER                              a
chris\a ENTER                        1
```

setGraph() CATALOG

setGraph(*Cadeia do nome do modo***,** *Cadeia de definição***)** ⇒ *cadeia*

Define o modo Graph de *Cadeia do nome* do modo para *Cadeia de definiçã*o, e devolve a definição anterior do modo. O armazenamento da definição anterior permite restaurá-la posteriormente.

Cadeia do nome do modoé uma cadeia de caracteres que especifica o modo que pretende definir. Tem de ser um dos nomes dos modos da tabela abaixo.

```
setGraph("Graph Order","Seq")
ENTER                           "SEQ"
setGraph("Coordinates","Off")
ENTER                          "RECT"
```

Nota: A capitalização e os espaços em branco são opcionais quando introduzir os nomes dos modos.

Cadeia de definição é uma cadeia de caracteres que especifica a nova definição para o modo. Tem de ser uma das definições listadas abaixo para o modo específico que está a definir.

Nome do modo	Definições
"Coordinates"	"Rect", "Polar", "Off"
"Graph Order"	"Seq", "Simul" [1]
"Grid"	"Off', "On"[2]
"Axes"	"Off", "On"(não no modo de representação gráfica 3D) "Off", "Axes", "Box"(modo de representação gráfica 3D)
"Leading Cursor"	"Off', "On"[2]
"Labels"	"Off", "On"
Style	"Wire Frame", "Hidden Surface", "Contour Levels", "Wire and Contour", "Implicit Plot" [3]
"Seq Axes"	"Time", "Web", "U1-vs-U2" [4]
"DE Axes"	"Time", "t-vs-y' ", "y-vs-y' ", "y1-vs-y2", "y1-vs-y2' ", "y1'-vs-y2' " [5] **Sugestão:** Para digitar um símbolo de primo ('), prima 2nd [′].
"Solution Method"	"RK", "Euler" [5]
"Fields"	"SlpFld", "DirFld", "FldOff" [5]
"Discontinuity Detection"	"Off', "On"[6]

[1]Não disponível no modo de representação gráfica Sequence, 3D ou Diff Equations. Também indisponível no modo de representação gráfica Function com "Discontinuity Detection" definida para "On."
[2]Indisponível no modo de representação gráfica 3D.
[3]Aplica-se apenas ao modo de representação gráfica 3D.
[4]Aplica-se ao modo de representação gráfica Sequence.
[5]Aplica-se ao modo de representação gráfica Diff Equations.
[6]Aplica-se apenas ao modo de representação gráfica Function, quando "Graph Order" for definido para "Seq."

setMode() CATALOG

setMode(Cadeia do nome do modo, Cadeia de definição) ⇒ *cadeia*
setMode(*lista***)** ⇒ *Lista de cadeias*

Define o modo *Cadeia do nome* do modo para a nova definição *Cadeia de definição* e devolve a definição actual desse modo.

Cadeia do nome do modoé uma cadeia de caracteres que especifica o modo que pretende definir. Tem de ser um dos nomes dos modos da tabela abaixo.

Cadeia de definição é uma cadeia de caracteres que especifica a nova definição para o modo. Tem de ser uma das definições listadas abaixo para o modo específico que está a definir.

setMode("Angle","Degree") ENTER "RADIAN"

sin(45) ENTER $\frac{\sqrt{2}}{2}$

setMode("Angle","Radian") ENTER "DEGREE"

sin(π/4) ENTER $\frac{\sqrt{2}}{2}$

setMode("Angle","Gradian") ENTER "RADIAN"

lista contém pares de cadeias de palavras-chave e define-as todas de uma só vez. É recomendado para várias alterações de modos. O exemplo mostrado pode não funcionar se cada um dos pares for introduzido com um **setMode()** independente pela ordem mostrada.

Utilize o**setMode(*var*)** para restaurar as definições guardadas com **getMode("ALL")→** *var*.

Nota: Para definir ou devolver informações sobre o modo Unit System, utilize **setUnits()** ou **getUnits()** em vez de **setMode()** ou **getMode()**.

```
sin(50) ENTER                                  √2/2

setMode("Display Digits",
"Fix 2") ENTER                          "FLOAT"

π ◆ ENTER                                   3.14

setMode ("Display Digits",
"Float") ENTER                          "FIX 2"

π ◆ ENTER                                 3.141...

setMode ({"Split Screen",
"Left-Right","Split 1 App",
"Graph","Split 2 App","Table"})
ENTER
                    {"Split 2 App" "Graph"
                      "Split 1 App" "Home"
                    "Split Screen" "FULL"}
```

Nota: A capitalização e os espaços em branco são opcionais quando introduzir os nomes dos modos. Os resultados nestes exemplos podem também ser diferentes na unidade.

Nome do modo	**Definições**
"Graph"	"Function", "Parametric", "Polar", "Sequence", "3D", "Diff Equations"
Display Digitos (Visualizar dígitos)	"Fix 0", "Fix 1", ..., "Fix 12", "Float", "Float 1", ..., "Float 12"
Angle (Ângulo)	"Radian", "Degree", "Gradian"
Exponential Format (Formato exponencial)	"Normal", "Scientific", "Engineering"
Complex Format (Formato complexo)	"Real", "Rectangular", "Polar"
Vector Format (Formato do vector)	"Rectangular", "Cylindrical", "Spherical"
Pretty Print	"Off", "On"
Split Screen (Ecrã dividido)	"Full", "Top-Bottom", "Left-Right"
"Split 1 App"	"Home", "Y= Editor", "Window Editor", "Graph", "Table", "Data/Matrix Editor", "Program Editor", "Text Editor", "Numeric Solver", "*Flash App*"
"Split 2 App"	"Home", "Y= Editor", "Window Editor", "Graph", "Table", "Data/Matrix Editor", "Program Editor", "Text Editor", "Numeric Solver", "*Flash App*"
Number of Graphs (Número de gráficos)	"1", "2"
"Graph2"	"Function", "Parametric", "Polar", "Sequence", "3D", "Diff Equations"
"Split Screen Ratio"	"1:1", "1:2", "2:1" (Apenas na Voyage™ 200)
Exact/Approx (Exact/Aprox)	"Auto", "Exact", "Approximate"

Base	"Dec", "Hex", "Bin"
Language (Idioma)	"English", "*Alternate Language*"
"Apps Desktop"	"Off", "On"

setTable() CATALOG

setTable(*Cadeia do nome do modo***,** *Cadeia de definição***)** ⇒ *cadeia*

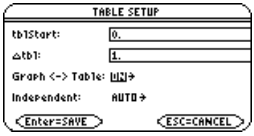
Define o parâmetro da tabela de *Cadeia do nome do modo* para *Cadeia de definição* e devolve a definição anterior do parâmetro. O armazenamento da definição anterior permite restaurá-la posteriormente.

Cadeia do nome do modo é uma cadeia de caracteres que especifica o parâmetro que pretende definir. Tem de ser um dos parâmetros da tabela abaixo.

Cadeia de definição é uma cadeia de caracteres que especifica a nova definição para o parâmetro. Tem de ser uma das definições listadas abaixo para o parâmetro específico que está a definir.

```
setTable("Graph <-> Table","ON")
ENTER                          "OFF"
setTable("Independent","AUTO")
ENTER                          "ASK"
```

◆[TblSet]



Nota: A capitalização e os espaços em branco são opcionais quando introduzir os parâmetros.

Nome do parâmetro	Definições
"Graph <-> Table"	"Off", "On"
"Independent"	"Auto", "Ask"

setTime() CATALOG

setTime(*hora,minuto,segundo***)** ⇒ *listold*

Define o relógio para uma hora especificada no argumento e devolve uma lista. A lista tem o formato {*hourold,minuteold,secondold*}. A hora devolvida é o valor anterior do relógio.

Introduza a hora no formato de 24 horas, em que 13 = 1 p.m.

```
setTime(11,32,50)
                    {10  44  49}
```

setTmFmt() CATALOG

setTmFmt(*número inteiro***)** ⇒ *integerold*

Define o formato de hora para o ambiente de trabalho de acordo com o argumento e devolve o valor do formato de hora anterior.

Valores inteiros:

12 = relógio de 12 horas

24 = relógio de 24 horas

setTmZn() CATALOG

setTmZn(*número inteiro***)** ⇒ *integerold*

Define o fuso horário de acordo com o argumento e devolve o valor do fuso horário anterior.

O fuso horário é definido por um inteiro que calcula o desvio em minutos da hora média de Greenwich (GMT), como estabelecido em Greenwich, Inglaterra. Por exemplo, se o fuso horário se desviar de GMT cerca de duas horas, o dispositivo devolve 120 (minutos).

Os números inteiros para fusos horários a oeste de GMT são negativos.

Os números inteiros para fusos horários a este de GMT são positivos.

Se GMT for 14:07:07, serão:

7:07:07 a.m. em Denver, Colorado (Hora Diurna da Montanha)
(–420 minutos de GMT)

15:07:07 p.m. em Bruxelas, Bélgica (Hora padrão da Europa central)
(+60 minutos de GMT)

setUnits() CATALOG

setUnits() ⇒ *lista*

Define as unidades predefinidas para os valores especificados na lista 1 e devolve uma lista das predefinições anteriores.

- Para especificar o sistema integrado SI (métrico) ou ENG/US, a lista 1 utiliza a formato:

  {"SI"} or {"ENG/US"}

- Para especificar um conjunto de unidades predefinidas personalizadas, a lista 1 utiliza o formato:

  {"CUSTOM", "*cat1*", "*unidade 1*" [, "*cat2*", "*unidade 2*", ...]}

  em que cada par de *cat* e *unidade* especifica uma categoria e a unidade predefinida. (Pode especificar apenas as unidades integradas, não as unidades definidas pelos utilizadores.) Qualquer categoria não especificada utilizará a unidade personalizada anterior.

- Para voltar às unidades predefinidas personalizadas anteriores, a *lista 1* utiliza o formato:

  {"CUSTOM"}

Se quiser diferentes predefinições dependendo da situação, crie listas separadas e guarde-as para nomes de listas únicos. Para utilizar um conjunto de predefinições, especifique esse nome de lista em **setUnits()**.

Pode utilizar **setUnits()** para restaurar as definições guardadas previamente com **setUnits()** → *var* ou com **getUnits()** → *var*.

Todos os nomes da unidades têm de começar com um carácter de sublinhado _.

◆ [_]

Pode também seleccionar as unidade a partir de um menu, premindo:

2nd [UNITS]

```
setUnits({"SI"}) ENTER
                 {"SI"  "Area"  "NONE"
            "Capacitance"  "_F"  ...}
```

```
setUnits({"CUSTOM","Length",
"_cm","Mass","_gm"}) ENTER
                 {"SI"  "Length"  "_m"
                  "Mass"  "_kg"  ...}
```

Nota: O ecrã pode ver unidades diferentes.

Shade CATALOG

Shade *expr1*, *expr2*, [*baixoX*], [*altoX*], [*padrão*], [*patRes*]

Mostra o ecrã Graph, os gráficos *expr1* e *expr2*, e as áreas sombreadas em que *expr1* é menor que *expr2*. (*expr1* e *expr2* têm de ser expressões que utilizam x como a variável independente.)

baixoX e *altoX*, se incluídos, especificam os limites direito e esquerdo da sombra. As entradas válidas estão entre xmin e xmax. As predefinições são xmin e xmax.

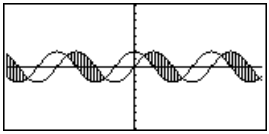
padrão especifica um dos quatro padrões de sombra:
1 = vertical (predefinição)
2 = horizontal
3 = negativo-declive 45°
4 = positivo-declive 45°

patRes especifica a resolução dos padrões de sombra:
1= sombra sólida
2= 1 espacejamento de 1 pixel (predefinição)
3= espacejamento de 2 pixéis
⋮
10= espacejamento de 9 pixéis

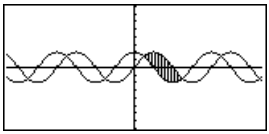
Nota: A sombra interactiva está disponível no ecrã Graph através da instrução **Shade**. A sombra automática de uma função específica está dsiponível através da instrução **Style**. **Shade** não é válida no modo de representação gráfica 3D.

Na janela de visualização **ZoomTrig**:

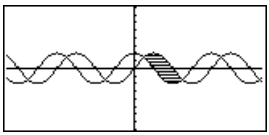
```
Shade cos(x),sin(x)
```



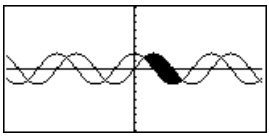
```
HOME
ClrDraw ENTER                    Done
Shade cos(x),sin(x),0,5 ENTER
```



```
HOME
ClrDraw ENTER                    Done
Shade cos(x),sin(x),0,5,2 ENTER
```



```
HOME
ClrDraw ENTER                    Done
Shade cos(x),sin(x),0,5,2,1 ENTER
```



shift() CATALOG

shift(*número inteiro 1*[,*# de rotações*]**)** ⇒ *número inteiro*

Roda os bits num número inteiro binário. Pode introduzir o *número inteiro 1* em qualquer base do número; é convertido para um formato binário de 32 bits. Se a magnitude do *número inteiro 1* é também muito grande para este formato, uma operação do módulo simétrico trá-lo para o intervalo.

Se *# de rotações* for positivo, a rotação é para a esquerda. Se *# de rotações* for negativo, a rotação é para a direita. A predefinição é ⁻1 (roda um bit para a direita).

Numa rotação para a direita, o bit mais à direita desaparece e 0 ou 1 é introduzido para corresponder ao bit mais à esquerda. Numa rotação para a esquerda, o bit mais à esquerda desaparece e 0 é introduzido como o bit mais à direita.

Por exemplo, numa rotação para a direita:

No modo base Bin:

```
shift(0b1111010110000110101) ENTER
                 0b111101011000011010
shift(256,1) ENTER
                           0b1000000000
```

No modo base Hex:

```
shift(0h78E) ENTER                0h3C7
shift(0h78E,⁻2) ENTER             0h1E3
shift(0h78E,2) ENTER             0h1E38
```

Importante: Para introduzir um número binário ou hexadecimal, utilize sempre o prefixo 0b ou 0h (zero, não a letra O).

Cada bit roda para a direita.

0b00000000000000111101011000011010 1

Introduz 0 se o bit mais à esquerda for 0, ou 1 se o bit mais à esquerda for 1.

Desaparecido

produz:

0b00000000000000011110101100001101 0

O resultado aparece de acordo com o modo Base. Os zeros à esquerda não aparecem.

---

shift([,*# de rotações*]**)** ⇒ *lista*

Devolve uma cópia de *lista 1* rodados para a direita ou esquerda pelos elementos *# de rotações*. Não altere *lista 1*.

Se *# de rotações* for positivo, a rotação é para a esquerda. Se *# de rotações* for negativo, a rotação é para a direita. A predefinição é ⁻1 (roda um elemento para a direita).

Os elementos introduzidos no início ou no fim da *lista* pela rotação são definidos para o símbolo "undef".

No modo base Dec:

```
shift({1,2,3,4}) ENTER
                        {undef 1 2 3}
shift({1,2,3,4},⁻2) ENTER
                    {undef undef 1 2}
shift({1,2,3,4},1) ENTER
                        {2 3 4 undef}
```

---

shift(*cadeia 1* [,*# de rotações*]**)** ⇒ *cadeia*

Devolve uma cópia de *cadeia 1* rodada para a direita ou esquerda pelos caracteres *# de rotações*. Não altere a *cadeia 1*.

Se *# de rotações* for positivo, a rotação é para a esquerda. Se *# de rotações* for negativo, a rotação é para a direita. A predefinição é ⁻1 (roda um carácter para a direita).

Os elementos introduzidos no início ou no fim da *cadeia* pela rotação são definidos para um espaço.

```
shift("abcd") ENTER          " abc"
shift("abcd",⁻2) ENTER       "  ab"
shift("abcd",1) ENTER        "bcd "
```

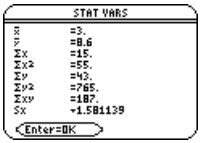
ShowStat CATALOG

ShowStat

Mostra uma caixa de diálogo com os últimos resultados estatísticos calculados se ainda forem válidos. Os resultados estatísticos são apagados automaticamente se alterar os dados para cálculo.

Utilize esta instrução depois de um cálculo estatístico, como **RegLin**.

```
{1,2,3,4,5}→L1 ENTER        {1 2 3 4 5}
{0,2,6,10,25}→L2 ENTER
                        {0 2 6 10 25}
TwoVar L1,L2 ENTER
ShowStat ENTER
```



sign() Menu MATH/Number

sign(*expressão1***)** ⇒ *expressão*
sign() ⇒ *lista*
sign(*matriz1***)** ⇒ *matriz*

Para a *expressão1*, real e complexa, devolve a *expressão1*/**abs(***expressão1***)** quando *expressão1*≠ 0.

Devolve 1 se a *expressão 1* for positiva.
Devolve ⁻1 if *expressão1*for negativa.
sign(0) devolve ±1 se o modo de formato completo for REAL; caso contrário, devolve-se a si próprio.
sign(0) representa a circunferência da unidade no domínio complexo.

Para uma lista ou matriz, devolve os sinais de todos os elementos.

```
sign(⁻3.2) ENTER                    ⁻1.
sign({2,3,4,⁻5}) ENTER
                          {1  1  1 ⁻1}
sign(1+abs(x)) ENTER                 1
```

Se o modo de formato complexo for **REAL**:

```
sign([⁻3,0,3]) ENTER        [⁻1 ±1 1]
```

simult() Menu MATH/Matrix

simult(*Matriz do coeficiente***,** *Vector constante***[,** *tol***])** ⇒ *matriz*

Devolve o vector de uma coluna que contém as soluções para um sistema de equações lineares.

matriz do coeficiente tem de ser uma matriz quadrada que contém os coeficientes das equações.

vector da constante tem de ter o mesmo número de linhas (mesma dimensão) como a *Matriz do coeficiente* e contém as constantes.

Opcionalmente, qualquer elemento da matriz é tratado como zero se o valor absoluto for menor do que *tol*. Esta tolerância só é utilizada se a matriz tiver entradas de ponto flutuante e não contiver quaisquer variáveis simbólicas que não tenham um valor. Caso contrário, *tol* é ignorado.

- Se utilizar ◆ ENTER ou definir o modo para Exact/Approx=APPROXIMATE, os cálculos são efectuados com a aritmética de ponto flutuante.
- ·Se *tol* for omitido ou não utilizado, a tolerância predefinida é calculada como:
  5ε⁻14 ∗ **max(dim(***Matriz do coeficiente***))**
  ∗ **rowNorm(***Matriz do coeficiente***)**

```
Solve for x and y:          x + 2y = 1
                                 sin()
simult([1,2;3,4],[1;⁻1]) ENTER
                              [⁻3 ]
                              [ 2 ]
The solution is x=⁻3 and y=2.
```

```
Solve:                      ax + by = 1
                             cx + dy = 2
[a,b;c,d]→matx1 ENTER          [a b]
                               [c d]
simult(matx1,[1;2]) ENTER
```

$$\begin{bmatrix} \frac{-(2\cdot b-d)}{a\cdot d-b\cdot c} \\ \frac{2\cdot a-c}{a\cdot d-b\cdot c} \end{bmatrix}$$

---

simult(*Matriz do coeficiente***,** *Matriz da constante***[,** *tol***])** ⇒ *matriz*

Resolve vários sistemas de equações lineares, em que cada sistema tem os mesmos coeficientes de equações, mas diferentes constantes.

Cada coluna em *Matriz da constante* tem de conter as constantes para um sistema de equações. Cada coluna da matriz resultante contém a solução para o sistema correspondente.

```
Solve:                      x + 2y = 1
                                 sin()
simult([1,2;3,4],[1,2;⁻1,⁻3])
ENTER
                          [⁻3   ⁻7 ]
                          [ 2   9/2]
```

Para o primeiro sistema, x=⁻3 e y=2. Para o segundo sistema, x=⁻7 e y=9/2.

sinh()

Teclas 2nd [SIN]

sin(*expressão1***)** ⇒ *expressão*
sin() ⇒ *lista*

sin(*expressão 1***)** devolve o seno do argumento como uma expressão.

sin(*lista 1***)** devolve uma lista dos senos de todos os elementos em *lista 1*.

Nota: O argumento é interpretado como um ângulo em graus, grados ou radianos, de acordo com o modo de ângulo actual. Pode utilizar °, ᵍ ou ʳ para substituir a definição do modo de ângulo temporariamente.

No modo de ângulo Degree:

sin((π/4)ʳ) ENTER $\frac{\sqrt{2}}{2}$

sin(45) ENTER $\frac{\sqrt{2}}{2}$

sin({0,60,90}) ENTER $\{0 \quad \frac{\sqrt{3}}{2} \quad 1\}$

No modo de ângulo Gradian:

sin(50) ENTER $\frac{\sqrt{2}}{2}$

No modo de ângulo Radian:

sin(π/4) ENTER $\frac{\sqrt{2}}{2}$

sin(45°) ENTER $\frac{\sqrt{2}}{2}$

---

sin(*Matriz quadrada 1***)** ⇒ *Matriz quadrada*

Devolve o seno da matriz de *Matriz quadrada 1*. Este procedimento não é a mesma coisa que calcular o seno de cada elemento. Para mais informações sobre o método de cálculo, consulte **cos()**.

*matriz quadrada 1*tem de estar na diagonal. O resultado contém sempre números de pontos flutuantes.

No modo de ângulo Degree:

sin([1,5,3;4,2,1;6,⁻2,1]) ENTER

$$\begin{bmatrix} .942... & ^-.045... & ^-.031... \\ ^-.045... & .949... & ^-.020... \\ ^-.048... & ^-.005... & .961... \end{bmatrix}$$

Teclas ◆ [SIN⁻¹]

sin⁻¹ (*expressão1***)** ⇒ *expressão*
sin⁻¹ () ⇒ *lista*

sin⁻¹ (*expressão1***)** returns the angle whose sine is *expressão1* as an expression.

sin⁻¹ () *lista 1* devolve uma lista de senos inversos de cada elemento de *lista 1*.

Nota: O resultado é devolvido como um ângulo em graus, grados ou radianos, de acordo com a definição do modo de ângulo actual.

No modo de ângulo Degree:

sin⁻¹(1) ENTER 90

No modo de ângulo Gradian:

sin⁻¹(1) ENTER 100

No modo de ângulo Radian:

sin⁻¹({0,.2,.5}) ENTER
{0 .201... .523...}

---

sin⁻¹(*Matriz quadrada 1***)** ⇒ *Matriz quadrada*

Devolve o seno hiperbólico da matriz de *Matriz quadrada 1*. Este procedimento *não* é a mesma coisa que calcular o seno inverso de cada elemento. Para mais informações sobre o método de cálculo, consulte **cos()**.

*matriz quadrada 1*tem de estar na diagonal. O resultado contém sempre números de pontos flutuantes.

No modo de ângulo Radian e no modo do formato complexo Rectangular:

sin⁻¹([1,5,3;4,2,1;6,⁻2,1]) ENTER

$$\begin{bmatrix} ^-.164...-.064...\cdot i & 1.490...-2.105...\cdot i & ... \\ .725...-1.515...\cdot i & .947...-.778...\cdot i & ... \\ 2.083...-2.632...\cdot i & ^-1.790...+1.271...\cdot i & ... \end{bmatrix}$$

sinh() Menu MATH/Hyperbolic

sinh(*expressão1***)** ⇒ *expressão*
sinh() ⇒ *lista*

sinh (*expressão 1***)** devolve o seno hiperbólico do argumento como uma expressão.

sinh (*lista***)** devolve uma lista dos senos hiperbólicos de cada elemento de lista 1.

```
sinh(1.2) ENTER                    1.509...
sinh(0,1.2,3.) ENTER
                    {0   1.509...   10.017...}
```

---

sinh(*Matriz quadrada 1***)** ⇒ *Matriz quadrada*

Devolve o seno hiperbólico da matriz de *Matriz quadrada 1*. Este procedimento *não* é a mesma coisa que calcular o seno hiperbólico de cada elemento. Para mais informações sobre o método de cálculo, consulte **cos()**.

*matriz quadrada 1*tem de estar na diagonal. O resultado contém sempre números de pontos flutuantes.

No modo de ângulo Radian:

```
sinh([1,5,3;4,2,1;6,-2,1]) ENTER
```

$$\begin{bmatrix} 360.954 & 305.708 & 239.604 \\ 352.912 & 233.495 & 193.564 \\ 298.632 & 154.599 & 140.251 \end{bmatrix}$$

$\sinh^{-1}()$ Menu MATH/Hyperbolic

$\sinh^{-1}$ (*expressão1***)** ⇒ *expressão*
$\sinh^{-1}$ () ⇒ *lista*

$\sinh^{-1}$ (*expressão 1***)** devolve o seno hiperbólico do argumento como uma expressão.

$\sinh^{-1}$ () *lista 1* devolve uma lista dos senos hiperbólicos inversos de cada elemento de *lista 1*.

```
sinh⁻¹(0) ENTER                    0
sinh⁻¹({0,2.1,3}) ENTER
                    {0   1.487...   sinh⁻¹(3)}
```

$\sinh^{-1}$(*Matriz quadrada 1***)** ⇒ *Matriz quadrada*

Devolve o seno hiperbólico da matriz de *Matriz quadrada 1*. Este procedimento não é a mesma coisa que calcular o seno hiperbólico inverso de cada elemento. Para mais informações sobre o método de cálculo, consulte **cos()**.

*matriz quadrada 1*tem de estar na diagonal. O resultado contém sempre números de pontos flutuantes.

No modo de ângulo Radian:

```
sinh⁻¹([1,5,3;4,2,1;6,-2,1]) ENTER
```

$$\begin{bmatrix} .041\ldots & 2.155\ldots & 1.158\ldots \\ 1.463\ldots & .926\ldots & .112\ldots \\ 2.750\ldots & -1.528\ldots & .572\ldots \end{bmatrix}$$

SinReg — Menu MATH/Statistics/Regressions

SinReg , *lista 2* [, [*repetições*], [*período*] [, *lista 3*, *lista 4*]]

Calcula a regressão sinusoidal e actualiza todas as variáveis estatísticas do sistema.

Todas as listas têm de ter dimensões iguais, excepto a *lista 4*.

lista 1 representa xlist.
lista 2 representa ylist.
lista 3 representa os còdigos de categorias.
lista 4 representa a lista de inclusão de categorias.

repetições especificam o número máximo de vezes (De 1 a 16)que uma solução é tentada. Se omitido, o 8 é utilizado. Geralmente, os valores maiores resultam em melhor precisão, mas em tempos de execução mais longos, e vice-versa.

período especifica um período estimado. Se omitido, a diferença entre valores em *lista 1* deve ser igual e estar na ordem sequencial. Se especificar o *período*, as diferenças entre os valores x podem ser desiguais.

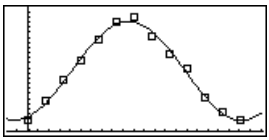
Nota: De *lista 1* a *lista 3* têm de ser um nome da variável ou c1–c99 (colunas da última variável de dados apresentada no Editor de Dados/Matriz). *lista 4* não tem de ser o nome de uma variável e não pode ser c1–c99.

A saída de **RegSin** é sempre em radianos, independentemente da definição do modo de ângulo.

No modo de representação gráfica da função.

```
seq(x,x,1,361,30)→L1 [ENTER]
                                {1 31 61 …}
{5.5,8,11,13.5,16.5,19,19.5,17,
14.5,12.5,8.5,6.5,5.5}→L2 [ENTER]
                              {5.5 8 11 …}
SinReg L1,L2 [ENTER]                  Done
ShowStat [ENTER]
```

```
[ENTER]
regeq(x)→y1(x) [ENTER]                Done
NewPlot 1,1,L1,L2 [ENTER]             Done
[◆][GRAPH]
[F2]9
```



solve() — Menu MATH/Algebra

solve(*equação*, *var***)** ⇒ *Expressão booleana*
solve(*desigualdade*, *var***)** ⇒ *Expressão booleana*

Devolve soluções reais candidatas de uma equação ou desigualdade para *var*. O objectivo é devolver candidatos para todas as soluções. No entanto, podem existir equações ou desigualdades para as quais o número de soluções é infinito.

```
solve(a∗x^2+b∗x+c=0,x) [ENTER]
```

$$x = \frac{\sqrt{b^2 - 4 \cdot a \cdot c} - b}{2 \cdot a}$$

or x=

Os candidatos a soluções podem não ser soluções finitas reais para algumas combinações de valores para variáveis indefinidas.

```
ans(1)| a=1 and b=1 and c=1
[ENTER]
Error: Non-real result (Resultado não
```

Para a definição AUTO do modo Exact/Approx, o objectivo é produzir soluções exactas quando forem concisas e complementadas por procuras repetidas com a aritmética aproximada quando as soluções exactas não forem práticas.

```
solve((x−a)e^(x)=⁻x∗(x−a),x) [ENTER]
                        x = a or x = ⁻.567…
```

Devido ao cancelamento predefinido do maior divisor comum do numerador e do denominador dos rácios, as soluções podem ser soluções apenas no limite de um dos lados.

```
(x+1)(x−1)/(x−1)+x−3 [ENTER]
                                    2 • x−2
solve(entry(1)=0,x) [ENTER]           x = 1
entry(2)|ans(1) [ENTER]             undef
limit(entry(3),x,1) [ENTER]             0
```

Para desigualdades de tipos ≥, ≤, <, ou >, as soluções explícitas são improváveis, excepto se a desigualdade for linear e contiver apenas *var*.

```
solve(5x−2 ≥ 2x,x) [ENTER]          x ≥ 2/3
```

Para a definição EXACT do modo Exact/Approx,as partes que não podem ser resolvidas são devolvidas como desigualdade ou equação implícita.

```
exact(solve((x−a)e^(x)=⁻x∗
(x−a),x)) [ENTER]
                        e^x + x = 0 or x = a
```

Utilize o operador "|" para restringir o intervalo de solução e/ou outras variáveis que ocorram na equação ou desigualdade. Quando localizar uma solução num intervalo, pode utilizar os operadores de desigualdade para excluir esse intervalo de procuras subsequentes.

No modo de ângulo Radian:

```
solve(tan(x)=1/x,x)|x>0 and x<1 ENTER x
```

falso é devolvido quando não forem encontradas soluções reais. Verdadeiro é devolvido se **solve()** conseguir determinar que qualquer valor real finito de *var* satisfaz a equação ou a desigualdade.

```
solve(x=x+1,x) ENTER                false
solve(x=x,x) ENTER                   true
```

Como **solve()** devolve sempre um resultado Booleano, pode utilizar "and," "or," e "not" para combinar os resultados de **solve()** uns com os outros ou com outras expressões booleanas.

```
2x-1≤1 and solve(x^2≠9,x) ENTER
                        x ≤ 1 and x ≠ -3
```

As soluções podem conter uma variável indefinida nova única da forma @*j* com o *j* sendo um número inteiro no intervalo 1–255. Essas variáveis indicam um número inteiro arbitrário.

No modo de ângulo Radian:

```
solve(sin(x)=0,x) ENTER          x = @1 • π
```

No modo real, as potências fraccionárias com denominadores ímpares indicam apenas a derivação real. Caso contrário, as várias expressões com derivações, como as potências fraccionárias, os logaritmos e as funções trigonométricas inversas indical apenas a derivação principal. Por consequência, **solve()** só produz soluções correspondente a essa derivação real ou principal.

```
solve(x^(1/3)=-1,x) ENTER        x = -1
solve(√(x)=-2,x) ENTER           false
solve(-√(x)=-2,x) ENTER          x = 4
```

Nota: Consulte também **cSolve(), cZeros()**, **nSolve()**, e **zeros()**.

---

solve(*equação 1* **e** *equação 2* [**e** …], **{***varOrGuess1***,** *varOrGuess2* [**,** …]**}) ⇒** *Expressão booleana*

```
solve(y=x^2-2 and
x+2y=-1,{x,y}) ENTER
                        x=1 and y=-1
                or x=-3/2 and y=1/4
```

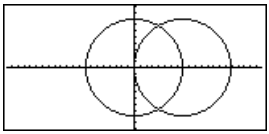
Devolve soluções reais candidatas para equações algébricas simultâneas, em que cada *varOrGuess* especifica uma variável que pretende resolver.

Optionalmente, pode especificar uma hipótese inicial para uma variável. Cada *varOrGuess* tem de ter o formato:

variável
– ou –
variável = número real ou não real

Por exemplo, x é válido e, por isso, é x=3.

Se todas as equações forem polinomiais e não especificar quaisquer hipóteses iniciais, **cSolve()** utilize o método de eliminação Gröbner/Buchberger lexical para tentar determinar **todas** as soluções complexas.



Por exemplo, suponha que tem uma circunferência de raio r na origem e outra circunferência de raio r centrada onde a primeira circunferência cruza com o eixo x positivo. Utilize **solve()** para localizar as intercepções.

Como ilustrado pelo r no exemplo à direita, as equações *polininomiais* simultâneas podem ter variáveis adicionais que não têm valores, mas representam valores numéricos dados que podem ser substituídos posteriormente.

```
solve(x^2+y^2=r^2 and
(x-r)^2+y^2=r^2,{x,y}) ENTER
```

$$x=\frac{r}{2} \text{ and } y=\frac{\sqrt{3}\cdot r}{2}$$

$$\text{or } x=\frac{r}{2} \text{ and } y=\frac{-\sqrt{3}\cdot r}{2}$$

Pode também (ou em vez de) incluir variáveis de solução que não apareçam nas equações. Por exemplo, pode incluir z como uma variável de solução para ampliar o exemplo anterior para dois cilindros de intercepção paralelos de raio r.

```
solve(x^2+y^2=r^2 and
(x-r)^2+y^2=r^2,{x,y,z}) ENTER
```

$$x=\frac{r}{2} \text{ and } y=\frac{\sqrt{3}\cdot r}{2} \text{ and } z=@1$$

$$\text{or } x=\frac{r}{2} \text{ and } y=\frac{-\sqrt{3}\cdot r}{2} \text{ and } z=@1$$

As soluções dos cilindros ilustram como as famílias de soluções podem conter constantes arbitrárias da forma *@k*, em que é o sufixo de um número inteiro de 1 a 255. O sufixo reinicia-se 1 quando utilizar **ClrHome** ou F1 8:Clear Home.

Para sistemas polinomiais, o tempo de cálculo e o consumo da memória podem depender fortemente da ordem em que lista as variáveis de solução. Se a escolha inicial consumir a memória ou a sua paciência, tente redispor as variáveis nas equações e/ou na lista *varOrGuess*.

Se não incluir quaisquer hipóteses e se qualquer equação for não polinomial em qualquer variável, mas todas as equações forem lineares em todas as variáveis de solução, **solve()** utiliza a eliminação Gaussian para tentar determinar todas as soluções.

```
solve(x+e^(z)*y=1 and
x-y=sin(z),{x,y}) ENTER
```

$$x=\frac{e^z\cdot\sin(z)+1}{e^z+1} \text{ and } y=\frac{-(\sin(z)-1)}{e^z+1}$$

Se um sistema não for polinomial em todas variávies nem linear nas variáveis de solução,determina a solução **solve()** mais pretendida com um método de repetição aproximada. Para o fazer, o número de variáveis de solução tem de ser igual ao múmero de equações, e todas as outras variáveis das equações têm de ser simplificadas para números.

```
solve(e^(z)*y=1 and -y=sin(z),{y,z})
ENTER
```

y=.041… and z=3.183…

Cada variável de solução começa no valor de hipótese se existir um; caso contrário, começa em 0.0.

Utilize as hipóteses para procurar soluções adicionais uma a uma. Para convergência, uma hipótese pode estar próxima de uma solução.

```
solve(e^(z)*y=1 and
-y=sin(z),{y,z=2π}) ENTER
                y=.001… and z=6.281…
```

SortA

Menu MATH/List

SortA *d 1*[, *d 2*] [, *d 3*] **...**
SortA *Nome do vector 1*[, *Nome do vector 2*] [, *Nome do vector 3*] **...**

Ordena os elementos do primeiro argumento na ordem ascendente.

Se incluir argumentos adicionais, ordena os elementos de cada um para que as novas posições correspondam às novas posições dos elementos no primeiro argumento.

Todos os argumentos têm de ser nomes de listas ou vectores. Todos os argumentos têm de ter dimensões iguais.

```
{2,1,4,3}→ ENTER          {2,1,4,3}
SortA  ENTER                   Done

 ENTER                    {1 2 3 4}
{4,3,2,1}→list2 ENTER     {4 3 2 1}
SortA list2, ENTER             Done

list2 ENTER               {1 2 3 4}
 ENTER                    {4 3 2 1}
```

SortD

Menu MATH/List

SortD *d 1*[, *d 2*] [, *d 3*] **...**
SortD *Nome do vector 1*[, *Nome do vector 2*] [, *Nome do vector 3*] **...**

Idêntico a **SortA**, excepto **SortD** que ordena os elementos na ordem descendente.

```
{2,1,4,3}→ ENTER          {2 1 4 3}
{1,2,3,4}→list2 ENTER     {1 2 3 4}
SortD ,list2 ENTER             Done
 ENTER                    {4 3 2 1}
list2 ENTER               {3 4 1 2}
```

▸Sphere — Menu MATH/Matrix/Vector

vector▸**Sphere**

Mostra o vector de linha ou coluna na forma esférica [ρ ∠θ ∠ϕ].

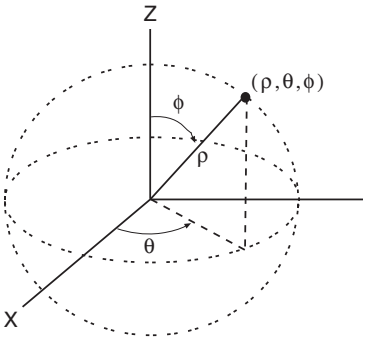
vector tem de ser de dimensão 3 e pode ser um vector de linha ou coluna.

Nota: ▸**Sphere** é uma instrução de formato de visualização, não uma função de conversão. Pode utilizá-la apenas no fim de uma linha de entrada.

