TI-Nspire™ CX
參考手冊

請透過 education.ti.com/eguide 的線上說明，瞭解更多有關 TI 技術的資訊。

重要資訊

除伴隨程式的授權中明確陳述之外，德州儀器概不提供有關任何程式或書籍資料的明示或暗示保證，包括但不限於任何可售性和特別目的適合性的暗示保證，並且僅按「原樣」提供此等資料。無論任何情況，德州儀器皆不負責與購買或使用這類資料有關或所致的任何特殊、附屬、附帶或衍生損害賠償，且無論行動的形式，德州儀器的唯一責任不會超過程式授權中載明的金額。此外，德州儀器不承擔任何種類的賠償責任，不管是否有任何其他當事人因使用這些材料而索賠。

© 2020 Texas Instruments Incorporated

產品實物可能與所提供的圖片資料略有不同。

目錄

運算式範本 **1**

依字母順序列表 **7**

A 7
B 15
C 19
D 33
E 41
F 48
G 54
I 63
L 70
M 83
N 91
O 98
P 100
Q 106
R 109
S 122
T 139
U 151
V 151
W 152
X 154
Z 155

符號 **161**

TI-Nspire™ CX II - 繪圖指令 **183**

函數繪圖程式設計 183
函數繪圖畫面 183
預設檢視與設定 184
函數繪圖畫面錯誤訊息 185
函數繪圖模式中的無效指令 185
C 186
D 187
F 190
G 192
P 193
S 195
U 197

空元素 **198**

輸入數學式的快速鍵 **200**

EOS™(方程式作業系統)階層 **202**

TI-Nspire CX II - TI-Basic 程式設計功能 **204**

程式設計編輯器中的自動縮排 204
改進 TI-Basic 的錯誤訊息 204

常數和數值 **207**

錯誤代碼和訊息 **208**

警告代碼和訊息 **216**

一般資訊 **218**

線上說明 218
連絡 TI 技術支援部門 218
服務與保固資訊 218

索引 **219**

運算式範本

運算式範本讓您能以標準的數學符號輕鬆輸入數學式。插入範本時，範本會出現在輸入線上，並在可輸入要素的位置顯示小方塊。游標會顯示您可輸入的要素。

使用方向鍵或按 tab 讓游標移到每個要素的位置，然後輸入數值或運算式。按 enter 或 ctrl enter 開始對運算式求值。

分數範本

ctrl ÷ 鍵

$\frac{\square}{\square}$

附註：另請參考 **/(除)**，頁碼：163。

範例：

$\frac{12}{8\cdot 2}$ $\quad$ $\frac{3}{4}$

指數範本

^ 鍵

$\square^{\square}$

附註：輸入第一個值、按 ^，然後輸入指數。如果要讓游標回到基線，請按向右鍵(▶)。

附註：另請參考 **^(次方)**，頁碼：164。

範例：

2^3 $\quad$ 8

平方根範本

ctrl x² 鍵

$\sqrt{\square}$ **附註：**另請參考 **√()(平方根)**，頁碼：173。

範例：

$\sqrt{4}$ $\quad$ 2

$\sqrt{\{9,16,4\}}$ $\quad$ $\{3,4,2\}$

N 次方根範本

ctrl ^ 鍵

$\square$

$\sqrt[\square]{\square}$ **附註：**另請參考 **root()**，頁碼：119。

範例：

$\sqrt[3]{8}$ $\quad$ 2

$\sqrt[3]{\{8,27,15\}}$ $\quad$ $\{2,3,2.46621\}$

e 指數範本

$e^{\square}$

自然指數 *e* 提升到某乘冪

附註：另請參考 **e^()**，頁碼：41。

範例：

e^{1} 2.71828182846

對數範本

$\log_{\square}(\square)$

計算指定基底的對數。如果是預設基底 10，則可省略基底。

附註：另請參考 **log()**，頁碼：79。

範例：

$\log_{4}(2.)$ 0.5

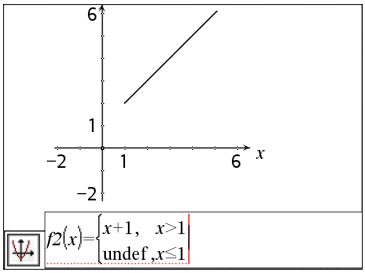
分段函數範本(2 段)

$\begin{cases}\square, & \square \\ \square, & \square\end{cases}$

讓您建立二段函數的運算式和條件。-如果要增加條件，請按一下範本並重複範本程序。

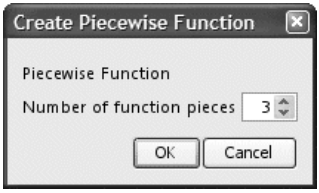
附註：另請參考 **piecewise()**，頁碼：102。

範例：



分段函數範本(N 段)

讓您建立 *N*- 段函數的運算式和條件。系統要求您輸入 *N*。



附註：另請參考 **piecewise()**，頁碼：102。

範例：

請參考分段函數範本(2 段) 的範例。

2 個聯立方程式範本

目錄 >

$\begin{cases} \square \\ \square \end{cases}$

建立兩個線性聯立方程式。如果要在現有聯立方程式中加入一列，請按一下範本並重複範本程序。

附註：另請參考 **system()**，頁碼：139。

範例：

$\text{solve}\left(\begin{cases} x+y=0 \\ x-y=5 \end{cases}, x, y\right) \qquad x=\frac{5}{2} \text{ and } y=\frac{-5}{2}$

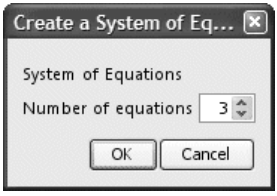
$\text{solve}\left(\begin{cases} y=x^2-2 \\ x+2\cdot y=-1 \end{cases}, x, y\right)$

$x=\frac{-3}{2} \text{ and } y=\frac{1}{4} \text{ or } x=1 \text{ and } y=-1$

N 個聯立方程式範本

目錄 >

讓您建立 *N* 個線性聯立方程式。系統要求您輸入 *N*。



附註：另請參考 **system()**，頁碼：139。

範例：

請參考聯立方程式(2 個方程式) 範本的範例。

絕對值範本

目錄 >

$|\square|$ **附註：**另請參考 **abs()**，頁碼：7。

範例：

$\left|\left\{2,-3,4,-4^3\right\}\right| \qquad \{2,3,4,64\}$

dd°mm'ss.ss" 範本

目錄 >

$\square°\square'\square''$

讓您以 **dd°mm'ss.ss''** 的格式輸入角度。**dd** 代表十進位度數、**mm** 代表分、**ss.ss** 代表秒。

範例：

30°15'10" 0.528011

矩陣範本 (2 x 2)

目錄 >

$$\begin{bmatrix} \square & \square \\ \square & \square \end{bmatrix}$$

建立 2 x 2 的矩陣。

範例：

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \cdot 5 \qquad \begin{bmatrix} 5 & 10 \\ 15 & 20 \end{bmatrix}$$

矩陣範本 (1 x 2)

目錄 >

$$\begin{bmatrix} \square & \square \end{bmatrix}.$$

範例：

$$\text{crossP}\left(\begin{bmatrix} 1 & 2 \end{bmatrix}, \begin{bmatrix} 3 & 4 \end{bmatrix}\right) \qquad \begin{bmatrix} 0 & 0 & -2 \end{bmatrix}$$

矩陣範本 (2 x 1)

目錄 >

$$\begin{bmatrix} \square \\ \square \end{bmatrix}$$

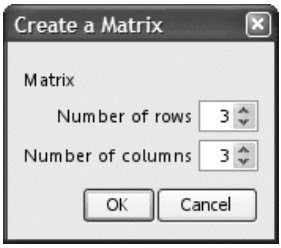
範例：

$$\begin{bmatrix} 5 \\ 8 \end{bmatrix} \cdot 0.01 \qquad \begin{bmatrix} 0.05 \\ 0.08 \end{bmatrix}$$

矩陣範本 (m x n)

目錄 >

系統要求您指定列數和行數後，就會顯示範本。



附註：如果您建立的矩陣有大量行列，可能就要等一下才會出現矩陣。

範例：

$$\text{diag}\left(\begin{bmatrix} 4 & 2 & 6 \\ 1 & 2 & 3 \\ 5 & 7 & 9 \end{bmatrix}\right) \qquad \begin{bmatrix} 4 & 2 & 9 \end{bmatrix}$$

總和範本 (Σ)

目錄 >

$$\sum_{\square=\square}^{\square}(\square)$$

附註：另請參考 **Σ()(sumSeq)**，頁碼：174。

範例：

$$\sum_{n=3}^{7}(n) \qquad 25$$

乘積範本 (Π)

目錄 >

$$\prod_{\square=\square}^{\square}(\square)$$

附註：另請參考 **Π()(prodSeq)**，頁碼：173。

範例：

$$\prod_{n=1}^{5}\left(\frac{1}{n}\right) \qquad \frac{1}{120}$$

一階導數範本

目錄 >

$$\frac{d}{d\square}(\square)$$

一階導數範本利用自動微分法，即可計算某個點的一階數值導數。

附註：另請參考 **d()(導數)**，頁碼：172。

範例：

$$\frac{d}{dx}(|x|)|x=0 \qquad \text{undef}$$

二階導數範本

目錄 >

$$\frac{d^2}{d\square^2}(\square)$$

二階導數範本利用自動微分法，即可計算某個點的二階數值導數。

附註：另請參考 **d()(導數)**，頁碼：172。

範例：

$$\frac{d^2}{dx^2}(x^3)|x=3 \qquad 18$$

定積分範本

$$\int_{\square}^{\square} \square \, d\square$$

定積分範本可利用與 nInt() 相同的方法，執行定積分的數值計算作業。

附註：另請參考**nInt()**，頁碼：94。

範例：

$$\int_{0}^{10} x^2 \, dx \qquad 333.333$$

依字母順序列表

名稱非字母的項目 (例如 +、!、>) 列在本節最後面，從第 頁碼：161 頁開始。除非另行指定，否則本節中的所有範例皆是以預設的重置模式執行，並假設所有變數都是未定義。

A

abs()

目 錄 >

abs(*值 1***)⇒***值*

abs(*列表 1***)⇒***列表*

abs(*矩陣 1***)⇒***矩陣*

傳回引數的絕對值。

附註：另請參考**絕對值範本**，頁碼：3。

如果引數是複數，則傳回數字的模數。

$$\left|\left\{\frac{\pi}{2},\frac{-\pi}{3}\right\}\right| \qquad \{1.5708,1.0472\}$$

$$|2-3\cdot i| \qquad 3.60555$$

amortTbl()

目 錄 >

amortTbl(*NPmt***,***N***,***I***,***PV***, [***Pmt***], [***FV***], [***PpY***], [***CpY***], [***PmtAt***], [***四捨五入值***])⇒***矩陣***

根據一組貨幣時間價值引數，傳回攤銷表矩陣的攤銷函數。

NPmt 是要加入表中的付款期數。表格以第一期付款開始。

N,I,PV,Pmt,FV,PpY,CpY,PmtAt 的說明位於貨幣時間價值引數表 (第 頁碼：148 頁) 。

- 如省略 *Pmt*，則會以 *Pmt*=**tvmPmt (***N,I,PV,FV,PpY,CpY,PmtAt***)** 為預設值。
- 如省略 *FV*，則會以 *FV*=0 為預設值。
- *PpY*、*CpY*、*PmtAt* 的預設值和貨幣時間價值函數相同。

*四捨五入值*用於指定四捨五入的小數位數。預設值 = 2。

amortTbl(12,60,10,5000,,,12,12)

0	0.	0.	5000.
1	-41.67	-64.57	4935.43
2	-41.13	-65.11	4870.32
3	-40.59	-65.65	4804.67
4	-40.04	-66.2	4738.47
5	-39.49	-66.75	4671.72
6	-38.93	-67.31	4604.41
7	-38.37	-67.87	4536.54
8	-37.8	-68.44	4468.1
9	-37.23	-69.01	4399.09
10	-36.66	-69.58	4329.51
11	-36.08	-70.16	4259.35
12	-35.49	-70.75	4188.6

amortTbl()