```
[1,2,3]▸Sphere
◆ ENTER        [3.741... ∠1.107... ∠.640...]

[2,∠π/4,3]▸Sphere
◆ ENTER        [3.605... ∠.785... ∠.588...]
```

ENTER $[\sqrt{13} \quad \angle\frac{\pi}{4} \quad \angle\cos^{-1}\left(\frac{3\cdot\sqrt{13}}{13}\right)]$



startTmr() — CATALOG

startTmr() ⇒ *número inteiro*

Devolve o valor actual do relógio com a representação de inteiro, especificando *starttime* para um temporizador. Pode introduzir starttime como um argumento em **checkTmr()** para determinar o número de segundos decorridos.

Pode executar vários temporizadores em simultâneo.

Nota: Consulte também **checkTmr()** e **timeCnv()**.

```
startTmr() ENTER                    148083315
checkTmr(148083315)34                      34

startTmr()→Timer1
⋮
startTmr()→Timer2
⋮
checkTmr(Timer1)→Timer1Value
⋮
checkTmr(Timer2)→Timer2Value
```

stdDev() Menu MATH/Statistics

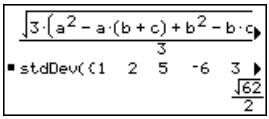
stdDev(*lista***[**, *lista de frequência***])** ⇒ *expressão*

Devolve o desvio standard dos elementos em *lista*.

Cada elemento da lista de frequência conta o número de ocorrências consecutivas do elemento correspondente na *lista*.

Nota: *lista* tem de ter pelo menos dois elementos.

stdDev({a,b,c}) ENTER
stdDev({1,2,5,⁻6,3,⁻2}) ENTER



stdDev({1.3,2.5,-6.4},{3,2,5}) ENTER 4.

---

stdDev(*matriz1***[**, *matriz de frequência***])** ⇒ *matriz*

Devolve um vector de linha dos desvios-padrão das colunas em matriz 1.

Cada elemento da *matriz de frequência* conta o número de ocorrências consecutivas do elemento correspondente na *matriz 1*.

Nota: *matriz 1* tem de ter pelo menos duas linhas.

stdDev([1,2,5;-3,0,1;.5,.7,3])
1.014... 2]

stdDev([1.2,5.3;2.5,7.3;6,4],[4,2;3,3;1,7])

stdDevPop() Menu MATH/Statistics

stdDevPop(*lista***[**, *lista de frequência***])** ⇒ *expressão*

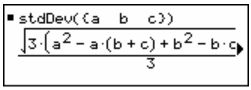
Devolve o desvio standard da população dos elementos em *lista*.

Cada elemento da *lista de frequência* conta o número de ocorrências consecutivas do elemento correspondente na *lista*.

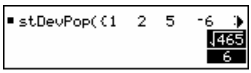
Nota: *lista* tem de ter pelo menos dois elementos.

No modo de ângulo Radian ou auto:

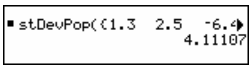
stdDevPop({a,b,c}) ENTER



stdDevPop({1,2,5,⁻6,3,⁻2}) ENTER



stdDevPop({1.3,2.5,-6.4},{3,2,5}) ENTER



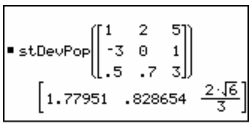
stdDevPop(*matriz1***[**, *matriz de frequência***])** ⇒ *matriz*

Devolve um vector de linha dos desvios-padrão da população das colunas em *matriz 1*.

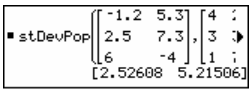
Cada elemento da *matriz de frequência* conta o número de ocorrências consecutivas do elemento correspondente na *matriz 1*.

Nota: *matriz 1* tem de ter pelo menos duas linhas.

stdDevPop([[1,2,5][-3,0,1][.5,.7,3]])



stdDevPop([1.2,5.3;2.5,7.3;6,4],[4,2;3,3;1,7])



StoGDB CATALOG

StoGDB *GDBvar*

Cria uma variável da base de dados gráfica (GDB) que contém:

* Modo de representação gráfica
* Funções Y=
* Variáveis da janela
* Definições do formato do gráfico
  Definição de 1 ou 2 gráficos (definições de ecrã dividido e rácio se for o modo 2-Graph)
  Modo Angle
  Modo Real/complex
* Condições iniciais se o modo Sequence ou Diff Equations
* Sinalizadores da tabela
* tblStart, Δtbl, tblInput

Pode utilizar **RclGDB** *GDBvar* para restaurar o ambiente gráfico.

Nota: Estes itens foram guardados para ambos os gráficos no modo 2-Graph.

Stop CATALOG

Stop

Utilizado como instrução do programa para parar a execução do programa.

Segmento do programa:

```
⋮
For i,1,10,1
  If i=5
  Stop
EndFor
⋮
```

StoPic CATALOG

StoPic *picVar* [, *LinhaPxl*, *ColPxl*] [, *largura*, *altura*]

Mostra o ecrã Graph e copia a área rectangular do ecrã para uma variável *picVar*.

LinhaPxl e *ColPxl*, se incluídas, especificam o canto superior esquerdo da área a copiar (as predefinições são 0, 0).

largura e *altura*, se incluídas, especificam as dimensões da área em pixéis. As predefinições são a largura e a altura em píxeis do ecrã Graph actual.

Store

Consulte → (store), na página 938.

string() Menu MATH/String

string(*expressão***)** ⇒ *cadeia*

Simplifica a *expressão* e devolve o resultado como uma cadeia de caracteres.

```
string(cos(x)+(1.2345))
string(cos(x)+(3))
string(cos(x)+√(3)) ENTER
                    "cos(x)+√(3)"
```

Style — CATALOG

Style *número da equação, Cadeia da propriedade de estilo*

Define a função de representação gráfica número da equação no modo Graph actual para utilizar a propriedade de representação gráfica Cadeia de propriedade de estilo.

número da equação tem de ser um número inteiro de 1–99 e a função já tem de existir.

Cadeia de propriedade de estilo tem de ser: "Line", "Dot", "Square", "Thick", "Animate", "Path", "Above", ou "Below".

Não se esqueça de que na representação gráfica paramétrica, apenas a metade do par *xt* contém informações estilo.

Nomes de estilo válido vs. modo de representação gráfica:

Função	todos os estilos
Parametric/Polar:	linha, ponto, quadrado, grosso animação, caminho
Sequence:	linha, ponto, quadrado, grosso
3D:	nenhum
Diff Equations:	linha, ponto, quadrado, grosso animação, caminho

Nota: A capitalização e os espaços em branco são opcionais quando introduzir os nomes de *Cadeia de propriedade de estilo*.

```
Style 1,"thick" ENTER                    Done
Style 10,"path" ENTER                    Done
```

Nota: No modo de representação gráfica da função, estes exemplos definem o estilo de **y1(x)** para **"Thick"** para **y10(x)** para **"Path"**.

subMat() — CATALOG

subMat(*matriz 1***[,** *Linha inicial***] [,** *Coluna inicial***] [,** *Linha final***] [,** *Coluna final***]) ⇒** *matriz*

Devolve a submatriz especificada de *matriz 1*.

Predefinições: *startRow*=1, *startCol*=1, *endRow*=last row, *endCol*=last column.

```
[1,2,3;4,5,6;7,8,9]→m1 ENTER
                          [1 2 3]
                          [4 5 6]
                          [7 8 9]
subMat(m1,2,1,3,2) ENTER
                            [4 5]
                            [7 8]
subMat(m1,2,2) ENTER
                            [5 6]
                            [8 9]
```

Sum (Sigma) — Consulte Σ() na página 964

sum() — Menu MATH/List

soma(*lista***[,** *início***[,** *fim***]]) ⇒** *expressão*

Devolve a soma dos elementos em *lista*.

Início e *fim* são opcionais. Especificam um intervalo de elementos.

```
sum({1,2,3,4,5}) ENTER                  15
sum({a,2a,3a}) ENTER                  6•a
sum(seq(n,n,1,10)) ENTER                55
sum({1,3,5,7,9},3) ENTER                21
```

soma(*matriz 1***[,** *início***[,** *fim***]]) ⇒** *matriz*

Devolve um vector de linha com as somas dos elementos nas colunas em *matriz 1*.

Início e *fim* são opcionais. Especificam um intervalo de linhas.

```
sum([1,2,3;4,5,6]) ENTER        [5  7  9]
sum([1,2,3;4,5,6;7,8,9]) ENTER
                          [12 15 18]
sum([1,2,3;4,5,6;7,8,9],2,3)
ENTER
                          [11,13,15]
```

switch() CATALOG

switch([*número inteiro 1***])** ⇒ *número inteiro*

Devolve o número da janela activa. Pode também definri a janela activa.

Nota: Janela 1 é esquerda ou topo; Janela 2 é direita ou fundo.

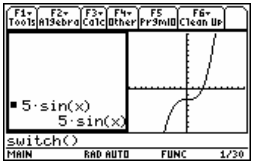
Se o *número inteiro 1* = 0, devolve o número da janela activa.

Se o *número inteiro 1* = 1, activa a janela 1 e devolve o número da janela activa previamente.

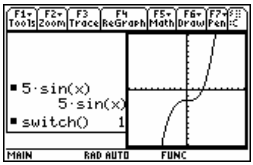
Se o *número inteiro 1* = 2, activa a janela 2 e devolve o número da janela activa previamente.

Se o *número inteiro 1* for omitido, muda a janela e devolve a número da janela activa previamente.

*número inteiro 1*é ignorado se a TI-89 Titanium/Voyage™ 200 não apresentar um ecrã dividido.



switch()



T(transposição) Menu MATH/Matrix

*matriz1*T ⇒ *matriz*

Devolve a transposição conjugada complexa de *matriz 1*.

[1,2,3;4,5,6;7,8,9]→mat1 ENTER

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$$

mat1T ENTER

$$\begin{bmatrix} 1 & 4 & 7 \\ 2 & 5 & 8 \\ 3 & 6 & 9 \end{bmatrix}$$

[a,b;c,d]→mat2 ENTER $\begin{bmatrix} a & b \\ c & d \end{bmatrix}$

mat2T ENTER $\begin{bmatrix} a & c \\ b & d \end{bmatrix}$

[1+*i*,2+*i*;3+*i*,4+*i*]→mat3 ENTER

$$\begin{bmatrix} 1+i & 2+i \\ 3+i & 4+i \end{bmatrix}$$

mat3T ENTER $\begin{bmatrix} 1-i & 3-i \\ 2-i & 4-i \end{bmatrix}$

Table CATALOG

Table *expressão1*[, *expressão2*] [, *var1*]

Constrói uma tabela das funções ou expressões especificadas.

As expressões da tabela podem também ser representadas graficamente. As expressões introduzidas com os comandos **Table** ou **Graph** são atribuídas, aumentando os números das funções começando por 1. As expressões podem ser modificadas ou eliminadas individualmente com as funções de edição disponíveis quando a tabela aparecer, premindo F4 Header. As funções seleccionadas actualmente no Editor Y= são ignoradas temporariamente.

Para apagar as funções criadas por **Table** ou **Graph**, executa o comando **ClrGraph** ou visualiza o Editor Y=.

Se o parâmetro *var* for omitido, a variável independente do modo gráfico actual é assumida. Algumas variações válidas desta instrução são:

Representação gráfica da função: **Table** *expr*, *x*

Representação gráfica paramétrica: **Table** *xExpr*, *yExpr*, *t*

Representação gráfica polar: **Table** *expr*, θ

Nota: O comando **Table** não é válido para a representação gráfica 3D, sequências ou equações diferenciais. Como alternativa, pode querer utilizar **BldData**.

No modo de representação gráfica da função.

Table 1.25x∗cos(x) ENTER

x	1		
0.	0.		
1.	.67538		
2.	-1.04		
3.	-3.712		
4.	-3.268		

Table cos(time),time ENTER

x	1	2	3
0.	0.	1.	
1.	.67538	.5403	
2.	-1.04	-.4161	
3.	-3.712	-.99	
4.	-3.268	-.6536	

Teclas 2nd [TAN]

tan(*expressão1***)** ⇒ *expressão*
tan() ⇒ *lista*

tan(*expressão 1***)** devolve a tangente do argumento como uma expressão.

tan(*lista 1***)** devolve uma lista das tangentes de todos os elementos em *lista 1*.

Nota: O argumento é interpretado como um ângulo em graus, grados ou radianos, de acordo com o modo de ângulo actual. Pode utilizar °, ᴳ ou ʳ para substituir a definição do modo de ângulo temporariamente.

No modo de ângulo Degree:

tan((π/4)ʳ) ENTER 1

tan(45) ENTER 1

tan({0,60,90}) ENTER
{0 √3 undef}

No modo de ângulo Gradian:

tan((π/4)ʳ) ENTER $\dfrac{200 \bullet \tan\left(\frac{\pi}{4}\right)}{\pi}$

tan(50) ENTER 1

tan({0,50,100}) ENTER
{0 1 undef}

No modo de ângulo Radian:

tan(π/4) ENTER 1

tan(45°) ENTER 1

tan({π,π/3,-π,π/4}) ENTER
{0 √3 0 1}

tan(*Matriz quadrada 1***)** ⇒ *Matriz quadrada*

Devolve a tangente da matriz de *matriz quadrada 1*. Este procedimento *não* é a mesma coisa que calcular a tangente de cada elemento. Para mais informações sobre o método de cálculo, consulte **cos()**.

*matriz quadrada 1*tem de estar na diagonal. O resultado contém sempre números de pontos flutuantes.

No modo de ângulo Radian:

`Devolve a tangente inversa da matriz de matriz quadrada 1.`

$$\begin{bmatrix} -28.291\ldots & 26.088\ldots & 11.114\ldots \\ 12.117\ldots & -7.835\ldots & -5.481\ldots \\ 36.818\ldots & -32.806\ldots & -10.459\ldots \end{bmatrix}$$

Teclas ◆ [TAN⁻¹]

tan⁻¹ (*expressão1***)** ⇒ *expressão*
tan⁻¹ () ⇒ *lista*

tan⁻¹ (*expressão1***)** devolve o ângulo cuja tangente é a *expressão1* como uma expressão.

tan⁻¹ () *lista1* devolve a lista das tangentes inversas de cada elemento de *lista1*.

Nota: O resultado é devolvido como um ângulo em graus, grados ou radianos, de acordo com a definição do modo de ângulo actual.

No modo de ângulo Degree:

`tan⁻¹(1)` ENTER 45

No modo de ângulo Gradian:

`tan⁻¹(1)` ENTER 50

No modo de ângulo Radian:

`tan⁻¹({0,.2,.5})` ENTER
{0 .197... .463...}

tan⁻¹(*Matriz quadrada 1***)** ⇒ *Matriz quadrada*

Devolve a tangente hiperbólica da matriz de *Matriz quadrada 1*. Este procedimento *não* é a mesma coisa que calcular a tangente inversa de cada elemento. Para mais informações sobre o método de cálculo, consulte **cos()**.

*matriz quadrada 1*tem de estar na diagonal. O resultado contém sempre números de pontos flutuantes.

No modo de ângulo Radian:

`tan⁻¹([1,5,3;4,2,1;6,-2,1])` ENTER

$$\begin{bmatrix} -.083\ldots & 1.266\ldots & .622\ldots \\ .748\ldots & .630\ldots & -.070\ldots \\ 1.686\ldots & -1.182\ldots & .455\ldots \end{bmatrix}$$

Menu MATH/Hyperbolic

tanh(*expressão1***)** ⇒ *expressão*
tanh() ⇒ *lista*

tanh(*expressão 1***)** devolve a tangente hiperbólica do argumento como uma expressão.

tanh(*lista***)** devolve uma lista de tangentes hiperbólicas de cada elemento de *lista 1*.

`tanh(1.2)` ENTER .833...

`tanh({0,1})` ENTER {0 tanh(1)}

tanh(*Matriz quadrada 1***)** ⇒ *Matriz quadrada*

Devolve a tangente hiperbólica da matriz de *Matriz quadrada 1*. Este procedimento não é a mesma coisa que calcular a tangente hiperbólica de cada elemento. Para mais informações sobre o método de cálculo, consulte **cos()**.

*matriz quadrada 1*tem de estar na diagonal. O resultado contém sempre números de pontos flutuantes.

No modo de ângulo Radian:

`tanh([1,5,3;4,2,1;6,-2,1])` ENTER

$$\begin{bmatrix} -.097\ldots & .933\ldots & .425\ldots \\ .488\ldots & .538\ldots & -.129\ldots \\ 1.282\ldots & -1.034\ldots & .428\ldots \end{bmatrix}$$

Menu MATH/Hyperbolic

tanh^{-1}(*expressão1***)** ⇒ *expressão*
tanh^{-1}() ⇒ *lista*

tanh(*expressão 1***)** devolve a tangente hiperbólica do argumento como uma expressão.

tanh^{-1}() *lista 1* devolve uma lista das tangentes hiperbólicas inversas de cada elemento de *lista 1*.

No modo de formato complexo Rectangular:

tanh^{-1}(0) ENTER 0

tanh^{-1}({1,2.1,3}) ENTER

$$\{\infty \quad .518\ldots - 1.570\ldots \cdot i \quad \frac{\ln(2)}{2} - \frac{\pi}{2} \cdot i\}$$

---

tanh^{-1}(*Matriz quadrada 1***)** ⇒ *Matriz quadrada*

Devolve a tangente hiperbólica da matriz de *Matriz quadrada 1*. Este procedimento *não* é a mesma coisa que calcular a tangente hiperbólica inversa de cada elemento. Para mais informações sobre o método de cálculo, consulte **cos()**.

*matriz quadrada 1*tem de estar na diagonal. O resultado contém sempre números de pontos flutuantes.

No modo de ângulo Radian e no modo do formato complexo Rectangular:

tanh^{-1}([1,5,3;4,2,1;6,⁻2,1]) ENTER

$$\begin{bmatrix} -.099\ldots+.164\ldots\cdot i & .267\ldots-1.490\ldots\cdot i & \ldots \\ -.087\ldots-.725\ldots\cdot i & .479\ldots-.947\ldots\cdot i & \ldots \\ .511\ldots-2.083\ldots\cdot i & -.878\ldots+1.790\ldots\cdot i & \ldots \end{bmatrix}$$

taylor()

Menu MATH/Calculus

taylor(*expressão1***,** *var***,** *order***[,** *point***]**) ⇒ *expressão*

Devolve o polinómio Taylor requerido. O polinómio inclui termos não zero de graus de números inteiros de *ordem* em (*var* menos *ponto*). **taylor()** devolve-se a si próprio se não existir nenhuma série de potência truncada desta ordem ouse se quiser expoentes negativos ou fraccionários. Utilize a substituição e/ou a multiplicação temporária por uma potência de (*var* menos *ponto*) para determinar séries de potências gerais.

ponto predefine para zero e é o ponto de expansão.

taylor(*e*^(√(x)),x,2) ENTER
taylor(*e*^(t),t,4)|t=√(x) ENTER

$$\blacksquare\ \text{taylor}\left(e^{\sqrt{x}}, x, 2\right)$$
$$\text{taylor}\left(e^{\sqrt{x}}, x, 2, 0\right)$$
$$\blacksquare\ \text{taylor}\left(e^{t}, t, 4\right) \mid t = \sqrt{x}$$
$$\frac{x^2}{24} + \frac{x^{3/2}}{6} + \frac{x}{2} + \sqrt{x} + 1$$

taylor(1/(x∗(x−1)),x,3) ENTER

$$\blacksquare\ \text{taylor}\left(\frac{1}{x\cdot(x-1)}, x, 3\right)$$
$$\text{taylor}\left(\frac{1}{x\cdot(x-1)}, x, 3, 0\right)$$

expand(taylor(x/(x∗(x−1)),
x,4)/x,x) ENTER

$$\blacksquare\ \text{expand}\left(\frac{\text{taylor}\left(\frac{x}{x\cdot(x-1)}, x\right.}{x}\right.$$
$$-x^3 - x^2 - x - \frac{1}{x} - 1$$

tCollect()

MATH\Algebra\Trig menu

tCollect(*expressão1***)** ⇒ *expressão*

Devolve uma expressão em que os produtos e as potências de números inteiros dos senos e co-senos são convertidas para uma combinação linear de senos e co-senos de vários ângulos, somas de ângulos e diferenças de ângulo. A transformação converte polinómios trigonométricos para uma combinação linear da harmónica.

Por vezes **tCollect()** conseguirá responder aos seus objectivos quando a simplificação trigonométrica predefinida não conseguir. **tCollect()** tende para transformações inversas efectuadas por **tExpand()**. Por vezes, a aplicação de **tExpand()** num resultado de **tCollect()**, ou vice-versa, em dois passos separados simplifica uma expressão.

tCollect((cos(α))^2) ENTER

$$\frac{\cos(2\cdot\alpha)+1}{2}$$

tCollect(sin()cos())

tExpand() MATH\Algebra\Trig menu

tExpand(*expressão1***)** ⇒ *expressão*

Devolve uma expressão em que os senos e os co-senos dos vários ângulos de números inteiros, somas de ângulos e diferenças de ângulos são expendidas. Como a identidade $(\sin(x))^2+(\cos(x))^2=1$, existem muitos resultados equivalentes possíveis. Por consequência, um resultado pode diferir de um resultado mostrado noutras publicações.

Por vezes **tCollect()** conseguirá responder aos seus objectivos quando a simplificação trigonométrica predefinida não conseguir. **tCollect()** tende para transformações inversas efectuadas por **tExpand()**. Por vezes, a aplicação de **tExpand()** num resultado de **tCollect()**, ou vice-versa, em dois passos separados simplifica uma expressão.

Nota: A escala do modo Degree por π/180 interfere na capacidade de **tExpand()** para reeconhecer formatos expansíveis. Para obter melhores resultados, **tExpand()** deve estar no modo Radian.

```
tExpand(sin(3ϕ)) ENTER
        4·sin(ϕ)·(cos(ϕ))² − sin(ϕ)

tExpand(cos(α−β)) ENTER
        cos(α)·cos(β)+sin(α)·sin(β)
```

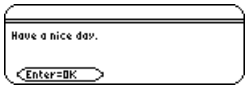
Text CATALOG

Text *Cadeia de comando*

Mostra a *caixa de diálogo da cadeia* de caracteres Cadeia de comando.

Se utilizada como parte de um bloco **Dialog...EndDlog**, *Cadeia de comando* aparece na caixa de diálogo. Se utilizada como uma instrução autónoma, **Texto** cria uma caixa de diálogo para ver a cadeia.

```
Text "Have a nice day." ENTER
                                Done
```



Then

Consulte **If,** na página 881.

timeCnv() CATALOG

timeCnv(*segundos***)** ⇒ *lista*

Converte segundos em unidades de tempo que possam ser compreendidas mais facilmente para fins de avaliação. A lista tem o formato {*dias,horas,minutos,segundos*}.

Nota: Consulte também **checkTmr()** e **startTmr()**.

```
timeCnv(152442117)
                    {1764  9  1  57}
```

Title CATALOG

Title *Cadeia do título, [Lbl]*

Cria o título de um menu pendente ou caixa de diálogo quando utilizado numa construção **Toolbar** or **Custom** ou num bloco **Dialog...EndDlog**.

Nota: *Lbl* só é válida na construção **Toolbar**. Quando presente, permite escolher o menu para derivar para uma etiqueta especificada dentro do programa.

Segmento do programa:

```
⋮
:Dialog
:Title          "This is a dialog box"
:Request              "Your name",Str1
:Dropdown       "Month you were born",
            seq(string(i),i,1,12),Var1
:EndDlog
⋮
```

tmpCnv() CATALOG

tmpCnv(*expressão1_ °Unidade de temperatura 1***,** *_ °Unidade de temperatura 2***)**

⇒ *expressão _Unidade de temperatura 2*

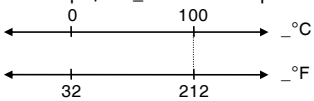
Converte o valor da temperatura especificado pela *expressão* 1 de uma unidade para outra. As unidades de temperatura válidas são:

_°C	Celsius
_°F	Fahrenheit
_°K	Kelvin
_°R	Rankine

```
tmpCnv(100_°c,_°f) ENTER        212.•_°F
tmpCnv(32_°f,_°c) ENTER           0.•_°C
tmpCnv(0_°c,_°k) ENTER        273.15•_°K
tmpCnv(0_°f,_°r) ENTER        459.67•_°R
```

Nota: Para seleccionar as unidades de temperatura num menu, prima 2nd [UNITS]

Por exemplo, 100_°C converte-se para 212_°F:



Para °, prima 2nd [°].

Para converter um intervalo de temperatura, utilize **ΔtmpCnv()**.

ΔtmpCnv() CATALOG

ΔtmpCnv(*expressão1_ °Unidade de temperatura 1***,** *_ °Unidade de temperatura 2***)**

⇒ *expressão _Unidade de temperatura 2*

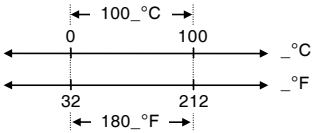
Converte um intervalo de temperatura (a diferença entre dois valores de temperatura) especificado pelas *expressão 1* de uma unidade para outra. As unidades de temperatura válidas são:

_°C	Celsius
_°F	Fahrenheit
_°K	Kelvin
_°R	Rankine

Para °, prima [2nd] [°].
Para _, prima [◆] [_].

1_°C e 1_°K têm a mesma magnitude como 1_°F e 1_°R. No entanto, 1_°C é 9/5 tão grande como 1_°F.

Por exemplo, um intervalo 100_°C (de 0_°C a 100_°C) é equivalente a intervalo de 180_°F:



Pasra converter um valor de temperatuta específici em vez de um intervalo, utilize **tmpCnv()**.

Para obter Δ, pode premir [◆] [(] [↑] [D] (ou [2nd] [CHAR] 1 5).

```
ΔtmpCnv(100_°c,_°f) ENTER
                          180.•_°F
ΔtmpCnv(180_°f,_°c) ENTER
                          100.•_°C
ΔtmpCnv(100_°c,_°k) ENTER
                          100.•_°K
ΔtmpCnv(100_°f,_°r) ENTER
                          100.•_°R
ΔtmpCnv(1_°c,_°f) ENTER
                           1.8•_°F
```

Nota: Para seleccionar as unidades de temperatura num menu, prima [2nd] [UNITS]

Toolbar CATALOG

Toolbar
bloco
EndTBar

Cria um menu da barra de ferramentas.

bloco pode ser uma instrução individual ou uma sequência de instruções separadas com o carácter ":" . As instruções podem ser Title ou Item.

Itens têm de ter etiquetas. Um Title tem de ter também uma etiqueta se não tiver um item.

Segmento do programa:

```
⋮
:Toolbar
:  Title "Examples"
:  Item "Trig", t
:  Item "Calc", c
:  Item "Stop", Pexit
:EndTbar
⋮
```

Nota: Quando executar num programa, este segmento cria um menu com três escolhas que derivam para três locais no programa.

Trace CATALOG

Trace

Desenha uma Smart Graph e coloca o cursor de traço na primeira função Y= definida na posição do cursor definida previamente ou na posição de reiniciação se for necessário uma nova representação gráfica.

Permite utilizar o cursor e maioria das teclas quando editar valores de coordenadas. Algumas teclas, como as teclas de função, [APPS], e [MODE], não são activadas durante o desenho.

Nota: Prima [ENTER] para retomar a operação.

:Try
bloco 1
Else
bloco 2
EndTry

Executa o *bloco 1* excepto se ocorrer um erro. A execução do programa transfere-se para o *bloco 2* se ocorrer um erro em *bloco 1*. A variável errornum contém o número de erro para que o programa possa efectuar a recuperação do erro.

bloco 1 e *bloco 2* podem ser ums instrução individual ou uma série de instruções separadas com o carácter ":" .

Segmento do programa:

```
⋮
:Try
:  NewFold(temp)
:  Else
:  ◉Already exists
:  ClrErr
:EndTry
⋮
```

Nota: Consulte **ClrErr** e **PassErr**.

TwoVar — Menu MATH/Statistics

TwoVar , *lista 2*[**,** [*lista 3*] [**,** *lista 4*, *lista 5*]]

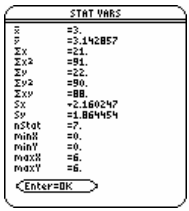
Calcula a estatística **TwoVar** e actualiza todas as variáveis estatísticas do sistema.

Todas as listas têm de ter dimensões iguais, excepto a *lista 5*.

lista 1 representa xlist.
lista 2 representa ylist.
lista 3 representa a frequência.
lista 4 representa os códigos de categorias.
lista 5 representa a lista de inclusão de categorias.

Nota: De *lista1* a *list4* têm de ser o nome de uma variável ou c1–c99 (colunas na última variável de dados apresentada no Editor de Dados/Matriz)). *lista5* não tem de ser o nome de uma variável e não pode ser c1–c99.

```
{0,1,2,3,4,5,6}→L1 [ENTER]
                          {0 1 2 ...}
{0,2,3,4,3,4,6}→L2 [ENTER]
                          {0 2 3 ...}
TwoVar L1,L2 [ENTER]            Done
ShowStat [ENTER]
```



Unarchiv — CATALOG

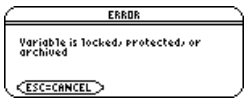
Unarchiv *var1* [, *var2*] [, *var3*] …

Move as variáveis especificadas da memória de arquivo de dados do utilizador para a RAM.

Pode aceder a uma variável arquivada da mesma forma que acederia a uma variável da RAM. No entanto, não pode eliminar, mudar o nome ou guardar numa variável arquivada porque é bloqueada automaticamente.

Para arquivar variáveis, utilize **Archive**.

```
10→arctest [ENTER]              10
Archive arctest [ENTER]       Done
5*arctest [ENTER]               50
15→arctest [ENTER]
```




```
[ESC]
Unarchiv arctest [ENTER]      Done
15→arctest [ENTER]              15
```

unitV — Menu MATH/Matrix/Vector

unitV(*vector1***)** ⇒ *vector*

Devolve um vector de unidade de coluna ou linha, dependendo da forma de *vector1*.

vector1 tem de ser uma matriz de linha ou coluna individual.

unitV([a,b,c]) [ENTER]

$$\left[\frac{a}{\sqrt{a^2+b^2+c^2}}\ \frac{b}{\sqrt{a^2+b^2+c^2}}\ \frac{c}{\sqrt{a^2+b^2+c^2}}\right]$$

unitV([1,2,1]) [ENTER]

$$\left[\frac{\sqrt{6}}{6}\ \frac{\sqrt{6}}{3}\ \frac{\sqrt{6}}{6}\right]$$

unitV([1;2;3])

Unlock — CATALOG

Unlock *var1*[, *var2*][, *var3*]...

Desbloqueia as variáveis especificadas.

Nota: As variáveis podem ser bloqueadas com o comando **Lock**.

variance — Menu MATH/Statistics

variance(*lista*[, *lista de frequência*]**)** ⇒ *expressão*

Devolve a variação de *lista*.

```
variance({a,b,c}) ENTER
```

$$\frac{a^2-a\cdot(b+c)+b^2-b\cdot c+c^2}{3}$$

Cada elemento da *lista de frequência* conta o número de ocorrências consecutivas do elemento correspondente na *lista*.

```
variance({1,2,5,⁻6,3,⁻2}) ENTER
```

31/2

Nota: *lista* tem de conter pelo mneos dois elementos.

```
variance({1,3,5},{4,6,2}) ENTER
```

68/33

---

variance(*matriz1*[, *matriz de frequência*]**)** ⇒ *matriz*

Devolve um vector de linha com a variação de cada coluna em *matriz 1*.

```
variance([1,2,5;⁻3,0,1;
.5,.7,3]) ENTER
```

[4.75 1.03 4]

Cada elemento da *matriz de frequência* conta o número de ocorrências consecutivas do elemento correspondente na *matriz 1*.

```
variance([-1.1,2.2;3.4,5.1;
-2.3,4.3],[6,3;2,4;5,1]) ENTER
```

[3.91731,2.08411]

Nota: *matriz 1* tem de conter pelo menos duas linhas.

when() — CATALOG

when(*condição, Resultado verdadeiro* [, *Resultado falso*] [, *Resultado desconhecido*]**)** ⇒ *expressão*

Devolve *Resultado verdadeiro*, *Resultado falso*, ou *Resultado desconhecido*, dependendo se a *condição* é verdadeira, falsa ou desconhecida. Devolve a entrada se existirem poucos argumentos para especificar o resultado adequado.

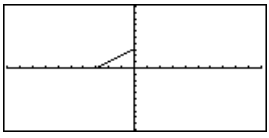
Omita Resultado falso e *Resultado desconhecido* para definir uma condição apenas na região em que a condição é verdadeira.

```
when(x<0,x+3)|x=5 ENTER
```

when(x<0,3+x)

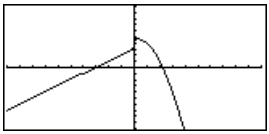
Utilize um undef *Resultado falso* para definir uma expressão que é representada graficamente apenas num intervalo.

```
ClrGraph ENTER
Graph when(x≥⁻π and x<0,x+3,undef)
ENTER
```



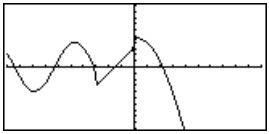
Omita apenas o *Resultado desconhecido* para definir uma expressão de duas partes.

```
Graph when(x<0,x+3,5- x^2) ENTER
```



Aninhe **when()** para definir expressões com mais de duas partes.

```
HOME
ClrGraph ENTER                          Done
Graph when(x<0,when(x<⁻π,
4*sin(x),2x+3),5-x^2) ENTER
```



when() é muito útil para definir funções recursivas.

```
when(n>0,n*factoral(n-1),1)
→factoral(n) ENTER                      Done
factoral(3) ENTER                          6
3! ENTER                                   6
```

While — CATALOG

Condição **While**
bloco
EndWhile

Executa as instruções no *bloco* desde que a *condição* seja verdadeira.

bloco pode ser uma instrução individual ou uma sequência de instruções separadas com o carácter ":".

Segmento do programa:

```
⋮
:1→i
:0→temp
:While i<=20
:  temp+1/i→temp
:  i+1→i
:EndWhile
:Disp "sum of reciprocals up to 20",te
⋮
```

"With"

Consulte | na página 968.

xor — Menu MATH/Test

Expressão booleana 1 **xor** *Expressão booleana 2* ⇒ *Expressão booleana*

Devolve verdadeiro se a *Expressão booleana 1* for verdadeira e *Expressão booleana 2* for falsa, ou vice-versa.
Devolve falso se *Expressão booleana 1* e a *Expressão booleana 2* forem ambas verdadeiras ou falsas. Devolve uma expressão booleana simplificada se não for possível resolver as Expressões booleanas originais para verdadeiro ou falso.

Nota: Consulte **or**.

```
true xor true ENTER                     false
(5>3) xor (3>5) ENTER                    true
```

número inteiro1 **xor** *número inteiro2* ⇒ *número inteiro*

Compara dois números inteiros reais bit a bit com uma operação **xor**. Internalmente, ambos os números inteiros são convertidos para números binários de 32 bits. Quando bits correspondentes forem comparados, o resultado é 1 se um dos bits for 1; o resultado é 0 se ambos os bits forem 0 ou 1.O valor devolvido representa os resultados das partes e aparece de acordo com o modo Base.

Pode introduzir os números inteiros em qualquer base do número. Para uma introdução binária ou hexadecimal, tem de utilizar o prefixo 0b ou 0h, respectivamente. Sem um prefixo, os números inteiros são tratados como decimal (base 10).

Se introduzir um número inteiro decimal muito grande para uma forma binária de 32 bits, é utilizada uma operação do módulo simétrico para colocar o valor no intervalo adequado.

Nota: Consulte **or**.

No modo base Hex:

```
0h7AC36 xor 0h3D5F ENTER        0h79169
```

Importante: Zero, not the letter O.

No modo base Bin:

```
0b100101 xor 0b100 ENTER       0b100001
```

Nota: Uma entrada binária pode ter até 32 dígitos (não contar o prefixo 0b). Uma entrada hexadecimal pode ter até 8 dígitos.

XorPic — CATALOG

XorPic *picVar*[, *linha*] [, *coluna*]

Mostra a imagem guardada em *picVar* no ecrã Graph actual.

Utiliza **xor** lógica para cada pixel. Apenas as posições de pixéis exclusivas do ecrã ou da imagem são ligadas. Esta instrução desliga os pixéis ligados a ambas as imagens.

picVar tem de conter um tipos de dados de imagem.

linha e *coluna*, se incluídas, especificam as coordenadas dos pixéis para o canto superior esquerdo da imagem. As predefinições são (0, 0).

zeros() — Menu MATH/Algebra

zeros(*expressão***,** *var***)** ⇒ *lista*

Devolve uma lista de valores reais candidatos de *var* que tornam a *expressão*=0. **zeros()** faz isto através do cálculo de **exp▶list(solve(***expressão*=0,*var***),***var***)**.

Para alguns efeitos, a forma de resultado para **zeros()** é mais conveniente do que de **solve()**. No entanto, a forma do resultado de **zeros()** não pode expressar soluções implícitas, soluções que requerem desigualdade ou soluções que não envolvam *var*.

Nota: Consulte também **cSolve()**, **cZeros()**, e **solve()**.

```
zeros(a*x^2+b*x+c,x) ENTER
```

$$\left\{\frac{-\left(\sqrt{b^2-4\cdot a\cdot c}+b\right)}{2\cdot a}\ \frac{\sqrt{b^2-4\cdot a\cdot c}-b}{2\cdot a}\right\}$$

```
a*x^2+b*x+c|x=ans(1)[2] ENTER        0
```

```
exact(zeros(a*(e^(x)+x)
(sign (x)-1),x)) ENTER                {}
```

```
exact(solve(a*(e^(x)+x)
(sign (x)-1)=0,x)) ENTER
```

$e^x + x = 0$ or $x>0$ or $a = 0$

zeros({*expressão1***,** *expressão2***},** **{***varOrGuess1***,** *varOrGuess2* [, …]**})** ⇒ *matriz*

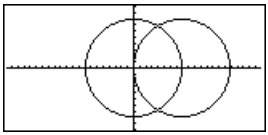
Devolve zeros reais candidatos das *expressões* algébricas simultâneas, em que cada *varOrGuess* especifica um desconhecido cujo valor procura.

Optionalmente, pode especificar uma hipótese inicial para uma variável. Cada varOrGuess tem de ter o formato:

variável
– ou –
variável = r*eal ou número não real*

Por exemplo, x é válido e, por isso, é x=3.

Se todas as expressões forem polinomiais e não especificar quaisquer hipóteses iniciais, **zeros()** utilize o método de eliminação Gröbner/Buchberger lexical para tentar determinar **todos** os zeros reais.



Por exemplo, suponha que tem uma circunferência de raio r na origem e outra circunferência de raio r centrada onde a primeira circunferência cruza com o eixo x positivo. Utilize **zeros()** para localizar as intercepções.

Como ilustrado pelo r no exemplo à direita, as expressões *polinomiais* simultâneas podem ter variáveis adicionais que não têm valores, mas representam valores numéricos dados que podem ser substituídos posteriormente.

```
zeros({x^2+y^2-r^2,
(x-r)^2+y^2-r^2},{x,y}) ENTER
```

$$\begin{bmatrix} \frac{r}{2} & \frac{\sqrt{3}\cdot r}{2} \\ \frac{r}{2} & \frac{-\sqrt{3}\cdot r}{2} \end{bmatrix}$$

Cada linha da matriz resultante representa um zero alternativo, com os componentes ordenados como na lista *varOrGuess*. Para extrair uma linha, indexe a matriz por [*linha*].

Extraia a linha 2:

```
ans(1)[2]
```

Pode também incluir variáveis desconhecidas que não aparecem nas expressões. Por exemplo, pode incluir z como um desconhecido para ampliar o exemplo anterior para dois cilindros de intercepção paralelos de raio r. Os zeros do cilindro ilustram como as famílias de zeros podem conter constantes arbitrárias na forma *ck*, em que *k* é um sufixo do número inteiro de 1 a 255. O sufixo reinicia-se 1 quando utilizar **ClrHome** ou F1 8:Clear Home.

```
zeros({x^2+y^2-r^2,
(x-r)^2+y^2-r^2},{x,y,z}) ENTER
```

$$\begin{bmatrix} \frac{r}{2} & \frac{\sqrt{3}\cdot r}{2} & c1 \\ \frac{r}{2} & \frac{-\sqrt{3}\cdot r}{2} & c1 \end{bmatrix}$$

Para sistemas polinomiais, o tempo de cálculo ou o consumo da memória podem depender muito da ordem em que lista os desconhecidos. Se a escolha inicial consumir a memória ou a sua paciência, tente redispor as variáveis nas expressões e/ou na lista *varOrGuess*.

Se não incluir quaisquer hipóteses e se qualquer expressão for não polinomial em qualquer variável, mas todas as expressões forem lineares em todos os desconhecidos, **cZeros()** utiliza a eliminação Gaussian para tentar determinar todos os zeros.

```
zeros({x+e^(z)*y-1,x-y-sin(z)},{x,y
}) ENTER
```

$$\begin{bmatrix} \frac{e^z\cdot\sin(z)+1}{e^z+1} & \frac{-(\sin(z)-1)}{e^z+1} \end{bmatrix}$$

Se um sistema não for polinomial em todas variávies nem linear nos desconhecidos, **zeros()** determina o zero mais pretendido com um método de repetição aproximada. Para o fazer, o número de desconhecidos tem de ser igual ao número de expressões e todas as outras variáveis nas expressões têm de ser simplificadas para números.

```
zeros({e^(z)*y-1,-y-sin(z)},
{y,z}) ENTER
```

[.041… 3.183…]

Cada desconhecido começa no valor de hipótese se existir um; caso contrário, começa em 0.0.

Utilize as hipóteses para procurar zeros adicionais um a um. Para convergência, uma hipótese pode ter de ser próxima de um zero.

```
zeros({e^(z)*y-1,-y-sin(z)},
{y,z=2π}) ENTER
```

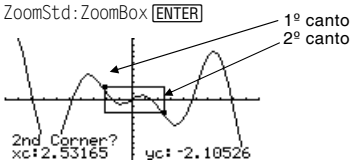
ZoomBox CATALOG

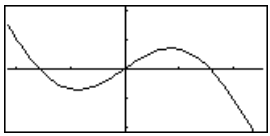
ZoomBox

Mostra o ecrã Graph, permite desenhar uma caixa que define uma nova janela de visualização e actualiza a janela.

No modo de representação gráfica da função.

1.25x*cos(x)→y1(x) [ENTER] Done
ZoomStd:ZoomBox [ENTER]





O ecrã depois de definir **ZoomBox**, premindo [ENTER] a segunda vez.

ZoomData CATALOG

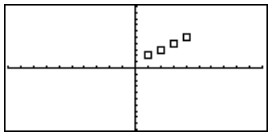
ZoomData

Ajusta as definições da janela baseadas nos desenhos definidos actualmente (e dados) para todos os pontos de dados estatísticos sejam exemplificados e apresentados no ecrã Graph.

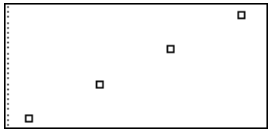
Nota: Não ajuste ymin e ymax para histogramas.

No modo de representação gráfica da função.

{1,2,3,4}→L1 [ENTER] {1 2 3 4}
{2,3,4,5}→L2 [ENTER] {2 3 4 5}
newPlot 1,1,L1,L2 [ENTER] Done
ZoomStd [ENTER]



[HOME]
ZoomData [ENTER]



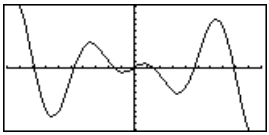
ZoomDec CATALOG

ZoomDec

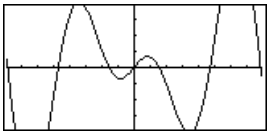
Ajusta a janela de visualização para que Δx e $\Delta y = 0.1$ e apresente o ecrã Graph com a origem centrada no ecrã.

No modo de representação gráfica da função.

1.25x∗cos(x)➔y1(x) [ENTER] Done
ZoomStd [ENTER]



[HOME]
ZoomDec [ENTER]



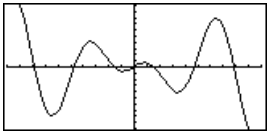
ZoomFit CATALOG

ZoomFit

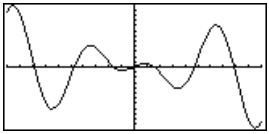
Mostra o ecrã Graph e calcula as dimensões necessárias da janela para as variáveis dependentes para ver a imagens de todas as definições das variáveis independentes actuais.

No modo de representação gráfica da função.

1.25x∗cos(x)➔y1(x) [ENTER] Done
ZoomStd [ENTER]



[HOME]
ZoomFit [ENTER]



ZoomIn CATALOG

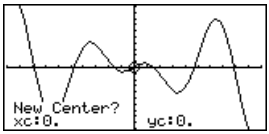
ZoomIn

Mostra o ecrã Graph, permite definir um ponto central para aumentar o zoom e actualiza a janela de visualização.

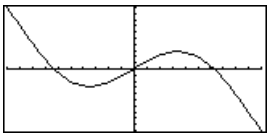
A magnitude do zoom depende de Zoom, xFact e yFact. No modo de representação gráfica 3D, a magnitude depende de xFact, yFact, and zFact.

No modo de representação gráfica da função.

1.25x∗cos(x)➔y1(x) [ENTER] Done
ZoomStd:ZoomIn [ENTER]



[ENTER]



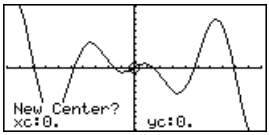
ZoomInt CATALOG

ZoomInt

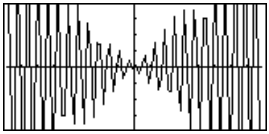
Mostra o ecrã Graph, permite definir um ponto central para o zoom e ajusta as definições da janela para que cada pixel seja um número inteiro em todas as direcções.

No modo de representação gráfica da função.

1.25x∗cos(x)➔y1(x) ENTER Done
ZoomStd:ZoomInt ENTER



ENTER



ZoomOut CATALOG

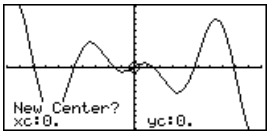
ZoomOut

Mostra o ecrã Graph, permite definir um ponto central para diminuir o zoom e actualiza a janela de visualização.

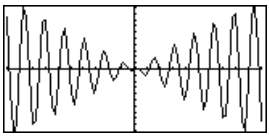
A magnitude do zoom depende de Zoom, xFact e yFact. No modo de representação gráfica 3D, a magnitude depende de xFact, yFact, and zFact.

No modo de representação gráfica da função.

1.25x∗cos(x)➔y1(x) ENTER Done
ZoomStd:ZoomOut ENTER



ENTER



ZoomPrev CATALOG

ZoomPrev

Mostra o ecrã Graph e actualiza a janela de visualização com as definições em utilização antes do último zoom.

ZoomRcl CATALOG

ZoomRcl

Mostra o ecrã Graph e actualiza a janela de visualização com as definições guardadas com a instrução **ZoomSto**.

ZoomSqr CATALOG

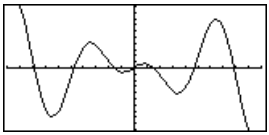
ZoomSqr

Mostra o ecrã Graph, ajusta as definições da janela x ou y para que cada pixel represente uma altura e uma largura iguais no sistema de coordenadas e actualiza a janela de visualização.

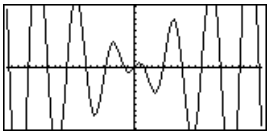
No modo de representação gráfica 3D, **ZoomSqr** aumenta os dois eixos mais pequenos para ficarem iguais ao eixo mais longo.

No modo de representação gráfica da função.

1.25x∗cos(x)→y1(x) ENTER Done
ZoomStd ENTER



HOME
ZoomSqr ENTER



ZoomStd CATALOG

ZoomStd

Define as variáveis da janela para os valores standard seguintes e actualiza a janela de visualização.

Representação gráfica da função:
x: [⁻10, 10, 1], y: [⁻10, 10, 1] and xres=2

Representação gráfica paramétrica:
t: [0, 2π, π/24], x: [⁻10, 10, 1], y:[⁻10, 10, 1]

Representação gráfica polar:
θ: [0, 2π, π/24], x: [⁻10, 10, 1], y: [⁻10, 10, 1]

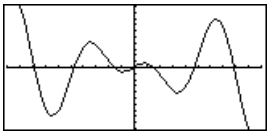
Representação gráfica da sequência:
nmin=1, nmax=10, plotStrt=1, plotStep=1, [⁻10, 10, 1], y: [⁻10, 10, 1]

Representação gráfica 3D:
eye°=20, eye°=70, eye°=0 [⁻10, 10, 14], y: [⁻10, 10, 14],
z: [⁻10, 10], ncontour=5

Representação gráfica de equações diferenciais:
t: [0, 10, .1, 0], x: [⁻1, 10, 1], y: [⁻10, 10, 1], ncurves=0, Estep=1, diftol=.001, fldres=14, dtime=0

No modo de representação gráfica da função.

1.25x∗cos(x)→y1(x) ENTER Done
ZoomStd ENTER



ZoomSto CATALOG

ZoomSto

Guarda as definições da janela actual na memória Zoom. Pode utilizar **ZoomRcl** para restaurar as definições.

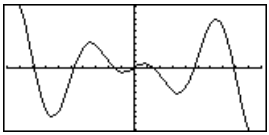
ZoomTrig CATALOG

ZoomTrig

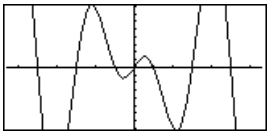
Mostra o ecrã Graph, defines Δx para π/24, e xscl para π/2, centra a origem, define as definições para [⁻4, 4, .5], e actualiza a janela de visualização.

No modo de representação gráfica da função.

1.25x∗cos(x)→y1(x) ENTER Done
ZoomStd ENTER



HOME
ZoomTrig ENTER



+ (adicionar) Tecla +

expressão1 + *expressão2* ⇒ *expressão*

Devolve a soma de *expressão 1* e *expressão 2*.

56 ENTER 56
ans(1)+4 ENTER 60
ans(1)+4 ENTER 64
ans(1)+4 ENTER 68
ans(1)+4 ENTER 72

+ *list2* ⇒ *list*
matriz1 + *matriz2* ⇒ *matriz*

Devolve uma lista (ou matriz) com as somas dos elementos correspondentes em *lista 1* e *lista 2* (ou *matriz 1* e *matriz 2*).

As dimensões dos argumentos têm de ser iguais.

{22,π,π/2}→L1 ENTER {22 π π/2}
{10,5,π/2}→L2 ENTER {10 5 π/2}
L1+L2 ENTER {32 π+5 π}

ans(1)+{π,⁻5,⁻π} ENTER
{π+32 π 0}

[a,b;c,d]+[1,0;0,1] ENTER

$$\begin{bmatrix} a+1 & b \\ c & d+1 \end{bmatrix}$$

---

expressão + ⇒ *lista*
+ *expressão* ⇒ *lista*

Devolve uma lista com as somas de *expressão* e de cada elemento em *lista 1*.

15+{10,15,20} ENTER {25 30 35}

{10,15,20}+15 ENTER {25 30 35}

---

expressão + *matriz1* ⇒ *matriz*
matriz1 + *expressão* ⇒ *matriz*

Devolve uma matriz com a *expressão* adicionada a cada elemento na diagonal de *matriz 1*. *matriz 1* tem de ser quadrada.

Nota: Utilize .+ (ponto adição) para adicionar uma expressão a cada elemento.

20+[1,2;3,4] ENTER

$$\begin{bmatrix} 21 & 2 \\ 3 & 24 \end{bmatrix}$$

− (subtrair) Tecla −

expressão1 - *expressão2* ⇒ *expressão*

Devolve a *expressão 1* menos a *expressão 2*.

6−2 ENTER 4

π−π/6 ENTER $\frac{5 \cdot \pi}{6}$

– lista 2 ⇒ *lista*
matriz1 – matriz2 ⇒ *matriz*

Subtrai cada elemento de *lista 2* (ou *matriz2*) do elemento correspondente em *lista 1* (ou *matriz1*), e devolve os resultados.

As dimensões dos argumentos têm de ser iguais.

{22,π,π/2}–{10,5,π/2} ENTER → {12 π–5 0}

[3,4]–[1,2] ENTER → [2 2]

expressão – ⇒ *lista*
– expressão ⇒ *lista*

Subtrai cada elemento de *lista 1* da *expressão* ou subtrai a *expressão* de cada elemento de *lista 1* e devolve uma lista dos resultados.

15–{10,15,20} ENTER → {5 0 -5}

{10,15,20}–15 ENTER → {-5 0 5}

expressão – matriz1 ⇒ *matriz*
matriz1 – expressão ⇒ *matriz*

expressão – matriz1 devolve uma matriz de *expressão* vezes a matriz de identidade menos a *matriz1*. *matriz1* tem de ser quadrada.

matriz1 – expressão devolve uma matriz de *expressão* vezes a matriz de identidade substraída de *matriz1*. *matriz1* tem de ser quadrada.

Nota: Utilize **.–** (ponto subtracção) para subtrair uma expressão de cada elemento.

20–[1,2;3,4] ENTER → $\begin{bmatrix} 19 & -2 \\ -3 & 16 \end{bmatrix}$

∗ (multiplicar) **Tecla** ⊠

expressão1 ∗ expressão2 ⇒ *expressão*

Devolve a soma de *expressão 1* e *expressão 2*.

2∗3.45 ENTER → 6.9

x∗y∗x ENTER → $x^2 \cdot y$

∗ lista 2 ⇒ *lista*

Devolve uma lista com as somas dos elementos correspondentes em *lista 1* e *lista 2*.

As dimensões das listas têm de ser iguais.

{1.0,2,3}∗{4,5,6} ENTER → {4. 10 18}

{2/a,3/2}∗{a²,b/3} ENTER → $\{2 \cdot a \ \ \frac{b}{2}\}$

matriz1 ∗ matriz2 ⇒ *matriz*

Devolve o produto da matriz de *matriz1* e *matriz2*.

O número de linhas em *matriz 1* tem de ser igual ao número de colunas em *matriz 2*.

[1,2,3;4,5,6]∗[a,d;b,e;c,f] ENTER

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix} \cdot \begin{bmatrix} a & d \\ b & e \\ c & f \end{bmatrix}$$

$$\begin{bmatrix} a+2\cdot b+3\cdot c & d+2\cdot e+3 \\ 4\cdot a+5\cdot b+6\cdot c & 4\cdot d+5\cdot e \end{bmatrix}$$

expressão ∗ ⇒ *lista*
∗ expressão ⇒ *lista*

Devolve uma lista com as somas de expressão e de cada elemento em *lista 1*.

π∗{4,5,6} ENTER → {4·π 5·π 6·π}

expressão ∗ matriz1 ⇒ *matriz*
matriz1 ∗ expressão ⇒ *matriz*

Devolve uma matriz com os produtos de *expressão* e de cada elemento em *matriz1*.

Nota: Utilizee **.∗** (ponto multiplicação) para multiplicar uma expressão por cada elemento.

[1,2;3,4]∗.01 ENTER → $\begin{bmatrix} .01 & .02 \\ .03 & .04 \end{bmatrix}$

λ∗identity(3) ENTER → $\begin{bmatrix} \lambda & 0 & 0 \\ 0 & \lambda & 0 \\ 0 & 0 & \lambda \end{bmatrix}$

/ (dividir) **Tecla ÷**

expressão1 / expressão2 ⇒ *expressão*

Devolve o quociente de *expressão 1* dividido por *expressão 2*.

```
2/3.45 ENTER                    .57971
x^3/x ENTER                     x²
```

/ lista 2 ⇒ *lista*

Devolve uma lista com os quocientes de lista 1 divididos por *lista 2*.

As dimensões das listas têm de ser iguais.

```
{1.0,2,3}/{4,5,6} ENTER
                    {.25 2/5 1/2}
```

expressão / ⇒ *lista*
/ expressão ⇒ *lista*

Devolve uma lista com os quocientes de *expressão* dividida por *lista1* ou *lista1* dividida por *expressão*.

```
a/{3,a,√(a)} ENTER
```

$$\left\{\frac{a}{3}\ 1\ \sqrt{a}\right\}$$

```
{a,b,c}/(a*b*c) ENTER
```

$$\left\{\frac{1}{b\cdot c}\ \frac{1}{a\cdot c}\ \frac{1}{a\cdot b}\right\}$$

matriz1 / expressão ⇒ *matriz*

Devolve uma matriz com os quocientes de *matriz 1/ expressão*.

Nota: Utilize **./** (ponto divisão) para dividir uma expressão por cada elemento.

```
[a,b,c]/(a*b*c) ENTER
```

$$\left[\frac{1}{b\cdot c}\ \frac{1}{a\cdot c}\ \frac{1}{a\cdot b}\right]$$

^ (potência) **Tecla ^**

expressão 1 ^ expressão 2 ⇒ *expressão*
lista 1 ^ lista 2 ⇒ *lista*

Devolve o primeiro argumento elevado à potência do segundo argumento.

Para uma lista, devolve os elementos da *lista1* elevados à potência dos elementos correspondentes na *lista2*.

No domínio real, as potências fraccionárias que tenham expoentes reduzidos com denominadores ímpares, utilizam a derivação real versus a derivação principal para o modo complexo.

```
4^2 ENTER                       16
{a,2,c}^{1,b,3} ENTER           {a 2^b c³ }
```

expressão ^ lista 1 ⇒ *lista*

Devolve a *expressão* elevada à potência dos elementos na *lista 1*.

```
p^{a,2,-3} ENTER
```

$$\left\{p^a\ \ p^2\ \ \frac{1}{p^3}\right\}$$

lista 1 ^ expressão ⇒ *lista*

Devolve os elementos da *lista1* elevados à potência da *expressão*.

```
{1,2,3,4}^-2 ENTER
                    {1  1/4  1/9  1/16}
```

Matriz quadrada 1 ^ número inteiro ⇒ *matriz*

Devolve a *Matriz quadrada 1* elevada à potência do *número inteiro*.

A *Matriz quadrada 1* tem de ser uma matriz quadrada.

Se o *número inteiro* = ⁻1, calcula a matriz inversa. Se o *número inteiro* < ⁻1, calcula a matriz inversa para uma potência positiva adequada.

```
[1,2;3,4]^2 ENTER
[1,2;3,4]^-1 ENTER
[1,2;3,4]^-2 ENTER
```

$$\blacksquare\begin{bmatrix}1 & 2\\3 & 4\end{bmatrix}^2 \qquad \begin{bmatrix}7 & 10\\15 & 22\end{bmatrix}$$

$$\blacksquare\begin{bmatrix}1 & 2\\3 & 4\end{bmatrix}^{-1} \qquad \begin{bmatrix}-2 & 1\\3/2 & -1/2\end{bmatrix}$$

$$\blacksquare\begin{bmatrix}1 & 2\\3 & 4\end{bmatrix}^{-2} \qquad \begin{bmatrix}11/2 & -5/2\\-15/4 & 7/4\end{bmatrix}$$

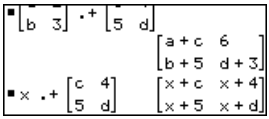
.+ (ponto adição) **Teclas** [.] [+]

matriz 1 **.+** *matriz 2* ⇒ *matriz*
expressão **.+** *matriz 1* ⇒ *matriz*

matriz 1 **.+** *matriz 2* devolve uma matriz que é a soma de cada par de elementos correspondentes da *matriz 1* e *matriz 2*.

expressão **.+** *matriz 1* devolve uma matriz que é a soma de *expressão* e cada elemento da *matriz 1*.

[a,2;b,3].+[c,4;5,d] [ENTER]
x.+[c,4;5,d] [ENTER]



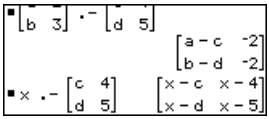
.− (ponto subtracção) **Teclas** [.] [−]

matriz 1 **.−** *matriz 2* ⇒ *matriz*
expressão **.−** *matriz 1* ⇒ *matriz*

matriz 1 **.−** *matriz 2* devolve uma matriz que é a diferença entre cada par de elementos correspondentes da *matriz 1* e *matriz 2*.

expressão **.−** *matriz 1* devolve uma matriz que é a diferença da *expressão* e cada elemento da *matriz 1*.

[a,2;b,3].−[c,4;d,5] [ENTER]
x.−[c,4;d,5] [ENTER]



.∗ (ponto multiplicação) **Teclas** [.] [×]

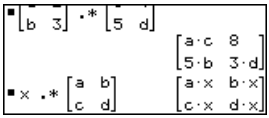
matriz 1 **.∗** *matriz 2* ⇒ *matriz*
expressão **.∗** *matriz 1* ⇒ *matriz*

matriz 1 **.∗** *matriz 2* devolve uma matriz que é o produto de cada par dos elementos correspondentes da *matriz 1* e *matriz 2*.

expressão **.∗** *matriz 1* devolve uma matriz com os produtos da *expressão* e cada elemento da *matriz 1*.

[a,2;b,3].∗[c,4;5,d] [ENTER]

x.∗[a,b;c,d] [ENTER]



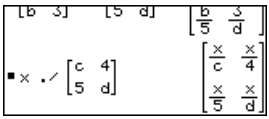
./ (ponto divisão) **Teclas** [.] [÷]

matriz 1 **./** *matriz 2* ⇒ *matriz*
expressão **./** *matriz 1* ⇒ *matriz*

matriz1 **./** *matriz 2* devolve uma matriz que é o quociente de cada par de elementos correspondentes da *matriz 1* e *matriz 2*.

expressão **./** *matriz 1* devolve uma matriz que é o quociente da *expressão* e cada elemento da *matriz 1*.

[a,2;b,3]./[c,4;5,d] [ENTER]
x./[c,4;5,d] [ENTER]



.^ (ponto potência) **Teclas** [.] [^]

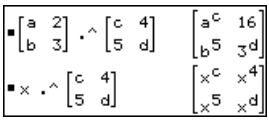
matriz 1 **.^** *matriz 2* ⇒ *matriz*
expressão **.^** *matriz 1* ⇒ *matriz*

matriz 1 **.^** *matriz 2* devolve uma matriz em que cada elemento da *matriz 2* é o expoente do elemento correspondente da *matriz 1*.

expressão **.^** *matriz 1* devolve uma matriz em que cada elemento da *matriz 1* é o expoente para a *expressão*.

[a,2;b,3].^[c,4;5,d] [ENTER]

x.^[c,4;5,d] [ENTER]



⁻ (negar) — Tecla [(-)] e menu MATH/Base

⁻*expressão 1* ⇒ *expressão*
⁻*lista 1* ⇒ *lista*
⁻*matriz 1* ⇒ *matriz*

Devolve a negação do argumento.

Para uma lista ou matriz, devolve todos os elementos negados.

Se a *expressão 1* for um número inteiro binário ou hexadecimal, a negação fornece o complemento de dois.

```
⁻2.43 ENTER                                  ⁻2.43
⁻{⁻1,0.4,1.2E19} ENTER
                                  {1 ⁻.4 ⁻1.2E19}
⁻a*⁻b ENTER                                   a•b
```

No modo base Hex:

└ **Importante:** Zero, não a letra O.

```
0b100101▸dec ENTER                             37

⁻0b100101 ENTER
0b11111111111111111111111111011011
ans(1)▸dec ENTER                              ⁻37
```

Nota: Para digitar ▸, prima [2nd] [▸].

% (percentagem) — Menu CHAR/Punctuation

expressão 1 **%** ⇒ *expressão*
lista 1 **%** ⇒ *lista*
matriz 1 **%** ⇒ *matriz*

Devolve $\frac{argument}{100}$.

Para uma lista ou matriz, devolve uma lista ou matriz com cada elemento divido por 100.

```
13% ◆ ENTER                                   .13
{1, 10, 100}% ◆ ENTER
                                   {.01 .1  1.}
```

= (igual) — Tecla [=]

expressão 1 = *expressão 2* ⇒ *Expressão booleana*
lista 1 = *lista 2* ⇒ *Lista booleana*
matriz 1 = *matriz 2* ⇒ *Matriz booleana*

Devolve verdadeiro se a *expressão 1* for determinada para ser igual à *expressão 2*.

Devolve falso se a *expressão 1* for determinada para não ser igual à *expressão 2*.

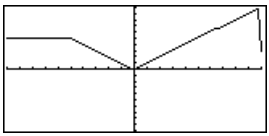
Qualquer outra coisa devolve uma forma simplificada da equação.

Para listas e matriz, devolve comparações elemento a elemento.

Exemplo de lista de funções com símbolos de teste matemáticos: =, ≠, <, ≤, >, ≥

```
:g(x)
:Func
:If x≤⁻5 Then
:  Return 5
:  ElseIf x>⁻5 and x<0 Then
:  Return ⁻x
:  ElseIf x≥0 and x≠10 Then
:  Return x
:  ElseIf x=10 Then
:  Return 3
:EndIf
:EndFunc

Graph g(x) ENTER
```



≠ Teclas ◆ [=]

expressão 1 ≠ *expressão 2* ⇒ *Expressão booleana*
lista 1 ≠ *lista 2* ⇒ *Lista booleana*
matriz 1 ≠ *matriz 2* ⇒ *Matriz booleana*

Devolve verdadeiro se a *expressão 1* for determinada para ser não igual à *expressão 2*.

Devolve falso se a *expressão 1* for determinada para ser igual à *expressão 2*.

Qualquer outra coisa devolve uma forma simplificada da equação.

Para listas e matriz, devolve comparações elemento a elemento.

Consulte o exemplo "=" (igual).

< Teclas [2nd] [<]

expressão 1 < *expressão 2* ⇒ *Expressão booleana*
lista 1 < *lista 2* ⇒ *Lista booleana*
matriz 1 < *matriz 2* ⇒ *Matriz booleana*

Devolve verdadeiro se a *expressão 1* for determinada para ser menor que a *expressão 2*.

Devolve falso se a *expressão 1* for determinada para ser maior ou igual à *expressão 2*.

Qualquer outra coisa devolve uma forma simplificada da equação.

Para listas e matriz, devolve comparações elemento a elemento.

Consulte o exemplo "=" (igual).

≤ Teclas ◆ [0]

expressão 1 ≤ *expressão 2* ⇒ *Expressão booleana*
lista 1 ≤ *lista 2* ⇒ *Lista booleana*
matriz 1 ≤ *matriz 2* ⇒ *Matriz booleana*

Devolve verdadeiro se a *expressão 1* for determinada para ser igual ou menor que a *expressão 2*.

Devolve falso se a *expressão 1* for determinada para ser maior que a *expressão 2*.

Qualquer outra coisa devolve uma forma simplificada da equação.

Para listas e matriz, devolve comparações elemento a elemento.

Consulte o exemplo "=" (igual).

> Teclas [2nd] [>]

expressão 1 > *expressão 2* ⇒ *Expressão booleana*
lista 1 > *lista 2* ⇒ *Lista booleana*
matriz 1 > *matriz 2* ⇒ *Matriz booleana matriz*

Devolve verdadeiro se a *expressão 1* for determinada para ser maior que a *expressão 2*.

Devolve falso se a *expressão 1* for determinada para ser igual ou menor que a *expressão 2*.

Qualquer outra coisa devolve uma forma simplificada da equação.

Para listas e matriz, devolve comparações elemento a elemento.

Consulte o exemplo "=" (igual).

≥ **Teclas** [◆] [.]

expressão 1 ≥ *expressão 2* ⇒ *Expressão booleana*
lista 1 ≥ *lista 2* ⇒ *Lista booleana*
matriz 1 ≥ *matriz 2* ⇒ *Matriz booleana*

Devolve verdadeiro se a *expressão1* for determinada para ser maior ou igual à *expressão2*.

Devolve falsa se a *expressão 1* for determinada para ser menor que a *expressão 2*.

Qualquer outra coisa devolve uma forma simplificada da equação.

Para listas e matriz, devolve comparações elemento a elemento.

Consulte o exemplo "=" (igual).

! (factorial) **Teclas** [◆] [÷]

expressão 1! ⇒ *expressão*
lista 1! ⇒ *lista*
matriz 1! ⇒ *matriz*

Devolve o factorial do argumento.

Para uma lista ou matriz, devolve uma lista ou matriz de factoriais dos elementos.

A TI-89 calcula um valor numérico apenas para valores de números inteiros não negativos.

5! [ENTER] 120

{5,4,3}! [ENTER] {120 24 6}

[1,2;3,4]! [ENTER] $\begin{bmatrix} 1 & 2 \\ 6 & 24 \end{bmatrix}$

& (acrescentar) **Teclas** [◆] [×]

cadeia1 & *cadeia2* ⇒ *cadeia*

Devolve uma cadeia de texto que é a *cadeia 2* acrescentada à *cadeia 1*.

"Hello " & "Nick" [ENTER] "Hello Nick"

∫() (integrar) **Teclas** [2nd] [∫]

∫(*expressão 1*, *var*[, *inferior*] [,*superior*]) ⇒ *expressão*
∫(*lista 1*,*var* [,*ordem*]) ⇒ *lista*
∫(*matriz 1*,*var* [,*ordem*]) ⇒ *matriz*

Devolve o integral da *expressão 1* no que respeita à variável *var* de *inferior* a *superior*.

Devolve uma primitiva se omitir *inferior* e *superior*. Uma constante simbólica da integração como C é omitida.

No entanto, *inferior* é adicionada como uma constante da integração se omitir apenas *superior*.

∫(x^2,x,a,b) [ENTER] $\frac{b^3}{3} - \frac{a^3}{3}$

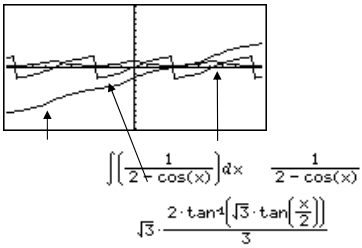
∫(x^2,x) [ENTER] $\frac{x^3}{3}$

∫(a*x^2,x,c) [ENTER] $\frac{a \cdot x^3}{3} + c$

As primitivas igualmente válidas podem diferir por uma constante numérica. Como uma constante pode estar disfarçada, em especial, quando uma primitiva contiver funções trigonométricas inversas ou logaritmos. Além disso, as expressões constantes por partes são por vezes adicionadas para tornar uma primitiva válida num intervalo maior do que a fórmula usual.

∫(1/(2−cos(x)),x)→tmp(x) [ENTER]

ClrGraph:Graph tmp(x):Graph 1/(2−cos(x)):Graph √(3) (2tan⁻¹(√(3)(tan(x/2)))/3) [ENTER]

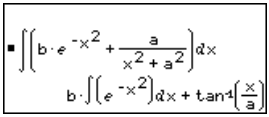


∫() devolve-se para partes da *expressão 1* que não podem ser determinadas como uma combinação finita explícita das funções e dos operadores integrados.

Quando *inferior* e *superior* estiverem presentes, é efectuada uma tentativa para localizar quaisquer descontinuidades ou derivadas descontínuas no intervalo *inferior* < *var* < *superior* e para subdividir o intervalo dessas casas.

Para a definição AUTO do modo Exact/Approx, a integração numérica é utilizada onde aplicável quando não conseguir determinar um limite ou uma primitiva.

∫(b∗*e*^(⁻x^2)+a/(x^2+a^2),x) [ENTER]



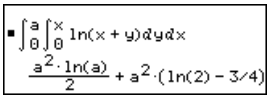
Para a definição APPROX, a integração numérica é tentada primeiro, se aplicável. As primitivas são procuradas apenas onde essa integração numérica não é aplicável ou falha.

∫(*e*^(⁻x^2),x,⁻1,1)◆ [ENTER] 1.493...

∫() pode ser aninhado para fazer integrais múltiplos. Os limites de integração podem depender das variáveis de integração fora delas.

Nota: Consulte também **nInt()**.

∫(∫(ln(x+y),y,0,x),x,0,a) [ENTER]



√() (raiz quadrada) **Teclas** [2nd] [√]

√ (*expressão 1*) ⇒ *expressão*
√ (*lista 1*) ⇒ *lista*

Devolve a raiz quadrada do argumento.

Para uma lista, devolve as raízes quadradas de todos os elementos da *lista 1.*

√(4) [ENTER] 2

√({9,a,4}) [ENTER] {3 √a 2}

Π() (produto) **Menu MATH/Calculus**

Π(*expressão 1*, *var*, *baixo*,*alto*) ⇒ *expressão*

Avalia a *expressão 1* para cada valor de *var* de *baixo* a *alto*, e devolve o produto dos resultados.

Π(1/n,n,1,5) [ENTER] $\frac{1}{120}$

Π(k^2,k,1,n) [ENTER] $(n!)^2$

Π({1/n,n,2},n,1,5) [ENTER]

$\left\{\frac{1}{120}\ 120\ 32\right\}$

Π(*expressão 1***,** *var***,** *baixo***,** *baixo*– 1**) ⇒** 1

Π(*expressão 1***,** *var***,** *baixo***,** *alto***) ⇒** 1*Λ*Π(*expressão 1, var, alto*+1, *baixo*– 1) *se alto* < *baixo*– 1

```
Π(k,k,4,3) ENTER                    1
Π(1/k,k,4,1) ENTER                  6
Π(1/k,k,4,1)*Π(1/k,k,2,4) ENTER
                                    1/4
```

Σ() (soma) **Menu MATH/Calculus**

Σ(*expressão 1***,** *var***,** *baixo***,** *alto***) ⇒** *expressão*

Avalia a *expressão 1* para cada valor de *var* de *baixo* a *alto*, e devolve a soma dos resultados.