目錄 >

所產生的矩陣行順序為：付款期數、支付利息金額、支付本金金額和結算餘額。

第 *n* 列顯示的餘額是付款 *n* 次後的餘額。

您可使用輸出矩陣當作其他攤銷函數的輸入，包括 **ΣInt()** 和 **ΣPrn()**(第 頁碼：174 頁) 與 **bal()**(第 頁碼：15 頁) 。

and

目錄 >

布林運算式1 **and** *布林運算式2*⇒*布林運算式*

布林列表1 **and** *布林列表2*⇒*布林列表*

布林矩陣1 **and** *布林矩陣2*⇒*布林矩陣*

傳回真偽值或原始輸入的化簡形式。

*整數1***and***整數2*⇒*整數*

利用 **and** 功能個別位元比較兩個實際整數。系統內部會把兩個整數轉換成有正負號的 64 位元二進位數字。比較對應的位元時，如果兩個位元都是 1，則結果是 1，否則結果是 0。傳回的值代表位元結果，並會以基底模式為顯示依據。

您可以用任何整數做為基底。如果要輸入二進位或十六進位數字，則必須在前面分別加上 0b 或 0h 字首。如果沒有加上字首，就會把整數視為十進位 (以 10 為底) 。

十六進位基底模式：

0h7AC36 and 0h3D5F	0h2C16

重要訊息：數字零，而非英文字母 O。

二進位基底模式：

0b100101 and 0b100	0b100

十進位基底模式：

37 and 0b100	4

附註：您可輸入最長 64 位的二進位數字(不含 0b 字首) 。您可輸入最長 16 位的十六進位數字。

angle()

目錄 >

angle(*值1***)**⇒*值*

傳回引數的角度，並將引數解譯成複數。

度數角模式：

$\text{angle}(0+2\cdot i)$	90

angle()

目錄 > 📖

梯度角模式：

angle(0+3·*i*)	100

弧度角模式：

angle(1+***i***)	0.785398
angle({1+2·***i***,3+0·***i***,0−4·***i***})	{1.10715,0.,⁻1.5708}

angle(*列表1***)⇒***列表***

angle(*列表1***)⇒***矩陣***

傳回*列表1*或*矩陣1*中各元素的角度列表或矩陣，將每個元素解譯成複數，代表二維直角座標點。

ANOVA

目錄 > 📖

ANOVA *列表1*,*列表2*[,*列表3*,...,*列表20*][,*旗標*]

執行單因子變異數分析，比較二至 20 個母群體的平均值。*stat.results* 變數會儲存結果摘要(請參閱第 頁碼：134 頁)。

旗標=0 時用於資料，*旗標*=1 時用於統計

輸出變數	說明
stat.F	F 統計值
stat.PVal	無效假說被否定之最低意義標準
stat.df	群組自由度
stat.SS	群組的平方和
stat.MS	群組的均方
stat.dfError	誤差的自由度
stat.SSError	誤差的平方和
stat.MSError	誤差的均方

輸出變數	說明
stat.sp	合併標準差
stat.xbarlist	所輸入列表的平均值
stat.CLowerList	每個輸入列表平均值的 95% 信賴區間
stat.CUpperList	每個輸入列表平均值的 95% 信賴區間

ANOVA2way

目錄 >

ANOVA2way *列表1*,*列表2*[,*列表3*,...,*列表10*][,*levRow*]

計算雙因子變異數分析，比較二至 10 個母群體的平均值。*stat.results* 變數會儲存結果摘要(請參閱第 頁碼：134 頁)。

LevRow=0 時用於區集

LevRow=2,3,...,*Len*-1 時用於雙因子，其中 *Len*=長度(*列表1*)=長度(*列表2*) = ... = 長度(*列表10*) 與 *Len* / *LevRow* ∈ {2,3,...}

輸出：區集設計 (Block Design)

輸出變數	說明
stat.F	F 統計量
stat.PVal	無效假說被否定之最低意義標準
stat.df	行因子的自由度
stat.SS	行因子的平方和
stat.MS	行因子的均方
stat.FBlock	F 統計量
stat.PValBlock	無效假說被否定之最低機率
stat.dfBlock	因子的自由度
stat.SSBlock	因子的平方和
stat.MSBlock	因子的均方
stat.dfError	誤差的自由度
stat.SSError	誤差的平方和

輸出變數	說明
stat.MSError	誤差的均方
stat.s	誤差的標準差

行因子輸出

輸出變數	說明
stat.Fcol	F 統計量
stat.PValCol	行因子的機率值
stat.dfCol	行因子的自由度
stat.SSCol	行因子的平方和
stat.MSCol	行因子的均方

列因子輸出

輸出變數	說明
stat.FRow	F 統計量
stat.PValRow	列因子的機率值
stat.dfRow	列因子的自由度
stat.SSRow	列因子的平方和
stat.MSRow	列因子的均方

相互影響的輸出

輸出變數	說明
stat.FInteract	F 統計量
stat.PValInteract	相互影響的機率值
stat.dfInteract	相互影響的自由度
stat.SSInteract	相互影響的平方和
stat.MSInteract	相互影響的均方

誤差的輸出

輸出變數	說明
stat.dfError	誤差的自由度

輸出變數	說明
stat.SSError	誤差的平方和
stat.MSError	誤差的均方
s	誤差的標準差

Ans

ctrl (–) 鍵

Ans⇒*值*

傳回最近的運算式求值的結果。

56	56
56+4	60
60+4	64

approx()

目錄 >

approx(*值 1***)**⇒*數字*

若可能則以包含十進位值的運算式，傳回引數的求值，不管目前的**自動或近似值**模式。

相當於輸入引數並按 ctrl enter。

$\text{approx}\left(\frac{1}{3}\right)$	0.333333
$\text{approx}\left(\left\{\frac{1}{3},\frac{1}{9}\right\}\right)$	{0.333333,0.111111}
approx({sin(π),cos(π)})	{0.,-1.}
approx([√2 √3])	[1.41421 1.73205]
$\text{approx}\left(\left[\frac{1}{3}\ \frac{1}{9}\right]\right)$	[0.333333 0.111111]

approx(*列表 1***)**⇒*列表*

approx(*矩陣 1***)**⇒*矩陣*

傳回列表或*矩陣*，若可能則將所有元素都求值成十進位值。

approx({sin(π),cos(π)})	{0.,-1.}
approx([√2 √3])	[1.41421 1.73205]

▸approxFraction()

目錄 >

值 ▸*approxFraction(*[*Tol*]*)*⇒*值*

列表 ▸**approxFraction(**[*Tol*]**)**⇒*列表*

矩陣 ▸**approxFraction(**[*Tol*]**)**⇒*矩陣*

使用 *Tol* 為容許值，傳回輸入量的分數值。如果省略 *Tol* 設定，則使用 5.E-14 為容許值。

附註：如果要從電腦鍵盤插入本函數，可輸入 `@>approxFraction(…)`。

$\frac{1}{2}+\frac{1}{3}+\tan(\pi)$	0.833333
0.83333333333333▸approxFraction(5.E⁻14)	$\frac{5}{6}$
{π,1.5}▸approxFraction(5.E⁻14)	$\left\{\frac{5419351}{1725033},\frac{3}{2}\right\}$

approxRational()

目錄 > 📖

approxRational(*值***[, *Tol*])⇒***值***

approxRational(*列表***[, *Tol*])⇒***列表***

approxRational(*矩陣***[, *Tol*])⇒***矩陣***

使用 *Tol* 為容許值，傳回引數的分數值。如果省略 *Tol* 設定，則使用 5.E-14 為容許值。

$$\text{approxRational}\left(0.333,5\cdot 10^{-5}\right) \qquad \frac{333}{1000}$$

$$\text{approxRational}\left(\left\{0.2,0.33,4.125\right\},5.\text{E}^{-}14\right) \qquad \left\{\frac{1}{5},\frac{33}{100},\frac{33}{8}\right\}$$

arccos()

請參考 $\cos^{-1}()$，頁碼：25。

arccosh()

請參考 $\cosh^{-1}()$，頁碼：26。

arccot()

請參考 $\cot^{-1}()$，頁碼：27。

arccoth()

請參考 $\coth^{-1}()$，頁碼：28。

arccsc()

請參考 $\csc^{-1}()$，頁碼：30。

arccsch()

請參考 $\text{csch}^{-1}()$，頁碼：31。

arcsec()

請參考 $\sec^{-1}()$，頁碼：122。

arcsech()

請參考 $\text{sech}^{-1}()$，頁碼：123。

arcsin()

請參考 $\sin^{-1}$()，頁碼：130。

arcsinh()

請參考 $\sinh^{-1}$()，頁碼：131。

arctan()

請參考 $\tan^{-1}$()，頁碼：140。

arctanh()

請參考 $\tanh^{-1}$()，頁碼：141。

augment()

目錄 >

augment(*列表1, 列表2***)⇒***列表*

傳回將*列表2*接到*列表1*結尾的新列表。

augment({1,-3,2},{5,4}) {1,-3,2,5,4}

augment(*矩陣1***,** *矩陣2***)⇒***矩陣*

傳回將*矩陣2*加到*矩陣1*的新矩陣。使用「,」字元時，矩陣的列數必須相同，並將*矩陣2*加到*矩陣1*中成為新的行。並不會更改*矩陣1*或*矩陣2*。

$\begin{bmatrix}1 & 2\\3 & 4\end{bmatrix}\rightarrow m1$ $\begin{bmatrix}1 & 2\\3 & 4\end{bmatrix}$

$\begin{bmatrix}5\\6\end{bmatrix}\rightarrow m2$ $\begin{bmatrix}5\\6\end{bmatrix}$

augment(*m1*,*m2*) $\begin{bmatrix}1 & 2 & 5\\3 & 4 & 6\end{bmatrix}$

avgRC()

目錄 >

avgRC(*運算式1***,** *Var* [=*值*] [**,** *步階*]**)⇒** *運算式*

avgRC(*運算式1***,** *Var* [=*值*] [**,** *列表1*]**)⇒** *列表*

avgRC(*列表1***,** *Var* [=*值*] [**,** *步階*]**)⇒***列表*

avgRC(*矩陣1***,** *Var* [=*值*] [**,** *步階*]**)⇒***矩陣*

傳回向前差商(平均變化率)。

x:=2 2

avgRC(x^2-x+2,x) 3.001

avgRC(x^2-x+2,x,.1) 3.1

avgRC(x^2-x+2,x,3) 6

avgRC()

目錄 > 📖

運算式1 可以是使用者定義的函數名稱(請參考 **Func**) 。

指定*值*時，此值會取代之前賦值的所有變數，或目前以「|」取代的所有該變數。

Step 代表步階值。如果省略*步階*，則會以 0.001 為預設值。

請注意，類似函數 **centralDiff()** 使用中央差商。

B

bal()

目錄 > 📖

bal(*NPmt,N,I,PV* **,[***Pmt***], [***FV***], [***PpY***], [***CpY***], [***PmtAt***], [***四捨五入值***])⇒***值***

bal(*NPmt,amortTable***)⇒***值***

攤銷函數可計算在某次付款後的每期餘額時間表。

N,I,PV,Pmt,FV,PpY,CpY,PmtAt 的說明位於貨幣時間價值引數表(第 頁碼：148 頁) 。

NPmt 代表付款期數，您就是要計算該期後的資料。

N,I,PV,Pmt,FV,PpY,CpY,PmtAt 的說明位於貨幣時間價值引數表(第 頁碼：148 頁) 。

- 如省略 *Pmt*，則會以 *Pmt*=**tvmPmt (***N,I,PV,FV,PpY,CpY,PmtAt***)** 為預設值。
- 如省略 *FV*，則會以 *FV*=0 為預設值。
- *PpY*、*CpY*、*PmtAt* 的預設值和貨幣時間價值函數相同。