```
Σ(1/n,n,1,5) ENTER                  137/60
Σ(k^2,k,1,n) ENTER
                                    n•(n+1)•(2•n+1)/6
Σ(1/n^2,n,1,∞) ENTER                π²/6
```

Σ(*expressão 1***,** *var***,** *baixo***,** *baixo*– 1) ⇒ 0

Σ(*expressão 1***,** *var***,** *baixo***,** *alto***) ⇒** ⁻**Σ(***expressão 1, var, alto*+1, *baixo*– 1) se *alto* < *baixo*– 1

```
Σ(k,k,4,3) ENTER                    0
Σ(k,k,4,1) ENTER                    -5
Σ(k,k,4,1)+Σ(k,k,2,4) ENTER         4
```

(acção indirecta) **CATALOG**

*varNameString*

Refere-se à variável cujo nome é *varNameString*. Permite criar e modificar variáveis a partir de um programa com cadeias.

Segmento do programa:

```
⋮
:Request "Enter Your Name",str1
:NewFold #str1
⋮
⋮
:For i,1,5,1
:  ClrGraph
:  Graph i*x
:  StoPic #("pic" & string(i))
:EndFor
⋮
```

G (gradiano) **Menu MATH/Angle**

*expressão 1*G **°** ⇒ *expressão*
*lista 1*G **°** ⇒ *lista*
*matriz 1*G **°** ⇒ *matriz*

Esta função proporciona uma forma para utilizar um ângulo em grados enquanto estiver nos modos Degree e Radian.

No modo de ângulo Radian, multiplica a *expressão* por π/200. No modo de ângulo Degree, multiplica a expressão 1 por g/100.
No modo de ângulo Gradian, devolve a expressão1 inalterada.

No modo Degree, Gradian ou Radian:

```
cos(50G) ENTER                      √2/2
cos({0,100G,200G}) ENTER            {1,0.-1}
```

ʳ (radiano) **Menu MATH/Angle**

*expressão 1*ʳ ⇒ *expressão*
*lista 1*ʳ ⇒ *lista*
*matriz 1*ʳ ⇒ *matriz*

No modo de ângulo Degree, multiplica a *expressão 1* por 180/π. No modo de ângulo Radian, devolve a *expressão 1* inalterada. No modo de ângulo Gradian, multiplica a *expressão 1* por 200/π.

Esta função proporciona uma forma para utilizar um ângulo em radianos enquanto estiver nos modos Degree e Gradian.

Sugestão: Utilize ʳ se quiser forçar radianos numa definição ou definição de programa, independentemente do modo que prevaleça quando a função ou o programa for utilizado.

No modo de ângulo Degree, Gradian ou Radian:

cos((π/4)ʳ) ENTER

$$\frac{\sqrt{2}}{2}$$

cos({0ʳ,(π/12)ʳ,⁻πʳ}) ENTER

$$\{1\ \frac{(\sqrt{3}+1)\cdot\sqrt{2}}{4}\ -1\}$$

° (grau) **Teclas** 2nd [°]

expressão 1 ⇒ *expressão*
lista 1° ⇒ *lista*
matriz 1° ⇒ *matriz*

No modo de ângulo Radian, multiplica a *expressão* por π/180. No modo de ângulo Degree, devolve a *expressão* inalterada. No modo de ângulo Gradian, multiplica a *expressão 1* por 10/9.

Esta função proporciona uma forma de utilizar o ângulo em graus enquanto estiver nos modos Gradian e Radian.

No modo de ângulo Degree, Gradian ou Radian:

cos(45°) ENTER

rectangular
$$\frac{\sqrt{2}}{2}$$

cos({0,π/4,90°,30.12°}) ◆ ENTER

cilíndrico
{1 .707… 0 .864…}
esférico

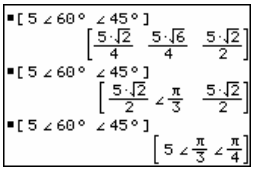
∠ (ângulo) **Teclas** 2nd [∠]

[*raio*,∠θ_*ângulo*] ⇒ *vector* (entrada polar)
[*raio*,∠θ_*ângulo*,*Z_coordenada*] ⇒ *vector* (entrada cilíndrica)
[*raio*,∠θ_*ângulo*,∠ϕ_*ângulo*] ⇒ *vector* (entrada esférica)

Devolve as coordenadas como um vector dependendo da definição do modo Vector Format: rectangular, cilíndrico ou esférico.

[5,∠60°,∠45°] ENTER

No modo Radian e o formato do vector definidos para:



(*magnitude* ∠ *ângulo*) ⇒ *Valor complexo* (entrada polar)

Introduz um valor complexo na forma polar (r∠θ). O *ângulo* é interpretado de acordo com a definição do modo Angle actual.

No modo de ângulo Radian e no modo do formato complexo Rectangular:

5+3*i*-(10∠π/4) ENTER

$$5-5\cdot\sqrt{2}+(3-5\cdot\sqrt{2})\cdot i$$

◆ ENTER

$$-2.071\ldots-4.071\ldots\cdot i$$

°, ', " **Teclas** 2nd [°] (°), **teclas** 2nd [′] ('), **teclas** 2nd ["] (")

dd° *mm*' *ss.ss*" ⇒ *expressão*

DD Um número positivo ou negativo
mm Um número não negativo
ss.ss Um número não negativo

Devolve *dd*+(*mm*/60)+(*ss.ss*/3600).

Este formato de entrada de base 60 permite:

- Introduzir um ângulo em graus/minutos/segundos, independentemente do modo de ângulo actual.
- Introduza a hora como horas/minutos/segundos.

No modo de ângulo Degree:

25°13'17.5" ENTER 25.221...

25°30' ENTER 51/2

' (primo) **Teclas** 2nd [′]

variável'
variável''

Introduz um símbolo de primo numa equação diferencial. Um símbolo de primo indica uma equação diferencial de 1ª ordem, dois símbolos de primo indicam uma de 2ª ordem, etc.

deSolve(y''=y^(-1/2) and y(0)=0 and y'(0)=0,t,y) ENTER

$$\frac{2\cdot y^{3/4}}{3}=t$$

_ (carácter sublinhado) **Teclas** ◆ [_]

unidade_expressão

Indica as unidades para uma *expressão*. Todos os nomes das unidades têm de começar com um carácter sublinhado.

Pode utilizar as unidades predefinidas ou criar as suas próprias unidades. Para uma lista de unidades predefinidas, consulte o módulo sobre constantes e unidades de medição. Pode premir 2nd [UNITS] para seleccionar unidades a partir de um menu, ou pode digitar os nomes das unidades directamente.

3_m▸_ft ENTER 9.842...·_ft

Nota: Para digitar ▸, prima 2nd [▸].

---

*variável_*

Quando a *variável* não tiver valor, é tratada como se representasse um número complexo. Por predefinição, sem o _, a variável é tratada como real.

Se a *variável* tiver um valor, o _ é ignorado e a *variável* retém o tipo de dados original.

Nota: Pode guardar um número complexo numa variável sem utilizar o _. No entanto, para obter melhores resultados nos cálculos como **cSolve()** e **cZeros()**, o _ é recomendado.

Partindo do princípio que z é indefinido:

real(z) ENTER z
real(z_) ENTER real(z_)

imag(z) ENTER 0
imag(z_) ENTER imag(z_)

▶ (converter) **Teclas** 2nd [▶]

unidade_expressão 1 ▶ *_unidade 2* ⇒ *unidade_expressão 2*

3_m▶_ft ENTER → 9.842…•_ft

Converte uma expressão de uma unidade para outra. As unidades têm de ser da mesma categoria.

O carácter sublinhado _ indica as unidades. Para uma lista de unidades predefinidas válidas, consulte o módulo sobre constantes e unidades de medição. Pode premir 2nd [UNITS] para seleccionar unidades a partir de um menu, ou pode digitar os nomes das unidades directamente.

Para obter o carácter sublinhado _ quando digitar as unidades directamente, prima ◆ [_]

Nota: O operador de conversão ▶ não processa unidades de temperatura. Utilize **tmpCnv()** e **ΔtmpCnv()**.

10^() **CATALOG**

10^ (*expressão 1***)** ⇒ *expressão*
10^ (*lista 1***)** ⇒ *lista*

10^(1.5) ENTER → 31.622…

10^{0,-2,2,a} ENTER → $\{1\ \frac{1}{100}\ 100\ 10^a\}$

Devolve 10 elevado à potência do argumento.

Para uma lista, devolve 10 elevado à potência dos elementos na *lista 1*.

10^(*Matriz quadrada 1***)** ⇒ *Matriz quadrada*

10^([1,5,3;4,2,1;6,-2,1]) ENTER

$$\begin{bmatrix} 1.143\ldots E7 & 8.171\ldots E6 & 6.675\ldots E6 \\ 9.956\ldots E6 & 7.115\ldots E6 & 5.813\ldots E6 \\ 7.652\ldots E6 & 5.469\ldots E6 & 4.468\ldots E6 \end{bmatrix}$$

Devolve 10 elevado à potência de *Matriz quadrada 1*. Este procedimento não é a mesma coisa que calcular 10 elevado à potência de cada elemento. Para mais informações sobre o método de cálculo, consulte **cos()**.

A *Matriz quadrada 1* tem de estar na diagonal. O resultado contém sempre números de pontos flutuantes.

x^{-1} **CATALOG** (^-1)

expressão 1 x^{-1} ⇒ *expressão*
lista 1 x^{-1} ⇒ *lista*

3.1^-1 ENTER → .322581

{a,4,-.1,x-2}^-1 ENTER → $\{\frac{1}{a}\ \frac{1}{4}\ -10.\ \frac{1}{x-2}\}$

Devolve o recíproco do argumento.

Para uma lista, devolve os recíprocos dos elementos da *lista 1*.

Matriz quadrada 1 x^{-1} ⇒ *Matriz quadrada*

[1,2;3,4]^-1 ENTER
[1,2;a,4]^-1 ENTER

$$\blacksquare \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}^{-1} \qquad \begin{bmatrix} -2 & 1 \\ 3/2 & -1/2 \end{bmatrix}$$

$$\blacksquare \begin{bmatrix} 1 & 2 \\ a & 4 \end{bmatrix}^{-1}$$

$$\begin{bmatrix} \frac{-2}{a-2} & \frac{1}{a-2} \\ \frac{a}{2\cdot(a-2)} & \frac{-1}{2\cdot(a-2)} \end{bmatrix}$$

Devolve o inverso de *Matriz quadrada 1*.

A *Matriz quadrada 1* tem de ser uma matriz quadrada não singular.

| ("with") **Tecla** [I]

expressão | *Expressão booleana1* [e *Booleana expressão 2*]...[*e Expressão booleanaN*]

O símbolo "with" (|) serve como um operador binário. O operando à esquerda de | é uma expressão. O operando à direita de | especifica uma ou mais relações que se destinam a afectar a simplificação da expressão. As várias relações depois de | têm de ser juntas com um "and" lógico.

```
x+1| x=3 ENTER                    4
x+y| x=sin(y) ENTER          sin(y) + y
x+y| sin(y)=x ENTER               x + y
```

O operador "with" fornece três tipos de funcionalidades básicas: substituições, limitações de intervalo e exclusão.

As substituições são efectuadas na forma de uma igualdade, como x=3 ou y=sin(x). Para ser mais eficaz, o lado esquerdo deve ser uma variável simples. *expressão* | *variável* = *valor* substituirá *valor* para todas as ocorrências de *variável* na *expressão*.

```
x^3- 2x+7→ f(x) ENTER             Done
f(x)| x=√(3) ENTER                √3 + 7
(sin(x))^2+2sin(x)- 6| sin(x)=d ENTER
                                  d² +2d- 6
```

As limitações de intervalo têm a forma de uma ou mais desigualdades juntas por operadores "and" lógicos. As limitações de intervalo permitem também a simplificação que caso contrário pode ser inválida ou não calculável.

```
solve(x^2- 1=0,x)|x>0 and x<2 ENTER
                                  x = 1
√(x)∗√(1/x)|x>0 ENTER              1
√(x)∗√(1/x) ENTER                  √(1/x) · √x
```

As exclusões utilizam o operador relacional "not equals" (/= ou ≠) para excluir um valor específico de consideração. São utilizados principalmente para excluir uma solução exacta quando utilizar **cSolve()**, **cZeros()**, **fMax()**, **fMin()**, **solve()**, **zeros()**, etc.

```
solve(x^2- 1=0,x)| x≠1 ENTER       x = - 1
```

→ (guardar) **Tecla** [STO▸]

expressão → *var*
lista → *var*
matriz → *var*
expressão → *fun_name(parâmetro 1,...)*
lista → *fun_name(parâmetro 1,...)*
matriz → *fun_name(parâmetro 1,...)*

```
π/4→myvar ENTER                   π/4
2cos(x)→Y1(x) ENTER               Done
{1,2,3,4}→Lst5 ENTER              {1 2 3 4}
[1,2,3;4,5,6]→MatG ENTER          [1 2 3; 4 5 6]
"Hello"→str1 ENTER                "Hello"
```

Se a variável *var* não existir, crie a *var* e inicialize-a para *expressão*, *lista* ou *matriz*.

Se a *var* já existir e não estiver bloqueada ou protegida, substitua o conteúdo por *expressão*, *lista* ou *matriz*.

Sugestão: Se quiser fazer cálculos simbólicos com variáveis indefinidas, evite guardar qualquer coisa nas variáveis de uma letra mais utilizadas, como a, b, c, x, y, z, etc.

◉ (comentário) **Menu Program Editor/Control ou teclas** [◆] [)]

◉ [*texto*]

◉ processa *texto* como uma linha de comentário, que pode ser utilizada para anotar instruções do programa.

◉ pode estar no início ou em qualquer parte da linha. Tudo à direita de ◉, no fim da linha, é o comentário.

Segmento do programa:

```
                                  ⋮
:◉ Get 10 points from the Graph
                                screen
:For i,1,10 ◉ This loops 10 times
                                  ⋮
```

0b, 0h

Teclas [0] [alpha] [B]

Teclas [0] [alpha] [H]

0b *Número binário*
0h *Número hexadecimal*

Indica um número binário ou hexadecimal, respectivamente. Para introduzir um número binário ou hex, tem de introduzir o prefixo 0b ou 0h, independentemente do modo Base. Sem um prefixo, o número é tratado como decimal (base 10).

Os resultados aparecem de acordo com o modo Base.

No modo base Dec:

`0b10+0hF+10` [ENTER] 27

No modo base Bin:

`0b10+0hF+10` [ENTER] `0b11011`

No modo base Hex:

`0b10+0hF+10` [ENTER] `0h1B`

Anexo B: Referência técnica

Esta secção contém uma lista completa dos códigos de caracteres e das mensagens de erro da TI-89 Titanium / Voyage™ 200 . Inclui também informações sobre como são calculadas determinadas operações da TI-89 Titanium / Voyage™ 200.

Mensagens de erro da TI-89 Titanium / Voyage™ 200

Esta secção lista as mensagens de erro que podem ser visualizadas quando ocorrerem erros internos ou de introdução. O número à esquerda de cada mensagem de erro representa um número de erro interno que não é visualizado. Se o erro ocorrer dentro de um bloco Try...EndTry, o número do erro é guardado na variável do sistema *errornum*. Muitas das mensagens de erro são explicações e não requerem informações descritivas. No entanto, foram adicionadas informações adicionais para algumas mensagens de erro.

Número de erro	Descrição
10	**A function did not return a value (Uma função não devolve um valor)**
20	**A test did not resolve to TRUE or FALSE (Um teste não resolveu para VERDADEIRO ou FALSO)** Geralmente, não é possível comparar as variáveis indefinidas. Por exemplo, o teste If a<b provocará um erro se a ou b não estiver definido quando executar a instrução If.
30	**Argument cannot be a folder name (O argumento não pode ser um nome de pasta)**
40	**Argument error (Erro de argumento)**
50	**Argument mistmatch (Argumento não coincidente)** Dois ou mais argumentos têm de ser do mesmo tipo. Por exemplo, **PtOn** *expression1,expression2* e **PtOn** *list1,list2* são ambas válidas, mas **PtOn** *expression,list* é uma má combinação.
60	**Argument must be a Boolean expression or integer (O argumento tem de ser um número inteiro ou uma expressão Booleana)**
70	**Argument must be a decimal number (O argumento tem de ser um número decimal)**
80	**Argument must be a label name (O argumento tem de ser um nome de etiqueta)**
90	**Argument must be a list (O argumento tem de ser uma lista)**
100	**Argument must be a matrix (O argumento tem de ser uma matriz)**
110	**Argument must be a Pic (O argumento tem de ser uma imagem)**
120	**Argument must be a Pic or string (O argumento tem de ser uma imagem ou cadeia)**
130	**Argument must be a string (O argumento tem de ser uma cadeia)**

Número de erro	Descrição
140	**Argument must be a variable name (O argumento tem de ser o nome de uma variável)** Por exemplo, DelVar 12 não é válido porque um número não pode ser o nome de uma variável.
150	**Argument must be an empty folder name (O argumento tem de ser o nome de uma pasta vazia)**
160	**Argument must be an expression (O argumento tem de ser uma expressão)** Por exemplo, zeros(2x+3=0,x) não é válido porque o primeiro argumento é uma equação.
161	**ASAP or Exec string too long (Cadeia ASAP ou Exec muito longa)**
163	**Attribute (8-digit number) of object (8-digit number) not found (Atributo (número de 8 dígitos) do objecto (número de 8 dígitos) não localizado)**
165	**Batteries too low for sending or receiving (Pilhas muito fracas para envio ou recepção)** Instale pilhas novas antes de enviar ou receber.
170	**Bound (Limite)** Para funções matemáticas gráficas interactivas como 2:Zero, o limite inferior tem de ser inferior ao limite superior para definir o intervalo de procura.
180	**Break (Interrupção)** A tecla [+] foi premida durante um cálculo longo ou a execução do programa.
185	**Checksum (Erro da soma de verificação)**
190	**Circular definition (Definição circular)** Esta mensagem aparece para evitar o esgotamento da memória durante a substituição infinita de valores das variáveis durante a simplificação. Por exemplo, a+1→a, em que a é uma variável indefinida, provocará este erro.
200	**Constraint expression invalid (Expressão de restrição inválida)** Por exemplo, solve(3x^2[°] 4=0, x) | x<0 ou x>5 produziria esta mensagem de erro porque a restrição é separada por “or” e não por “and.”
205	**Data is too big to save to a variable. (Os dados são muito grandes para guardar uma variável.) Utilize F6 Util para reduzir o tamanho.** O tamanho dos dados do editor excede o tamanho máximo que pode ser guardado numa variável. O menu F6 Util fornece as operações que podem ser utilizadas na redução do tamanho dos dados.
210	**Data type (Tipo de dados)** Um argumento é do tipo de dados incorrecto.
220	**Dependent limit (Limite dependente)** Um limite de integração depende da variável de integração. Por exemplo,](x^2,x,1,x) não é permitido.
225	**Diff Eq setup (Configurar equação diferencial)**
230	**Dimension (Dimensão)** Um índice de lista ou matriz não é válido. Por exemplo, se a lista {1,2,3,4} estiver guardada em L1, L1[5] é um erro de dimensão porque L1 contém apenas quatro elementos.

Número de erro	Descrição
240	**Dimension mismatch (Dimensão não coincidente)** Dois ou mais argumentos têm de ter a mesma dimensão. Por exemplo, [1,2]+[1,2,3] é uma dimensão não coincidente porque as matrizes contêm um diferente número de elementos .
250	**Divide by zero (Dividir por zero)**
260	**Domain error (Erro do domínio)** Um argumento tem de ter um domínio especificado. Por exemplo, ans(100) não é válido porque o argumento para **ans()** tem de estar no intervalo 1–99.
270	**Duplicate variable name (Nome de variável duplicado)**
280	**Else and ElseIf invalid outside of If..EndIf block (Else e ElseIf inválidos fora do bloco If..EndIf)**
290	**EndTry is missing the matching Else statement (EndTry não existe na instrução Else coincidente)**
295	**Excessive iteration (Repetição excessiva)**
300	**Expected 2 or 3-element list or matrix (Matriz ou lista de 2 ou 3 elementos prevista)**
307	**Flash application extension (function or program) not found (Extensão da aplicação Flash (função ou programa) não localizada)**
308	**Flash application not found (Aplicação Flash não localizada)**
310	**First argument of nSolve must be a univariate equation (O primeiro argumento de nSolve tem de ser uma equação de uma variável)** O primeiro argumento tem de ser uma equação, e a equação não pode conter uma variável sem valor diferente da variável de interesse. Por exemplo, nSolve(3x^2[◦] 4=0, x) é uma equação válida; no entanto, nSolve(3x^2[◦] 4, x) não é uma equação, e nSolve(3x^2[◦] y=0,x) não é uma equação de uma variável porque y não tem um valor neste exemplo.
320	**First argument of solve or cSolve must be an equation or inequality (O primeiro argumento de solve ou cSolve tem de ser uma equação ou desigualdade)** Por exemplo, solve(3x^2[◦] 4, x) não é válido porque o primeiro argumento não é uma equação.
330	**Folder (Pasta)** Foi efectuada uma tentativa no menu VAR-LINK para guardar uma variável numa pasta que não existe.
335	**Funções gráficas y1(x)...y99(x) não disponíveis no modo Diff Equations (Equações diferenciais)**
345	**Inconsistent units (Unidades inconsistentes)**
350	**Index out of range (Fora do índice)**
360	**Indirection string is not a valid variable name (A cadeia de acção indirecta não é um nome de variável válido)**
380	**Invalid ans() (ans() inválida)**
390	**Invalid assignment (Atribuição inválida)**
400	**Invalid assignment value (Valor de atribuição inválido)**

Número de erro	Descrição
405	**Invalid axes (Eixos inválidos)**
410	**Invalid command (Comando inválido)**
420	**Invalid folder name (Nome de pasta inválido)**
430	**Invalid for the current mode settings (Inválido para as definições do modo actual)**
440	**Invalid implied multiply (Multiplicação implícita inválida)** Por exemplo, x(x+1) não é válido; em que, x (x+1) é a sintaxe correcta. Destina-se a evitar a confusão entre a multiplicação implícita e as chamadas de funções.
450	**Invalid in a function or current expression (Inválida numa função ou expressão actual)** Apenas determinados comandos são válidos numa função definida pelo utilizador. As entradas efectuadas no Editor de janelas, Editor de tabelas, Editor de dados/Matriz e Solucionador de problemas, e nas linhas de comando do sistema, como Lower Bound, não podem conter quaisquer comandos ou dois pontos (:). Consulte também “Criar e avaliar funções definidas pelo utilizador” no módulo *Ecrã inicial da calculadora*.
460	**Invalid in Custom..EndCustm block (Inválido no bloco Custom..EndCustm)**
470	**Invalid in Dialog..EndDlog block (Inválido no bloco Dialog..EndDlog)**
480	**Invalid in Toolbar..EndTBar block (Inválido no bloco Toolbar..EndTBar)**
490	**Invalid in Try..EndTry block (Inválido no bloco Try..EndTry)**
500	**Invalid label (Etiqueta inválida)** Os nomes das etiquetas têm de seguir as mesmas regras utilizadas para dar nome às variáveis.
510	**Invalid list or matrix (Matriz ou lista inválida)** Por exemplo, uma lista no interior de uma lista, como {2,{3,4}} não é válida.
520	**Invalid outside Custom..EndCustm or ToolBar..EndTbar blocks (Inválido fora dos blocos Custom..EndCustm or ToolBar..EndTbar)** Por exemplo, tentou utilizar um comando **Item** fora de uma estrutura **Custom** ou **ToolBar**.
530	**Invalid outside Dialog..EndDlog, Custom..EndCustm, or ToolBar..EndTBar blocks (Inválido fora dos blocos Dialog..EndDlog, Custom..EndCustm ou ToolBar..EndTBar)** Por exemplo, tentou utilizar um comando Title fora de uma estrutura **Dialog**, **Custom** ou **ToolBar**.
540	**Invalid outside Dialog..EndDlog block (Inválido fora do bloco Dialog..EndDlog)** Por exemplo, tentou utilizar o comando **DropDown** fora de uma estrutura **Dialog**.
550	**Invalid outside function or program (Inválido fora da função ou programa)** Um número de comandos não são válidos fora de um programa ou de uma função. Por exemplo, Local não pode ser utilizado excepto se for num programa ou função.

Número de erro	Descrição
560	**Invalid outside Loop..EndLoop, For..EndFor, or While..EndWhile blocks (Inválido fora dos blocos Loop..EndLoop, For..EndFor ou While..EndWhile)** Por exemplo, o comando **Exit** só é válido dentro destes blocos de espirais.
570	**Invalid pathname (Nome de caminho inválido)** Por exemplo, \\var não é válido.
575	**Invalid polar complex (Complexo polar inválido)**
580	**Invalid program reference (Referência de programa inválida)** Os programas não podem ser referenciados nas funções ou expressões, como 1+p(x) em que p é um programa.
585	**Invalid relocation data in ASM program (Dados de deslocação inválidos no programa ASM)** Os dados de deslocação necessários no programa ASM (Assembly) não existem ou estão danificados.
590	**Invalid syntax block (Bloco de sintaxe inválido)** Um bloco **Dialog..EndDlog** está vazio ou tem mais de um título. Um bloco **Custom..EndCustm** não pode conter variáveis PIC e os itens têm de ser antecedidos por um título. Um bloco **Toolbar..EndTBar** tem de ter um segundo argumento se não existirem itens; ou os itens têm de ter um segundo argumento e têm de ser antecedidos por um título.
600	**Invalid table (Tabela inválida)**
605	**Invalid use of units (Utilização de unidades inválidas)**
610	**Invalid variable name in a Local statement (Nome da variável inválido numa instrução Local)**
620	**Invalid variable or function name (Nome da função ou variável inválido)**
630	**Invalid variable reference (Referência da variável inválida)**
640	**Invalid vector syntax (Sintaxe do vector inválida)**
650	**Link transmission (Transmissão de ligações)** Uma transmissão entre as duas unidades não foi concluída. Verifique se o cabo de ligação está ligado correctamente a ambas as unidades.
665	**Matrix not diagonalizable (Matriz sem diagonal)**
670 673	**Memory (Memória)** O cálculo requereu mais memória do que a disponível nessa altura. Se aparecer este erro quando executar um programa grande, pode ser necessário dividir o programa em funções ou programas mais pequenos e independentes (em que um programa ou função chama outro).
680	**Missing (Inexistente) (**
690	**Missing (Inexistente))**
700	**Missing (Inexistente) "**
710	**Missing (Inexistente)]**
720	**Missing (Inexistente) }**

Número de erro	Descrição
730	**Missing start or end of block syntax (Início ou fim inexistente da sintaxe do bloco)**
740	**Missing Then in the If..EndIf block (Bloco Then in the If..EndIf inexistente)**
750	**Name is not a function or program (Nome não é um programa ou função)**
765	**No functions selected (Nenhuma função seleccionada)**
780	**No solution found (Nenhuma solução localizada)** A utilização das funções matemáticas interactivas (F5:Math) na aplicação Graph pode originar este erro. Por exemplo, se tentar localizar um ponto de inflexão da parábola y1(x)=x , que não existe, este erro aparece.
790	**Non-algebraic variable in expression (Variável não algébrica em expressão)** Se a for o nome de uma PIC, GDB, MAC, FIG, etc., a+1 não é válida. Use a different variable name in the expression or delete the variable. (Utilize um nome de variável diferente ou elimine a variável.)
800	**Non-real result (Resultado não real)** Por exemplo, se a unidade estiver na definição REAL do modo Complex Format (Formato complexo), ln([≈] 2) não é válido.
810	**Not enough memory to save current variable (Memória insuficiente para guardar a variável actual). Elimine as variáveis desnecessárias no ecrã Var-Link e reabra o editor como actual ou reabra o editor e utilize F1 8 para apagar o editor.** Esta mensagem de erro é provocada pelas condições de pouca memória dentro do editor de dados/matriz.
830	**Overflow (Excesso)**
840	**Plot Setup (Configuração do desenho)**
850	**Program not found (Programa não localizado)** Uma referência de programa dentro de outro programa não pode ser localizada no caminho fornecido durante a execução.
855	**Rand type functions not allowed in graphing (Funções de tipo aleatório não permitidas no gráfico)**
860	**Recursion is limited to 255 calls deep (A recursão está limitada a 255 chamadas)**
870	**Reserved name or system variable (Variável do sistema ou nome reservado)**
875	**ROM-resident routine not available (Rotina residente da ROM indisponível)**
880	**Sequence setup (Configuração da sequência)**
885	**Signature error (Erro de assinatura)**
890	**Singular matrix (Matriz singular)**
895	**Slope fields need one selected function and are used for 1st-order equations only (Os campos do declive necessitam de uma função seleccionada e são utilizados para equações de 1ª ordem)**
900	**Stat (Estatística)**

Número de erro	Descrição
910	**Syntax (Sintaxe)** A estrutura da entrada está incorrecta. Por exemplo, x+[°] y (x mais menos y) não é válido; em que, x+[≈] y (x mais negativo y) está correcto.
930	**Too few arguments (Poucos argumentos)** A expressão ou equação falta num ou mais argumentos. Por exemplo, *d*(f(x)) não é válido; em que, *d*(f(x),x) é a sintaxe correcta.
940	**Too many arguments (Muitos argumentos)** A expressão ou equação contém um número excessivo de argumentos e não pode ser avaliada.
950	**Too many subscripts (Demasiados subscritos)**
955	**Too many undefined variables (Demasiadas variáveis indefinidas)**
960	**Undefined variable (Variável indefinida)**
965	**Unlicensed OS or Flash application (Aplicação Flash ou SO não licenciado)**
970	**Variable in use so references or changes are not allowed (Variável em utilização invalida alterações ou referências)**
980	**Variable is locked, protected, or archived (Variável bloqueada, protegida ou arquivada)**
990	**Variable name is limited to 8 characters (Nome da variável limitada a 8 caracteres)**
1000	**Window variables domain (Domínio de variáveis da janela)**
1010	**Zoom**
	Aviso: ^0 ou undef^0 substituído por 1
	Aviso: 0^0 substituído por 1
	Aviso: 1^ ou 1^undef substituído por 1
	Aviso: cSolve pode especificar mais zeros
	Aviso: Pode produzir uma equação falsa
	Aviso: Expressão a integrar real finita prevista
	Aviso: Pode não ser completamente simplificada
	Aviso: Podem existir mais soluções
	Aviso: Pode introduzir soluções falsas
	Aviso: A operação pode perder soluções
	Aviso: Requer e devolve um valor de 32 bits
	Aviso: Excesso substituído por ou [≈]
	Aviso: Precisão questionável
	Aviso: Solução questionável
	Aviso: Solve pode especificar mais zeros
	Aviso: Argumento Trig muito grande para reduzir
	Aviso: O domínio pode ser maior

Número de erro	Descrição
	Aviso: O domínio pode ser menor

Modos TI-89 Titanium / Voyage™ 200

Esta secção descreve os modos da TI-89 Titanium /Voyage™ 200 e lista as definições possíveis de cada modo. Estas definições do modo aparecem quando premir MODE.

Graph (Gráfico)

Especifica o tipo de gráficos que pode desenhar.

1:FUNCTION	Funções y(x)
2:PARAMETRIC	Equações paramétricas x(t) e y(t)
3:POLAR	Equações polares r(θ)
4:SEQUENCE	Sequências u(n)
5:3D	Equações 3D z(x,y) 3D
6:DIFF EQUATIONS	Equações diferenciais y'(t)

Nota: Se utilizar um ecrã dividido com Number of Graphs = 2, Graph destina-se à parte superior ou esquerda do ecrã e Graph 2 destina-se à parte inferior ou direita.

Current folder (Pasta actual)

Especifica a pasta actual. Pode configurar várias pastas com configurações únicas de variáveis, bases de dados de gráficos, programas, etc.

Nota: Para mais informações sobre como utilizar as pastas, consulte o ecrã inicial da calculadora.

1:main	Pasta predefinida incluída com a TI-89 Titanium / Voyage™ 200.
2: — (pastas personalizadas)	Estão disponíveis outras pastas apenas se tiverem sido criadas por um utilizador.

Display Digits (Visualizar dígitos)

Selecciona o número de dígitos. Estas definições decimais só afectam a forma de visualização dos resultados—pode introduzir um número em qualquer formato.

Internamente, a TI-89 Titanium / Voyage™ 200 retém números decimais com 14 dígitos significativos. Para efeitos de visualização, esses números são arredondados para um máximo de 12 dígitos significativos.

1:FIX 0 2:FIX 1 … D:FIX 12	Os resultados aparecem sempre com o número seleccionado de casas decimais.
E:FLOAT	O número de casas decimais depende do resultado.

F:FLOAT 1 G:FLOAT 2 … Θ:ΦΛΟΑΤ 12	Se a parte do número inteiro for maior do que o número de dígitos seleccionado, o resultado é arredondado e visualizado em notação científica. Por exemplo, em FLOAT 4: 12345. aparece como 1.235E4

Angle (Ângulo)

Especifica as unidades em que os valores dos ângulos são interpretados e visualizados em funções trigonométricas e conversões polares/rectangulares.

1:RADIAN
2:DEGREE
3:GRADIAN

Exponential Format (Formato exponencial)

Especifica o formato de notação que deve ser utilizado. Estes formatos só afectam a forma de visualização de uma resposta; pode introduzir um número em qualquer formato. As respostas numéricas podem ser visualizadas até 12 dígitos e um expoente de 3 dígitos.

1:NORMAL	Expressa os números em formato standard. Por exemplo, 12345.67
2:SCIENTIFIC	Expressa os números em duas partes: • Os dígitos significativos aparecem com um dígito à esquerda do decimal. • A potência de 10 aparece à direita de E. Por exemplo, 1.234567E4 significa 1.234567×10^4
3:ENGINEERING	Similar à notação científica. No entanto: • O número pode ter um, dois ou três dígitos antes do decimal. • O expoente da potência de 10 é um múltiplo de três. Por exemplo, 12.34567E3 significa 12.34567×10^3

Nota: Se seleccionar NORMAL, mas não for possível visualizar a resposta no número de dígitos seleccionados pelo Display Digits, a TI-89 Titanium / Voyage™ 200 mostra a resposta em notação científica. Se Display Digits = FLOAT, a notação científica é utilizada para expoentes de 12 ou mais expoentes de -4 ou menos.

Complex Format (Formato complexo)

Especifica a forma de visualização de resultados complexos e o formato.

1:REAL	Não mostra resultados complexos. (Se o resultado for um número complexo e a entrada não contiver a unidade complexa ***i***, aparece uma mensagem de erro.)
2:RECTANGULAR	Mostra os números complexos na formato: a+b***i***
3:POLAR	Mostra os números complexos na formato: $re^{i\theta}$

Vector Format (Formato do vector)

Determina a forma de visualização dos vectores de 2 e 3 elementos. Pode introduzir vectores em qualquer formato dos sistemas de coordenadas.

1:RECTANGULAR	As coordenadas são em termos de x, y e z. Por exemplo, [3,5,2] representa x = 3, y = 5 e z = 2.
2:CYLINDRICAL	As coordenadas são em termos de r, θ e z. Por exemplo, [3,∠45,2] representa r = 3, θ = 45 e z = 2.
3:SPHERICAL	As coordenadas são em termos de r, θ e ϕ. Por exemplo, [3, ∠45, ∠90] representa r = 3, θ = 45 e ϕ = 90.

Pretty Print

Determina a forma de visualização dos resultados no ecrã inicial.

1:OFF	Os resultados aparecem numa forma unidimensional linear. Por exemplo, π^2, π/2, ou ⁄((x-3)/x)
2:ON	Os resultados aparecem num formato matemático convencional. Por exemplo, π^2, 2 ou x3x

Nota: Para obter uma descrição completa destas definições, consulte “Formatos de resultados visualizados” no módulo *Utilizar a calculadora*.

Split Screen (Ecrã dividido)

Permite dividir o ecrã em duas partes. Por exemplo, pode visualizar um gráfico e ver o Editor Y= em simultâneo.

1:FULL	O ecrã não está dividido.
2:TOP-BOTTOM	As aplicações aparecem em dois ecrãs que ficam um por cima do outro.
3:LEFT-RIGHT	As aplicações aparecem em dois ecrãs que ficam à esquerda e à direita de cada um.

Para determinar como e que informações aparecem num ecrã dividido, utilize este modo juntamente com outros modos, como Split 1 App, Split 2 App, Number of Graphs e Split Screen Ratio. (Split Screen Ratio só está disponível na Voyage™ 200 .)

Split (Divisão) 1 App e Split (Divisão) 2 App

Especifica que aplicação aparece no ecrã.

- Para um ecrã completo, só Split (Divisão) 1 App está activa.
- Para um ecrã dividido, Split 1 App é a parte superior ou esquerda do ecrã e Split 2 App é a parte inferior ou direita.

As aplicações disponíveis aparecem quando premir B no ecrã do modo Page 2 ou quando premir [APPS]. Deve ter diferentes aplicações em cada ecrã excepto se tiver no modo de 2 gráficos.

Number of Graphs (Número de gráficos)

Especifica se ambas as partes de um ecrã dividido podem mostrar gráficos em simultâneo.

1	Apenas uma parte pode mostrar gráficos.
2	Ambas as partes podem mostrar um ecrã de gráfico independente (Definição Graph ou Graph 2) com definições independentes.

Especifica o tipo de gráficos que pode desenhar para o segundo gráfico no ecrã dividido de dois gráficos. Este está activo apenas quando Number of Graphs = 2. Nesta definição de dois gráficos, Graph define o tipo de gráfico para a parte superior ou esquerda do ecrã dividido, e Graph 2 define a parte direita ou inferior. As escolhas disponíveis são iguais a Graph.

Split Screen Ratio (Rácio do ecrã dividido) (Apenas na Voyage 200)

Especifica os tamanhos proporcionais das duas partes do ecrã dividido.

1:1	O ecrã está dividido uniformemente.
1:2	A parte inferior ou direita é cerca de duas vezes maior do que a parte superior ou esquerda.
2:1	A parte superior ou esquerda é cerca de duas vezes maior do que a parte inferior ou direita.

Exact/Approx (Exact/Aprox)

Especifica a forma de cálculo e visualização de expressões simbólicas e fraccionais. Ao reter formas racionais e simbólicas na definição EXACT, a TI-89 Titanium / Voyage™ 200 aumenta a precisão através da eliminação da maioria dos números de arredondamento numérico.

1:AUTO	Utiliza a definição EXACT na maioria dos casos. No entanto, utiliza APPROXIMATE se a entrada contiver um ponto decimal.
2:EXACT	Mostra os resultados de números não inteiros na forma simbólica ou racional.
3:APPROXIMATE	Mostra os resultados numéricos na forma de ponto flutuante.

Nota: Para obter uma descrição completa destas definições, consulte “Formatos de resultados visualizados” no módulo *Utilizar a calculadora*.

Base

Permite efectuar cálculos através da introdução de números em formato decimal, binário ou hexadecimal.

1:DEC	Os números decimais utilizam 0 - 9 no formato base 10
2:HEX	Os números hexadecimais utilizam 0 - 9 e A - F no formato base 16.

3:BIN	Os números binários utilizam 0 e 1 no formato base 2.

Unit System (Sistema de unidades)

Permite introduzir uma unidade para valores numa expressão, como 6_m * 4_m ou 23_m/_s * 10_s, converter valores de uma unidade para a outra dentro da mesma categoria, e criar unidades definidas pelo utilizador.

1:SI	Seleccione SI para o sistema métrico de medições
2:ENG/US	Seleccione ENG/US para o sistema não métrico de medições
3:CUSTOM	Permite seleccionar as predefinições personalizadas.

Custom Units (Unidades personalizadas)

Permite seleccionar as predefinições personalizadas. Este modo fica escurecido até seleccionar Unit System, 3:CUSTOM.

Language (Idioma)

Permite localizar a TI-89 Titanium / Voyage™ 200 num dos vários idiomas, dependendo do idioma em que as aplicações Flash são instaladas.

1:English (Inglês)	Idioma predefinido incluído com o sistema operativo (SO) da TI-89 Titanium / Voyage™ 200.
2: — (Aplicações Flash multilingues)	Os idiomas alternativos estão disponíveis se instalar as respectivas aplicações Flash multilingues.

Apps Desktop

Permite activar ou desactivar a visualização do ambiente de trabalho das Apps.

ON	Mostra o ambiente de trabalho das Apps navegável. O ambiente de trabalho das Apps aparece quando: • Premir [APPS]. • Ligar a unidade depois de a desligar, premindo [2nd] [1]. • Premir [2nd] [QUIT] a partir de uma App visualizada no modo de ecrã completo.
OFF	Não mostra o ambiente de trabalho das Apps navegável. A unidade está predefinida para o ecrã inicial da calculadora. O ecrã inicial da calculadora aparece quando [2nd] [QUIT]. O menu APPLICATIONS aparece quando premir [APPS].

Códigos de caracteres da TI-89 Titanium / Voyage™ 200

A função **char()** permite referir qualquer carácter através do respectivo código de carácter numérico. Por exemplo, para ver ◆ no ecrã Program I/O, utilize `Disp char(127)`. Pode utilizar **ord()** para localizar o código numérico de um carácter. Por exemplo, `ord("A")` devolve 65.

1.	SOH	38.	&	76.	L	113.	q	148.	ω	186.	º	222.	Þ
2.	STX	39.	'	77.	M	114.	r	149.	E	187.	»	223.	ß
3.	ETX	40.	(	78.	N	115.	s	150.	*e*	188.	*d*	224.	à
4.	EOT	41.	)	79.	O	116.	t	151.	***i***	189.	∫	225.	á
5.	ENQ	42.	*	80.	P	117.	u	152.	ʳ	190.	∞	226.	â
6.	ACK	43.	+	81.	Q	118.	v	153.	ᵀ	191.	¿	227.	ã
7.	BELL	44.	,	82.	R	119.	w	154.	$\bar{x}$	192.	À	228.	ä
8.	BS	45.	[°]	83.	S	120.	x	155.	$\bar{y}$	193.	Á	229.	å
9.	TAB	46.	.	84.	T	121.	y	156.	≤	194.	Â	230.	æ
10.	LF	47.	/	85.	U	122.	z	157.	≠	195.	Ã	231.	ç
11.	[0]	48.	0	86.	V	123.	{	158.	≥	196.	Ä	232.	è
12.	FF	49.	1	87.	W	124.	|	159.	∠	197.	Å	233.	é
13.	CR	50.	2	88.	X	125.	}	160.	…	198.	Æ	234.	ê
14.	🔒	51.	3	89.	Y	126.	~	161.	¡	199.	Ç	235.	ë
15.	[X]	52.	4	90.	Z	127.	◆	162.	¢	200.	È	236.	ì
16.	[▶]	53.	5	91.	[	128.	α	163.	£	201.	É	237.	í
17.	◀	54.	6	92.	\	129.	β	164.	¤	202.	Ê	238.	î
18.	▶	55.	7	93.	]	130.	Γ	165.	¥	203.	Ë	239.	ï
19.	▲	56.	8	94.	^	131.	γ	166.	¦	204.	Ì	240.	ð
20.	▼	57.	9	95.	_	132.	Δ	167.	§	205.	Í	241.	ñ
21.	←	58.	:	96.	`	133.	δ	168.	√	206.	Î	242.	ò
22.	→	59.	;	97.	a	134.	ε	169.	◐	207.	Ï	243.	ó
23.	↑	60.	<	98.	b	135.	ζ	170.	ª	208.	Ð	244.	ô
24.	↓	61.	=	99.	c	136.	θ	171.	«	209.	Ñ	245.	õ
25.	◀	62.	>	100.	d	137.	λ	172.	¬	210.	Ò	246.	ö
26.	▶	63.	?	101.	e	138.	ξ	173.	-	211.	Ó	247.	÷
27.	↑	64.	@	102.	f	139.	Π	174.	®	212.	Ô	248.	ø
28.	∪	65.	A	103.	g	140.	π	175.	¯	213.	Õ	249.	ù
29.	∩	66.	B	104.	h	141.	ρ	176.	°	214.	Ö	250.	ú
30.	⊂	67.	C	105.	i	142.	Σ	177.	±	215.	×	251.	û
31.	∈	68.	D	106.	j	143.	σ	178.	²	216.	Ø	252.	ü
32.	SPACE	69.	E	107.	k	144.	τ	179.	³	217.	Ù	253.	ý
33.	!	70.	F	108.	l	145.	ϕ	180.	⁻¹	218.	Ú	254.	þ
34.	"	71.	G	109.	m	146.	ψ	181.	µ	219.	Û	255.	ÿ
35.	#	72.	H	110.	n	147.	Ω	182.	¶	220.	Ü		
36.	$	73.	I	111.	o			183.	•	221.	Ý		
37.	%	74.	J	112.	p			184.	¸				
		75.	K					185.	¹				

Códigos de teclas da TI-89 Titanium

A função **getKey()** devolve um valor que corresponde à última tecla premida de acordo com as tabelas mostradas nesta secção. Por exemplo, se o programa contiver uma função **getKey()**, premindo [2nd] [F6], devolve um valor de 273.

Tabela 1: Códigos de teclas para teclas primárias

Tecla	Modificador							
	Nenhum		[↑]		[2nd]		[◆]	
	Assoc.	Valor	Assoc.	Valor	Assoc.	Valor	Assoc.	Valor
[F1]	F1	268	F1	268	F1	268		8460
[F2]	F2	269	F2	269	F2	269		8461
[F3]	F3	270	F3	270	F3	270		8462
[F4]	F4	271	F4	271	F4	271		8463
[F5]	F5	272	F5	272	F5	272		8464
[F6]	F6	273	F6	273	F6	273		8465
[F7]	F7	274	F7	274	F7	274		8466
[F8]	F8	275	F8	275	F8	275		8467
[MODE]	MODE (Modo)	266	MODE (Modo)	266	MODE (Modo)	266		8458
[CLEAR]	CLEAR (Apagar)	263	CLEAR (Apagar)	263	CLEAR (Apagar)	263		8455
[LN]	LN	262	LN	262	e^x	4358		8454
[ESC]	ESC	264	ESC	264	QUIT (Sair)	4360		8456
[APPS]	APPS	265	APPS	265	SWITCH (Mudar)	4361		8457
[ENTER]	CR	13	CR	13	ENTRY (Entrada)	4109	APPROX (Aprox)	8205
[SIN]	SIN (Seno)	259	SIN (Seno)	259	SIN $(\text{Seno})^{-1}$	4355		8451
[COS]	COS (Co-seno)	260	COS (Co-seno)	260	COS $(\text{Co-seno})^{-1}$	4356		8452
[TAN]	TAN (Tangente)	261	TAN (Tangente)	261	TAN $(\text{Tangente})^{-1}$	4357		8453
[^]	^	94	^	94	π	140		8286
[(]	(	40	(	40	{	123		8232
[)]	)	41	)	41	}	125		8233
[,]	,	44	,	44	[	91		8236
Tecla	Modificador							

Tabela 1: Códigos de teclas para teclas primárias

	Nenhum				2nd		ON	
	Assoc.	**Valor**	**Assoc.**	**Valor**	**Assoc.**	**Valor**	**Assoc.**	**Valor**
÷	/	47	/	47	]	93		8239
×	*	42	*	42	√	4138		8234
−	-	45	-	45	VAR-LNK	4141	Contraste	
+	+	43	+	43	CHAR (Carácter)	4139	Contraste +	
STO▸	STO▸	258	STO▸	258	RCL	4354		8450
SPACE		32		32		32		8224
=	=	61	=	61	\	92		8253
←	BS	257	BS	257	INS	4353	DEL	8449
[θ]	θ	136	θ	136	:	58		8328
(-)	-	173	-	173	ANS	4372		8365
.	.	46	.	46	>	62		8238
0	0	48	0	48	<	60		8240
1	1	49	1	49	E	149		8241
2	2	50	2	50	CATALOG (Catálogo)	4146		8242
3	3	51	3	51	CUST (Custo)	4147		8243
4	4	52	4	52	Σ	4148		8244
5	5	53	5	53	MATH (Matemática)	4149		8245
6	6	54	6	54	MEM	4150		8246
7	7	55	7	55	∫	4151		8247
8	8	56	8	56	*d*	4152		8248
9	9	57	9	57	x^{-1}	4153		8249
A	a	97	A	65	Tabela 3			8257
B	b	98	B	66	‘	39		8258
C	c	99	C	67	Tabela 4		COPY (Copiar)	8259
Tecla	**Modificador**							
	Nenhum				2nd		ON	
	Assoc.	**Valor**	**Assoc.**	**Valor**	**Assoc.**	**Valor**	**Assoc.**	**Valor**

Tabela 1: Códigos de teclas para teclas primárias

D	d	100	D	68	°	176		8260
E	e	101	E	69	Tabela 5		WINDOW (Janela)	8261
F	f	102	F	70	∠	159	FORMAT (Formatar)	8262
G	g	103	G	71	Tabela 6			8263
H	h	104	H	72	&	38		8264
I	i	105	I	73	i	151		8265
J		106	J	74	∞	190		8266
K	k	107	K	75	|	124	KEY (Tecla)	8267
L	l	108	L	76	“	34		8268