*四捨五入值*用於指定四捨五入的小數位數。預設值 = 2。

bal(*NPmt,amortTable***)** 可根據 *amortTable* 攤銷表計算付款第 *NPmt* 期後的餘額。*amortTable* 引數必須是 **amortTbl()** 一節所描述的矩陣形式(第 頁碼：7 頁) 。

bal(5,6,5.75,5000,,12,12) 833.11

tbl:=amortTbl(6,6,5.75,5000,,12,12)

0	0.	0.	5000.
1	-23.35	-825.63	4174.37
2	-19.49	-829.49	3344.88
3	-15.62	-833.36	2511.52
4	-11.73	-837.25	1674.27
5	-7.82	-841.16	833.11
6	-3.89	-845.09	-11.98

bal(4,*tbl*) 1674.27

bal()

目錄 > 📖

附註：另請參考 **ΣInt()** 與 **ΣPrn()**，頁碼：174。

▶Base2

目錄 > 📖

整數1 **▶Base2⇒***整數*

附註：如果要從電腦鍵盤插入本運算子，可輸入 `@>Base2`。

把*整數1*轉換成二進位數字。二進位或十六進位數字前面一定分別有 0b 或 0h 字首。數字零，而非英文字母 O，後面跟著 b 或 h。

0b *二進位數字*

0h *十六進位數字*

二進位數字最長可到 64 位。十六進位數字最長可到 16 位。

如果沒有加上字首，就會把*整數1*視為十進位(以 10 為底) 。無論使用何種基底模式，都以二進位制顯示結果。

負數以「2的補數」形式顯示。例如：

−1 顯示為

0hFFFFFFFFFFFFFFFF (十六進位基底模式)

0b111…111(64 個 1) (二進位基底模式)

-2^{63} 顯示為

0h8000000000000000 (十六進位基底模式)

0b100…000(63 個零) (二進位基底模式)

如果您輸入的十進位整數，超出具正負號 64 位元二進位格式的範圍，系統會以對稱模數運算法使數值落入適當範圍。請看以下超出範圍的數值範例。

2^{63} 變成 -2^{63}，並顯示為

256▶Base2	0b100000000
0h1F▶Base2	0b11111

▶Base2

目錄 > 📖

0h8000000000000000（十六進位基底模式）

0b100...000(63 個零）（二進位基底模式）

2^{64} 變成 0，並顯示為

0h0（十六進位基底模式）

0b0（二進位基底模式）

$-2^{63}-1$ 變成 $2^{63}-1$，並顯示為

0h7FFFFFFFFFFFFFFF（十六進位基底模式）

0b111...111(64 個 1)（二進位基底模式）

▶Base10

目錄 > 📖

整數1 ▶**Base10**⇒*整數*

附註：如果要從電腦鍵盤插入本運算子，可輸入 **@>Base10**。

把 *整數1* 轉換成十進位(以 10 為底) 數字。輸入二進位或十六進位數字時，前面一定分別要有 0b 或 0h 字首。

0b *二進位數字*

0h *十六進位數字*

數字零，而非英文字母 O，後面跟著 b 或 h。

二進位數字最長可到 64 位。十六進位數字最長可到 16 位。

如果沒有加上字首，就會把 *整數1* 視為十進位。無論使用何種基底模式，都以十進位制顯示結果。

0b10011▶Base10	19
0h1F▶Base10	31

▶Base16

目錄 > 📖

整數1 ▶**Base16**⇒*整數*

附註：如果要從電腦鍵盤插入本運算子，可輸入 **@>Base16**。

256▶Base16	0h100
0b111100001111▶Base16	0hF0F

▶Base16

目錄 > 📖

把*整數1*轉換成十六進位數字。二進位或十六進位數字前面一定分別有 0b 或 0h 字首。

0b *二進位數字*

0h *十六進位數字*

數字零，而非英文字母 O，後面跟著 b 或 h。

二進位數字最長可到 64 位。十六進位數字最長可到 16 位。

如果沒有加上字首，就會把*整數1*視為十進位(以 10 為底) 。無論使用何種基底模式，都以十六進位制顯示結果。

如果您輸入的十進位整數太大，超出具正負號 64 位元二進位格式的範圍，系統會以對稱模數運算法使數值落入適當範圍。如需詳細資訊，請參閱 **▶Base2**，頁碼: 16。

binomCdf()

目錄 > 📖

binomCdf(*n,p***)**⇒*列表*

binomCdf(*n,p,下限,上限***)**⇒*數字*(若*下限*和*上限*是數字) 或*列表*(若*下限*和*上限*是列表)

binomCdf(*n,p,上限***)**for P(0≤X≤*上限*)⇒*數字*(若*上限*是數字) 或*列表*(若*上限*是列表)

計算 *n* 次試驗和每次試驗成功機率 *p* 的離散二項分布累積機率。

對於 P(X ≤ *上限*)，請設定 *下限*=0

binomPdf()

目錄 > 📖

binomPdf(*n,p***)**⇒*列表*

binomPdf(*n,p,XVal***)**⇒*數字*(若 *XVal* 是數字) 或*列表*(若 *XVal* 是列表)

計算 *n* 次試驗和每次試驗成功機率 *p* 的離散二項分布機率。

ceiling() 無條件進入法()

目錄 >

ceiling(*值1***)⇒***值*

傳回大於 ≥ 引數且最接近的整數。

引數可以是實數也可以是複數。

附註：另請參考 **floor()**。

ceiling(.456)	1.

ceiling(*列表1***)⇒***列表*

ceiling(*矩陣1***)⇒***矩陣*

傳回各元素無條件進入後的列表或矩陣。

ceiling({-3.1,1,2.5})	{-3.,1,3.}
ceiling([0 -3.2·i; 1.3 4])	[0 -3.·i; 2. 4]

centralDiff()

目錄 >

centralDiff(*運算式1***,***Var* [=*值*][,*步階*])⇒***運算式***

centralDiff(*運算式1***,***Var* [,*步階*])|***Var*=*值*⇒***運算式***

centralDiff(*運算式1***,***Var* [=*值*][,*列表*])⇒***列表***

centralDiff(*列表1***,***Var* [=*值*][,*步階*])⇒***列表***

centralDiff(*矩陣1***,***Var* [=*值*][,*步階*])⇒***矩陣***

使用中央差商公式傳回數值導數。

指定*值*時，此值會取代之前賦值的所有變數，或目前以「|」取代的所有該變數。

Step 代表步階值。如果省略*步階*，則會以 0.001 為預設值。

使用*列表1*或*矩陣1*時，運算作業會對應到列表中的值或矩陣元素。

附註：另請參考 **avgRC()**。

$\text{centralDiff}(\cos(x),x)\|x=\frac{\pi}{2}$	-1.

char()

目錄 >

char(*Integer***)⇒***字元*

傳回計算機字元集中屬於 *Integer* 編號字元的字元字串。*Integer* 的有效範圍是 0–65535。

char(38)	"&"
char(65)	"A"

χ22way

目錄 >

χ22way *obsMatrix*

chi22way *obsMatrix*

計算 χ^2 檢定，以用於 *obsMatrix* 觀察矩陣中計數的雙向交叉表之結合。*stat.results* 變數會儲存結果摘要(請參閱第 頁碼: 134 頁) 。

如要了解矩陣中有空元素時的影響，請參考°ß空元素°®，頁碼: 198。

輸出變數	說明
stat.χ^2	卡方檢定統計: sum (觀察 - 預期)^2/預期
stat.PVal	無效假說被否定之最低意義標準
stat.df	卡方檢定統計的自由度
stat.ExpMat	預期基本計數表的矩陣，假定無效假說
stat.CompMat	基本卡方統計貢獻的矩陣

χ2Cdf()

目錄 >

χ2Cdf(*下限***,***上限***,***df***)⇒***數字*(若 *下限*和 *上限*是數字) 或 *列表*(若 *下限*和 *上限*是列表)

chi2Cdf(*下限***,***上限***,***df***)⇒***數字*(若 *下限*和 *上限*是數字) 或 *列表*(若 *下限*和 *上限*是列表)

針對指定自由度 *df* 的 *下限*和 *上限*之間，計算 χ^2 分布機率。

對於 P($X \le$ *上限*)，請設定 *下限*=0

如要了解列表中有空元素時的影響，請參考°ß空元素°®，頁碼: 198。

χ^2GOF

χ^2GOF *觀察列表,預期列表,df*

chi2GOF *觀察列表,預期列表,df*

執行檢定，確認資料樣本的確來自符合特定分布的母群體。*觀察列表*是計數的列表，必須包含整數。*stat.results* 變數會儲存結果摘要(請參閱第 頁碼：134 頁) 。

如要了解列表中有空元素時的影響，請參考°ß空元素°®，頁碼：198。

輸出變數	說明
stat.χ^2	卡方檢定統計：sum (觀察 - 預期)^2/預期
stat.PVal	無效假說被否定之最低意義標準
stat.df	卡方檢定統計的自由度
stat.CompList	基本卡方統計貢獻

χ^2Pdf()

χ^2Pdf(*XVal*,*df*)⇒*數字*(若 *XVal* 是數字) 或*列表*(若 *XVal* 是列表)

chi2Pdf(*XVal*,*df*)⇒*數字*(若 *XVal* 是數字) 或*列表*(若 *XVal* 是列表)

針對指定自由度 *df*，計算 χ^2 分布在指定 *XVal* 值的機率密度函數(pdf) 。

如要了解列表中有空元素時的影響，請參考°ß空元素°®，頁碼：198。

ClearAZ

ClearAZ

清除目前問題空間中的所有單字元變數。

如果一個以上的變數已遭鎖定，本指令會顯示錯誤訊息，並只刪除未鎖定的變數。請參考 **unLock**，頁碼：151。

$5 \rightarrow b$	5
b	5
ClearAZ	*Done*
b	"Error: Variable is not defined"

ClrErr

ClrErr

清除錯誤狀態，並將系統變數 *errCode* 設定成零。

Try...Else...EndTry 段的 **Else** 子句應該使用 **ClrErr** 或 **PassErr**。若要處理或忽略錯誤，請使用 **ClrErr**。若不知如何處理錯誤，請用 **PassErr** 送到下一個錯誤處理程式。若已經沒有待處理的 **Try...Else...EndTry** 錯誤處理程式，則錯誤對話方塊將顯示為正常。

附註：另請參考第 頁碼：101 頁的 **PassErr** 和第 頁碼：145 頁的 **Try**。

輸入 範例的注意事項：關於輸入多行程式和函數定義的說明，請參閱產品手冊中的「計算工具」章節。

若需 **ClrErr** 的參考範例，請參考第 頁碼：145 頁 **Try 指令下的例 2。**

colAugment()

colAugment(*矩陣1***, *矩陣2***)⇒***矩陣***

傳回將*矩陣2*加到*矩陣1*的新矩陣。矩陣的行數必須相同，並將*矩陣2*加到*矩陣1*中成為新的列。並不會更改*矩陣1*或*矩陣2*。

$$\begin{bmatrix}1 & 2\\3 & 4\end{bmatrix}\rightarrow m1 \qquad \begin{bmatrix}1 & 2\\3 & 4\end{bmatrix}$$

$$\begin{bmatrix}5 & 6\end{bmatrix}\rightarrow m2 \qquad \begin{bmatrix}5 & 6\end{bmatrix}$$

$$\text{colAugment}(m1,m2) \qquad \begin{bmatrix}1 & 2\\3 & 4\\5 & 6\end{bmatrix}$$

colDim()

colDim(*矩陣***)⇒***運算式***

傳回*矩陣*包含的行數。

附註：另請參考 **rowDim()**。

$$\text{colDim}\left(\begin{bmatrix}0 & 1 & 2\\3 & 4 & 5\end{bmatrix}\right) \qquad 3$$

colNorm()

colNorm(*矩陣***)⇒***運算式***

傳回*矩陣*中各行元素取絕對值後加總之最大值。

附註：不得有未定義的矩陣元素。另請參考 **rowNorm()**。

$$\begin{bmatrix}1 & -2 & 3\\4 & 5 & -6\end{bmatrix}\rightarrow mat \qquad \begin{bmatrix}1 & -2 & 3\\4 & 5 & -6\end{bmatrix}$$

$$\text{colNorm}(mat) \qquad 9$$

conj()

目錄 > 📖

conj(*值 1***)⇒***值*

conj(*列表 1***)⇒***列表*

conj(*矩陣 1***)⇒***矩陣*

傳回引數的共軛複數。

$\text{conj}(1+2\cdot i)$	$1-2\cdot i$
$\text{conj}\left(\begin{bmatrix} 2 & 1-3\cdot i \\ -i & -7 \end{bmatrix}\right)$	$\begin{bmatrix} 2 & 1+3\cdot i \\ i & -7 \end{bmatrix}$

constructMat()

目錄 > 📖

constructMat (*Expr***,***Var1***,***Var2***,***numRows***,***numCols***) ⇒***矩陣*

根據引數傳回矩陣。

Expr 代表以 *Var1* 與 *Var2* 為變數的運算式。所產生矩陣中的各元素，是由對 *Var1* 與 *Var2* 求值的每個增量值之 *Expr* 而來。

Var1 是自動由第 **1** 列到第 *numRows* 列增量而成。*Var2* 是在每一列中從第 **1** 行到第 *numCols 行增量而成*。

$\text{constructMat}\left(\frac{1}{i+j},i,j,3,4\right)$	$\begin{bmatrix} \frac{1}{2} & \frac{1}{3} & \frac{1}{4} & \frac{1}{5} \\ \frac{1}{3} & \frac{1}{4} & \frac{1}{5} & \frac{1}{6} \\ \frac{1}{4} & \frac{1}{5} & \frac{1}{6} & \frac{1}{7} \end{bmatrix}$

CopyVar

目錄 > 📖

CopyVar *Var1*, *Var2*

CopyVar *Var1*., *Var2*.

CopyVar *Var1, Var2* 會把 *Var1* 變數的值複製到 *Var2* 變數，必要時會建立 *Var2*。*Var1* 變數必須有值。

若 *Var1* 是既有使用者定義函數的名稱，則將該函數的定義複製到 *Var2* 函數。您必須定義 *Var1* 函數。

Var1 必須符合變數命名規則，或必須是能化簡成符合規則之變數名稱的間接取值運算式。

Define $a(x)=\frac{1}{x}$	*Done*
Define $b(x)=x^2$	*Done*
CopyVar a,c: $c(4)$	$\frac{1}{4}$
CopyVar b,c: $c(4)$	16

CopyVar

目錄 >

CopyVar *Var1*., *Var2*.可把 *Var1*.變數群組的所有成員複製到 *Var2*.群組，必要時會建立 *Var2*.。

Var1.必須是既有變數群組的名稱(例如 *stat.nn* 統計結果) ，或是用 **LibShortcut()** 函數建立的變數。若 *Var2*. 已經存在，本指令就會取代兩個群組的所有共同成員，並加入還不存在的成員。若一個以上的 *Var2*.成員已遭鎖定，則 *Var2*.的所有成員都會保留原樣。

aa.a:=45	45
aa.b:=6.78	6.78
CopyVar *aa.*,*bb.*	*Done*
getVarInfo()	*aa.a* "NUM" "" 0 *aa.b* "NUM" "" 0 *bb.a* "NUM" "" 0 *bb.b* "NUM" "" 0

corrMat()

目錄 >

corrMat(*列表1*,*列表2*[,...[,*列表20*]])

計算擴增矩陣 [*列表1, 列表2, ..., 列表20*] 的相關矩陣。

cos()

trig 鍵

cos(*值1***)⇒***值*

cos(*列表1***)⇒***列表*

cos(*值1***)** 可將引數的餘弦傳回為值。

cos(*列表1***)**：傳回*列表1* 中各元素之餘弦的列表。

附註：系統會根據目前的角度模式設定，將引數解譯為度數角、梯度角或弧度角。您可使用 °、G 或 r 來暫時覆寫角度模式。

度數角模式：

$\cos\left(\left(\frac{\pi}{4}\right)^{r}\right)$	0.707107
$\cos(45)$	0.707107
$\cos(\{0,60,90\})$	$\{1.,0.5,0.\}$

梯度角模式：

$\cos(\{0,50,100\})$	$\{1.,0.707107,0.\}$

弧度角模式：

$\cos\left(\frac{\pi}{4}\right)$	0.707107
$\cos(45°)$	0.707107

cos(*方陣1***)⇒***方陣*

傳回*方陣1* 的矩陣餘弦。這和計算各元素的餘弦不同。

純量函數 f(A) 對*方陣1* (A) 進行運算時，計算結果時所用的算法如下：

弧度角模式：

cos()

trig 鍵

計算 A 的特徵值 (λ_i) 和特徵向量 (V_i)。

方陣1 必需可對角化。另外不能有尚未賦值的符號變數。

建立矩陣：

$$B = \begin{bmatrix} \lambda_1 & 0 & \dots & 0 \\ 0 & \lambda_2 & \dots & 0 \\ 0 & 0 & \dots & 0 \\ 0 & 0 & \dots & \lambda_n \end{bmatrix} \text{ and } X = [V_1, V_2, \dots, V_n]$$

則 $A = X\,B\,X^{-1}$且 $f(A) = X\,f(B)\,X^{-1}$。例如 $\cos(A) = X\cos(B)\,X^{-1}$，其中：

$\cos(B) =$

$$\begin{bmatrix} \cos(\lambda_1) & 0 & \dots & 0 \\ 0 & \cos(\lambda_2) & \dots & 0 \\ 0 & 0 & \dots & 0 \\ 0 & 0 & \dots & \cos(\lambda_n) \end{bmatrix}$$

執行所有計算作業時，都使用浮點運算法。

$$\cos\left(\begin{bmatrix} 1 & 5 & 3 \\ 4 & 2 & 1 \\ 6 & -2 & 1 \end{bmatrix}\right)$$

$$\begin{bmatrix} 0.212493 & 0.205064 & 0.121389 \\ 0.160871 & 0.259042 & 0.037126 \\ 0.248079 & -0.090153 & 0.218972 \end{bmatrix}$$

$\cos^{-1}()$

trig 鍵

**$\cos^{-1}$(*值1*)⇒*值*

**$\cos^{-1}$(*列表1*)⇒*列表*

$\cos^{-1}$(*值1*)：傳回餘弦是值1 的角度。

$\cos^{-1}$(*列表1*)：傳回*列表1* 各元素之反餘弦的列表。

附註：系統會根據目前的角度模式設定，將結果傳回為度數角、梯度角或弧度角。

附註：如果要從鍵盤插入本函數，可輸入 `arccos(...)`。

$\cos^{-1}$(*方陣1*)⇒*方陣*

傳回*方陣1* 的矩陣反餘弦。這和計算各元素的反餘弦不同。若要了解計算方式，請參考 **cos()**。

度數角模式：

$\cos^{-1}(1)$ 0.

梯度角模式：

$\cos^{-1}(0)$ 100.

弧度角模式：

$\cos^{-1}(\{0,0.2,0.5\})$

$\{1.5708, 1.36944, 1.0472\}$

弧度角模式與直角複數格式：

$\cos^{-1}()$ trig 鍵

方陣1 必需可對角化。結果一定會包含浮點數。

$$\cos^{-1}\left(\begin{bmatrix}1 & 5 & 3\\ 4 & 2 & 1\\ 6 & -2 & 1\end{bmatrix}\right)$$

$$\begin{bmatrix}1.73485+0.064606\cdot i & -1.49086+2.10514\\ -0.725533+1.51594\cdot i & 0.623491+0.77836\cdot\\ -2.08316+2.63205\cdot i & 1.79018-1.27182\cdot\end{bmatrix}$$

若要看到完整結果，請按 ▲，然後使用 ◀ 與 ▶ 移動游標。

cosh() 目錄 > 📖

cosh(*值1***)⇒***值*

cosh(*列表1***)⇒***列表*

cosh(*值1***)**：傳回引數的雙曲餘弦。

cosh(*列表1***)**：傳回*列表1* 各元素之雙曲餘弦的列表。

cosh(*方陣1***)⇒***方陣*

傳回*方陣1* 的矩陣雙曲餘弦。這和計算各元素的雙曲餘弦不同。若要了解計算方式，請參考 **cos()**。

方陣1 必需可對角化。結果一定會包含浮點數。

度數角模式：

$$\cosh\left(\left(\frac{\pi}{4}\right)^{r}\right) \qquad 1.74671\text{E}19$$

弧度角模式：

$$\cosh\left(\begin{bmatrix}1 & 5 & 3\\ 4 & 2 & 1\\ 6 & -2 & 1\end{bmatrix}\right)$$

$$\begin{bmatrix}421.255 & 253.909 & 216.905\\ 327.635 & 255.301 & 202.958\\ 226.297 & 216.623 & 167.628\end{bmatrix}$$

$\cosh^{-1}()$ 目錄 > 📖

cosh⁻¹(*值1***)⇒***值*

cosh⁻¹(*列表1***)⇒***列表*

$$\cosh^{-1}(1) \qquad 0$$

$$\cosh^{-1}(\{1,2.1,3\}) \qquad \{0,1.37286,1.76275\}$$

cosh⁻¹(*值1***)**：傳回引數的反雙曲餘弦。

cosh⁻¹(*列表1***)**：傳回*列表1* 各元素之反雙曲餘弦的列表。

附註：如果要從鍵盤插入本函數，可輸入 `arccosh(…)`。

cosh⁻¹(*方陣1***)⇒***方陣*

弧度角模式與直角複數格式：

cosh⁻¹()

目錄 >

傳回*方陣1*的矩陣反雙曲餘弦。這和計算各元素的反雙曲餘弦不同。若要了解計算方式，請參考 **cos()**。

*方陣1*必需可對角化。結果一定會包含浮點數。

$\cosh^{-1}\left(\begin{bmatrix}1 & 5 & 3\\ 4 & 2 & 1\\ 6 & -2 & 1\end{bmatrix}\right)$

$$\begin{bmatrix}2.52503+1.73485\cdot i & -0.009241-1.4908\\ 0.486969-0.725533\cdot i & 1.66262+0.623491\\ -0.322354-2.08316\cdot i & 1.26707+1.79018\end{bmatrix}$$

若要看到完整結果，請按 ▲，然後使用 ◀ 與 ▶ 移動游標。

cot()

trig 鍵

cot(*值1***)** ⇒ *值*

cot(*列表1***)** ⇒ *列表*

傳回 *Value1* 的餘切，或傳回*列表1*中各元素之餘切的列表。

附註：系統會根據目前的角度模式設定，將引數解譯為度數角、梯度角或弧度角。您可使用 °、G 或 r 來暫時覆寫角度模式。

度數角模式：

cot(45) 1.

梯度角模式：

cot(50) 1.

弧度角模式：

cot({1,2.1,3})

{0.642093,-0.584848,-7.01525}

cot⁻¹()

trig 鍵

cot⁻¹(*值1***)**⇒*值*

cot⁻¹(*列表1***)**⇒*列表*

傳回餘切是*值1*的角度，或傳回包含*列表1*中各元素之反餘切的列表。

附註：系統會根據目前的角度模式設定，將結果傳回為度數角、梯度角或弧度角。

附註：如果要從鍵盤插入本函數，可輸入 `arccot(…)`。

度數角模式：

cot⁻¹(1) 45.

梯度角模式：

cot⁻¹(1) 50.