M	m	109	M	77	;	59		8269
N	n	110	N	78	Tabela 7		NEW (Novo)	8270
O	o	111	O	79	Tabela 8		OPEN (Abrir)	8271
P	p	112	P	80	_	95	UNITS (Unidades)	8272
Q	q	113	Q	81	?	63	CALCHOME (Ecrã inicial da calculadora)	8273
R	r	114	R	82	@	64	GRAPH (Gráfico)	8274
S	s	115	S	83	β	223	SAVE (Guardar)	8275
T	t	116	T	84	#	35	TBLSET	8276
U	u	117	U	85	Tabela 9			8277
V	v	118	V	86	≠	157	PASTE (Colar)	8278
W	w	119	W	87	!	33	Y=	8279
X	x	120	X	88	◐	169	CORTAR (Cortar)	8280
Y	y	121	Y	89	▸	18	TABLE (Tabela)	8281
Z	z	122	Z	90	CAPS			8282

Tabela 2: Teclas de setas (incluindo movimento diagonal)

Tecla	Normal	Z	2nd	ON	alpha
⊕	338	16722	4434	8530	33106
⊕	340	16724	4436	8532	33108
⊕	344	16728	4440	8536	33112
⊕	337	16721	4433	8529	33105
⊕ e ⊕	339	16723	4435	8531	33107
⊕ e ⊕	342	16726	4438	8534	33110
⊕ e ⊕	345	16729	4441	8537	33113
⊕ e ⊕	348	16732	4444	8540	33116

Tabela 3: Letras gregas (antecedido por ON ( )

Teclas	Segundo modificador			
	alpha		Z	
	Assoc.	Valor	Assoc.	Valor
[A]	α	128		
([B]	β	129		
, [D]	δ	133	Δ	132
÷ [E]	ε	134		
-1 [F]	ϕ	145		
7 [G]	γ	131	Γ	130
4 [L]	λ	137		
5 [M]	μ	181		
↑ [P]	π	140	Π	139
2 [R]	ρ	141		
[S]	σ	143	Σ	142
F4 [T]	τ	144		
◉ [W]	ω	148	Ω	147
r	ξ	138		
3	ψ	146		
°	ζ	135		

Códigos de teclas da Voyage™ 200

A função **getKey()** devolve um valor que corresponde à última tecla premida de acordo com as tabelas mostradas nesta secção. Por exemplo, se o programa contiver uma função **getKey()**, premindo 2nd F1, devolve um valor de 268.

Tabela 1: Códigos de teclas para teclas primárias

Tecla	Modificador							
	Nenhum				2nd		ON	
	Assoc.	Valor	Assoc.	Valor	Assoc.	Valor	Assoc.	Valor
F1	F1	268	F1	268	F1	268		8460
F2	F2	269	F2	269	F2	269		8461
F3	F3	270	F3	270	F3	270		8462
F4	F4	271	F4	271	F4	271		8463
F5	F5	272	F5	272	F5	272		8464
F6	F6	273	F6	273	F6	273		8465
F7	F7	274	F7	274	F7	274		8466
F8	F8	275	F8	275	F8	275		8467
MODE	MODE (Modo)	266	MODE (Modo)	266	MODE (Modo)	266		8458
CLEAR	CLEAR (Apagar)	263	CLEAR (Apagar)	263	CLEAR (Apagar)	263		8455
LN	LN	262	LN	262	e^x	4358		8454
ESC	ESC	264	ESC	264	QUIT (Sair)	4360		8456
APPS	APPS	265	APPS	265	SWITCH (Mudar)	4361		8457
ENTER	CR	13	CR	13	ENTRY (Entrada)	4109	APPROX (Aprox)	8205
SIN	SIN (Seno)	259	SIN (Seno)	259	SIN $(\text{Seno})^{-1}$	4355		8451
COS	COS (Co-seno)	260	COS (Co-seno)	260	COS $(\text{Co-seno})^{-1}$	4356		8452
TAN	TAN	261	TAN	261	TAN $(\text{Tangente})^{-1}$	4357		8453
^	^	94	^	94	π	140		8286
(	(	40	(	40	{	123		8232
)	)	41	)	41	}	125		8233
,	,	44	,	44	[	91		8236

÷	/	47	/	47	]	93		8239
×	*	42	*	42	√	4138		8234
−	-	45	-	45	VAR-LNK	4141	Contraste	
+	+	43	+	43	CHAR (Carácter)	4139	Contraste +	
STO▸	STO▸	258	STO▸	258	RCL	4354		8450
SPACE		32		32		32		8224
=	=	61	=	61	\	92		8253
←	BS	257	BS	257	INS	4353	DEL	8449
[θ]		136	θ	136	:	58		8328
(-)	-	173	-	173	ANS	4372		8365
.	.	46	.	46	>	62		8238
Tecla	**Modificador**							
	Nenhum				2nd		ON	
	Assoc.	**Valor**	**Assoc.**	**Valor**	**Assoc.**	**Valor**	**Assoc.**	**Valor**
0	0	48	0	48	<	60		8240
1	1	49	1	49	E	149		8241
2	2	50	2	50	CATALOG (Catálogo)	4146		8242
3	3	51	3	51	CUST (Custo)	4147		8243
4	4	52	4	52	Σ	4148		8244
5	5	53	5	53	MATH (Matemática)	4149		8245
6	6	54	6	54	MEM	4150		8246
7	7	55	7	55	∫	4151		8247
8	8	56	8	56	*d*	4152		8248
9	9	57	9	57	x^{-1}	4153		8249
A	a	97	A	65	Tabela 3			8257
B	b	98	B	66	‘	39		8258
C	c	99	C	67	Tabela 4		COPY (Copiar)	8259
D	d	100	D	68	°	176		8260
E	e	101	E	69	Tabela 5		WINDOW (Janela)	8261

F	f	102	F	70	∠	159	FORMAT (Formatar)	8262
G	g	103	G	71	Tabela 6			8263
H	h	104	H	72	&	38		8264
I	i	105	I	73	i	151		8265
J		106	J	74	∞	190		8266
K	k	107	K	75	|	124	KEY (Tecla)	8267
L	l	108	L	76	“	34		8268
M	m	109	M	77	;	59		8269
N	n	110	N	78	Tabela 7		NEW (Novo)	8270
O	o	111	O	79	Tabela 8		OPEN (Abrir)	8271
P	p	112	P	80	_	95	UNITS (Unidades)	8272
Q	q	113	Q	81	?	63	CALCHOME (Ecrã inicial da calculadora)	8273
R	r	114	R	82	@	64	GRAPH (Gráfico)	8274
S	s	115	S	83	β	223	SAVE (Guardar)	8275
T	t	116	T	84	#	35	TBLSET	8276
U	u	117	U	85	Tabela 9			8277
V	v	118	V	86	≠	157	PASTE (Colar)	8278
W	w	119	W	87	!	33	Y=	8279
X	x	120	X	88	◐	169	CORTAR (Cortar)	8280
Y	y	121	Y	89	▸	18	TABLE (Tabela)	8281
Z	z	122	Z	90	CAPS			8282

Tabela 2: Teclas de setas (incluindo movimento diagonal)

Tecla	Normal	Z	2nd	ON	
⊖	338	16722	4434	8530	33106
⊕	340	16724	4436	8532	33108
⊖	344	16728	4440	8536	33112
⊖	337	16721	4433	8529	33105

⊕ e ⊖	339	16723	4435	8531	33107
⊕ e ⊕	342	16726	4438	8534	33110
⊖ e ⊖	345	16729	4441	8537	33113
⊖ e ⊕	348	16732	4444	8540	33116

Nota: O modificador Grab ( ) só afecta as teclas de setas.

Tabela 3: Letras com acentos graves (antecedido por 2nd A)

Tecla	Assoc.	Normal	
A	à	224	192
E	è	232	200
I	ì	236	204
O	ò	242	210
U	ù	249	217

Tabela 4: Letras cedilhadas (antecidido por 2nd C)

Tecla	Assoc.	Normal	
C	ç	231	199

Tabela 5: Letras com acentos agudos (antecedido por 2nd E)

Tecla	Assoc.	Normal	
A	á	225	193
E	é	233	201
I	í	237	205
O	ó	243	211
U	ú	250	218
Y	ý	253	221

Tabela 6: Letras gregas (antecedido por 2nd G)

Tecla	Assoc.	Normal	
A	α	128	
B	β	129	
D	δ	133	132
E	ε	134	
F	ϕ	145	
G	γ	131	130

L	λ	137	
M	μ	181	
P	π	140	139
R	ρ	141	
S	σ	143	142
T	τ	144	
W	ω	148	147
X	ξ	138	
Y	ψ	146	
Z	ζ	135	

Tabela 7: Letras com til (antecedido por 2nd N)

Tecla	Assoc.	Normal	
N	ñ	241	209
O	õ	245	

Tabela 8: Letras de omissão (antecedido por 2nd O)

Tecla	Assoc.	Normal	
A	â	226	194
E	ê	234	202
I	î	238	206
O	ô	244	212
U	û	251	219

Tabela 9: Letras com trema (antecedido por 2nd U)

Tecla	Assoc.	Normal	
A	ä	228	196
E	ë	235	203
I	ï	239	207
O	ö	246	214
U	ü	252	220
Y	ÿ	255	

Introduzir números complexos

Pode introduzir números complexos no formato polar ($r\angle\theta$), em que r é a magnitude e θ é o ângulo, ou no formato polar $re^{i\theta}$. Pode também introduzir números complexos no formato rectangular a+b*i*

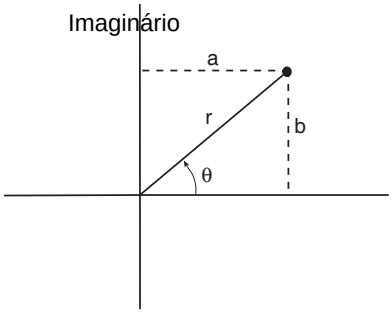
Introdução aos números complexos

Um número complexo tem componentes reais e imaginários que identificam um ponto no plano complexo. Estes componentes são medidos ao longo dos eixos reais e imaginários, que são similares aos eixos x e y no plano real.

O ponto pode ser expresso no formato rectangular ou num dos dois polares

O símbolo i representa o número imaginário $\sqrt{-1}$

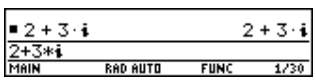
f

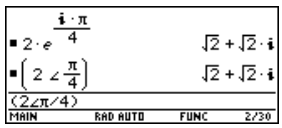


Como mostrado em seguida, o formato que pode introduzir depende do modo Angle (Ângulo) actual.

Pode utilizar o formato:	Quando a definição do modo Angle (Ângulo) for:
a+b*i*	Radian (Radiano), Degree (Grau) ou Gradian (Grado)
$re^{i\theta}$	Apenas no modo de ângulo Radian (Radiano) (No modo de ângulo Degree (Grau) ou Gradian (Grado), este formato provoca um erro de domínio.)
($r\angle\theta$)	Radian (Radiano), Degree (Grau) ou Gradian (Grado)

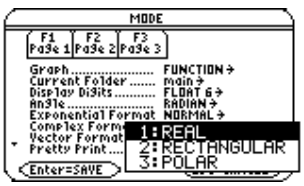
Utilize um dos seguintes métodos para introduzir um número complexo.

Para introduzir o:	Efectuar este procedimento:
Formato rectangular a+b*i*	Substitua os nomes das variáveis ou os valores aplicáveis para a e b. a [+] b [2nd] [i] **Nota:** Para obter o símbolo ***i*** , prima [2nd] [i], não digite apenas um i alfabético. Por exemplo: 

Para introduzir o:	Efectuar este procedimento:
Formato polar $re^{i\theta}$ ou $(r\angle\theta)$ Os parêntesis são necessários para o formato $(r\angle\theta)$.	Substitua os nomes das variáveis ou os valores aplicáveis para r e θ, em que θ é interpretado de acordo com a definição do modo de ângulo. **eq TI89 Titanium:** [alpha] [R] [◆] [e^x] [2nd] [i] [◆] [θ] [)] ou – [(] [alpha] [R] [2nd] [∠] [◆] [θ] [)] **Importante:** Não utilize o formato polar $re^{i\theta}$ no modo de ângulo Degree (Grau). Provoca um erro de domínio. **Nota:** Para obter o símbolo e, prima: eq TI89 Titanium: [◆] [e^x]. Voyage™ 200**:** [2nd] [e^x] Não digite apenas um e alfabético. **Sugestão:** Para obter o símbolo ∠, prima [2nd] [∠]. **Sugestão:** Para introduzir θ em graus para $(r\angle\theta)$, pode digitar um símbolo ° (como 45°). Para obter o símbolo °, prima [2nd] [°]. Não deve utilizar graus ou grados para $re^{i\theta}$. **Voyage™ 200:** R [2nd] [e^x] [2nd] [i] [θ] [)] ou – [(] R [2nd] [i] [θ] [)] Por exemplo:  Os resultados aparecem em formato rectangular, mas pode

Modo Complex Format (Formato complexo) para visualizar resultados

Utilize [MODE] para definir o modo Complex Format (Formato complexo) para uma das três definições.



Pode introduzir um número complexo a qualquer momento, independentemente da definição do modo Complex Format (Formato complexo). No entanto, a definição do modo determina a forma de visualização do resultados.

Se Complex Format i(Formato complexo) for:	The TI-89 Titanium / Voyage™ 200:

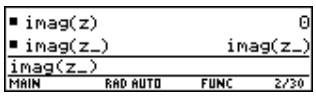
REAL	Não mostra resultados complexos excepto se: • Introduzir um número complexo. – ou – • Utilizar uma função complexa, como **cFactor()**, **cSolve()** ou **cZeros()**. Se os resultados complexos aparecerem, aparecerão no formato a+b***i*** ou $re^{i\theta}$ **Nota:** Pode introduzir números complexos em qualquer formato (ou uma mistura de todos os formatos) dependendo do modo de ângulo.
RECTANGULAR	Mostra os resultados complexos como a+b***i***.
POLAR	Mostra os resultados complexos como: • $re^{i\theta}$ se o modo de ângulo = Radian (Radiano) – ou – • ($r\angle\theta$) se o modo de ângulo = Degree (Grau) ou Gradian (Grado)

Utilizar variáveis complexas em cálculos simbólicos

Independentemente da definição do modo Complex Format (Formato complexo), as variáveis que não tenham nenhum valor guardado e não termimem com um carácter sublinhado (_) são tratadas como números reais. Para efectuar a análise simbólica complexa, pode utilizar um dos seguintes métodos para configurar uma variável complexa.

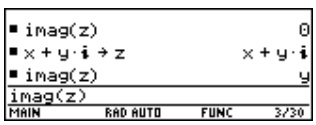
Método 1: Utilize um carácter sublinhado _
(**eq TI89 Titanium:** [ON] **Voyage™ 200** [2nd]) como o último carácter no nome da variável para indicar uma variável complexa. Por exemplo:

z_ é tratado como uma variável complexa se não tiver um valor guardado.



Método 2: Guarde um valor não real em qualquer variável. Por exemplo:

x+y***i***→z
z é tratado como uma variável complexa.



Nota: para obter melhores resultados em cálculos como **cSolve()** e **cZeros()**, utilize o Método 1.

Número complexos e modo Degree (Grau)

O modo de ângulo Radian (Radiano) é recomendado para cálculos de números complexos. Internamente, a TI-89 Titanium / Voyage™ 200 converte todos os valores trig introduzidos para radianos, mas não converte os valores para funções exponenciais, logarítmicas ou hiperbólicas.

No modo de ângulo Degree (Grau), as identidades do complexo como e^(iq) = cos(q) + i sin(q) não são geralmente verdadeiras porque os valores para cos e sin são convertidos para radianos, enquanto os valores para e^( ) não são. Por exemplo, e^(i45) = cos(45) + i sin(45) é tratado internamente como e^(i45) = cos(p/4) + i sin(p/4). As identidades do complexo são sempre verdadeiras no modo de ângulo Radian (Radiano).

Nota: Se utilizar o modo de ângulo Degree (Grau), tem de efectuar as entradas polares no formato ($r\angle\theta$). No modo de ângulo Degree (Grau) ou Gradian (Grado), uma entrada $r\,e^{i\theta}$ provoca um erro.

Informações de precisão

Para maximizar a precisão, a TI-89 Titanium / Voyage™ 200 tem mais dígitos internamente do que os visualizados.

Precisão computacional

Os valores de ponto flutuante (decimais) na memória são guardados utilizando até 14 dígitos com um expoente de 3-dígitos.

- Para as variáveis da janela min e max (xmin, xmax, ymin, ymax, etc.), pode guardar valores utilizando até 12 dígitos. As outras variáveis da janela utilizam 14 dígitos.
- Quando aparecer um valor de ponto flutuante, o valor apresentado é arredondado como especificado pelas definições do modo aplicável (Display Digits (Visualizar dígitos), Exponential Format (Formato exponencial), etc.), com um máximo de 12 dígitos e um expoente de 3 dígitos.
- RegEQ mostra coeficientes até 14 dígitos.

Os valores inteiros na memória são guardados até 614 dígitos.

Precisão gráfica

A variável da janela xmin é o centro do pixel utilizado mais à esquerda, e xmax é o centro do pixel utilizado mais à direita. Δx é a distância entre os centros de dois pixéis adjacentes na horizontal.

- Δx é calculado como (xmax − xmin) / (# de x pixéis −1).
- Se introduzir Δx a partir do ecrã inicial ou de um programa, xmax é calculado como xmin + Δx *(# de x pixéis − 1).

Nota: Para obter uma tabela que liste o número de pixéis num ecrã dividido ou completo, consulte “Definir e sair do modelo de ecrã dividido em ecrãs divididos.

A variável da janela ymin é o centro do pixel utilizado no fundo, e ymax é o centro do pixel utilizado no topo. Δx é a distância entre os centros de dois pixéis adjacentes na vertical.

- Δy é calculado como (ymax − ymin) / (# de y pixéis − 1).
- Se introduzir Δy a partir do ecrã inicial ou de um programa, ymax é calculado como ymin + Δy * (# de y pixéis − 1).

As coordenadas do cursor aparecem como oito caracteres (que pode incluir um sinal negativo, ponto decimal e expoente). Os valores das coordenadas (xc, yc, zc, etc.) são actualizados com um máximo de eq 12 dígitos de precisão.

Variáveis do sistema e nomes reservados

Esta secção lista os nomes das variáveis do sistema e os nomes das funções reservadas utilizados pela TI-89 Titanium / Voyage™ 200. Apenas as variáveis do sistema e os nomes de funções reservadas identificadas com um asterisco (*) podem ser eliminados com **DelVar** *var* na linha de entrada.

Gráfico

y1(x)–y99(x)*	y1'(t)–y99'(t)*	yi1–yi99*	r1(θ)–r99(θ)*
xt1(t)–xt99(t)*	yt1(t)– yt99(t)*	z1(x,y)–z99(x,y)*	u1(n)–u99(n)*
ui1–ui99*	xc	yc	zc
tc	rc	θc	nc
xfact	yfact	zfact	xmin
xmax	xscl	xgrid	ymin
ymax	yscl	ygrid	xres
Δx	Δy	zmin	zmax
zscl	eyeθ	eyeϕ	eyeψ
ncontour	θmin	θmax	θstep
tmin	tmax	tstep	t0
tplot	ncurves	diftol	dtime
Estep	fldpic	fldres	nmin
nmax	plotStrt	plotStep	sysMath

Zoom do gráfico

zxmin	zxmax	zxscl	zxgrid
zymin	zymax	zyscl	zygrid
zxres	zθmin	zθmax	zθstep
ztmin	ztmax	ztstep	zt0de
ztmaxde	ztstepde	ztplotde	zzmin
zzmax	zzscl	zeyeθ	zeyeϕ
zeyeψ	znmin	znmax	zpltstrt
zpltstep			

Estatística

$\bar{x}$	$\bar{y}$	Σx	σx
Σx^2	Σxy	Σy	σy
Σy^2	corr	maxX	maxY
medStat	medx1	medx2	medx3

medy1	medy2	medy3	minX
minY	nStat	q1	q3
regCoef*	regEq(x)*	seed1	seed2
Sx	Sy	R^2	

Tabela

tblStart	ΔtbI	tblInput

Dados/Matriz

c1–c99	sysData*

Diversos

principal	ok	errornum

Solver

eqn*	exp*

Hierarquia do EOS (Sistema Operativo de Equações)

Esta secção descreve o Sistema Operativo de Equações (EOS) utilizado pela TI-89 Titanium / Voyage™ 200. Os números, as variáveis e as funções são introduzidos numa sequência simples e clara. O EOS avalia as expressões e as equações através do agrupamento parentético e de acordo com as prioridades descritas abaixo.

Ordem de avaliação

Nível	Operador
1	Parêntesis curvos ( ), parêntesis curvos [ ], chavetas { }
2	Acção indirecta (#)
3	Chamadas de funções
4	Pós-operadores: graus-minutos-segundos (,',"), factorial (!), percentagem (%), radiano ([a-lock]), subscrito ([ ]), transposição ( )
5	Exponenciação, operador de potência (^)
6	Negação ([≈])
7	Concatenação de cadeias (&)
8	Multiplicação ( ), divisão (/)
9	Adição (+), subtracção ([°])
10	Relações de igualdade: igual (=), não igual (ou /=), menor que (<), igual ou menor que (ou <=), maior que (>), igual ou maior que (〉 ou >=)
11	Lógico **not**
12	Lógico **and**
13	Lógico **or**, lógico exclusivo **xor**
14	Restrição do operador “with” (|)
15	Guardar (→)

Parêntesis curvos, parêntesis rectos e chavetas

Todos os cálculos que estejam entre parêntesis curvos, parêntesis rectos ou chavetas são avaliados primeiro. Por exemplo, na expressão 4(1+2), o EOS avalia primeiro a parte da expressão que está entre parêntesis, 1+2, e, em seguida, multiplica o resultado, 3, por 4.

O número de parêntesis curvos, parêntesis rectos e chavetas de abertura e fecho tem de ser igual numa expressão ou equação. Se não, aparece uma mensagem de erro que indica o elemento inexistente. Por exemplo, (1+2)/(3+4 mostram a mensagem de erro “Missing).”

Nota: Como a TI-89 Titanium / Voyage™ 200 permite definir as suas próprias funções, um nome da variável seguido por uma expressão entre parêntesis é considerado uma “chamada de função” em de vez de uma multiplicação implícita. Por exemplo a(b+c) é a função a avaliada por b+c. Para multiplicar a expressão b+c pela variável a, utilize a multiplicação explícita: aù(b+c).

Acção indirecta

O operador de acção indirecta (#) converte uma cadeia para um nome de função ou variável. Por exemplo, #(“x”&”y”&”z”) cria o nome da variável xyz. A acção indirecta permite também criar e modificar variáveis dentro de um programa. Por exemplo, se 10!r e “r”!s1, então #s1=10.

Pós-operadores

Os pós-operadores são operadores que vêm logo após um argumento, como 5!, 25%, ou 60ó15' 45". Os argumentos seguidos por um pós-operador são avaliados no quarto nível de prioridade. Por exemplo, na expressão 4^3!, 3! é avaliado primeiro. O resultado, 6, torna-se no expoente de 4 para dar 4096.

Exponenciação

A exponenciação (^) e a exponenciação de elemento a elemento (.^) são avaliadas da direita para a esquerda. Por exemplo, a expressão 2^3^2 é avaliada como 2^(3^2) para produzir 512. Este é diferente de (2^3)^2, que é 64.

Negação

Para introduzir um número negativo, prima · seguido pelo número. As pós-operações e a exponenciação são efectuadas antes da negação. Por exemplo, o resultado de ëx2 é um número negativo, e ë92 =ë81. Utilize os parêntesis para elevar ao quadrado um número negativo como (ë9)2 para produzir 81. Não se esqueça também de que o 5 negativo (ë5) é diferente de menos 5 (ì5), e ë3! avalia como ë(3!).

Restrição (|)

O argumento seguido pelo operador “with” (|) fornece um conjunto de restrições que afectam a avaliação do argumento que antecede o operador “with”.

Fórmulas de regressão

Esta secção descreve a forma de cálculo das regressões estatísticas.

Algoritmo de mínimo quadrado

A maioria das regressões utiliza as técnicas de mínimos quadrados recursivos não lineares para optimizar a função de custo seguinte, que é a soma dos quadrados dos erros residuais:

$$J = \sum_{i=1}^{N} \left[residualExpression \right]^2$$

onde:A expressão residual está em termos de xi e yi

x_i é a lista de variáveis independente
y_i é a lista de variáveis dependente
N é a dimensão das listas

Esta técnica tenta prever recursivamente as constantes da expressão do modelo para tornar J o mais pequeno possível.

Por exemplo, $y=a\ sin(bx+c)+d$ é a equação do modelo para **SinReg**. Por isso, a expressão residual é:

$a\ sin(bx_i+c)+d[°]\ y_i$

Por isso, para **SinReg**, o algoritmo do mínimo quadrado localiza as constantes *a*, *b*, *c* e *d* que minimizam a função:

$$J = \sum_{i=1}^{N} \left[a\ sin(bx_I + c) + d - y_i \right]^2$$

Regressões

Regressão	Descrição
CubicReg	Utiliza o algoritmo do mínimo quadrado para ajustar o polinómio de terceira ordem: $y=ax^3+bx^2+cx+d$ Para quatro pontos de dados, a equação é um ajuste polinomial; para cinco ou mais, é uma regressão polinomial. São necessários pelo menos quatro pontos de dados.
ExpReg	Utiliza o algoritmo do mínimo quadrado e os valores transformados *x* e ln(*y*) para ajustar a equação do modelo: $y=ab^x$
LinReg	Utiliza o algoritmo do mínimo quadrado para ajustar a equação do modelo: $y=ax+b$ em que *a* é o declive e *b* é a intersecção y.

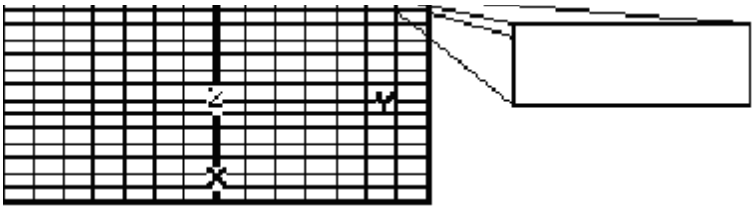
LnReg	Utiliza o algoritmo do mínimo quadrado e os valores transformados ln(x) e y para ajustar a equação do modelo: $y=a+b\ln(x)$
Logística	Utiliza o algoritmo do mínimo quadrado para ajustar a equação do modelo: $y=a/(1+b*e^{(c*x)})+d$
MedMed	Utilize a técnica da linha média-média (linha resistente) para calcular os pontos sumários x1, y1, x2, y2, x3 e y3, e ajuste a equação do modelo: $y=ax+b$ em que a é o declive e b é a intersecção y.
PowerReg	Utiliza o algoritmo do mínimo quadrado e os valores transformados ln(x) e ln(y) para ajustar a equação do modelo: $y=ax^b$
QuadReg	Utiliza o algoritmo do mínimo quadrado para ajustar o polinómio de segunda ordem: $y=ax^2+bx+c$ Para três pontos de dados, a equação é um ajuste polinomial; para quatro ou mais, é uma regressão polinomial. São necessários pelo menos três pontos de dados.
QuartReg	Utiliza o algoritmo do mínimo quadrado para ajustar o polinómio de quarta ordem: $y=ax^4+bx^3+cx^2+dx+e$ Para cinco pontos de dados, a equação é um ajuste polinomial; para seis ou mais, é uma regressão polinomial. São necessários pelo menos cinco pontos de dados.
SinReg	Utiliza o algoritmo do mínimo quadrado para ajustar a equação do modelo: $y=a\sin(bx+c)+d$

Níveis de contorno e algoritmo de desenho implícito

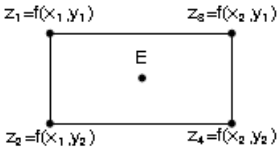
Os contornos são calculados e desenhados com o método seguinte. Um desenho implícito é igual a um contorno, excepto se um desenho implícito se destina apenas ao contorno z=0.

Algoritmo

Baseada nas variáveis da janela x e y, a distância entre xmin e xmax e entre ymin e ymax é dividida pelo número de linhas da grelha especificadas por xgrid e ygrid. Estas linhas da grelha intersectam-se para formar uma série de rectângulos.



Para cada rectângulo, a equação é avaliada em cada um dos quatro cantos (também chamados de vértices ou pontos da grelha) e é calculado um valor médio (E):

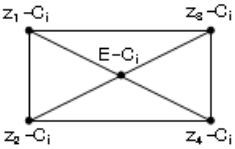


E = z1 + z2 + z3 + z44

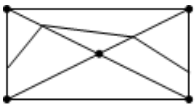
O valor E é tratado como o valor da equação no centro do rectângulo.

Para cada valor do contorno especificado (Ci)

- Em cada cinco pontos apresentados à direita, a diferença entre o valor z do ponto e o valor do contorno é calculada.



- Uma alteração de sinal entre dois pontos adjacentes quaisquer implica que um contorno cruza a linha que junta esses dois pontos. A interpolação linear é utilizada para aproximar onde o zero cruza a linha.
- No rectângulo, quaisquer cruzamentos zero estão ligados a linhas rectas.
- Este processo é repetido para cada valor do contorno.



Cada rectângulo da grelha é tratado similarmente.

Método de Runge-Kutta

Para integrações Runge-Kutta de equações diferenciais ordinárias, a TI-89 Titanium / Voyage™ 200 utiliza a fórmula Bogacki-Shampine 3(2) como encontrada no jornal *Applied Math Letters*, 2 (1989), pp. 1–9.

Fórmula Bogacki-Shampine 3(2)

A fórmula Bogacki-Shampine 3(2) fornece um resultado de precisão de 3ª ordem e uma previsão de erro baseada numa fórmula de 2ª ordem embebida. Para um problema da forma:

$y' = f(x, y)$

e um tamanho de passo fornecido h, a fórmula Bogacki-Shampine pode ser escrita:

$F_1 = f(x_n, y_n)$

$F_2 = f\left(x_n + h\,12\ ,\ y_n + h\,12\ F_1\right)$

$F_3 = f\left(x_n + h\,34\ ,\ y_n + h\,34\ F_2\right)$

$y_{n+1} = y_n + h\left(\ 29\ F_1 + 13\ F_2 + 49\ F_3\right)$

$x_{n+1} = x_n + h$

$F_4 = f\left(x_{n+1}\ ,\ y_{n+1}\right)$

$errest = h\left(\ 572\ F_1\,[\circ]\ 112\ F_2\,[\circ]\ 19\ F_3 + 18\ F_4\right)$

A previsão de erro *errest* é utilizada para controlar o tamanho do passo automaticamente. Para uma discussão mais aprofundada de como isto pode ser efectuado, consulte *Numerical Solution of Ordinary Differential Equations* de L. F. Shampine (New York: Chapman & Hall, 1994).

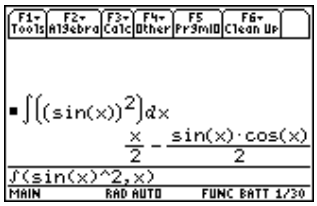
O software da TI-89 Titanium / Voyage™ 200 não ajusta o tamanho do passo para ir para pontos de saída específicos. Em vez disso, escolhe os passos maiores que forem possíveis (baseado no diftol de tolerância de erro) e obtém resultados para xn _ x _ xn+1 com o polinómio de interpolação cúbica que passa através do ponto (xn , yn) com o declive F1 e através de (xn+1 , yn+1) com o declive F4. O interpolante é eficiente e fornece resultados durante o passo que são tão precisos como os resultados no fim do passo.

Informações das pilhas

A TI-89 Titanium / Voyage™ 200 utiliza dois tipos de pilhas: quatro pilhas alcalinas e uma pilha de lítio que funciona como apoio para manter a memória durante a mudança das pilhas alcalinas.

Quando substituir as pilhas

À medida que as pilhas alcalinas acabam, o ecrã começar a escurecer (em especial durante os cálculos). Para compensar, é necessário ajustar o contraste para uma definição mais elevada. Se for necessário aumentar a definição do contraste frequentemente, é necessário substituir as pilhas alcalinas. Para o ajudar, um indicador BATT (BATT) mostra na área da lina de estado quando as pilhas acabaram e quando devem ser substituídas. Quando o indicador BATT aparecer com o texto ao contrário (BATT), tem de substituir as pilhas alcalinas imediatamente.



Para evitar a perda de dados, não remova a pilha de lítio, excepto se estiverem instaladas quatro pilhas alcalinas novas. Substitua a pilha de lítio de segurança todos os três ou quatro anos.

Nota: Para evitar a perda de informações guardadas na memória, tem de desligar a TI-89 Titanium / Voyage 200. Não remova as pilhas alcalinas e a pilha de lítio em simultâneo.

Efeitos da substituição das pilhas

Se não remover ambos os tipos de pilhas em simultâneo nem as deixar acabar completamente, pode mudar ambos os tipos de pilhas sem perder dados da memória.

Precauções com as pilhas

Tome estas precauções quando substituir as pilhas:

- Não deixe as pilhas ao alcance das crianças.
- Não misture pilhas novas com pilhas usadas. Não misture marcas (ou tipos) de pilhas.
- Não misture pilhas recarregáveis com pilhas não recarregáveis.
- Instale as pilhas de acordo com os diagramas de polaridade (+ e).
- Não coloque pilhas não recarregáveis num carregador de pilhas.
- Elimine as pilhas usadas imediatamente.
- Não incinere nem desmonte as pilhas.

Substituir as pilhas alcalinas da TI-89 Titanium

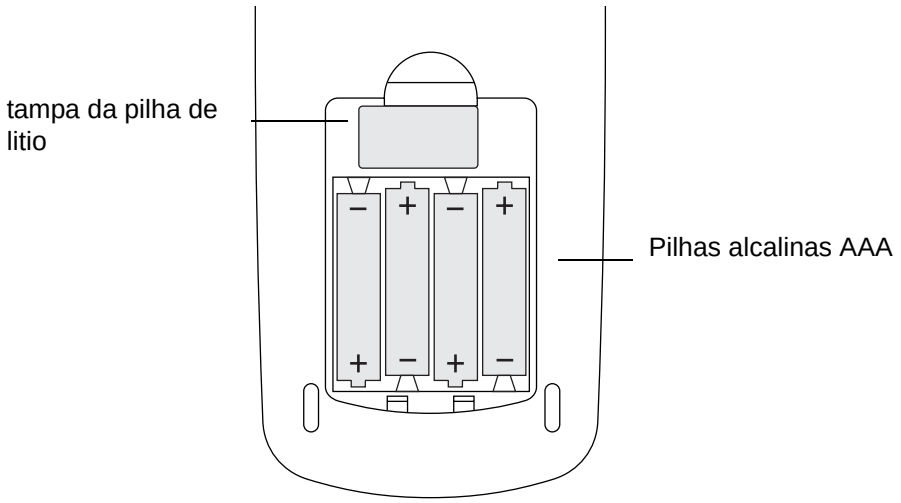
1. Se a TI-89 Titanium estiver ligada, desligue-a (prima [2nd] [OFF]) para evitar a perda de informações guardadas na memória.

2. Faça deslizar a tampa de protecção do teclado e vire o dispositivo ao contrário.
3. Empurre a lingueta da tampa do compartimento das pilhas e, em seguida, puxe para remover a tampa.
4. Remova as quatro pilhas AAA descarregadas.
5. Instale quatro pilhas alcalinas AAA novas, dispostas de acordo com o diagrama de polaridade (+ e -) existente no interior do compartimento de pilhas.
6. Reinstale a tampa do compartimento de pilhas, inserindo as duas patilhas nas duas ranhuras no fundo do compartimento de pilhas, e empurre a tampa até encaixar a lingueta.

Substituir a pilha de lítio na TI-89 Titanium

Para substituir a pilha de lítio de segurança, remova a tampa do compartimento de pilhas e desaparafuse o parafuso que fixa a tampa BACK UP BATTERY.

Remova a pilha antiga e instale uma pilha WR44SW ou 303 nova, lado positivo (+) virado para cima. Reinstale a tampa e o parafuso.



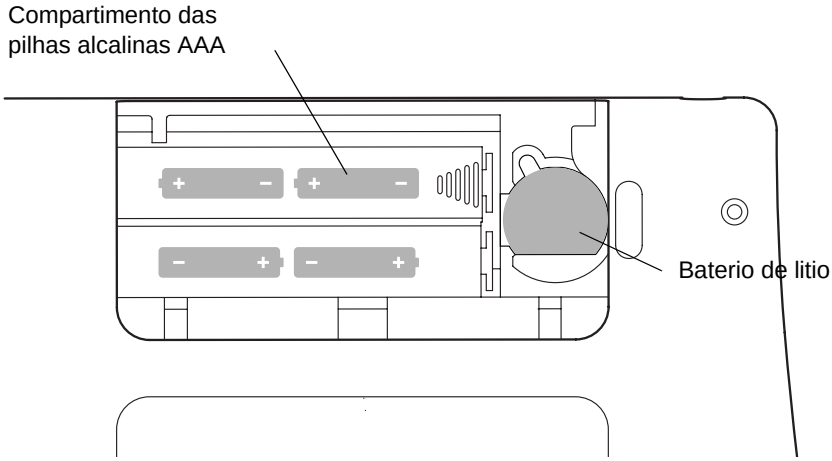
Substituir as pilhas alcalinas na Voyage 200

1. Se a Voyage™ 200 estiver ligada, desligue-a (prima 2 ®) para evitar a perda de informações guardadas na memória.
2. Faça deslizar a tampa de protecção do teclado e vire o dispositivo ao contrário.
3. Prima a tampa do compartimento de pilhas denteada e faça-a deslizar para fora do dispositivo.
4. Remova as quatro pilhas AAA descarregadas.
5. Instale quatro pilhas alcalinas AAA novas, dispostas de acordo com o diagrama de polaridade (+ e -) existente no interior do compartimento de pilhas.
6. Faça deslizar a tampa do compartimento de pilhas para o compartimento, começando pelas patilhas. Empurre a tampa até encaixar as patilhas.

Substituir a pilha de lítio na Voyage 200

Para substituir a pilha de lítio de segurança, remova a tampa do compartimento de pilhas. Introduza um objecto não afiado na indentação circular junto à pilha. Coloque um dedo na pilha de lítio e remova a pilha.

Faça deslizar uma pilha CR1616 ou CR1620 nova, lado positivo (+) virado para cima. Prima para encaixar a pilha de lítio nova.



Em caso de dificuldade

Se tiver dificuldades em utilizar a TI-89 Titanium / Voyage™ 200, as sugestões seguintes podem ajudá-lo a corrigir o problema.

Se:		Acção sugerida:
Não conseguir ver nada no ecrã.		Prima ◆ + para escurecer ou ◆ = clarear o contraste do ecrã.
Aparecer o indicador BATT.		Substitua as pilhas. Se BATT aparecer com o texto ao contrário (BATT), substitua as pilhas assim que for possível.
Aparecer o indicador BUSY.		Está um cálculo em progresso. Se quiser parar o cálculo, prima ON.
Aparecer o indicador PAUSE.		Um gráfico ou programa estiver interrompido e a TI-89 Titanium / Voyage 200 estiver à espera de uma introdução; prima ENTER.
Aparecer uma mensagem de erro.		Consulte a lista de mensagens de erro neste módulo. Prima ESC para apagar.
A TI-89 Titanium / Voyage 200 não estiver a trabalhar correctamente.		Prima várias vezes ESC para sair de qualquer menu ou caixa de diálogo e devolver o cursor à linha de introdução. — ou — Certifique-se de que as pilhas estão instaladas correctamente e são novas.

Se:		Acção sugerida:
A eq TI symbol 45 -89 Titanium parecer estar “bloqueada” e não responder à introdução do teclado.		A acção seguinte apaga a RAM. Esta acção apaga todos os dados, programas e variáveis definidas pelo utilizador, funções ou pastas. Prima sem soltar ◀, ▶ e [2nd]. Prima e liberte [ON]. A acção seguinte apaga a RAM *e a* Flash ROM. Esta acção apaga todos os dados, programas, variáveis definidas pelo utilizador, funções, pastas, aplicações Flash e o arquivo de dados do utilizador. 1. Remova uma das quatro pilhas AAA. 2. Prima sem soltar [(-)] e [)] quando reinstalar a pilha. 3. Continue a premir [(-)] e [)] durante cinco segundos antes de libertar.
A Voyage™ 200 parecer estar “bloqueada” e não responder à introdução do teclado.		A acção seguinte apaga a RAM. Esta acção apaga todos os dados, programas e variáveis definidas pelo utilizador, funções ou pastas. Prima sem soltar [2nd] e [⌫]. Prima e liberte [ON]. A acção seguinte apaga a RAM *e a* Flash ROM. Esta acção apaga todos os dados, programas, variáveis definidas pelo utilizador, funções, pastas, aplicações Flash e o arquivo de dados do utilizador. 1. Remova uma das quatro pilhas AAA. 2. Prima sem soltar [(-)] e [)] quando reinstalar a pilha. 3. Continue a premir [(-)] e [)] durante cinco segundos antes de libertar.

Apëndice C: Guia do Programador

As cadeias de caracteres de parâmetro/modo usadas nas funções reinModo( ), obtModo( ), reinGráf( ) e reinTab( ) não são traduzidas para outros idiomas quando usadas em um programa. Por exemplo, quando você escreve um programa no modo Idioma em francês e depois muda para o modo Idioma em italiano, o programa produzirá um erro. Para evitar esse erro, você precisa substituir os caracteres alfabéticos por dígitos. Esses dígitos funcionam em todos os idiomas. Este apêndice contém os dígitos que substituem as cadeias de caracteres.

Os exemplos a seguir mostram como substituir os dígitos na função reinModo( ).

Exemplo 1: Um programa usando cadeias de caracteres alfabéticas em parâmetro/modo:

```
reinModo("Gráfico","Sequence")
```

Exemplo 2: O mesmo programa, substituindo as cadeias de caracteres por dígitos:

```
reinModo("1","4")
```

reinModo( ) e obtModo( )

Configuração de Parâmetro/Modo	Cadeias de Caracteres
1. TODO	2. 0
Gráfico	3. 1
FUNÇÃO	4. 1
PARAMÉTRICO	5. 2
POLAR	6. 3
SEQUÊNCIA	7. 4
3D	8. 5
EQUAÇÕES DIF	9. 6
Pasta atual	2
FIX 0	1
FIX 1	2
FIX 2	3
FIX 3	4
FIX 4	5
FIX 5	6
FIX 6	7

FIX 7	8
FIX 8	9
FIX 9	10
FIX 10	11
FIX 11	12
FIX 12	13
FLUT	14
FLUT 1	15
FLUT 2	16
FLUT 3	17
FLUT 4	18
FLUT 5	19
FLUT 6	20
FLUT 7	21
FLUT 8	22
FLUT 9	23
Configuração de Parâmetro/Modo	**Cadeias de Caracteres**
FLUT 10	24

FLUT 11	25
FLUT 12	26
10. Ângulo	11. 3
RADIANO	12. 1
GRAU	13. 2
Formt exponencial	14. 4
NORMAL	15. 1
CIENTÍFICA	16. 2
ENGENHARIA	17. 3
Formt complexo	18. 5
REAL	19. 1
RETANGULAR	20. 2
POLAR	21. 3
Formt vetor	22. 6
RETANGULAR	23. 1
CILÍNDRICO	24. 2
ESFÉRICO	25. 3
Impress agrad	26. 7

DESATIV	27. 1
AT	28. 2
Dividir tela	29. 8
COMPLETO	30. 1
SUPER-INFER	31. 2
ESQUERD-DIR	32. 3
Dividir apl 1	33. 9
(as aplicações não são numeradas)	34.
Dividir apl 2	35. 10
(as aplicações não são numeradas)	36.
Nº de gráficos	37. 11
1	38. 1
2	39. 2
Configuração de Parâmetro/Modo	**Cadeias de Caracteres**
40. Gráfico 2	41. 12
FUNÇÃO	42. 1
PARAMÉTRICO	43. 2
POLAR	44. 3

SEQUÊNCIA	45. 4
3D	46. 5
EQUAÇÕES DIF	47. 6
48. Divid tela prop	49. 13
1:1	50. 1
1:2	51. 2
2:1	52. 3
Exato/aprox	53. 14
AUTO	54. 1
EXATO	55. 2
APROXIMADO	56. 3
Base	57. 15
DEC	58. 1
HEX	59. 2
BIN	60. 3

reinGráf( )

Configuração de Parâmetro/Modo	Cadeias de Caracteres
1. Coordenadas	2. 1
RETANG	3. 1
POLAR	4. 2
DESATIV	5. 3
6. Ord gráficos	7. 2
SEQ	8. 1
SIMUL	9. 2
Grade	3
DESATIV	1
AT	2
Eixos	4
No modo 3D:	
DESATIV	1
EIXOS	2
CAIXA	3
Não no modo 3D:	

DESATIV	1
AT	2
Cursor Princ	5
DESATIV	1
AT	2
Etiquetas	6
DESATIV	1
AT	1
Eixos Seq	7
TEMPO	1
TEIA	2
PERSON	3
Método de solução	8
RK	1
EULER	2
Configuração de Parâmetro/Modo	**Cadeias de Caracteres**
Campos	9
CMPINCL	1
CMPDIR	2
CMPDSTV	3

Eixos DE	10
TEMPO	1
Y1-VS-Y2	2
T-VS-Y'	3
Y-VS-Y'	4
Y1-VS-Y2'	5
Y1'-VS-Y2'	6
Estil XR	11
ESTRUT ARAME	1
SUPERF OCULTA	2
NÍVEIS CONTORNO	3
ARAME/CONTORNO	4
DESENHO IMPL	5

reinTab( )

Configuração de Parâmetro/Modo	Cadeias de Caracteres
1. Gráfico <-> Tbl	2. 1
DESATIV	3. 1
AT	4. 2
5. Independente	6. 2
AUTO	7. 1
PERGUNTAR	8. 2
Eixos	4

Apêndice D: Informações gerais

Informações sobre a Assistência aos Produtos e a Garantia TI

Apoio técnico e manutenção	Para obter apoio técnico relativamente a produtos Texas Instruments, incluíndo informações de uso e/ou manutenção/assistência técnica, por favor contacte-nos, E-mail: ti-cares@ti.com ou visite: education.ti.com
Garantia do produto	Para conhecer melhor os termos e a cobertura da garantia desta produto, por favor consulte o Termo de Garantia que o acompanha ou contacte o distribuidor/revendedor Texas Instruments mais próximo.

Precauções com as pilhas

Tome estas precauções quando substituir as pilhas.

- Não deixe as pilhas ao alcance das crianças.
- Não misture pilhas novas e velhas. Não misture marcas (ou tipos nas marcas) das pilhas.
- Não misture pilhas recarregáveis e normais.

- Instale as pilhas de acordo com os esquemas de polaridades (+ e -).
- Não coloque pilhas normais num carregador de pilhas.
- Coloque as pilhas usadas no lixo.
- Não incinere nem desmonte as pilhas.

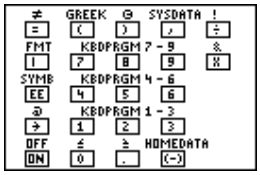
Teclas de atalho da TI-89 Titanium

Geral

◆ APPS	Lista de aplicações Flash
2nd [+⊞]	Alternar entre as duas últimas aplicações escolhidas ou telas divididas
◆ −, ◆ +	Clarear ou escurecer o contraste
◆ ENTER	Calcular o resultado aproximado
◆ ⊖, ◆ ⊕	Mover o cursor para o topo ou para a base (nos editores)
↑ ⊖, ↑ ⊕	Rolar os objetos altos do histórico
↑ ⊖, ↑ ⊕	Realçar à esquerda ou à direita do cursor
2nd ⊖, 2nd ⊕	Página para cima ou página para baixo (nos editores)
2nd ⊖, 2nd ⊕	Mover o cursor para a extrema esquerda ou para a extrema direita

Mapa do teclado na tela ◆ EE

Pressione ESC para sair do mapa.



O mapa do teclado exibe atalhos que não estão indicados no teclado. Pressione ◆, em seguida, a tecla apropriada, como mostrado abaixo.

◆ =	≠
◆ (	Acessar as letras gregas (veja próxima coluna)
◆)	◉ (comentário)
◆ ,	Copiar coordenadas do gráfico em sysdata
◆ ÷	! (fatorial)
◆ !	Exibir a caixa de diálogo FORMATS
◆ 1 – ◆ 6	Executar os programas de kbdprgm1( ) a kbdprgm9( )
◆ ×	& (anexar)
◆ EE	Mapa do teclado na tela
◆ STO▸	@
◆ ON	Desligar a unidade para que ela retorne à aplicação atual na próxima vez em que for ligada
◆ 0 (zero)	≤
◆ .	≥
◆ (-)	Copiar coordenadas do gráfico para o histórico da tela principal

Regras para teclas alfabéticas

alpha	Digitar uma letra minúscula
↑	Digitar uma letra maiúscula
2nd [a-lock]	Alpha-lock de minúsculas
↑ alpha	Alpha-lock de maiúsculas
alpha	Sair de alpha-lock

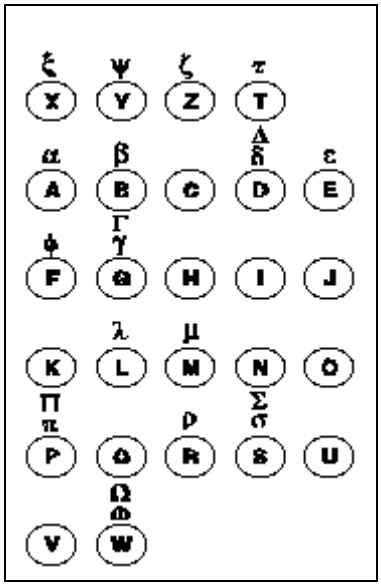
Para representação gráfica em 3D

⊖, ⊕, ⊖, ⊕	Animar gráfico
+, −	Alterar a velocidade da animação
X, Y, Z	Visualizar ao longo do eixo
0	Retornar à vista original
I	Alterar o estilo do formato do gráfico
×	Vista expandida/normal

Letras gregas

◆ (	Para acessar o conjunto de caracteres grego.
◆ (alpha + letter	Para acessar as letras gregas minúsculas. Exemplo: ◆ (alpha [W] exibe w.
◆ (↑ + letter	Para acessar as letras gregas maiúsculas. Exemplo: ◆ (Z [W] exibe W

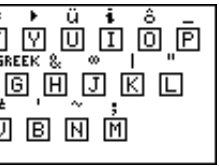
Se for pressionada uma combinação de teclas que não acesse uma letra grega, será obtida a letra normal daquela tecla.



Copiar coordenadas gráficas para o histórico da tela principal
Criar uma nova variável
Abrir uma variável existente
Salvar cópia como
Escurecer/clarear contraste
Calcular a resposta aproximada
Desligar a unidade para que ela volte para a aplicação atual na próxima vez que for ligada
Executar os programas kbdprgm1() a kbdprgm6()

clado na tela (◆ [KEY])

ara sair do mapa.



a abaixo os atalhos que não estão marcados
vage 200. Consulte na próxima coluna as
tos e as letras gregas.

© (commentario)
≠
≤
≥

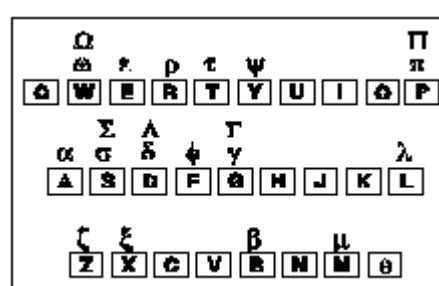
Letras acentuadas e cedilha

2nd A + letra	à, è, ì, ò, ù, À, È, Ì, Ò, Ù
2nd C + letra	ç, Ç
2nd E + letra	á, é, í, ó, ú, ý, Á, É, Í, Ó, Ú, Ý
2nd N + letra	ã, ñ, õ, Ã, Ñ, Õ
2nd O + letra	â, ê, î, ô, û, Â, Ê, Î, Ô, Û
2nd U + letra	ä, ë, ï, ö, ü, ÿ, Ä, Ë, Ï, Ö, Ü

Letras gregas

2nd G	Para acessar o conjunto de caracteres gregos
2nd G + letra	Para acessar as letras gregas minúsculas. Exemplo: 2nd G W exibe
2nd G ↑ + letra	Para acessar letras gregas maiúsculas. Exemplo: 2nd G ↑ W exibe

Se você pressionar uma combinação de teclas que não dê acesso a uma letra grega, você obterá a letra normal para aquela tecla.



nusculas	2nd [a-lock]	
letras minusculas	alpha	
aiusculas	↑ [a-lock]	2nd [CAPS]
etras maiusculas	alpha	2nd [CAPS]
UNÇÃO		
	2nd [F6]	F6
	2nd [F7]	F7
	2nd [F8]	F8
tos para cima ou para baixo no histórico	↑ ⊕, ↑ ⊖	⎙ ⊕, ⎙ ⊖
para a extrema direita ou esquerda da linha	2nd ⊲, 2nd ⊳	2nd ⊲, 2nd ⊳
onal	⊕ and ⊲ ⊕ and ⊳ ⊖ and ⊲ ⊖ and ⊳	⊕ and ⊲ ⊕ and ⊳ ⊖ and ⊲ ⊖ and ⊳
pal	HOME	◆ [CALC HOME]
	◆ [CUT]	◆ X
	◆ [COPY]	◆ C
	◆ [PASTE]	◆ V
	CATALOG	2nd [CATALOG]
dialogo das unidades	2nd [UNITS]	◆ [UNITS]
	2nd [SIN]	SIN
	2nd [COS]	COS
	2nd [TAN]	TAN

o	◆ [EE]	◆ [KEY]
no histórico da tela principal	◆ [(-)]	◆ H
, ú)	2nd [CHAR] 5	2nd A a, e, i, o, u
	2nd [CHAR] 5 6	2nd C c
, ú, ý)	2nd [CHAR] 5	2nd E a, e, i, o, u, y
	2nd [CHAR] 5 6	2nd N a, n, o
e, í, ó, ú)	2nd [CHAR] 5	2nd O a, e, i, o, u
, ú, ý)	2nd [CHAR] 5	2nd U a, e, i, o, u, y
rrogação)	2nd [CHAR] 3	2nd Q
	2nd [CHAR] 5 6	2nd S
direta)	2nd [CHAR] 3	2nd T
	◆ [×] (times)	2nd H
	◆ [STO▸]	2nd R
"diferente")	◆ [=]	2nd V
	◆ [÷]	2nd W
circulado)	◆ [)] ◉	2nd X ©
	F1 3	◆ N
	F1 1	◆ O
mo	F1 2	◆ S
o do formato	◆ [I]	◆ F

Índice remissivo

Símbolos

→, armazenar 966
!, factorial 80
!, fatorial 960
→, guardar 636
", notação de segundos 964
≠, /=, diferente de 644, 959
#, conversão indirecta 642
#, conversão indireta 962
√( ), raíz quadrada 961
%, porcentagem 958
&, anexar 642, 960
', notação de minutos 964
', primo 964
Σ( ), soma 288
Σ( ), somatório 962
∫( ), integrar 86, 263, 265, 266, 271, 288, 290, ... 960
*, multiplicar 955
∫f(x)dx (ferramenta matemática gráfica) . 355, ... 359
+, adicionar 954
F1–F8 (teclas de função)
localização FÌsica 12
mover entre os menus das barras de ferramentas 59
seleccionar categorias 32, 35
seleccionar menus 53
utilizações 16
°, notação de graus 818, 963
°-, notação de graus 964
∠, ângulo 963
.*, multiplicação de ponto 957
.+, adição de ponto 957
./, divisão de ponto 957
.^, potência de ponto 957
.−, subtração de ponto 957
/, dividir 956
<, menor que 644, 959
<<...>>, memória do visor insuficiente . . 252
=, igual 644, 958
>, maior que 644, 959
(-) (sinal de menos posicional) 17
@, inteiro arbitrário 299
Δlist( ), listar a diferença 890
Δtbl, tabela incrementar 490
ΔtmpCnv( ), conversão de intervalo de temperatura 309, 944
Δvariável
x de janela 349, 350
y de janela 349, 350
^, potência 956
_, sublinhado 964
≤, <=, menor ou igual que 644, 959
− (sinal de menos operacional) 17

≥, >=, maior ou igual que 644, 960
◉, comentário 622, 967
|, with 84, 91, 262, 272, 966
[←] / [◆] [DEL] (apagar carácter) 20
[✋] (tecla modificadora Hand)
descrição 16
estado 43
localização física 12
[2nd] (tecla modificadora Second)
descrição 15
estado 43
localização física 12
[2nd] [MEM] (MEMORY) 21
[2nd] [EE] (tecla de expoente) 18
[2nd] [▶] (conversões de medições) . . . 20
[2nd] [_] (unidades de medição) 20
[2nd] [RCL] (rechamar) 21
[2nd] [CATALOG] (Catálogo)
comando de tecla 21
comandos 26
descrição 25
sair 28
[2nd] [CUSTOM] (Custom)
comando de tecla 20
descrição 60
exemplo 61
[2nd] [CHAR] (Carácter)
comando de tecla 21
introduzir caracteres especiais . . . 12
seleccionar caracteres 12
[2nd] [QUIT]
desligar a calculadora 7
ecrã inicial da calculadora 28
introduzir comandos 26
sair do modo Split Screen 69
▸, conversão 306
▸, converter 965
▸Bin, exibir como binário 837
▸Bin, mostrar como binário 728
▸Cylind, exibir como vetor cilíndrico . . 854
▸DD, exibir como ângulo decimal . . . 856
▸Dec, exibir como decimal inteiro . . . 857
▸Dec, mostrar como decimal inteiro . . 728
▸DMS, exibir como grau/minuto/segundo 863
▸Hex, exibir como hexadecimal 882
▸Hex, mostrar como hexadecimal . . . 728
▸Polar, exibir como vetor polar 908
▸Rect, exibir como vetor retangular . . 917
▸Sphere, exibir como vetor esférico . . 936
[↑] (tecla modificadora Shift)
descrição 15
estado 43
localização física 12
[◆] (tecla modificadora Diamond)
descrição 15
estado 43
localização física 12
[◆] F (FORMATS/GRAPH FORMATS) 20
[◆] N (novo ficheiro) 20
[◆] O (abrir ficheiro) 20
[◆] S (SAVE COPY AS)

caixa de diálogo 20
descrição 20
exemplo 58
STO▸ tecla (store) 21
◀▶▲▼ (teclas de cursor)
abrir Apps 32
função 17
funções adicionais 17
introduzir comandos 27
localização física 12
seleccionar pares de entradas/ respostas 30
utilizada com a tecla Hand 16
utilizar o menu CHAR 12
E, expoente 866
-, negar 175, 958
−, subtrair 955
Π( ), produto 289, 962
θmax variáveis de janela 369
θmin variáveis de janela 369
θstep variáveis de janela 369
r, radiano 963
T, transpor 939

Numéricos

009AppB, página = 574 1004
0b, indicador binário 967
0h, indicador hexadecimal 967
10^( ), potência de dez 965
▸Grad( ) 881
▸ln( ) 890
▸logbase( ) 893
▸Rad( ) 916

A

Abrir ficheiro (◆ O) 20
abs( ), valor absoluto 821, 834
acção indirecta, # 998
Acentos
comandos das teclas 15
menu CHAR 21
actividades. *Consulte* exemplos, visualizações, actividades
actualizar sistema operativo (SO) . . 783, 784, ... 785
Adaptador de vídeo TI-Presenter
ligar 73
adicionar, + ... 954
Aleatória
matriz, randMat( ) 799
semente do número, RandSeed . 799
aleatória
norma, randNorm( ) 916
aleatório
número, rand( ) 916
polinômio, randPoly( ) 917
semente do número, RandSeed . 917
alternar, switch( ) 658, 939
Ambiente de trabalho das Apps
arranque inicial 2
categorias 32, 37
data e hora 48

desligar 44
desligar a calculadora 7
ecrã inicial da calculadora e 28
modo 22, 44
relógio 46
ampliar/concatenar, augment( ) 799, 837
and, e Booleano 273, 645, 732, 834
AndPic, e imagem gráfica 670, 835
anexar, & 642, 960
Angle mode 206
angle( ), ângulo 835
ângulo, ∠ 963
ângulo, angle( ) 835
ans( ), última resposta 242, 836
apagar
desenho, ClrDraw . . . 527, 672, 840
erro, ClrErr 678, 840
gráfico, ClrGraph 510, 668, 724, 841
home, ClrHome 841
I/O, ClrIO 616, 661, 841
Apagar carácter (← / ◆ [DEL]) 20
APD (Automatic Power Down) 162
durante o cálculo ou o programa . . 8
ligar após 7
no modo de transferência do sistema operativo 75
APD (Automatic Power Down) feature
during calculation or program 8
Aplicações de software portáteis (Apps)
Ícones 2
Aplicações Flash 738
eliminar 777
aplicações Flash 200, 214, 296
approx( ), aproximar 280, 836
Apps (Aplicações de software portáteis)
abrir 32, 63
apagar 71
atalhos 37
comutar 69
Ícones 2
Apps desktop
split-screen status 41
apresentação do ângulo 417
aproximar, approx( ) 280, 836
Arc (ferramenta matemática gráfica) 356, 361, ... 372
Archive, arquivar variáveis 636, 758, 836
arcLen( ), comprimento do arco 289, 836
área de transferência 234, 236, 697
área do histórico 228, 229, 710
Área do histórico, estado 44
armazenamento
base de dados gráficos, StoGDB . . . 545, ... 670
imagem gráfica, StoPic 670
armazenando
banco de dados de gráfico, StoGDB 937
imagem gráfica, StoPic 937
símbolo, → 966
arquivar variáveis, Archive 636, 758, 836
Arranque inicial 2

arredondar, round( ) 920
asymptotes, faux, detecting . . . 101, 122
augment( ), ampliar/concatenar 799, 837
auto independente/perguntar,
Independent AUTO/ASK . .491, 494, 499
Automatic Power Down (APD)
durante o cálculo ou o programa . . 8
ligar após 7
no modo de transferência do sistema operativo 75
Automatic Power Down (APD) feature
during calculation or program 8
avgRC( ), taxa de câmbio normal . . . 837
aviso, Prompt( ) 660, 911

B

Backspace (⇦) 20
barra de ferramentas
activada, CustmOn 253
ativada, CustmOn 853
definir, Custom 662
desactivada, CustmOff 253
desativada, CustmOff 852
barra de ferramentas personalizada
Consulte barra de ferramentas
bases numéricas 727
conversões 728
operações Booleanas 732
operações matemáticas . . .729, 730, 731
binário
deslocar, shift( ) 734
exibir, ▸Bin 837
girar, rotate( ) 734
indicador, 0b 967
mostrar, ▸Bin 728
BldData, montar dados . . .473, 636, 838
bloquear variável, Lock 637, 892
Booleano
e, and 273, 645, 732, 834
não, not 645, 732, 901
ou exclusivo, xor 645, 732, 948
ou, or 645, 732, 903
Box Plot 597
Build Web, construir rede .391, 392, 393
BUSY, indicador 341

C

Cabo 70
instalar Apps 70
TI Connectivity 70
Cabo de ligação de unidade a unidade 75
Cabo de unidade a unidade, ligar 73
Cabo TI Connectivity 766, 785, 789
Cabos 72
cabos 766, 785, 789
cadeias
acção indirecta, # 998
introduzir, InputSt 780
cadeias de caracteres
anexar, & 642, 960

cadeia de caracteres para expressão, expr( ) 871
cadeia de caracteres, char( ) . . . 839
char( ) 642
código de carácter, ord( ) 643
código de caráter, ord( ) 904
conversão indirecta, # 642
conversão indireta, # 962
dentro, InString 642, 884
deslocar, shift( ) 643, 929
dimensão, dim( ) 642, 861
direita, right( ) 643, 919
esquerda, left( ) 643, 887
expressão para cadeia de caracteres, string( ) 643, 937
formatar, format( ) . . . 642, 661, 875
girar, rotate( ) 643, 919
introdução, InputSt 641, 660
introduzindo, InputSt 884
operações 640, 642, 833
para expressão, expr( ) 641, 642, 661
semi-cadeia de caracteres, mid( ) . . 643, ... 896
caixa
de diálogo, definir, Dialog 861
Caixa de diálogo
◆ S (SAVE COPY AS) 20
CLOCK 46
Edit Categories 37
FORMATS 112, 119
FORMATS/GRAPH FORMATS (◆ F) 20
indicador do menu 57
MODE 21
para abrir as Apps 32
caixa de diálogo 197
definir, Dialog 662
FORMATS 338, 424, 426, 428, 437, 702
caixa zoom, ZoomBox 349
Calcular 83
calcular polinômio, polyEval( ) 908
calcular, factor( ) 798, 828
Calculator-Based Laboratory. *Consulte* CBL
Calculator-Based Ranger. *Consulte* CBR
Cálculo
actividade 827
campo
de direcção, DIRFLD 447, 455, 456, 484
de inclinação, SLPFLD . . . 447, 455, 456, ... 482, 483
desactivado, FLDOFF 447, 455, 457, 486
Caracteres
apagar 20
especiais 12, 15, 21
gregos 12, 21
internacionais/acentuados . . . 12, 21
maiúsculos 12, 15

matemáticos 12, 21
caracteres
acentuados 700, 702, 704
cadeia de caracteres, char( ) . . . 642, 839
código numérico, ord( ) . . . 643, 904
especiais 174, 700, 703
gregos 702, 705, 706
maiúsculas/minúsculas . . . 172, 693
menu 195
símbolos 703
Catalog (2nd [CATALOG])
comando de tecla 21
comandos 26
descrição 25
sair 28
Categoria
All 35
English 36
exemplo de edição 38
Graphing 36
Math 36
Organizr 36
personalizar 37
Science 36
seleccionar 35
seleccionar vazio 37
SocialSt 36
CBL
actividade 814
dados estatísticos 609, 610
enviar variável de lista, Send . . . 922
obter/retornar, Get 876
programas 674, 814
CBR
dados estatísticos 609, 610
enviar variável de lista, Send . . . 922
obter/retornar, Get 876
programas 674, 814
ceiling( ), limite superior 802, 838
certificado 777, 783, 784, 785, 787, 788, 789
cFactor( ), factor complexo 280
cFactor( ), fator complexo 839
cFactor(†), factor complexo 992
chamar
banco de dados de gráficos, RclGDB 917
base de dados gráficos, RclGDB . . . 545, ... 670
imagem gráfica, RclPic 670, 917
char( ), cadeia de caracteres . . . 642, 839
checkTmr( ), verificar temporizador . . 839
ciclo, Cycle 853
Circle, desenhar circunferência . 673, 840
círculo, desenho 531
circunferência, Circle 673, 840
classificando
ascendente, SortA 935
descendente, SortD 936
ClockOff, desligar relógio 840
ClockOn, ligar relógio 840

ClrDraw
apagar desenho 672
eliminação de desenho 527
ClrDraw, apagar desenho 840
ClrErr, apagar erro 678, 840
ClrGraph
apagar gráfico 668, 724
eliminação de gráfico 510
ClrGraph, apagar gráfico 841
ClrHome, apagar tela principal 841
ClrIO, apagar I/O 616, 661, 841
colagem 234, 236, 237, 697
automática 234, 243, 244
Colar automático 30
colDim( ), dimensão da coluna matriz 842
coleção trigonométrica, tCollect( ) . . 280, 942
colNorm( ), coluna matriz norm 842
Comandos
Flash Apps 25
Tecla 12, 13
Comandos das teclas
caracteres especiais 12
mapa do teclado 13
combinações, nCr( ) 898
comDenom( ), denominador comum 280, 281, ... 286, 843
comentário, ◉ 622, 967
complexa, solução, cSolve( ) 263
Complexo
factor, cFactor( ) 828
números 79
complexo
factor, cFactor(†) 992
modo, Complex Format (Formato complexo) 977
resolver, cSolve(†) 992
superfície de módulo 433
tabelas 498
zeros, cZeros(†) 992
comprimento do arco, arcLen( ) 289, 836
Condições de erro após APD 8
condições iniciais 454
conectando e transmitindo 923
enviar variável de lista, Send . . . 922
obter/retornar valor de CBL/CBR, Get 876
configurar tabela, setTable( ) 493
conj( ), conjugação complexa 843
conjugação, complexa, conj( ) 843
Constant Memory 162
constantes 302
predefinidas 315
construir rede, Build Web . 391, 392, 393
Contraste
ajustar 76
contraste, ajuste 163
conversão
▸ 306
de temperatura, tmpCnv( ) 943
do intervalo de temperatura, ΔtmpCnv( ) 944

indirecta, # 642
indireta, # 962
intervalo de temperatura, ΔtmpCnv( ) 309
temperatura, tmpCnv( ) 308
Conversões de medições (2nd [▶]) . . . 20
converter hora, timeCnv( ) 943
Converter medições 20
converter, ▶ 965
coordenada x retangular, P▶Rx( ) . . . 905
coordenada y retangular, P▶Ry( ) . . . 905
cópia 234, 235, 236, 237, 697
copiar variável, CopyVar . 636, 752, 844
CopyVar, copiar variável 752, 844
corte 234, 235, 236, 697
cos( ), cosseno 844
cos⁻¹( ), arco-cosseno 845
cosec( ), co-secante 847
cosec⁻¹( ), co-secante inversa 847
co-secante hiperbólica, cosech( ) . . . 848
co-secante, cosec( ), ... 847
cosech( ), co-secante hiperbólica . . . 848
cosech⁻¹( ), co-secante hiperbólica inversa 850
cosh( ), cosseno hiperbólico 846
cosh⁻¹( ), arco-cosseno hiperbólico . . 846
cosseno, cos( ) 844
cotan( ), co-tangente 846
cotan⁻¹( ), co-tangente inversa 846
co-tangente hiperbólica, cotanh( ) . . 847
co-tangente, cotan( ), ... 846
cotanh( ), co-tangente hiperbólica . . . 847
cotanh⁻¹( ),co-tangente hiperbólica inversa 847
cover
removing 5
replacing 5
crossP( ), entre produtos 847
cSolve( ), solução complexa . . . 263, 850
cSolve(†), resolver complexo 992
CubicReg, regressão cúbica . . 587, 852, 1000
cumSum( ), soma cumulativa . . 575, 852
Cursor
apagar caracteres 20
apagar um par de entrada/resposta . 32
funcionalidade 17
localização a seguir APD 8
mover 17
na área do histórico 30
seleccionar um comando 27
ver entradas 30
cursor
deslocação 191
fora da curva 416
gráfico 3D 411
movendo 167
movimento livre . 342, 371, 379, 389, 410, ... 452
superfície oculta 414
traçado 344, 345, 346, 347, 348

CustmOff, barra de ferramentas do menu personalizado desactivada 253
CustmOff, personalizar barra de ferramentas desativada 852
CustmOn, barra de ferramentas do menu personalizado activada 253
CustmOn, personalizar barra de ferramentas ativada 853
CUSTOM
eixos (sucessão) 391
gráficos personalizados 384
menu 253
Custom, definir barra de ferramentas . . . 662, ... 853
Cycle, ciclo 853
CyclePic
repetição da imagem gráfica . . . 540
repetir imagem gráfica 670
CyclePic, repetir imagem gráfica . . . 853
cZeros( ), raízes complexas . . .263, 280, 854
cZeros(†), zeros complexo 992

D

d( ), primeira derivada 85, 271, 288, 290, 856
dados (novos), NewData 898
dados do sistema, sysData . . . 504, 505
Data
definir 45
reiniciar 53
Data/Matrix Editor 504
data▸mat( ) 856
dayOfWk( ), dia da semana 856
decimal
exibir como ângulo, ▸DD 856
exibir como inteiro, ▸Dec 857
decimal, mostrar como inteiro, ▸Dec . 728
decompor em factores 282
factor( ) 263, 279, 282
Define, definir . 294, 376, 385, 444, 480, 857
definição Axes 417, 425
definidas pelo utilizador, unidade . . . 312
Definir
pasta, setFold( ) 747
definir
barra de ferramentas, Custom . . 853
barra de ferramentas, Toolbar . 662, 944
Define 246, 294, 331, 385, 406, 444, 480, . . .508, 514, 631, 636, 668
gráfico, setGraph( ) . . .657, 669, 924
modo, setMode( ) 658, 669, 925
pasta, setFold( ) 657, 924
tabela, setTable( ) 658, 668, 926
unidades, setUnits( ) 658, 927
definir data, setDate( ) 924
definir formato de data, setDtFmt( ) . 924
definir formato de hora, setTmFmt( ) . 926
definir fuso horário, setTmZn( ) 927
definir hora, setTime( ) 926

definir, Define 794, 857
deleting
variables of type 754
DelFold, excluir pasta 636, 754, 858
DelType 636, 858
DelType command 754
DelVar, excluir variável . . .293, 636, 640, 754, ... 858
denominador 843
denominador comum, comDenom( ) 280, 281, ... 286, 843
dentro da cadeia de caracteres, inString( ) 642, 884
Derivadas 85
primeira derivada, d( ) 85
derivadas
derivadas numéricas, nDeriv( ) .289, 898
primeira derivada, d( ) 271, 288, 290, 856
Derivatives (ferramenta matemática gráfica) 355, 359, 372, 380
desarquivar variáveis, Unarchiv 637, 758, ... 945
desbloquear, Unlock 637, 946
desenhos e desenhar
apagando, ClrDraw 840
círculos 531
circunferência, Circle 673, 840
contorno, DrwCtour 674, 866
curva, DrawSlp 533, 673, 865
eliminação 529
eliminação, ClrDraw 672
função, DrawFunc 524, 673, 863
inverso, DrawInv 525, 674, 864
linha horizontal, LineHorz . .673, 889
linha tangente, LineTan . . .673, 889
linha vertical, LineVert 673, 889
linha, Line 673, 888
linhas 530, 532
mão livre 528
numa imagem gráfica 671
paramétrica, DrawParm . . 524, 674, 865
Pencil 528
polar, DrawPol 524, 674, 865
desenhos implícitos 1002
Desligar 7
após APD 7
seguir inactividade 7
desligar relógio, ClockOff 840
deslocação 498
deslocar, shift( ) 643, 734, 929
deSolve( ), solução 289, 479, 858
desvio padrão, stdDev( ) 936
det( ), matriz determinante 861
devolver, Return 513
dia da semana, dayOfWk( ) 856
diag( ), matriz diagonal 861
Dialog, definir caixa de diálogo .662, 861
diferente de, ≠, /= 959
diferente de, ≠, /= 644

dim( ), dimensão 642, 861
dimensão, dim( ) 642, 861
direita, right( ) 281, 643, 919
DIRFLD, campo de direcção . .447, 455, 456, ... 484
discontinuities
detecting 101, 122
Disp, exibir tela de I/O 862
Disp, mostrar ecrã de E/S 145
Disp, ver ecrã de I/O 623, 661, 677
DispG, exibir gráfico 862
DispG, ver gráfico 661, 668
DispHome, exibir tela principal 862
DispHome, ver ecrã principal 661
DispTbl, exibir tabela 863
DispTbl, ver tabela 661, 668
Distance (ferramenta matemática gráfica) 356, ... 360, 372, 380
dividir por um inteiro, intDiv( ) . . 731, 885
dividir, / 956
divisão de ecrã 522
dotP( ), produto de ponto 863
DrawFunc
desenhar função 673
desenho de função 524
DrawFunc, desenhar função 863
DrawInv
desenhar inverso 674
desenho de inversa 525
DrawInv, desenhar inverso 864
DrawParm
desenhar paramétrica 674
desenho de paramétrica 524
DrawParm, desenhar paramétrica . . 865
DrawPol
desenhar polar 674
desenho de polar 524
DrawPol, desenhar polar 865
DrawSlp
desenhar curva 673
desenho de curva 533
DrawSlp, desenhar curva 865
DropDown, menu drop-down 866
DropDown, menu pendente 662
DrwCtour, desenhar contorno . 431, 674, 866

E

e (Booleano), and . . .273, 645, 732, 834
E (símbolo de expoente) 18
e imagem gráfica, AndPic 670
E, expoente 866
e^( ), e elevado a uma potência 867
Ecrã ABOUT 70
ecrã dividido . . .518, 546–555, 710, 725
alternar, switch( ) 658
coordenadas de pixel 548
definições 546, 547
linha de entrada 553, 555
saída 550
trocar 553
Ecrã dividido da esquerda para a direita

definir 64
definir aplicação inicial 66
estado 40
Ecrã dividido de cima para baixo
definir 64
definir aplicação inicial 66
estado 40
Ecrã inicial da calculadora
2nd [QUIT] 21
comando da tecla 21
desligar a calculadora 7
introduzir comandos 26
menu Custom 60
menus das barras de ferramentas 53
teclas de função 16
voltar ao ambiente de trabalho das Apps) 45
Ecrã principal 226
edição 191
edição de texto 688
corte, cópia, colagem .234, 235, 236, 237, ... 697
localizar 698
realce 696
Editor de Dados/Matrizes
deslocar, shift( ) 929
editor de dados/matrizes
cabeçalho da coluna . 569, 571, 573
Cálculo automático 572
classificação de colunas . . 575, 576
deslocação 565
largura da célula 567
novos, NewData 637
preenchimento 566
traçados estatísticos . 591, 592, 594, 595
valores 564
eigenvalor, eigVl( ) 867
eigenvetor, eigVc( ) 867
eigVc( ), eigenvetor 867
eigVl( ), eigenvalor 867
eixo (sucessão), CUSTOM 391
else if, ElseIf 513, 649, 868
Else, else 883
else, Else 649, 883
ElseIf, else if 868
encerrar
barra de ferramentas, EndTBar 662, 944
dialog, EndDlog 662
diálogo, EndDlog 861
for, EndFor 623, 652, 874
função, EndFunc 514, 628, 875
if, EndIf 623, 646, 648, 649, 883
personalização, EndCustm .662, 853
programa, EndPrgm 631, 910
repetição, EndLoop 655, 894
tentar, EndTry 945
tentativa, EndTry 678
while, EndWhile 653, 947
encerrar programa, EndPrgm 142

EndCustm, encerrar personalização 662, 853
EndDlog, encerrar diálogo 662, 861
ende 631
EndFor, encerrar for 623, 652, 874
EndFunc, encerrar função 628, 875
EndIf, encerrar if 623, 646, 648, 649, 883
EndLoop, encerrar repetição . . 655, 894
EndPrgm, encerrar programa . .142, 631, 910
EndTBar, encerrar barra de ferramentas 662, ... 944
EndTry, encerrar tentativa 678, 945
EndWhile, encerrar while 653, 947
entrada
  entry( ) 241
  Input 660, 669
entrada, entry( ) 868
entrada, Input 884
entre produtos, crossP( ) 847
entry( ), entrada 868
enviar
  mensagem de bate-papo, SendChat 676, ... 923
  para calculadora, SendCalc 675, 923
  variável de lista, Send 676, 922
enviar conversação, SendChat 780
enviar para a calculadora, SendCalc 780
enviar para calculadora, SendCalc . . 780
EOS (Sistema Operativo de Equações) . 998
equações
  simultâneas, simult( ) 285
  solução 712
equações diferenciais
  condições iniciais 454
  DIRFLD, campo de direcção . . . 447, 455, ... 456, 484
  FLDOFF, campo desactivado . . 447, 455, ... 457, 486
  métodos de solução .446, 473, 1003
  primeira ordem 458, 479
  representações gráficas 441
  segunda ordem 460, 479
  SLPFLD, campo de inclinação . 447, 455, ... 456, 482, 483
  solução de problemas 481
  terceira ordem 464
equações simultâneas, simult( ) 930
erro
  Circular definition 635
  Out-of-memory 296
Erro de memória 763, 764, 765
erros e resolução de problemas . . . 1006
  avisos 976
  transmissão 777, 787
Erros e solução de problemas
  erro de memória 763, 764, 765
erros e solução de problemas
  apagar erro, ClrErr 678, 840
  Circular definition 635
  Out-of-memory 296

passar erro, PassErr 678, 907
programas 676
Escrever
caracteres maiúsculos 12
esquerda, left( ) 281, 643, 887
Estado
bateria com carga baixa 77
ecrã dividido 40
modo APPROX (APROX) 43
modo AUTO 43
modo EXACT (EXACTA) 43
modo Graph 43
modo Graph number 43
no ambiente de trabalho das Apps 3
ocupada/pausa) 44
pasta actual 43
variável bloqueada/arquivada . . . 44
estatísticas 581
Consulte também regressões
Box Plot 597
Calculation Type 583, 584, 586
categorias 606
Category 583, 584
combinações, nCr( ) 898
desvio padrão, stdDev( ) 936
estatísticas de uma variável, OneVar 587, ... 903
fatorial, ! 960
Freq 583, 584
frequência 604
média, mean( ) 895
mediana, median( ) 895
mostrar resultados, ShowStat . . 589, 930
norma aleatória, randNorm( ) . . . 916
novo traçado, NewPlot 597, 900
número aleatório, rand( ) 916
operações 833
permutações, nPr( ) 902
resultado de duas variáveis, TwoVar 587, ... 588, 945
semente do número aleatório, RandSeed 917
traçados . . 591, 592, 594, 595, 596, 597, 598, 599, 600, 601, 602, 603
traçados activados, PlotsOn 333, 669
traçados ativados, PlotsOn 908
traçados desactivados, PlotsOff 333, 669
traçados desativados, PlotsOff . . 908
variância, variance( ) 946
variáveis 585, 589
visão geral 581
Estatísticas, semente do número aleatório, RandSeed 799
estatísticas, traçados
Histogram 598
Scatter 596
xyline 596
estilo, Style 335, 669, 938
etiqueta, Lbl 633, 646, 650, 656, 887
exact( ), exato 869

examples, previews, activities
angle modes 88
detecting discontinuities . . . 101, 122
finding roots 81
log to any base 87
examples,previews,activities
log to any base 87
exato, exact( ) 869
excluindo
pasta, DelFold 858
variável, DelVar 858
Excluir
pasta, DelFold 754
variável, DelVar 754
excluir
pasta, DelFold 636
variável, DelVar 293
exclusão variável, DelVar 636, 640
Exec, executar linguagem assembler 869
Exec, executar linguagem assembly 686
executar
linguagem assembly, Exec 686
programa, Prgm 631
executar linguagem assembler, Exec 869
executar programa, Prgm 142, 910
Exemplo
activar/desactivar o menu Custom (Custom) 61
alterar definições do modo 23
criar um programa novo 33
desligar o relógio 52
editar categorias 38
restaurar o menu Custom (Custom) predefinido 61
seleccionar opções dos menus . . . 55
utilizar as caixas de diálogo 58
utilizar o Catalog 26
utilizar o mapa do teclado 14
utilizar o menu CHAR 13
Exemplos, visualizações, actividades
actividade cos(x)=sin(x) 800
amostragem 829
anuidade padrão 823
árvores e florestas 108
bases numéricas 152
beisebol 817
cálculo de polinómios 83
caminho de uma bola 103
com anuidade padrão 823
com funções racionais 807
com o valor do dinheiro ao longo do tempo 825
constantes 92
de amostragem 829
decomposição de uma função racional 807
derivadas 85
desenhar o gráfico da função 98
do paralelepípedo 802
ecrã dividido 127, 817
editor de dados/matrizes 129
equações diferenciais 117

estatísticas 131
factores complexos 827
factores primos 79, 81
factores racionais 827
factores reais 827
factorial 79, 80
filtragem dos dados 810
fórmula quadrática 795
funções gráficas 95
funções por partes 122
gestão da memória 155
gestão das variáveis 155
integrais 86
manipulação simbólica 89
números complexos 79, 80
operações com texto 146
polar 106
polinómio cúbico 820
população 131
problema do poste e do canto . . 793
programa CBL 814
programação 142
raízes complexas 820
representação gráfica 3D . . 111, 802
representação gráfica paramétrica . 103, ... 817
representação gráfica sequencial 108
resolução de equações lineares . 83, 84
script do tutorial com o editor de texto 804
simplificação de expressões 82
solucionador numérico 149
tabelas 125
Teorema de Pitágoras 793
unidades de medida 92
valor do dinheiro ao longo do tempo . 825
exemplos, visualizações, actividades
decompor polinómios em factores . . 282
equação diferencial de segunda ordem 460, 479
equação diferencial de terceira ordem 464
gráficos de rede divergentes . . . 394, 395
modelo predador-presa 398, 469
oscilar gráficos de rede 397
programação 678, 679, 680, 681, 682
resolver equações lineares 284
sucessão de Fibonacci 401
superfície de módulo complexo . 433
traçados implícitos 439
exibir
gráfico, DispG 862
tabela, DispTbl 863
tela de I/O, Disp 862
tela principal, DispHome 862
exibir como
ângulo decimal, ▸DD 856
binário, ▸Bin 837

decimal inteiro, ▸Dec 857
grau/minuto/segundo, ▸DMS . . . 863
hexadecimal, ▸Hex 882
vetor cilíndrico, ▸Cylind 854
vetor esférico, ▸Sphere 936
vetor polar, ▸Polar 908
vetor retangular, ▸Rect 917
Exit, sair 869
exp▸list( ), lista de expressões 869
expand( ), expandir 279, 282, 870
expandir, expand( ) . 279, 282, 797, 820, 870
expansão trigonométrica, tExpand( ) 280, ... 942
expoente, E 866
expr( ), cadeia de caracteres para expressão 641, 642, 661, 871
ExpReg, regressão exponencial . . . . 587, 871, ... 1000
Expressões 28
simplificar 82
expressões 177, 179, 191
cadeia de caracteres para expressão, expr( ) 641, 642, 661, 871
lista de expressões, exp▸list( ) . . 869
eyeϕ variável de janela do eixo z . . . 407, 417, ... 419
eyeθ variável de janela do eixo x . . . 407, 417, ... 418
eyeψ variável de rotação da janela . 407, 417, ... 419

F

factor complexo, cFactor( ) 280
factor( ), calcular 81, 83
factor( ), decompor em factores 263, 282
factor( ), fatorar 871
factorial, ! 80
família de curvas 515
fator complexo, cFactor( ) 839
fatoração
QR 914
fatorar, factor( ) 871
fatorial, ! 960
Ficheiro, novo (◆ N) 20
Fill, matriz preencher 872
Filtragem dos dados 810
FLASH APPLICATIONS (◆ APPS)
aceder a Apps não listadas 64
descrição 54
se não estiverem instaladas 25
Flash, actualizar sistema operativo . 783, 784, ... 785
FLDOFF, campo desactivado . 447, 455, 457, ... 486
fldpic, imagem do campo 451
floor( ), limite inferior 802, 873
fMax( ), função máxima . . . 263, 289, 873
fMin( ), função mínima 263, 289, 873
FnOff, função desactivada 333, 668
FnOff, função desativada 874
FnOn, função activada 333, 669
FnOn, função ativada 874

For, for 874
for, For 623, 652, 874
forma escalonada de fila reduzida, rref( ) 285, ... 799, 921
forma escalonada de fila, ref( ) 918
format( ), formatar cadeia de caracteres 642, ... 661, 875
formatar cadeia de caracteres, format( ) 642, ... 661, 875
formato gráfico
Axes 339, 448, 468, 469
Coordinates 339, 370
Field 447
Graph Order 339, 446
Grid 339
Labels 339
Leading Cursor 339
Solution Method 446
FORMATS (FORMATOS) (◆ F)
caixa de diálogo 20
comando de tecla 20
Fórmula Bogacki-Shampine 1003
fPart( ), parte da função 875
fração própria, propFrac 911
fracção própria, propFrac . .82, 280, 286, 808
Fracções 807
fracções 280, 286
frações 911
Func, função do programa 514, 628, 875
Funções 25
definidas por partes 122
funções 177
activadas, FnOn 333, 669
ativadas, FnOn 874
definidas pelo usuário 857
definidas pelo utilizador 627
desactivadas, FnOff 333, 668
desativadas, FnOff 874
do programa, Func 514
função do programa, Func . 628, 875
máximas, fMax( ) 263, 289, 873
mínimas, fMin( ) 263, 289, 873
múltiplas declarações 513
partes, fPart( ) 875
representação gráfica 324
simplificação diferida 270
funções definidas
pelo utilização 626
pelo utilizador . . 215, 245, 294, 295, 406, ... 510, 513, 626, 628
por partes 511

G

gcd( ), máximo divisor comum 876
Get, obter/devolver valor de CBL/CBR . . 609, ... 676
Get, obter/retornar valor de CBL/CBR . . 876
GetCalc, obter/devolver à calculadora . . 675
GetCalc, obter/devolver calculadora . 780

GetCalc, obter/retornar da calculadora . 876
getConfg( ), obter/devolver configuração 657
getConfg( ), obter/retornar configuração 877
getDate( ), obter data actual 877
getDenom( ), obter/devolver denominador 281
getDenom( ), obter/retornar denominador 877
getDtFmt( ), obter formato de data . . 878
getDtStr( ), obter cadeia de data . . . 878
getFold( ), obter/voltar à pasta . 636, 657
getKey( ), obter/devolver à tecla 660
getKey( ), obter/retornar tecla 878
getKey(†), obter/devolver tecla . 982, 986
getMode( ), obter/devolverr ao modo 657
getMode( ), obter/retornar modo . . . 879
getNum( ), obter/devolver número . . 281
getNum( ), obter/retornar número . . . 879
getTime( ), obter hora actual 879
getTmFmt( ), obter formato de hora . 879
getTmStr( ), obter cadeia de hora . . 879
getTmZn( ), obter fuso horário 880
getType( ), obter/devolver tipo . 260, 636
getType( ), obter/retornar tipo 880
getUnits( ), obter/devolver unidades . 657
getUnits( ), obter/retornar unidades . 881
girar, rotate( ) 643, 734, 919
Goto, ir para . . . 633, 646, 650, 656, 881
GRAD(gradian) mode 88
Gradian angle mode 88
gradian,[G] 962
gráfico de nível de contorno 1002
Gráficos
desenhar 16
Número de 67, 69
gráficos e fazer gráficos
modos 976
pixéis 995
Gráficos e representação gráfica
circunferência 98
coordenadas 96
gráfico, Graph 122
Minimum 96
traçado 96
traçar, Trace 803, 814, 816, 820
Y= editor 99
gráficos e representação gráfica
∫f(x)dx 355, 359
3D 403
animação 540
apagando, ClrGraph 841
Arc 356, 361, 372
armazenar banco de dados de gráfico, StoGDB 937
bases de dados gráficos 543
chamar banco de dados de gráfico novamente, RclGDB 917
chamar base de dados gráficos novamente, RclGDB 670

coordenadas 342
dados de matriz 504
de tempo 469
definição, setGraph( ) 657, 669
definindo, setGraph( ) 924
Derivatives 355, 359, 380
desenho . . 526, 528, 529, 530, 531, 532, ... 533, 534, 535, 671
Distance 356, 360, 372, 380
divisão de ecrã 522
ecrã dividido 518, 546, 547
ecrã principal 506, 508
eixos personalizados 391
eliminação, ClrGraph 510, 668
equações diferenciais 441
estilo, Style 669, 938
estilos de linha . 333, 368, 377, 386, 406, ... 445
factores do zoom 351, 353
família de curvas 515
formatos 446
funções 324–364
funções aninhadas 511
gráfico, Graph . 331, 508, 516, 669, 882
gráficos de rede 384, 391, 392
gráficos personalizados 384
gráficos simultâneos 517
guardar base de dados gráficos, StoGDB 670
imagens gráficas 535, 537
Inflection 356, 360
Intersection 355, 358