弧度角模式：

cot⁻¹(1) 0.785398

coth()

目錄 >

coth(*值 1***)⇒***值*

coth(*列表 1***)⇒***列表*

傳回*值 1* 的雙曲餘切，或傳回*列表 1* 中各元素之雙曲餘切的列表。

coth(1.2)	1.19954
coth({1,3.2})	{1.31304,1.00333}

coth⁻¹()

目錄 >

coth⁻¹(*值 1***)⇒***值*

coth⁻¹(*列表 1***)⇒***列表*

傳回*值 1* 的反雙曲餘切，或傳回包含*列表 1* 中各元素之反雙曲餘切的列表。

附註：如果要從鍵盤插入本函數，可輸入 **arccoth(...)**。

$\coth^{-1}(3.5)$	0.293893
$\coth^{-1}(\{-2,2.1,6\})$	{-0.549306,0.518046,0.168236}

count()

目錄 >

count(*值 1 或列表 1* [,*值 2 或列表 2* [,...]])**⇒***值*

傳回引數(會求值成數值) 中所有元素的累積個數。

每個引數都可以是運算式、值、列表或矩陣。您可混合資料類型，並使用不同長度的引數。

系統會對列表、矩陣或儲存格範圍的各元素求值，判斷是否應該加入個數之計算。

您可在 Lists & Spreadsheet 應用程式中用儲存格範圍取代任何引數。

空元素會被忽略。如需空元素的詳細資訊，請參考第 頁碼：198 頁。

count(2,4,6)	3
count({2,4,6})	3
$\text{count}\left(2,\{4,6\},\begin{bmatrix}8 & 10\\12 & 14\end{bmatrix}\right)$	7

countif()

目錄 >

countif(*列表,準則***)⇒***值*

傳回*列表*中符合指定*準則*之所有元素的累積個數。

countIf({1,3,"abc",undef,3,1},3)	2

計算等於 3 的元素之個數。

countif()

*準則*可以有以下形式：

- 值、運算式、字串。例如 **3** 代表只計算*列表*中可化簡成 3 這個值的元素個數。
- 包含 **?**符號當作各元素預留位置的布林運算式。例如 **?<5** 代表只計算*列表*中小於 5 的元素個數。

您可在 Lists & Spreadsheet 應用程式中用儲存格範圍取代*列表*。

列表中的空元素會遭到忽略。如需空元素的詳細資訊，請參考第 頁碼：198 頁。

附註：另請參考第 頁碼：138 頁的 **sumIf ()** 和第 頁碼：52 頁的 **frequency()**。

countIf({"abc","def","abc",3},"def") 1

計算等於「def」的元素之個數。

countIf({1,3,5,7,9},?<5) 2

計算 1 和 3 的個數。

countIf({1,3,5,7,9},2<?<8) 3

計算 3、5、7 的個數。

countIf({1,3,5,7,9},?<4 or ?>6) 4

計算 1、3、7、9 的個數。

cPolyRoots()

cPolyRoots(*Poly*,*Var***)⇒***列表*

cPolyRoots(*係數表***)⇒***列表*

第一個語法 cPolyRoots(*Poly*,*Var*) 可傳回 *Poly* 多項式的複數根列表(對 *Var*) 。

Poly 必須是一個變數中的展開多項式。請勿使用未展開的形式，例如 $y^2 \cdot y + 1$ 或 $x \cdot x + 2 \cdot x + 1$

第二個語法 **cPolyRoots(***係數表***)** 可傳回*係數表中係數的複數根列表*。

附註：另請參考 **polyRoots()**，頁碼：103。

polyRoots(y^3+1,y) {-1}

cPolyRoots(y^3+1,y) $\left\{-1,\frac{1}{2}-\frac{\sqrt{3}}{2}\cdot i,\frac{1}{2}+\frac{\sqrt{3}}{2}\cdot i\right\}$

polyRoots($x^2+2\cdot x+1$,x) {-1,-1}

polyRoots(y^3+1,y) {-1}

cPolyRoots(y^3+1,y) {-1,0.5−0.866025·*i*,0.5+0.866025·*i*}

polyRoots($x^2+2\cdot x+1$,x) {-1,-1}

cPolyRoots({1,2,1}) {-1,-1}

crossP()

crossP(*列表1*, *列表2***)⇒***列表*

crossP({0.1,2.2,-5},{1,-0.5,0}) {-2.5,-5.,-2.25}

crossP()

目錄 >

將*列表1*和*列表2*的向量外積傳回為列表。

*列表1*和*列表2*的維數必須相同，且必須是 2 維或 3 維。

crossP(*向量1***, *向量2***)⇒***向量*

所傳回的列向量或行向量(依引數而定)，是*向量1*和*向量2*的向量外積(叉積)。

*向量1*和*向量2*必須同為列向量或同為行向量。兩個向量的維數必須相同，且必須是 2 維或 3 維。

crossP([1 2 3],[4 5 6])	[-3 6 -3]
crossP([1 2],[3 4])	[0 0 -2]

csc()

鍵

csc(*值1***)⇒***值*

csc(*列表1***)⇒***列表*

傳回*值1*的餘割，或傳回包含*列表1*中各元素之餘割的列表。

度數角模式：

csc(45)	1.41421

梯度角模式：

csc(50)	1.41421

弧度角模式：

$\csc\left(\left\{1,\frac{\pi}{2},\frac{\pi}{3}\right\}\right)$	{1.1884,1.,1.1547}

csc⁻¹()

鍵

csc⁻¹(*值1***)** ⇒ *值*

csc⁻¹(*列表1***)** ⇒ *列表*

傳回餘割是*值1*的角度，或傳回包含*列表1*中各元素之反餘割的列表。

附註：系統會根據目前的角度模式設定，將結果傳回為度數角、梯度角或弧度角。

附註：如果要從鍵盤插入本函數，可輸入 `arccsc(...)`。

度數角模式：

$\csc^{-1}(1)$	90.

梯度角模式：

$\csc^{-1}(1)$	100.

弧度角模式：

csc⁻¹()

trig 鍵

$\csc^{-1}(\{1,4,6\})$	$\{1.5708,0.25268,0.167448\}$

csch()

目錄 >

csch(*值 1***)** ⇒ *值*

csch(*列表 1***)** ⇒ *列表*

傳回*值 1* 的雙曲餘割，或傳回*列表 1* 中各元素之雙曲餘割的列表。

$\mathrm{csch}(3)$	0.099822
$\mathrm{csch}(\{1,2.1,4\})$	$\{0.850918,0.248641,0.036644\}$

csch⁻¹()

目錄 >

csch⁻¹(*值***)** ⇒ *值*

csch⁻¹(*列表 1***)** ⇒ *列表*

傳回*值 1* 的反雙曲餘割，或傳回包含*列表 1* 中各元素之反雙曲餘割的列表。

附註：如果要從鍵盤插入本函數，可輸入 **arccsch(…)**。

$\mathrm{csch}^{-1}(1)$	0.881374
$\mathrm{csch}^{-1}(\{1,2.1,3\})$	$\{0.881374,0.459815,0.32745\}$

CubicReg

目錄 >

CubicReg *X*, *Y*[, [*Freq*] [, *類別*, *包含*]]

計算 *X* 列表與 *Y* 列表的 y = a ·x³+b ·x²+c ·x+d 三次多項式迴歸(頻率為 *Freq*) 。*stat.results* 變數會儲存結果摘要(請參閱第 頁碼：134 頁) 。

所有列表的維數都必須相同，*包含*除外。

X、*Y* 是自變數和因變數列表。

Freq 是頻率值列表(非必要) 。*Freq* 的每個元素，可用於指定各 *X*、*Y* 對應資料點的出現頻率。預設值是 1。所有元素都必須是 ≥ 0 的整數。

*類別*是 *X*、*Y* 對應資料的數字或字串類別代號列表。

*包含*是一個以上類別代號的列表。只有本列表中類別代號所對應的資料項目，才會包含在計算作業中。

CubicReg

如要了解列表中有空元素時的影響，請參考°ß空元素°®，頁碼：198。

輸出變數	說明
stat.RegEqn	迴歸方程式：a ·x³+b ·x²+c ·x+d
stat.a、stat.b、stat.c、stat.d	迴歸係數
stat.R²	判定係數
stat.Resid	迴歸殘差
stat.XReg	實際用於迴歸的已修改 *X 列表*中的資料點列表(根據 *Freq*、*類別列表*、*包含類別的限制)*
stat.YReg	實際用於迴歸的已修改 *Y 列表*中的資料點列表(根據 *Freq*、*類別列表*、*包含類別的限制)*
stat.FreqReg	*stat.XReg* 與 *stat.YReg 對應的頻率列表*

cumulativeSum()

cumulativeSum(*列表 l***)⇒***列表*

傳回*列表 l*(從 1 元素開始) 中各元素的累積總和列表。

cumulativeSum({1,2,3,4}) {1,3,6,10}

cumulativeSum(*矩陣 l***)⇒***矩陣*

傳回*矩陣 l* 中各元素的累積總和矩陣。每個元素都是該行從上到下的累積總和。

列表 l 或*矩陣 l* 若有空元素，則會在得出的列表或矩陣中產生空元素。如需空元素的詳細資訊，請參考第 頁碼：198 頁。

$$\begin{bmatrix}1&2\\3&4\\5&6\end{bmatrix}\rightarrow m1 \qquad \begin{bmatrix}1&2\\3&4\\5&6\end{bmatrix}$$

cumulativeSum(*m1*) $\begin{bmatrix}1&2\\4&6\\9&12\end{bmatrix}$

Cycle

Cycle

立刻把控制權轉到目前迴圈的下一輪迴圈(**For**、**While**、**Loop**) 。

Cycle 不可用於三種迴圈結構以外(**For**、**While**、**Loop**) 。

以下函數列出從 1 到 100 但略過 50 的整數總和。

Cycle

目錄 > 📖

輸入範例的注意事項：關於輸入多行程式和函數定義的說明，請參閱產品手冊中的「計算工具」章節。

```
Define g()=Func                    Done
          Local temp,i
          0→temp
          For i,1,100,1
          If i=50
          Cycle
          temp+i→temp
          EndFor
          Return temp
          EndFunc
g()                                5000
```

▸Cylind

目錄 > 📖

向量 **▸Cylind**

附註：如果要從電腦鍵盤插入本運算子，可輸入 **@>Cylind**。

以圓柱座標形式 [r,∠θ, z] 顯示列向量或行向量。

*向量*必須剛好包含三個元素，可以是行向量也可以是列向量。

```
[2  2  3]▸Cylind
          [2.82843  ∠0.785398  3.]
```

D

dbd()

目錄 > 📖

dbd(*日期1,日期2***)⇒***值*

利用實際天數法傳回*日期1*和*日期2*之間的天數。

*日期1*與*日期2*可以是標準日曆日期範圍內的數字，也可以是數字列表。若*日期1*和*日期2*都是列表，則長度必須相同。

*日期1*和*日期2*必須在 1950 年到 2049 年範圍內。

您可輸入兩種格式的日期。小數點的位置可區分兩種日期格式。

MM.DDYY(美國常見格式)

DDMM.YY(歐洲常見格式)

```
dbd(12.3103,1.0104)                1
dbd(1.0107,6.0107)               151
dbd(3112.03,101.04)                1
dbd(101.07,106.07)               151
```

▸DD

目錄 > 📖

運算式1 **▸DD⇒***值*

列表1 **▸DD⇒***列表*

矩陣1 **▸DD⇒***矩陣*

附註：如果要從電腦鍵盤插入本運算子，可輸入 `@>DD`。

針對以度數表達的引數，傳回十進位相等值。系統根據角度模式設定，將引數解譯為梯度角、弧度角或度數角的數字、列表或矩陣。

度數角模式：

(1.5°)▸DD	1.5°
(45°22'14.3")▸DD	45.3706°
({45°22'14.3",60°0'0"})▸DD	{45.3706°,60°}

梯度角模式：

1▸DD	$\frac{9}{10}$°

弧度角模式：

(1.5)▸DD	85.9437°

▸Decimal

目錄 > 📖

數字1 **▸Decimal⇒***值*

列表1 **▸Decimal⇒***值*

矩陣1 **▸Decimal⇒***值*

附註：如果要從電腦鍵盤插入本運算子，可輸入 `@>Decimal`。

以十進位形式顯示引數。本運算子只能用於輸入線的末尾。

$\frac{1}{3}$▸Decimal	0.333333

Define

目錄 > 📖

Define *Var* **=** *運算式*

Define *Function*(*Param1*, *Param2*, ...)**=** *運算式*

定義變數 *Var* 或使用者定義的*函數*。

Param1 這類參數是將引數傳給函數時所用的預留位置。呼叫使用者定義的函數時，必須提供對應到參數的引數 (例如值或變數) 。呼叫時，函數會利用所提供的引數對*運算式*求值。

Define $g(x,y)=2\cdot x-3\cdot y$	*Done*
$g(1,2)$	-4
$1\rightarrow a$: $2\rightarrow b$: $g(a,b)$	-4
Define $h(x)=\text{when}(x<2,2\cdot x-3,-2\cdot x+3)$	*Done*
$h(-3)$	-9
$h(4)$	-5

Define

Var 和*函數*不能用系統變數或內建函數或指令的名稱。

附註：這個形式的 **Define** 與執行以下運算式相同：*運算式* → *函數*(*Param1,Param2*).

Define *Function*(*Param1*, *Param2*, ...)=
Func
區段
EndFunc

```
Define g(x,y)=Func                    Done
              If x>y Then
              Return x
              Else
              Return y
              EndIf
              EndFunc
g(3,-7)                                  3
```

Define *Program*(*Param1*, *Param2*, ...)=
Prgm
區段
EndPrgm

使用者定義的函數或程式，可利用這個形式執行整個區段的多條語句。

*區段*可以只是一個語句，也可以是分成多行的一連串語句。*區段*也可以包括運算式和指令（例如 **If**、**Then**、**Else**、**For**）。

```
Define g(x,y)=Prgm
              If x>y Then
              Disp x," greater than ",y
              Else
              Disp x," not greater than ",y
              EndIf
              EndPrgm
                                      Done
g(3,-7)
                      3 greater than -7
                                      Done
```

輸入範例的注意事項：關於輸入多行程式和函數定義的說明，請參閱產品手冊中的「計算工具」章節。

附註：另請參考第 頁碼：35 頁的 **Define LibPriv** 和第 頁碼：36 頁的 **Define LibPub**。

Define LibPriv

Define LibPriv *Var* = *運算式*

Define LibPriv *函數*(*Param1*, *Param2*, ...)= *運算式*

Define LibPriv *函數*(*Param1*, *Param2*, ...)=
Func
區段
EndFunc

Define LibPriv *程式*(*參數1*, *參數2*, ...)=
Prgm
區段
EndPrgm

Define LibPriv

目錄 > 📖

功能與 **Define** 相同，但只定義自訂的資料庫變數、函數或程式。自訂函數和程式並不會出現在目錄中。

附註：另請參考第 頁碼：34 頁的 **Define** 和第 頁碼：36 頁的 **Define LibPub**。

Define LibPub

目錄 > 📖

Define LibPub *Var* **=** *運算式*

Define LibPub *函數*(*Param1*, *Param2*, ...)= *運算式*

Define LibPub *函數*(*Param1*, *Param2*, ...)=
Func
區段
EndFunc

Define LibPub *程式*(*參數1*, *參數2*, ...)=
Prgm
區段
EndPrgm

功能與 **Define** 相同，但只定義公用的資料庫變數、函數或程式。儲存並重新整理資料庫後，公用函數和程式就會出現在目錄中。

附註：另請參考第 頁碼：34 頁的 **Define** 和第 頁碼：35 頁的 **Define LibPriv**。

deltaList()

請參考 ΔList()，頁碼：76。

DelVar

目錄 > 📖

DelVar *變數1*[, *變數2*] [, *變數3*] ...

DelVar *Var*.

刪除記憶體中的指定變數或變數群組。

$2 \rightarrow a$	2
$(a+2)^2$	16
DelVar a	*Done*
$(a+2)^2$	"Error: Variable is not defined"

DelVar

目錄 > 📖

如果一個以上的變數已遭鎖定，本指令會顯示錯誤訊息，並只刪除未鎖定的變數。請參考 **unLock，頁碼：151。**

DelVar Var. 會刪除 *Var*. 變數群組(例如 *stat.nn* 統計結果，或以 **LibShortcut()** 函數建立的變數*) 的所有成員*。這個形式中 **DelVar** 指令的點(.) 用途為只刪除變數群組；簡單變數 *Var* 則不受影響。

aa.a:=45	45
aa.b:=5.67	5.67
aa.c:=78.9	78.9
getVarInfo()	$\begin{bmatrix} aa.a & \text{"NUM"} & \text{"}\square\text{"} \\ aa.b & \text{"NUM"} & \text{"}\square\text{"} \\ aa.c & \text{"NUM"} & \text{"}\square\text{"} \end{bmatrix}$
DelVar *aa*.	*Done*
getVarInfo()	"NONE"

delVoid()

目錄 > 📖

delVoid(*列表1***)⇒***列表*

傳回*列表1* 內容的列表，但移除所有空元素。

如需空元素的詳細資訊，請參考第 頁碼：198 頁。

delVoid({1,void,3})	{1,3}

det()

目錄 > 📖

det(*方陣***[,** *容許值***])⇒***運算式*

傳回*方陣*的行列式值。

若任何矩陣元素的絕對值小於*容許值*，則亦可將此元素視為零。只有在矩陣中包含浮點數項目，而且不包含尚未賦值的任何符號變數時，才會使用此容許值。其他時候都會忽略*容許值*。

- 若您使用 ctrl enter 或將**自動或近似值**模式設定成近似值，則系統會利用浮點運算法執行計算作業。
- 若指令省略或未使用*容許值*，則預設容許值的計算方式如下：

  5E-14 ·**max(dim(***方陣***))** ·**rowNorm(***方陣***)**

$\det\left(\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}\right)$	-2
$\begin{bmatrix} 1.\text{E}20 & 1 \\ 0 & 1 \end{bmatrix} \rightarrow mat1$	$\begin{bmatrix} 1.\text{E}20 & 1 \\ 0 & 1 \end{bmatrix}$
det(*mat1*)	0
det(*mat1*,.1)	1.E20

diag()

目錄 > 📖

diag(*列表***)⇒***矩陣*

diag(*列陣***)⇒***矩陣*

diag(*行陣***)⇒***矩陣*

所傳回的矩陣，是由引數列表或矩陣主對角線上的值所構成。

```
diag([2 4 6])                    [2 0 0]
                                 [0 4 0]
                                 [0 0 6]
```

diag(*方陣***)⇒***列陣*

所傳回的列陣，包含*方陣*主對角線上的元素。

*方陣*必需是方陣。

```
[4 6 8]                          [4 6 8]
[1 2 3]                          [1 2 3]
[5 7 9]                          [5 7 9]
diag(Ans)                        [4 2 9]
```

dim()

目錄 > 📖

dim(*列表***)⇒***整數*

傳回*列表*的維數。

```
dim({0,1,2})                     3
```

dim(*矩陣***)⇒***列表*

以兩行列表 {列, 行} 傳回矩陣維度。

```
dim([1 -1])                      {3,2}
    [2 -2]
    [3  5]
```

dim(*String***)⇒***整數*

傳回 *String* 字元字串包含的字元數。

```
dim("Hello")                     5
dim("Hello "&"there")            11
```

Disp

目錄 > 📖

Disp *運算式或字串1* [, *運算式或字串2*] ...

顯示 *Calculator* 歷史記錄中的引數。畫面會連續顯示引數，並以短空格作區分。

主要用於程式與函數，以顯示計算過程。

輸入範例的注意事項：關於輸入多行程式和函數定義的說明，請參閱產品手冊中的「計算工具」章節。

```
Define chars(start,end)=Prgm
                        For i,start,end
                        Disp i," ",char(i)
                        EndFor
                        EndPrgm
                                        Done
chars(240,243)
                                    240 ð
                                    241 ñ
                                    242 ò
                                    243 ó
                                        Done
```

DispAt*int*, *運算式 1* [, *運算式 2* ...] ...

DispAt 可讓您指定特定運算式或字串顯示在螢幕上的哪一行。

您可使用運算式來指定行號。

請注意，行號不代表整個螢幕，而是緊接在指令/程式以後的區域。

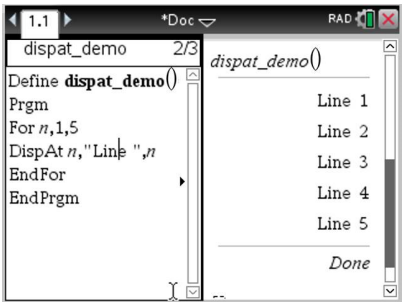
這個指令可讓程式產生類似儀表板的輸出畫面，將運算式的值或感應器的讀數更新在同一行中。

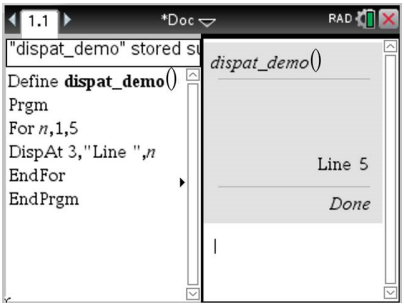
同一程式可同時使用 **DispAt** 和 **Disp**。

注意:最高行號設定為 8，因為此數值等於計算機螢幕的全螢幕行數，只要確定行中不含 2D 數學運算式即可。實際的行數必須根據螢幕所顯示的資訊內容而定。

DispAt

範例





示範範例:

Define z()=	輸出
Prgm	z()
For n,1,3	疊代 1:
DispAt 1,"N: ",n	行 1:N:1
Disp "Hello"	行 2:Hello
EndFor	
EndPrgm	疊代 2:
	行 1:N:2
	行 2:Hello
	行 3:Hello
	疊代 3:
	行 1:N:3

	行 2:Hello 行 3:Hello 行 4:Hello
Define z1()= Prgm For n,1,3 DispAt 1,"N: ",n EndFor For n,1,4 Disp "Hello" EndFor EndPrgm	z1() 行 1:N:3 行 2:Hello 行 3:Hello 行 4:Hello 行 5:Hello

錯誤狀況:

錯誤訊息	**說明**
DispAt 行號須介於 1 到 8 之間	運算式會計算超出 1 到 8 (含) 範圍的行號
引數太少	函數或指令缺少一個以上的引數。
沒有引數	與目前的「語法錯誤」對話方塊相同
引數太多	限制引數。錯誤與 Disp 相同。
資料類型無效	第一個引數必須是數字。
無效:DispAt 無效	"Hello World" 資料類型錯誤會在無效時發出 (如果已定義回調的話)

▸DMS

值 ▸*DMS*

列表 ▸**DMS**

矩陣 ▸**DMS**

附註: 如果要從電腦鍵盤插入本運算子，可輸入 `@>DMS`。

將引數解譯為角度，並顯示等值的 DMS(DDDDDD°MM'SS.ss'') 數字。請參考第 頁碼: 178 頁的 °、'、'' 說明 DMS(度、分、秒) 格式。

度數角模式:

(45.371)▸DMS	45°22'15.6"
({45.371,60})▸DMS	{45°22'15.6",60°}

▶DMS 目錄 >

附註：若用於弧度角模式，則 ▶DMS 可將弧度角轉成度數角。若輸入內容後面跟著度數符號 °，則不會轉換。**▶DMS** 只能用於輸入線的末尾。

dotP() 目錄 >

dotP(*列表1, 列表2***)⇒***運算式*

傳回兩個列表的「點」積。

$\text{dotP}(\{1,2\},\{5,6\})$ 17

dotP(*向量1, 向量2***)⇒***運算式*

傳回兩個向量的「點」積。

$\text{dotP}([1\ \ 2\ \ 3],[4\ \ 5\ \ 6])$ 32

兩者必須同為列向量或同為行向量。

E

e^() e^x 鍵

e^(*值1***)⇒***值*

傳回以 e 為底，*值1* 為冪的指數值。

e^1 2.71828

e^{3^2} 8103.08

附註：另請參考 **e 指數範本**，頁碼：2。

附註：按 e^x 可顯示 e^(與按鍵盤上的 E 字元不同）。

您可輸入 $re^{i\theta}$ 極座標式複數。但您只能在弧度角模式採用這種形式。若是在度數角或梯度角模式，則會導致值域錯誤。

e^(*列表1***)⇒***列表*

傳回以 e 為底，*列表1* 各元素為冪的指數值。

$e^{\{1,1.,0.5\}}$ $\{2.71828,2.71828,1.64872\}$

e^(*方陣1***)⇒***方陣*

傳回 *方陣1* 的矩陣指數。這和計算以 e 為底、以各元素為冪的指數值不同。若要了解計算方式，請參考 **cos()**。

$e^{\begin{bmatrix}1 & 5 & 3\\ 4 & 2 & 1\\ 6 & -2 & 1\end{bmatrix}}$ $\begin{bmatrix}782.209 & 559.617 & 456.509\\ 680.546 & 488.795 & 396.521\\ 524.929 & 371.222 & 307.879\end{bmatrix}$

方陣1 必需可對角化。結果一定會包含浮點數。

eff()

目錄 >

eff(*nominalRate,CpY***)⇒***值*

可將 *nominalRate* 名目利率轉成實質年利率的財務函數(已知 *CpY* 是每年複利期數) 。

nominalRate 必須是實數，*CpY* 必須是 > 0 的實數。

附註：另請參考 **nom()**，頁碼：94。

```
eff(5.75,12)                    5.90398
```

eigVc()

目錄 >

eigVc(*方陣***)⇒***矩陣*

所傳回的矩陣，包含實數或複數*方陣*的特徵向量。結果中的每一行代表一個特徵值。請注意，特徵向量並非唯一，可隨任何常數因式改變。特徵向量需正規化，亦即若 V = [x_1, x_2, … , x_n]，則：

$x_1^2 + x_2^2 + \ldots + x_n^2 = 1$

*方陣*首先要經由相似變換法予以平衡，直到列範數與行範數儘量接近同一值為止。接著*方陣*會簡化成上 Hessenberg 形式，並透過 Schur 因式分解法計算特徵向量。

直角複數格式：

```
[-1  2  5]
[ 3 -6  9]→m1                [-1  2  5]
[ 2 -5  7]                   [ 3 -6  9]
                             [ 2 -5  7]
eigVc(m1)
[-0.800906        0.767947                  (
 0.484029   0.573804+0.052258·i   0.5738▸
 0.352512   0.262687+0.096286·i   0.2626
```

若要看到完整結果，請按 ▲，然後使用 ◀ 與 ▶ 移動游標。

eigVl()

目錄 >

eigVl(*方陣***)⇒***列表*

傳回實數或複數*方陣*的特徵值列表。

*方陣*首先要經由相似變換法予以平衡，直到列範數與行範數儘量接近同一值為止。接著*方陣*會簡化成上 Hessenberg 形式，並從上 Hessenberg 矩陣計算特徵值。

直角複數格式模式：

```
[-1  2  5]
[ 3 -6  9]→m1                [-1  2  5]
[ 2 -5  7]                   [ 3 -6  9]
                             [ 2 -5  7]
eigVl(m1)
{-4.40941,2.20471+0.763006·i,2.20471−0.▸
```

若要看到完整結果，請按 ▲，然後使用 ◀ 與 ▶ 移動游標。

Else

請參考 If，頁碼：63。

ElseIf

目錄 > 📖

If *布林運算式1* **Then**
　區段1
ElseIf *布林運算式2* **Then**
　區段2
　⋮
ElseIf *布林運算式N* **Then**
　區段N
EndIf
　⋮

輸入範例的注意事項：關於輸入多行程式和函數定義的說明，請參閱產品手冊中的「計算工具」章節。

```
Define 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
                                Done
```

EndFor

請參考 **For**，頁碼：**50**。

EndFunc

請參考 **Func**，頁碼：**53**。

EndIf

請參考 **If**，頁碼：**63**。

EndLoop

請參考 **Loop**，頁碼：**82**。

EndPrgm

請參考 **Prgm**，頁碼：**105**。

EndTry

請參考 **Try**，頁碼：**145**。

EndWhile

請參考 **While**，頁碼：**154**。

euler(*Expr*, *Var*, *depVar*, {*Var0, VarMax*}, *depVar0*, *VarStep*[, *eulerStep*]**)** ⇒*矩陣*

euler(*SystemOfExpr*, *Var*, *ListOfDepVars*, {*Var0*, *VarMax*}, *ListOfDepVars0*, *VarStep* [, *eulerStep*]**)** ⇒*矩陣*

euler(*ListOfExpr*, *Var*, *ListOfDepVars*, {*Var0*, *VarMax*}, *ListOfDepVars0*, *VarStep* [, *eulerStep*]**)** ⇒**矩陣**

使用尤拉方法解出方程組

$$\frac{d\,depVar}{d\,Var} = Expr(Var, depVar)$$

其中，*depVar*(*Var0*)=*depVar0*，區間為 [*Var0,VarMax*]。傳回一個矩陣，其中第一列定義 *Var* 輸出值，且第二列定義位於對應 *Var* 值的第一個求解元素的值，依此類推。

Expr 是在右邊定義常微分方程式 (ODE)。

SystemOfExpr 是在右邊的方程組，定義 ODE 方程組 (對應於 *ListOfDepVars* 中的因變數順序)。

ListOfExpr 是在右邊的列表，定義 ODE 方程組 (對應於 *ListOfDepVars* 中的因變數順序)。

Var 是自變數。

ListOfDepVars 是因變數列表。

{*Var0, VarMax*} 是一個雙元素列表，表示函數從 *Var0* 積分至 *VarMax*。

ListOfDepVars0 是因變數初始值的列表。

VarStep 是非零數字，使 **sign(***VarStep***) = sign(***VarMax-Var0***)**，傳回 *Var0*+i·*VarStep* 對所有的 i=0,1,2,... 的解，使 *Var0*+i·*VarStep* 位於 [*var0,VarMax*] (在 *VarMax* 可能不會有解)。

微分方程式：

y'=0.001*y*(100-y) 與 y(0)=10

```
euler(0.001·y·(100−y),t,y,{0,100},10,1)
[ 0.    1.      2.        3.        4.  ]
[ 10.   10.9    11.8712   12.9174   14.042
```

若要看到完整結果，請按 ▲，然後使用 ◀ 與 ▶ 移動游標。

```
deSolve(y'=0.001·y·(100−y) and y(0)=10,t,y)
                                  100.·(1.10517)^t
                        y=─────────────────────────
                                (1.10517)^t+9.
```

```
seqGen(100.·(1.10517)^t/((1.10517)^t+9.),t,y,{0,100})
{10.,10.9367,11.9494,13.0423,14.2189
```

方程組：

$$\begin{cases} y1' = -y1 + 0.1 \cdot y1 \cdot y2 \\ y2' = 3 \cdot y2 - y1 \cdot y2 \end{cases}$$

其中，*y1*(0)=2 及 *y2*(0)=5

```
euler({−y1+0.1·y1·y2, 3·y2−y1·y2},t,{y1,y2},{0,5},{2,5},1)
[ 0.   1.   2.   3.   4.    5.    ]
[ 2.   1.   1.   3.   27.   243.  ]
[ 5.   10.  30.  90.  90.   −2070.]
```

euler()

eulerStep 是一個正整數 (預設值為 1)，定義在輸出值之間的尤拉步階數目。尤拉方法所用的實際步階大小是 *VarStep*/*eulerStep*。

eval ()

eval(*Expr***)** ⇒ *字串*

eval() 只有效在 TI-Innovator™ Hub 程式設計指令 **Get**、**GetStr** 和 **Send** 的指令引數中才有效。軟體會對運算式 *Expr* 求值，該結果會將 **eval()** 語句取代為字元字串。

引數 *Expr* 必須化簡成實數。

雖然 **eval()** 不會顯示其結果，但您可以在執行指令後，透過檢查以下任一項特殊變數來檢視求出的分享器指令字串。

iostr.SendAns
iostr.GetAns
iostr.GetAns

附註:另請參考 **Get** (頁碼: 55)、**GetStr** (頁碼: 61) 和 **Send** (頁碼: 123).

將 RGB LED 藍色元素的強度設為一半。

```
lum:=127                                   127
Send "SET COLOR.BLUE eval(lum)"            Done
```

將藍色元素重設為「關閉」。

```
Send "SET COLOR.BLUE OFF"                  Done
```

eval() 引數必須化簡成實數。

```
Send "SET LED eval("4") TO ON"
                    "Error: Invalid data type"
```

讓紅色元素淡入的程式

```
Define fadein()=
Prgm
For i,0,255,10
  Send "SET COLOR.RED eval(i)"
    Wait 0.1
  EndFor
  Send "SET COLOR.RED OFF"
EndPrgm
```

執行程式。

```
fadein()                                   Done
```

```
n:=0.25                                    0.25
m:=8                                       8
n·m                                        2.
Send "SET COLOR.BLUE ON TIME eval(n·m)"
                                           Done
iostr.SendAns    "SET COLOR.BLUE ON TIME 2"
```

Exit

目錄 >

Exit

結束目前的 **For**、**While** 或 **Loop** 區段。

Exit 不可用於三種迴圈結構以外(**For**、**While** 或 **Loop**) 。

輸入 範例的注意事項：關於輸入多行程式和函數定義的說明，請參閱產品手冊中的「計算工具」章節。

函數列表：

```
Define g()=Func                    Done
            Local temp,i
            0→temp
            For i,1,100,1
            temp+i→temp
            If temp>20 Then
            Exit
            EndIf
            EndFor
            EndFunc
g()                                21
```

exp()

e^x 鍵

exp(*值 1***)⇒***值*

傳回以 **e** 為底，*值 1* 為冪的指數值。

附註：另請參考 **e** 指數範本，頁碼：2。

您可輸入 $re^{i\theta}$ 極座標式複數。但您只能在弧度角模式採用這種形式。若是在度數角或梯度角模式，則會導致值域錯誤。

e^1 2.71828

e^{3^2} 8103.08

exp(*列表 1***)⇒***列表*

傳回以 **e** 為底，*列表 1* 各元素為冪的指數值。

$e^{\{1,1.,0.5\}}$ $\{2.71828,2.71828,1.64872\}$

exp(*方陣 1***)⇒***方陣*

傳回*方陣 1* 的矩陣指數。這和計算以 **e** 為底、以各元素為冪的指數值不同。若要了解計算方式，請參考 **cos()**。

方陣 1 必需可對角化。結果一定會包含浮點數。

$$e^{\begin{bmatrix}1&5&3\\4&2&1\\6&-2&1\end{bmatrix}} \quad \begin{bmatrix}782.209&559.617&456.509\\680.546&488.795&396.521\\524.929&371.222&307.879\end{bmatrix}$$

expr()

目錄 >

expr(*字串***)⇒***運算式*

將*字串*包含的字元字串傳回為運算式，並且立刻執行。

```
"Define cube(x)=x^3"→funcstr
                    "Define cube(x)=x^3"
expr(funcstr)                      Done
cube(2)                            8
```

ExpReg *X, Y* [, [*Freq*] [, *類別, 包含*]]

計算 *X* 列表與 *Y* 列表的 y = a ·(b)ˣ指數迴歸(頻率為 *Freq*) 。*stat.results* 變數會儲存結果摘要(請參閱第 頁碼：134 頁) 。

所有列表的維數都必須相同，*包含*除外。

X、*Y* 是自變數和因變數列表。

Freq 是頻率值列表(非必要) 。*Freq* 的每個元素，可用於指定各 *X*、*Y* 對應資料點的出現頻率。預設值是 1。所有元素都必須是 ≥ 0 的整數。