janela de visualização 336, 378
limpar, ClrGraph 724
Maximum 355, 357
Minimum 355, 357
modo de dois gráficos 518, 520, 546, 547
modos 206, 224, 326, 367, 383, 405, ...443
operações 831
panorâmica 347
paramétrica 373
pausa 341
polar 365
programas 668
QuickCenter 348
selecção de funções . .332, 377, 445
seleccionar funcões 385
Shade 356, 362
sombreamento, Shade 674, 928
sucessão 381
Tangent 356, 361, 372, 380
texto 534
traçado 371, 380, 390, 411, 453
traçados de contorno 427, 432
traçados de tempo . . .384, 391, 468
traçados implícitos . . .435, 438, 439
traçados personalizados . . .468, 469
traçar, Trace 344, 345, 346, 347, 348, ...669, 944

Value 355, 356, 372, 380, 390, 411, 453
variáveis de janela . . .336, 369, 378, 386, ... 407
variável independente 506
visão geral . 324, 365, 381, 403, 441
visualização da janela 369, 386, 407
Y= editor . . 328, 368, 375, 384, 406, 444, ... 506
Zero 355, 357
zoom 349, 371, 379, 389, 410, 452, 669
zoom Memory 350, 354
gráficos e representação gráfica, de rede
convergência 394
oscilar 397
WEB 384, 391, 392
gráficos e representação gráfica, funções
324– 364
activadas, FnOn 669
aninhadas 512
ativadas, FnOn 874
definidas por partes 511
desactivadas, FnOff 668
desativadas, FnOff 874
inversas 525
matemáticas 355
gráficos personalizados, CUSTOM . 384
gráfico-tabela, Graph<->Table 491
Graph
gráfico 122, 331, 669
modo 43
modo de número 43
modos 367, 375
GRAPH FORMATS (◆ F)
caixa de diálogo 20
Graph, gráfico 882
Graph<->Table, gráfico-tabela 491
grau/minuto/segundo exibe, ▸DMS . . 863
guardar símbolo, → 636
GUI, interface gráfica do utilizador . . 662

H

hexadecimal
exibir, ▸Hex 882
indicador, 0h 967
hexadecimal mostrar, ▸Hex 728
hiperbólico
cosseno, cosh( ) 846
seno, sinh( ) 932
tangente, tanh( ) 941
Histogram 598
Hora
definir 45
Reiniciar 53

I

Ícone principal 28
ID (identificador)
produto 256
unidade 257
identificação do produto 255

identity( ), matriz identidade 883
If, if 883
if, If . . 513, 623, 646, 647, 648, 649, 883
igual, = ... 644, 958
imag( ), parte imaginária 883
imagem do campo, fldpic 451
imagem gráfica ou exclusivo, XorPic 948
imagens gráficas 535, 537
armazenamento, StoPic 670
armazenando, StoPic 937
chamar novamente, RclPic 670, 917
ciclo, CyclePic 853
e, AndPic 670, 835
exclusão 538
novas, NewPic 637, 670, 899
ou exclusivo, XorPic 670, 948
repetir, CyclePic 670
substituir, RplcPic 921
ImpDif 883
Independent AUTO/ASK, auto independente/perguntar . . .491, 494, 499
indicador
BUSY 225, 614
PAUSE 225
Indicador do histórico 31
Inflection (ferramenta matemática gráfica) 356, 360
iniciar temporizador, startTmr( ) 936
Input, entrada 660, 669, 884
InputSt, cadeia de caracteres de entrada 641, ...660, 884
InputSt, introduzir cadeia 780
inString( ), dentro da cadeia de caracteres 642, 884
Instruções
Catalog (Catálogo) 25
ecrã inicial da calculadora 28
instruções 178
int( ), inteiro 885
intDiv( ), dividir por um inteiro . . 731, 885
integrar, ∫( ) 86, 263, 265, 266, 271, 288, 290, ...960
inteiro arbitrário, @ 299
inteiro, int( ) 885
interface gráfica do utilizador, GUI . . 662
interrupção de um cálculo 183
Intersection (ferramenta matemática gráfica) 355, 358
introduzir cadeia, InputSt 780
inverso, x^{-1} 965
iPart( ), parte inteira 109, 885
ir para, Goto . . .633, 646, 650, 656, 881
isArchiv() 636
isArchiv(), is archived 742
isArchive 885
isClkOn( ), relógio está ligado 885
isLocked 885
isLocked() 636
isLocked(), is locked 742
isPrime( ), teste de número primo . . . 885

isVar 885
isVar() 636
isVAR(), is variable 742
item do menu, Item 662, 665, 886
Item, item do menu 662, 665, 886

L

Lbl, etiqueta . . . 633, 646, 650, 656, 887
lcm, mínimo múltiplo comum 887
left( ), esquerda 281, 643, 887
ligação e transmissão
de calculadora 674
enviar mensagem de bate-papo, SendChat 676
enviar para calculadora, SendCalc . 675
enviar variável de lista, Send . . . 676
obter/devolver valor de CBL/CBR, Get 609, 676
prog 674
ligado/desligado 161, 162
Ligar
adaptador de vídeo TI-Presenter . 73
arranque inicial 2
computador 72
dispositivos 72
painel retroprojector TI ViewScreen . 73
Voyage 200 PLT 72
ligar e transmitir
Aplicações Flash 770, 771, 774, 778, ... 779
calculadora - calculadora . 766, 769, 770, ... 774, 780, 782, 783
cancelar 776
enviar conversação, SendChat . . 780
enviar para a calculadora, SendCalc 780
enviar para calculadora, SendCalc . . 780
erros 777, 787
pastas 770, 771, 775, 777
programa 780
variáveis 770, 771, 774, 775
ligar relógio, ClockOn 840
limit( ), limite 271, 288, 291, 888
limite inferior, floor( ) 802, 873
limite superior, ceiling( ) 802, 838
limite, limit( ) 271, 288, 291, 888
Line, desenhar linha 673, 888
LineHorz, desenhar linha horizontal 673, 889
LineTan, desenhar linha tangente . . 673, 889
LineVert, desenhar linha vertical 673, 889
linguagem assembler 869
linguagem assembly 683, 685, 686
Linha de entrada
cursor permanece em 30
inserir comandos 26
limpar a área do histórico 32

rechamar 30
Linha de estado
informações do histórico 31
parâmetros do comando 27
linha de estado 221, 222, 327
LinReg, regressão linear 587, 890, 1000
list▸mat( ), listar para a matriz . . 572, 890
lista de ID 788, 789, 791
listar a diferença, ∆list( ) 890
listas
ampliar/concatenar, augment( ) . 837
classificar em ordem ascendente, SortA 935
classificar em ordem descendente, SortD 936
diferença, ∆list( ) 890
dimensão, dim( ) 861
entre-produtos, crossP( ) 847
lista de expressões, exp▸list( ) . . 869
listar para a matriz, list▸mat( ) . . 890
matriz para lista, mat▸list( ) 894
máximo, max( ) 895
mínimo, min( ) 897
nova, newList( ) 899
novos dados, NewData . . . 637, 898
operações 831
produto de ponto, dotP( ) 863
produto, product( ) 910
semi-cadeia de caracteres, mid( ) . . 896
soma cumulativa, cumSum( ) . . . 852
somatório, sum( ) 919, 938
variáveis 560, 561, 562, 563
variáveis de tabela 501
ln( ), logaritmo natural 890
LnReg, regressão logarítmica . 587, 891, 1001
Local, variável local . 630, 634, 636, 637, 638, ... 639, 891
Lock, bloquear variável 637, 892
log to any base 87
log( ), logaritmo 892
logaritmo natural, ln( ) 890
logaritmo, log( ) 892
logaritmos 890, 892
Logistic, regressão logística . . 587, 893, 1001
loop , Loop 894
Loop, loop 655, 894
loop, Loop 655
LU, decomposição inferior-superior da matriz 894

M

maior ou igual que, ≥, >= 644, 960
maior que, > 644, 959
mais claro/mais escuro 163
manipulação simbólica 258
marca de comando 707
mat▸data( ) 894
mat▸list( ), matriz para lista 894
MATH menu 195

matriz
aleatória, randMat( ) 916
identidade, identity( ) 883
para lista, mat▸list( ) 894
Matrizes
aleatórias, randMat( ) 799
ampliar/concatenar, augment( ) . 799
matrizes
adição de ponto, .+ 957
adicionar linhas, rowAdd( ) 920
aleatórias, randMat( ) 916
ampliar/concatenar, augment( ) . 837
bloqueio 570
coluna norm, colNorm( ) 842
cópia 578
dados de um gráfico 504
decomposição inferior-superior, LU . 894
determinantes, det( ) 861
diagonal, diag( ) 861
dimensão da coluna, colDim( ) . . 842
dimensão da linha, rowDim( ) . . 921
dimensão, dim( ) 861
divisão de ponto, ./ 957
eigenvalor, eigVl( ) 867
eigenvetor, eigVc( ) 867
fatoração de QR, QR 914
forma escalonada de fila reduzida, rref( ) 285, 921
forma escalonada de fila, ref( ) . 918
identidade, identity( ) 883
listar para a matriz, list▸mat( ) . . . 890
matriz para lista, mat▸list( ) 894
máximo, max( ) 895
mínimo, min( ) 897
multiplicação de ponto, .* 957
multiplicação e adição de linhas, mRowAdd( ) 897
norma da linha, rowNorm( ) 921
novas, newMat( ) 899
novos dados, NewData 637, 898
operações 832
potência de ponto, .^ 957
preenchendo, Fill 872
pretty print 559
produto, product( ) 910
row operation, mRow( ) 897
soma cumulativa, cumSum( ) . . . 852
somatório, sum( ) 919, 938
submatriz, subMat( ) 938
subtração de ponto, .− 957
transpor, ᵀ 939
troca de linha, rowSwap( ) 921
max( ), máximo 895
máximo divisor comum, gcd( ) 876
máximo, max( ) 895
Maximum (ferramenta matemática gráfica) 355, 357
mean( ), média 895
média, mean( ) 895
median( ), mediana 895
mediana, median( ) 895

média-regressão de linha média, MedMed 588
Medição
unidades (2nd [_]) 20
MedMed, média-regressão de linha média 588, 896
MedMed, regressão de linha média-média 1001
Memória 736–765
arquivando, Archive 758
constante 21
desarquivar, Unarchiv 758
ecrã VAR LINK . . 742, 743, 757, 758
reinicializar 738, 739, 740, 741
verificar 736, 739, 740, 741
memória
arquivando, Archive 836
arquivo, Archive 636
desarquivar, Unarchiv 637, 945
memória do visor insuficiente, <<...>> 252
MEMORY (2nd [MEM]) 21
Memory (zoom) 350, 354
menor ou igual que, ≤, <= 644, 959
menor que, ... 959
menor que, < 644
Mensagem BATT 1004, 1006
Mensagens
Recolha de lixo . 759, 760, 761, 762, 763
mensagens
BATT 224, 1004, 1006
false 298
true 298
undef (indefinida) 300
mensagens Ver também erros e resolução de problemas
Menu APPLICATIONS (APLICAÇÕES) (APPS) 63
Menu CHAR (2nd [CHAR])
comando de tecla 21
introduzir caracteres especiais . . . 12
Menu CUSTOM (2nd [CUSTOM]) 61
comando de tecla 20
descrição 60
Menus
APPLICATIONS (APLICAÇÕES) (APPS) 63
cancelar 59
CHAR 12, 21
CUSTOM (2nd [CUSTOM]) . .20, 60, 61
FLASH APPLICATIONS (◆ APPS) . 20, ...54, 64
opções 15
opções dos submenus 56
seleccionar opções 55
menus 194
Algebra 278, 279, 281
APPLICATIONS 195, 199
barra de ferramentas 253
Calc 288
CATALOG 210

Clean Up 208
Complex 280
CUSTOM 196, 253
drop-down, DropDown 866
Extract 281
MATH 195, 355
pendente, DropDown 662
pendente, PopUp 660
personalizados 664, 667
pop-up, PopUp 909
Trig 280
utilização 194
Zoom 349
Menus das barras de ferramentas
ecrã inicial da calculadora 53
mover entre 59
seleccionar operações matemáticas 16, ... 28
substituídos pelo menu Custom . . 60
método
Euler 446, 473
Runge-Kutta 446, 469, 473
Método de Runge-Kutta 1003
mid( ), semi-cadeia de caracteres . . 643, 896
min( ), mínimo 897
mínimo múltiplo comum, lcm 887
mínimo, min( ) 897
Minimum (ferramenta matemática gráfica) 96, 355, 357
mod( ), módulo 897
modes
Angle 206
Modo 3D (tridimensional) 43
Modo Angle 22
estado 43
Modo Angle (Ângulo) 977
Modo Auto 978
Modo Base 22
Modo base 979
Modo Complex Format 22
Modo Complex Format (Formato complexo) 977
Modo Current 22
Modo Current folder (Pasta actual) . . 976
Modo Custom Units 22
Modo Custom Units (Unidades personalizadas) 979
Modo DE (DE) (equação diferencial) . 43
Modo DEG (GRA) (grau) 43
Modo Display Digits 22
Modo Display Digits (Visualizar dígitos) . 976
Modo Exact/Approx 22
Modo Exact/Approx (Exact/Aprox) . . 978
Modo Exponential Format 22
Modo Exponential Format (Formato exponencial) 977
Modo Full Screen
alterar a partir de ecrã dividido . . . 69
Ambiente de trabalho das Apps . . 41
ver as aplicações em 69

Modo FUNC (FUNC) (função) 43
Modo Graph 22
Modo Graph (Gráfico) 976
Modo Graph (Gráfico) 2 978
Modo Insert (2nd [INS]) 21
Modo Language
alterar a definição do modo 23
ver 22
Modo Language (Idioma) 979
Modo Number of Graphs (Número de gráficos) 978
Modo Overwrite(2nd [INS]) 21
Modo PAR (PAR) (paramétrico) 43
Modo POL (POL) (polar) 43
Modo Pretty Print 22, 978
Modo RAD (RAD) (radiano) 43
Modo SEQ (SEQ) (sequência) 43
Modo Split (Divisão) App 978
Modo Split Screen
definir 64
definir aplicação inicial 66
especificar aplicações apresentadas 67
estado 40
estado e Apps abertas 3
gráfico activo 43
número de gráficos 67
Ratio 67
sair 69
seleccionar aplicação activa 69
Split 1 App 67
Split 2 App 67
ver 22
voltar a uma App 35
Modo Split Screen (Ecrã dividido) . . . 978
Modo Unit System 22
Modo Unit System (Sistema de unidades) 979
Modo Vector Format 22
Modo Vector Format (Formato do vector) 978
Modos
3D (tridimensional) 43
a cinzento 22
Angle 22
Angle (Ângulo) 43
APPROX (APROX) 43
Apps desktop 22
AUTO 43
Base 22
Complex Format 22
Current 22
Custom Units 22
DE (DE) (equação diferencial) . . . 43
definições 21
DEG (GRA) (grau) 43
Display Digits 22
EXACT (EXACTA) 43
Exact/Approx 22
Exponential Format 22
Full Screen 35, 41, 66, 69
FUNC (FUNC) (função) 43

Graph 22
Graph Number 43
Graph type 43
insert ([2nd] [INS]) 21
Language 22, 23
overwrite ([2nd] [INS]) 21
PAR (PAR) (paramétrico) 43
POL (POL) (polar) 43
Pretty Print 22
RAD (radiano) 43
SEQ (SEQ) (sequência) 43
Split Screen 3, 22, 35, 40, 43, 64, 66, ... 67, 69
Unit System 22, 92
Vector Format 22
modos 204
Angle 327
Angle (Ângulo) 977
Approximate 185, 207, 223, 264
Approximate (Aproximada) 978
Auto 185, 207, 223, 265, 978
Base 207, 979
Complex Format 206
Complex Format (Formato complexo) 977
configuração de programas 657
Current folder 206
Current folder (Pasta actual) . . . 976
Custom Units 208
Custom Units (Unidades personalizadas) 979
definição, setMode( ) 658, 669
definindo, setMode( ) 925
Display Digits 188, 206
Display Digits (Visualizar dígitos) 976
Exact/Approx . . . 185, 207, 223, 263, 264, ... 265
Exact/Approx (Exact/Aprox) 978
Exponential Format 190, 206
Exponential Format (Formato exponencial) 977
Graph 206, 224, 326, 367, 375, 383, 405, ... 443
Graph (Gráfico) 2 978
Graph 2 207
Language 208
Language (Idioma) 979
Number of Graphs 207
Number of Graphs (Número de gráficos) 978
obter/devolver, getMode( ) 657
obter/retornar, getMode( ) 879
Pretty Print 184, 207, 978
Split (Divisão) App 978
Split App 207
Split Screen 207
Split Screen (Ecrã dividido) 978
Unit System 207
Unit System (Sistema de unidades) . 979
Vector Format 206

Vector Format (Formato do vector) . 978
módulo, mod( ) 897
montar
dados, BldData 473, 636, 838
tabela, Table 668, 939
Mostrar
ecrã de E/S, Disp 145
mostrar como
binário, ▸Bin 728
decimal inteiro, ▸Dec 728
hexadecimal, ▸Hex 728
mostrar resultados estatísticos, ShowStat 589, ... 930
mover variável, MoveVar 637, 897
MoveVar, mover variável 637, 897
mRow( ), matrix row operation 897
mRowAdd( ), multiplicação e adição de linhas de matriz 897
multiplicação implícita 178, 376
multiplicar, * 955

N

não (Booleano), not 645, 732, 901
nCr( ), combinações 898
nDeriv( ), derivada numérica . . 289, 898
negar, - 175, 958
NewData, novos dados . .558, 571, 637, 898
NewFold, nova pasta 637, 747, 899
newList( ), nova lista 899
newMat( ), nova matriz 899
NewPic, nova imagem gráfica . 637, 670, 899
NewPlot, novo traçado . . .597, 669, 899
NewProb, novo problema 209, 900
nInt( ), integral numérica 289, 900
nmax, variáveis de janela 386
nmin, variáveis de janela 386
nomes reservados 998
norm( ), norma de Frobenius 901
norma de Frobenius, norm( ) 901
not, não Booleano 645, 732, 901
notação
de graus, ° 963, 964
de minutos, ' 964
de segundos, ...964
Notação científica 17
de graus, ° 818
notação científica 176
Nova
pasta, NewFold 747
nova
imagem gráfica, NewPic . . 637, 670, 899
lista, newList( ) 899
matriz, newMat( ) 899
pasta, NewFold 637, 899
novo
dados, NewData 637
problema, NewProb 209, 900
traçado, NewPlot 597, 669, 900

Novo ficheiro (◆ N) 20
novos
dados, NewData 898
nPr( ), permutações 902
nSolve( ), solução numérica . . . 280, 902
numérica
derivada, nDeriv( ) 289, 898
integral, nInt( ) 289, 900
solução, nSolve( ) 902
número
obter/retornar, getNum( ) 879
número de ID . 783, 784, 785, 788, 789, 791
Números
complexos 80
negativos 17
primos 81
números
de identificação 256
de série 255, 256
irracionais 263, 264
negativos 175
racionais 263, 264, 265

O

obter cadeia de data, getDtStr( ) . . . 878
obter cadeia de hora, getTmStr( ) . . 879
obter data actual, getDate( ) 877
obter formato de data, getDtFmt( ) . . 878
obter formato de hora, getTmFmt( ) . 879
obter fuso horário, getTmZn( ) 880
obter hora actual, getTime( ) 879
obter/devolver
à calculadora, GetCalc 675
calculadora, GetCalc 780
configuração, getConfg( ) 657
denominador, getDenom( ) 281
modo, getMode( ) 657
número, getNum( ) 281
pasta, getFold( ) 636, 657
tecla, getKey( ) 660
tecla, getKey(†) 982, 986
tipo, getType( ) 260, 636
unidades, getUnits( ) 657
valor de CBL/CBR, Get 609, 676
obter/retornar
configuração, getConfg( ) 877
da calculadora, GetCalc 876
denominador, getDenom( ) 877
pasta, getFold( ) 878
valor de CBL/CBR, Get 876
OCUPADA 44
OneVar, estatística de uma variável 587, 903
operações
algébricas 831
de cálculo 831
matemáticas 832
operadores 177
or, ou Booleano 645, 732, 903
ord( ), código de carácter numérico . 643
ord( ), código de caráter numérico . . 904

ou (Booleano), or 645, 732, 903
ou exclusivo (Booleano), xor . .645, 732, 948
Output, saída 661, 904

P

P▸Rx( ), coordenada x retangular . . . 905
P▸Ry( ), coordenada y retangular . . . 905
Painel retroprojector TI ViewScreen
ligar 73
panorâmica 347
parar, Stop 620, 937
parêntesis curvos, parêntesis rectos e chavetas 998
parêntesis, chavetas e parêntesis rectos 179
Pares de entradas/respostas
estado 44
part( ), parte 905
parte imaginária, imag( ) 883
parte inteira, iPart( ) 109, 885
parte, part( ) 905
passar erro, PassErr 678, 907
PassErr, passar erro 678, 907
Pastas 744
atribuir nomes 748
bloquear/desbloquear 752
colar o nome 754, 755
definindo, setFold( ) 747
excluir 753
excluir, DelFold 754
novas, NewFold 747
VAR LINK 742, 743, 749
pastas 206, 976
configuração, setFold( ) 657
definindo, setFold( ) 924
excluindo, DelFold 858
excluir, DelFold 636
novas, NewFold 637, 899
obter/retornar, getFold( ) 878
transmitir 770, 771, 774, 775
PAUSA 44
pausa, Pause 661, 677, 907
Pause, pausa 907
Percorrer 30
permutações, nPr( ) 902
Pilhas
arranque inicial 2
precauções 75, 1018
prolongar duração 7
substituir 1, 74, 76
pilhas 164, 224, 1004, 1006
pixel
activado, PxlOn 535, 672
alterar, PxlChg 672, 912
ativado, PxlOn 913
círculo, PxlCrcl 673, 912
desactivado, PxlOff 672
desativado, PxlOff 913
linha horizontal, PxlHorz . . .673, 912
linha vertical, PxlVert 673, 914
linha, PxlLine 535, 673, 913

teste, pxlTest( ) 672, 913
texto, PxlText 673, 913
PlotsOff, traçados desactivados 333, 669
PlotsOff, traçados desativados 908
PlotsOn, traçados activados . . . 333, 669
PlotsOn, traçados ativados 908
plotStep, variáveis de janela 387
plotStrt, variáveis de janela 386
polar
coordenada, R▸Pr( ) 916
exibir vetor, ▸Polar 908
polar, representação gráfica 365
polinómio de Taylor, taylor( ) . . 289, 291
polinômio de Taylor, taylor( ) 941
Polinómios
actividade 820
polinómios 282, 291
polinômios
aleatórios, randPoly( ) 917
calcular, polyEval( ) 908
polyEval( ), calcular polinômio 908
ponto
activado, PtOn 672
adição, .+ 957
alterar, PtChg 672, 911
ativado, PtOn 912
desactivado, PtOff 672
desativado, PtOff 911
divisão, ./ 957
multiplicação, .* 957
potência, .^ 957
produto, dotP( ) 863
subtração, .− 957
teste, ptTest( ) 672, 912
texto, PtText 673, 912
PopUp, menu pendente 660
PopUp, menu pop-up 909
porcentagem, % 958
Porta
acessório 72
E/S 72
potência de dez, 10^( ) 965
potência, ^ 956, 998
PowerReg, regressão de potência . 588, 909
PowerReg, regressão potencial . . . 1001
precisão 995
Pretty print 95
pretty print 184, 227
Prgm, executar programa . 142, 631, 910
primo, ' 964
problemas (novo), NewProb . . . 209, 900
product( ), produto 910
produto, Π( ) 289, 962
produto, product( ) 910
Program Editor 33
Programas e programação
CBL 814
CBR 814
definir, Define 794
encerrar programa, EndPrgm . . . 142
executar programa, Prgm 142

mostrar ecrã de E/S, Disp 145
programas e programação 613–687
argumentos 625
aviso, Prompt( ) 660, 911
barra de ferramentas do menu personalizado activada, CustmOn 253
barra de ferramentas do menu personalizado desactivada, CustmOff 253
barra de ferramentas personalizada activada, CustmOn 662
barra de ferramentas personalizada ativada, CustmOn 853
barra de ferramentas personalizada desactivada, CustmOff 662
barra de ferramentas personalizada desativada, CustmOff 852
CBL 674
CBR 674
chamada de outro programa . . . 631
comentário, ◕ 622, 967
cópia 620
definir barra de ferramentas, Custom 662, ... 853
definir barra de ferramentas, Toolbar 662, ... 944
definir caixa de diálogo Dialog . . 662, 861
definir, Define 631, 668, 857
depuração 677
desvio 623, 646, 650
digitação . . 617, 620, 621, 622, 623, 624
else if, ElseIf 513, 649, 868
else, Else 649, 883
encerramento de função, EndFunc . . 514
encerrar barra de ferramentas, EndTBar 662
encerrar personalização, EndCustm . 662
encerrar tentativa, EndTry 678
entrada 615, 624, 660
entrada, Input 660, 669, 884
etiqueta, Lbl 633, 646, 650, 656
etiquetar, Lbl 887
exclusão 620
execução 613
executar linguagem assembler, Exec 869
executar linguagem assembly, Exec 686
executar programa, Prgm . . 631, 910
exibir gráfico, DispG 862
exibir tabela, DispTbl 863
exibir tela de I/O, Disp 862
exibir tela principal, DispHome . . 862
for, For 623, 652, 874
formatar cadeia de caracteres, format( ) 661, 875
função, Func 514, 628, 875

funções 617, 626, 627, 628
gráficos 668
if, If . . . 623, 646, 647, 648, 649, 883
interface gráfica do utilizador, GUI . . 662
ir para, Goto 633, 646, 650, 656, 881
item do menu, Item . . 662, 665, 886
linguagem assembly . 683, 685, 686
local, Local 630, 634, 636, 637, 638, 639, ... 891
loop, Loop 655
menu drop-down, DropDown . . . 866
menu pendente, DropDown 662
menu pendente, PopUp 660
menu pop-up, PopUp 909
menus 663, 667
obter/devolver calculadora, GetCalc 780
obter/devolver tecla, getKey(†) . 982, 986
operações 833
parar, Stop 620, 937
passar erro, PassErr 678, 907
pausa, Pause 661, 677, 907
repetição 623, 651, 652, 653
repetição, Loop 894
retornar, Return 919
saída 615, 623, 661, 662
saída, Output 661, 904
sair, Exit 869
se, If 513
solicitar, Request 660, 663, 919
subrotinas 631
tabelas 668
tentar, Try 678, 945
terminar diálogo, EndDlog 662
testes de condição 643
texto, Text 661, 663, 942
Then, Then 646, 648, 649, 883
título, Title 663, 943
transferência de valores 625
variáveis 633
ver ecrã de I/O, Disp . .623, 661, 677
ver ecrã principal, DispHome . . . 661
ver gráfico, DispG 661, 668
ver tabela, DispTbl 661, 668
voltar, Return 629, 632
while, While 653, 947
programas e programação, apagar
erro, ClrErr 678, 840
gráfico, ClrGraph 510, 668, 841
I/O, ClrIO 616, 661, 841
tabela, ClrTable 841
tela principal, ClrHome 841
programas e programação, encerrar
barra de ferramentas, EndTBar . 944
diálogo, EndDlog 861
for, EndFor 623, 652, 874
função, EndFunc 628, 875
if, EndIf 623, 646, 648, 649, 883
personalização, EndCustm 853
programa, EndPrgm 631, 910

repetição, EndLoop 655, 894
tentar, EndTry 945
while, EndWhile 653, 947
programas e programação, obter/devolver
à calculadora, GetCalc 675
configuração, getConfg( ) 657
modo, getMode( ) 657
pasta, getFold( ) 657
tecla, getKey( ) 660
programas e programação, obter/retornar
configuração, getConfg( ) 877
da calculadora, GetCalc 876
modo, getMode( ) 879
pasta, getFold( ) 878
tecla, getKey( ) 878
unidades, getUnits( ) 881
Prompt( ), aviso 660, 911
propFrac, fração própria 911
propFrac, fracção própria . .82, 280, 286, 808
PtChg, alterar ponto 672, 911
PtOff, ponto desactivado 672
PtOff, ponto desativado 911
PtOn, ponto activado 672
PtOn, ponto ativado 912
ptTest( ), teste do ponto 672, 912
PtText, texto do ponto 673, 912
PxlChg, alteração do pixel 672, 912
PxlCrcl, círculo do pixel 673, 912
PxlHorz, linha horizontal do pixel . . .673, 912
PxlLine, linha do pixel 535, 673, 913
PxlOff, pixel desactivado 672
PxlOff, pixel desativado 913
PxlOn, pixel activado 535, 672
PxlOn, pixel ativado 913
pxlTest( ), teste do pixel 672, 913
PxlText, texto do pixel 673, 913
PxlVert, linha vertical do pixel . .673, 914

Q

QR, fatoração de QR 914
QuadReg, regressão quadrática . . . 588, 915, ...1001
quando, when( ) 122, 511, 946
QuartReg, regressão quártica . 588, 915, 1001
QuickCenter 348

R

r, radiano 963
R▸Pθ( ), coordenada polar 916
R▸Pr( ), coordenada polar 916
radiano, r 963
raíz quadrada, √( ) 961
Raízes
actividade 820
raízes
complexos, cZeros( ) 854
rand( ), número aleatório 916
randMat( ), matriz aleatória 799, 916
randNorm( ), norma aleatória 916

randPoly( ), polinômio aleatório 917
RandSeed, semente do número aleatório 799, ... 917
RclGDB, chamar banco de dados de gráficos 917
RclGDB, chamar base de dados gráficos 545, ... 670
RclPic, chamar novamente imagem gráfica 670, 917
real( ), real 917
real, real( ) 917
Realçar
caracteres quando editar 15
para ver o nome completo da App . 3
realce do texto 696
Rechamar (2nd [RCL]) 21
recípra, x^{-1} 965
ref( ), forma escalonada de fila 918
regressão cúbica, CubicReg 1000
regressão de linha média-média, MedMed 1001
regressão exponencial, ExpReg . . . 1000
regressão linear, LinReg 1000
regressão logarítmica, LnReg 1001
regressão logística, Logistic 1001
regressão potencial, PowerReg . . . 1001
regressão quadrática, QuadReg . . 1001
regressão quártica, QuartReg 1001
regressão sinusoidal, SinReg 1001
Regressões
actividade com fórmula quadrática . . 795
regressões 890
cúbica, CubicReg 1000
cúbicas, CubicReg 587, 852
exponencial, ExpReg 587, 871, 1000
fórmulas 1000, 1002
linha média-média, MedMed . . . 588, 896, ... 1001
logarítmica, LnReg . . 587, 891, 1001
logística, Logistic 587, 893, 1001
quadrática, QuadReg 588, 915, 1001
quártica, QuartReg 1001
quárticas, QuartReg 588, 915
regressão de potência, PowerReg . . 588, ... 909
regressão linear, LinReg . . 587, 890, 1000
regressão potencial, PowerReg 1001
seleccionar 586
sinusoidais, SinReg 588, 933
sinusoidal, SinReg 1001
Relógio
desligar 51
funcionamento 45
ligar 53
relógio está ligado, isClkOn( ) 885
remain( ), resto 918
Rename, renomear 637, 918
renomear, Rename 637, 918

repetição da imagem gráfica, CyclePic . 540
repetir imagem gráfica, CyclePic . . . 670, 853
representação gráfica paramétrica . . 373
representações gráficas
nível de contorno 426
sequencial 381
wire-and-contour 426
wire-frame 426
Representações gráficas 3D
animação 114
representações gráficas 3D 403
animação 421
CONTOUR LEVELS 426
HIDDEN SURFACE 426
WIRE AND CONTOUR 426
WIRE FRAME 426
Request, solicitar 660, 663, 919
resolver
equações lineares 284
solve( ) 263, 265, 266, 271, 279, 284, ... 285, 481
Resolver equações lineares 83, 84
resolver, solve( ) 83, 90, 933
resposta (última), ans( ) 242, 836
resto, remain( ) 918
restrições de domínio 276
Resultados 28
resultados de duas variáveis, TwoVar . . 587, ... 588, 945
retornar *Consulte* obter/retornar
retornar, Return 919
Return, devolver 513
Return, retornar 919
Revisão do certificado (Cert. Rev.) . . 257
right( ), direita 281, 643, 919
root 919
roots 81
rotate( ), girar 643, 734, 919
round( ), arredondar 920
rowAdd( ), adicionar linha da matriz . 920
rowDim( ), dimensão da linha da matriz . 921
rowNorm( ), norma da linha da matriz 921
rowSwap( ), troca de linha da matriz . 921
RplcPic, substituir imagem gráfica . 670, 921
rref( ), forma escalonada de fila reduzida 285, ... 799, 921

S

saída, Output 661, 904
sair, Exit 869
SAVE COPY AS (◆ S)
caixa de diálogo 20
descrição 20
exemplo 58
Script de comandos
actividade 804
tutorial 804
script de comandos 232, 707, 710

sec( ), secante 922
sec^{-1}( ), secante inversa 922
secante hiperbólica, sech( ) 922
secante, sec( ), ... 922
sech( ), secante hiperbólica 922
sech^{-1}( ), secante hiperbólica inversa 922
Seleccionar categorias 35
semi-cadeia de caracteres, mid( ) . . 643, 896
Send, enviar variável de lista . . 676, 922
SendCalc, enviar para a calculadora 780
SendCalc, enviar para calculadora . . 675, 780, ... 923
SendChat, enviar conversação 780
SendChat, enviar mensagem de bate-papo 676, 923
seno, sin( ) 931
seq( ), seqüência 923
seqüência, seq( ) 923
Set factors (zoom) 351, 353
setDate( ), definir data 924
setDtFmt( ), definir formato de data . 924
setFold( ), definir pasta . . . 657, 747, 924
setGraph( ), definir gráfico 657, 669, 924
setMode( ), definir modo . . 658, 669, 925
setTable( ), definir tabela .493, 658, 668, 926
setTime( ), definir hora 926
setTmFmt( ), definir formato de hora 926
setTmZn( ), definir fuso horário 927
setUnits( ), definir unidades . . . 658, 927
Shade
(ferramenta matemática gráfica) 356, 362
sombra 674
Shade, sombra 928
shift( ), deslocar 574, 643, 734, 929
ShowStat, mostrar resultados estatísticos 589, ... 930
sign( ), sinal 930
Símbolo de expoente (E) 18
Símbolos 1
#, acção indirecta 998
^, potência 998
|, with 998
simplificação
diferida 270
interromper 270
regras 267
simult( ), equações simultâneas 285, 930
sin( ), seno 931
sin^{-1}( ), arco-seno 931
Sinal de menos
operacional (−) 17
posicional (⊖) 17
sinal, sign( ) 930
sinh( ), seno hiperbólico 932
sinh^{-1}( ), arco-seno hiperbólico 932
SinReg, regressão sinusoidal . 588, 933, 1001
Sistema Calculator-Based Laboratory
ligar 73

Sistema Calculator-Based Ranger
ligar 73
Sistema CBL 2
ligar 73
Sistema CBR
ligar 73
sistema operativo 785, 787
Sistema operativo (SO)
transferir 75
sistema operativo, actualizar . .783, 784, 785
SLPFLD, campo de inclinação .447, 455, 456, ... 482, 483
Smart Graph 341
SO 783, 784, 785
Software TI Connect 70
Software TI Connect 785
solicitar, Request 660, 663, 919
solução
complexa, cSolve( ) 850
deSolve( ) 289, 479
numérica, nSolve( ) 280
solução, deSolve( ) 858
solucionador numérico 712
ecrãs divididos 723
equações 712, 714, 715
representação gráfica 722, 723, 724, 725
variáveis 715
solve( ), resolver . 83, 90, 263, 265, 266, 271, ... 284, 285, 481, 933
soma cumulativa, cumSum( ) 852
soma, Σ( ) 288
somatório, Σ( ) 962
somatório, sum( ) 919, 938
sombra, Shade 674, 928
SortA, classificar em ordem ascendente . 935
SortD, classificar em ordem descendente 936
Sprachauswahl 1023
startTmr( ), iniciar temporizador 936
stdDev( ), desvio padrão 936
stdDevPop 936
StoGDB
armazenamento de base de dados gráficos 545
guardar base de dados gráficos . 670
StoGDB, armazenar banco de dados de gráfico 937
Stop, parar 620, 937
StoPic, armazenar imagem gráfica . . 937
StoPic, guardar imagem gráfica 670
string( ), expressão para cadeia de caracteres 643, 937
Style, estilo 335, 669, 938
sublinhado, _ 964
subMat( ), submatriz 938
submatriz, subMat( ) 938
submenus 197
substituições . . .272, 273, 274, 275, 277

substituir imagem gráfica, RplcPic . . 670, 921
subtrair, − 955
Sucessão de Fibonacci 401
sum( ), somatório 919, 938
superfície oculta 414, 426
switch( ), alternar 658, 939
sysData, dados do sistema . . . 504, 505

T

ᵀ, transpor 939
Tabela de transmissão e ligação . . . 792
tabelas 488
  ∆tbl 490
  apagando, ClrTable 841
  automáticas 494
  configuração 491, 492, 493
  configuração, TABLE SETUP . . 490
  definição, setTable( ) 658, 668
  definindo, setTable( ) 926
  equações diferenciais 487
  exibindo, DispTbl 863
  funções 498
  gerar em sucessão 401
  incremento, ∆tbl 490
  Independent AUTO/ASK . . 491, 494, 499
  iniciando, tblStart 490
  largura da célula 496, 502
  manual 499
  montar, Table 668, 939
  números complexos 498
  programas 668
  representação de gráfico, Graph<->Table 491
  setTable( ) 493
  tblStart 490
  visão geral 488
  visualização, DispTbl 661, 668
TABLE SETUP, configuração de tabela . 490
Table, montar tabela 668, 939
Tampa
  guardar 6
tan( ), tangente 940
tan⁻¹( ), arco-tangente 940
Tangent (ferramenta matemática gráfica) 356, ... 361, 372, 380
tangente, tan( ) 940
tanh( ), tangente hiperbólica 941
tanh⁻¹( ), arco-tangente hiperbólico . . 941
taxa de câmbio normal, avgRC( ) . . . 837
taylor( ), polinómio de Taylor 289
taylor( ), polinômio de Taylor 941
tblStart, iniciação de tabela 490
tCollect( ), coleção trigonométrica . . 280, 942
Tecla de expoente (2nd EE) 18
Tecla modificadora Diamond (◆)
  descrição 15
  estado 43
Tecla modificadora Hand (✋)

descrição 16
estado 43
Tecla modificadora Second (2nd)
descrição 15
estado 43
Tecla modificadora Shift (⇧)
descrição 15
estado 43
Tecla Store (STO▸) 21
tecla, obter/retornar, getKey( ) 878
Teclado
mapa 12, 13
QWERTY 12
teclado 165, 166
tecla (mão) 168
2nd (segunda) tecla 167
⇧ tecla (shift) 167
◆ tecla (losango) 167
atalhos 702
códigos das teclas 660
alpha tecla (alfa) 168
mapa 700, 703
Teclado numérico 17
localização física 12
Teclado QWERTY 12
Teclas
cursor 12, 17
função 12, 16
modificadoras 12, 15
outras 19
Teclas de cursor (◀▶▲▼)
abrir Apps 32
função 17
funções adicionais 17
introduzir comandos 27
localização fÌsica 12
seleccionar pares de entradas/
respostas 30
utilizada com a tecla Hand 16
utilizar o menu CHAR 12
Teclas de função (F1–F8)
localização fÌsica 12
mover entre os menus das barras de
ferramentas 59
seleccionar categorias 32, 35
seleccionar menus 53
Teclas modificadoras (2nd ◆ ⇧) . 15
estado 43
localização fÌsica 12
tela dividida
alternar, switch( ) 939
tentar, Try 678, 945
teste de número primo, isPrime( ) . . . 885
tExpand( ), expansão trigonométrica 280,
. 942
Text, texto 942
texto
realçado 191
Text 661, 663
texto, Text 942
Then, Then 646, 648, 649, 883
TIME

gráfico de tempo 469
traçados de tempo . . . 384, 391, 468
timeCnv( ), converter hora 943
tipo, obter/retornar, getType( ) 880
Title, título 943
título, Title 943
tmax, variáveis de janela 378
tmin, variáveis de janela 378
tmpCnv( ), conversão de temperatura . .
308, ... 943
Todas as categorias 35
Toolbar, barra de ferramentas . 662, 944
traçado em colmeia. *Consulte também*
gráficos de rede
Traçados 96
dados 131
traçados 371, 380, 390, 411, 453
activados, PlotsOn 333, 669
ativados, PlotsOn 908
de tempo, TIME 384, 391, 468
desactivados, PlotsOff 333, 669
desativados, PlotsOff 908
eliminação 595
implícitos 435, 438, 439
novos, NewPlot 597, 669, 900
personalizados, CUSTOM . 468, 469
Scatter 596
selecção 601
traçando 603
visualização da janela 602
xyline 596
Y=Editor 600
traçados de contorno 427, 431, 432
DrwCtour, desenhar contornos . . 431
traçar, Trace . . 344, 345, 346, 347, 348,
669, ...803, 814, 816, 820, 944
Trace, traçar 944
transmissão. *Consulte* ligação e
transmissão
transpor, T 939
Try, tentar 678, 945
tstep, variáveis de janela 378
TwoVar, resultado de duas variáveis 945

U

última
entrada 171, 239, 241
resposta 171, 182, 239, 242
Unarchiv, desarquivar variáveis 637, 758,
. 945
Unidades
modos 92
unidades 302
conversão 306
definição, setUnits( ) 658
definidas pelo utilizador 312
definindo, setUnits( ) 927
formatação 309
medida 302
modos 208, 979
obter/retornar, getUnits( ) 881
predefinição 315

predefinições 309
unitV( ), vetor unitário 946
Unlock, desbloquear 637, 946

V

valor absoluto, abs( ) 821, 834
Value (ferramenta matemática gráfica) . 355, ... 356, 372, 380, 411, 453
variables
deleting
DelType 754
variance( ), variância 946
variância, variance( ) 946
Variáveis 44
armazenar 744
arquivando, Archive 758
arquivar e desarquivar 756
atribuir nomes 748
bloquear/desbloquear 752
colar o nome 754, 755
consultar ficheiros da App 33
copiar 752
copiar, CopyVar 752
desarquivar, Unarchiv 758
excluir, DelVar 754
nas aplicações 754, 755
recall 21
store 21
VAR LINK . . 742, 743, 749, 757, 758
variáveis 217, 218, 219, 220, 221
apagar 725
arquivando, Archive 836
arquivo, Archive 636
bloquear, Lock 637
bloqueio/desbloqueio 225
copiar, CopyVar 636, 844
dados 557
definidas 258, 716, 717
desarquivar, Unarchiv 637, 945
desbloquear, Unlock 637
desconhecidas, resolver 720
eliminar 777
estatísticas 585, 589
excluir, DelVar . . . 293, 636, 640, 858
globais 640
indefinidas 258, 260, 717
lista 556
locais, Local 630, 634, 636, 637, 638, ... 639, 891
matriz 559
mover, MoveVar 637
nomes reservados 996, 998
prevalecência 262
simplificação diferida 270
sistema 996, 998
texto 232
transmitir 766, 769, 775
variáveis da janela
(x 995
(y 995
xmax 995
xmin 995

ymax 995
ymin 995
variáveis de janela
diftol 450
dtime 450
Estep 450
eyeϕ (eixo z) 407, 417, 419
eyeθ (eixo x) 407, 417, 418
eyeψ (rotação) 407, 417, 419
fldres 450
ncontour 408
ncurves 449
nmax 386
nmin 386
plotStep 387
plotStrt 386
θmax 369
θmin 369
θstep 369
t0 448
tmax 378, 449
tmin 378
tplot 449
tstep 378, 449
xgrid 408
xmax . . 336, 369, 378, 387, 407, 449
xmin . . 336, 369, 378, 387, 407, 449
xres 337
xscl 336, 369, 378, 387, 449
ygrid 408
ymax . . 336, 369, 378, 387, 407, 449
ymin . 336, 369, 378, 387, 407, 449
yscl 336, 369, 378, 387, 449
zmax 407
zmin 407
variáveis do sistema 996, 998
variável da janela (x 995
variável da janela (y 995
variável da janela xmax 995
variável da janela xmin 995
variável da janela xscl 995
variável da janela ymax 995
variável da janela ymin 995
variável local, Local . 630, 634, 636, 637, 638, ... 639, 891
vectores
Modo Vector Format (Formato do vector) 978
vectores, Vector Format 206
ver
ecrã de I/O, Disp 623, 661, 677
ecrã principal, DispHome 661
gráfico, DispG 661, 668
tabela, DispTbl 661, 668
verificar temporizador, checkTmr( ) . . 839
versão
hardware 256
sistema operativo (SO) 256
SO (sistema operativo) 256
software 255, 256
vetor unitário, unitV( ) 946
vetores

entre produtos, crossP( ) 847
exibir como vetor cilíndrico, ▸Cylind . 854
produto de ponto, dotP( ) 863
unitários, unitV( ) 946
visualização 250
da órbita 422
visualizações. *Consulte* exemplos, visualizações, actividades
voltar, Return 629, 632

W

WEB, gráficos de rede . . . 384, 391, 392
when( ), quando 122, 511, 946
While, while 947
while, While 653, 947
Window Editor 63
with, | 84, 91, 262, 272, 966, 998

X

x^{-1}, recípra 965
xmax, variáveis de janela .336, 369, 378, 387
xmin, variáveis de janela .336, 369, 378, 387
xor, ou exclusivo Booleano 645, 732, 948
XorPic, imagem gráfica ou exclusivo 670, ... 948
xres, variáveis de janela 337
xscl, variáveis de janela . . 336, 369, 378

Y

Y= editor . . 99, 328, 368, 375, 384, 406, 444, ... 506
ymax, variáveis de janela 336, 369, 378, 387
ymin, variáveis de janela . 336, 369, 378, 387
yscl, variáveis de janela . .336, 369, 378

Z

Zero (ferramenta matemática gráfica) . . . 355, ... 357
zeros
complexos, cZeros( ) 263, 280
zeros( ) 263, 279, 286
zeros( ), zeros 263, 279, 286, 948
zeros, zeros( ) 795, 948
zoom
ajustar, ZoomFit 350, 951
ampliar, ZoomIn 349, 352, 951
anterior, ZoomPrev 354, 952
armazenar, ZoomSto 354, 953
caixa, ZoomBox 351, 950
chamar à memória novamente, ZoomRcl 354, 952
dados, ZoomData 350, 950
decimal, ZoomDec 349, 951
factores 351, 353
inteiro, ZoomInt 350, 952
Memory 350, 354
padrão, ZoomStd 953

predefinição, ZoomStd 350
quadrado, ZoomSqr 350, 953
reduzir, ZoomOut 349, 352, 952
trig, ZoomTrig 350, 954
ZoomBox, caixa de zoom 950
ZoomData, dados do zoom 950
ZoomDec, zoom decimal 951
ZoomFit, ajustar zoom 951
ZoomIn, ampliar 951
ZoomInt, zoom inteiro 952
ZoomOut, reduzir 952
ZoomPrev, zoom anterior 952
ZoomRcl, chamar zoom à memória
novamente 952
ZoomSqr, zoom quadrado 953
ZoomStd, zoom padrão 953
ZoomSto, armazenar zoom 953
ZoomTrig, zoom trig 954