*類別*是 *X*、*Y* 對應資料的數字或字串類別代號列表。

*包含*是一個以上類別代號的列表。只有本列表中類別代號所對應的資料項目，才會包含在計算作業中。

如要了解列表中有空元素時的影響，請參考°ß空元素°®，頁碼：198。

輸出變數	說明
stat.RegEqn	迴歸方程式：a ·(b)ˣ
stat.a°Bstat.b	迴歸係數
stat.r2	所轉換資料之線性判定的係數
stat.r	所轉換資料 (x, ln(y)) 的相關係數
stat.Resid	與指數模型相關的殘差
stat.ResidTrans	與所轉換資料之線性擬合相關的殘差
stat.XReg	實際用於迴歸的已修改 *X 列表*中的資料點列表(根據 *Freq*、*類別列表*、*包含類別的限制)*
stat.YReg	實際用於迴歸的已修改 *Y 列表*中的資料點列表(根據 *Freq*、*類別列表*、*包含類別的限制)*
stat.FreqReg	*stat.XReg* 與 *stat.YReg 對應的頻率列表*

factor()

目錄 >

factor(*有理數***)**：傳回已因式分解成質數的有理數。若是合成數，則計算時間會隨著第二大因式的位數而大幅增加。例如對 30 位整數進行因式分解可能要花超過一天，對 100 位數進行因式分解可能要花超過一世紀。

factor(152417172689)	123457·1234577
isPrime(152417172689)	false

若要手動停止計算，

- **計算機**：按住 on 鍵並重複按 · 鍵。
- **Windows®**：按住 **F12** 鍵並重複按 **Enter** 鍵。
- **Macintosh®**：按住 **F5** 鍵並重複按 **Enter** 鍵。
- **iPad®**：應用程式顯示提示。您可以繼續等待或取消。

若您只想知道某數字是否為質數，請改用 **isPrime()**。此法速度快得多，若*有理數*不是質數，或如果第二大因數超過五位數，則能省更多時間。

FCdf()

目錄 >

FCdf(*下限*,*上限*,*dfNumer*,*dfDenom***)**⇒*數字*(*若下限和上限是數字*) 或*列表*(*若下限和上限是列表*)

FCdf(*下限*,*上限*,*dfNumer*,*dfDenom***)**⇒*數字*(*若下限和上限是數字*) 或*列表*(*若下限和上限是列表*)

針對所指定 *dfNumer*(自由度) 和 *dfDenom* 的*下限*和*上限*之間，計算 F 分布機率。

對於 P($X \le$)，請設定*下限*=0。

Fill

目錄 > 📖

Fill *值, 矩陣變數*⇒*矩陣*

將*矩陣變數*中的各元素替換成*值*。

*矩陣變數*必須已經存在。

$\begin{bmatrix}1 & 2\\3 & 4\end{bmatrix}\rightarrow amatrix$ $\begin{bmatrix}1 & 2\\3 & 4\end{bmatrix}$

Fill 1.01,*amatrix* *Done*

amatrix $\begin{bmatrix}1.01 & 1.01\\1.01 & 1.01\end{bmatrix}$

Fill *值, 列表變數*⇒*列表*

將*列表變數*中的各元素替換成*值*。

*列表變數*必須已經存在。

$\{1,2,3,4,5\}\rightarrow alist$ $\{1,2,3,4,5\}$

Fill 1.01,*alist* *Done*

alist $\{1.01,1.01,1.01,1.01,1.01\}$

FiveNumSummary

目錄 > 📖

FiveNumSummary *X*[,[*Freq*][,*類別*,*包含*]]

提供 *X* 列表的簡化版單元統計。*stat.results* 變數會儲存結果摘要(請參閱第 頁碼：134 頁)。

X 代表包含資料的列表。

Freq 是頻率值列表(非必要)。*Freq* 的每個元素，可用於指定各 *X*、*Y* 對應資料點的出現頻率。預設值是 1。

*類別*是 *X* 對應資料的數字類別代號列表。

*包含*是一個以上類別代號的列表。只有本列表中類別代號所對應的資料項目，才會包含在計算作業中。

X、*Freq*、*類別*的任何列表中若有空元素，則所有這些列表的對應元素就會是空元素。如需空元素的詳細資訊，請參考第 頁碼：198 頁。

輸出變數	說明
stat.MinX	x 值的最小值。
stat.Q_1X	x 的第一四分位數。
stat.MedianX	x 的中位數。
stat.Q_3X	x 的第三四分位數。
stat.MaxX	x 值的最大值。

floor()

目錄 >

floor(*值 1***)⇒***整數*

傳回小於 ≤ 引數的最大整數。本功能與 **int()** 相同。

引數可以是實數也可以是複數。

floor(⁻2.14)	⁻3.

floor(*列表 1***)⇒***列表*

floor(*矩陣 1***)⇒***矩陣*

傳回各元素無條件捨去值的列表或矩陣。

floor({3/2,0,⁻5.3})	{1,0,⁻6.}
floor([1.2 3.4; 2.5 4.8])	[1. 3.; 2. 4.]

附註：另請參考 **ceiling()** 與 **int()**。

For

目錄 >

For *Var*, *低*, *高* [, *步階*]

區段

EndFor

針對 *Var* 各個值反覆執行*區段*中的語句，順序為由*低*到*高*，*步階*則是增加單位。

Var 不得是系統變數。

*步階*值可正可負。預設值是 1。

*區段*可以只是一個語句，也可以是由「:」字元分隔的一連串語句。

輸入 範例的注意事項：關於輸入多行程式和函數定義的說明，請參閱產品手冊中的「計算工具」章節。

```
Define g()=Func                    Done
        Local tempsum,step,i
        0→tempsum
        1→step
        For i,1,100,step
        tempsum+i→tempsum
        EndFor
        EndFunc
```

g()	5050

format()

目錄 >

format(*值*[, *格式字串*])⇒*字串*

根據格式範本將*值*傳回為字元字串。

*格式字串*的格式必須如下：「F[n]」、「S[n]」、「E[n]」、「G[n][c]」，[ ] 代表非必要部份。

F[n]：固定格式。n 代表小數點後要顯示的位數。

format(1.234567,"f3")	"1.235"
format(1.234567,"s2")	"1.23E0"
format(1.234567,"e3")	"1.235E0"
format(1.234567,"g3")	"1.235"
format(1234.567,"g3")	"1,234.567"
format(1.234567,"g3,r:")	"1:235"

format()

目錄 >

S[n]：科學格式。n 代表小數點後要顯示的位數。

E[n]：工程格式。n 代表第一個有效位數後要顯示的位數。指數調整成 3 的倍數，且小數點往右移 0、1 或 2 位。

G[n][c]：與固定格式相同，但也將數基點左邊的數字分為 3 個一組。c 用於指定分組字元，預設為逗號。若 c 是句點，則將數基點顯示為逗號。

[Rc]：以上任何指定符號都可加上 Rc 數基點標示的下標，其中 c 是單一字元，用於指定取代數基點的符號。

fPart()

目錄 >

fPart(*運算式1***)⇒***運算式*

fPart(*列表1***)⇒***列表*

fPart(*矩陣1***)⇒***矩陣*

傳回引數的分數部份。

若是列表或矩陣，則傳回各元素的分數部份。

引數可以是實數也可以是複數。

```
fPart(-1.234)                    -0.234
fPart({1,-2.3,7.003})      {0,-0.3,0.003}
```

FPdf()

目錄 >

FPdf(*XVal***,***dfNumer***,***dfDenom***)⇒***數字*(若 *XVal* 是數字) 或*列表*(若 *XVal* 是列表)

針對指定 *dfNumer*(自由度) 和 *dfDenom*，計算 *XVal* 的 F 分布機率。

freqTable▸list()

目錄 >

freqTable▸list(*列表1***,***頻率整數表***)⇒***列表*

所傳回列表包含*列表1* 中的元素，且依*頻率整數表*中的頻率展開這些元素。本函數可用於製作 Data & Statistics 應用程式的次數表。

```
freqTable▸list({1,2,3,4},{1,4,3,1})
                    {1,2,2,2,2,3,3,3,4}
freqTable▸list({1,2,3,4},{1,4,0,1})
                          {1,2,2,2,2,4}
```

freqTable▸list()

列表1 可以是任何有效列表。

*頻率整數表*和*列表1* 的維數必須相同，而且只能包含非負整數的元素。各元素可用於指定*列表1* 對應元素在結果列表中的重複次數。值如果是零，代表排除*列表1* 的對應元素。

附註：如果要從電腦鍵盤插入本函數，可輸入 **`freqTable@>list(…)`**。

空元素會被忽略。如需空元素的詳細資訊，請參考第 頁碼：198 頁。

frequency()

frequency(*列表1,binsList***)⇒***列表*

傳回*列表1* 中各元素出現次數的列表。次數是以您在 *binsList* 中定義的範圍 (bins) 為準。

若 *binsList* 是 {b(1), b(2), …, b(n)}，指定範圍是 {**?**≤b(1), b(1)<**?**≤b(2),…,b(n-1)<**?**≤b(n), b(n)>**?**}。所求出的列表是超過 *binsList* 單一元素長度。

結果中的各元素，皆與*列表1* 中在該組距範圍內的元素數目相同。若以 **countIf()** 函數表示，則結果是 { countIf(list, **?**≤b(1)), countIf(list, b(1)<**?**≤b(2)), …, countIf(list, b(n-1)<**?**≤b(n)), countIf(list, b(n)>**?**)}。

系統會略過*列表1* 無法「放在組距」內的元素。空元素也會遭到忽略。如需空元素的詳細資訊，請參考第 頁碼：198 頁。

您可在 Lists & Spreadsheet 應用程式中用儲存格範圍取代兩個引數。

附註：另請參考 **countIf()**，頁碼：28。

$datalist:=\{1,2,e,3,\pi,4,5,6,\text{"hello"},7\}$

$\{1,2,2.71828,3,3.14159,4,5,6,\text{"hello"},7\}$

$\text{frequency}(datalist,\{2.5,4.5\})$ $\{2,4,3\}$

解釋結果：

2 個元素(*Datalist* 中) ≤2.5

4 個元素(*Datalist* 中) >2.5 且 ≤4.5

3 個元素(*Datalist* 中) >4.5

「hello」元素是字串，無法放在既定的任何組距中。

FTest_2Samp

FTest_2Samp *列表1*,*列表2*[,*Freq1*[,*Freq2*[,*Hypoth*]]]

FTest_2Samp *列表1*,*列表2*[,*Freq1*[,*Freq2*[,*Hypoth*]]]

FTest_2Samp

目錄 > 📖

(輸入資料列表)

FTest_2Samp *sx1*,*n1*,*sx2*,*n2*[,*Hypoth*]

FTest_2Samp *sx1*,*n1*,*sx2*,*n2*[,*Hypoth*]

(輸入統計摘要)

執行二樣本 F 檢定。*stat.results* 變數會儲存結果摘要(請參閱第 頁碼：134 頁) 。

H_a:σ1 > σ2 時，設定 *Hypoth*>0

H_a:σ1 ≠ σ2(預設值) 時，設定 *Hypoth* =0

H_a:σ1 < σ2 時，設定 *Hypoth*<0

如要了解列表中有空元素時的影響，請參考°ß空元素°®，頁碼：198。

輸出變數	說明
stat.F	計算資料序列的 Û 統計
stat.PVal	無效假說被否定之最低意義標準
stat.dfNumer	分子自由度 = n1-1
stat.dfDenom	分母自由度 = n2-1
stat.sx1、stat.sx2	*列表 1 和列表 2 中資料序列的標準差樣本*
stat.x1_bar stat.x2_bar	*列表 1 和列表 2 中資料序列的平均值樣本*
stat.n1、stat.n2	樣本大小

Func

目錄 > 📖

Func
區段
EndFunc

製作使用者自行定義函數的範本。

*區段*可以只是一個語句、由「:」字元分隔的一連串語句，也可以是分成多行的一連串語句。本函數可利用 **Return** 指令傳回特定結果。

定義分段函數：

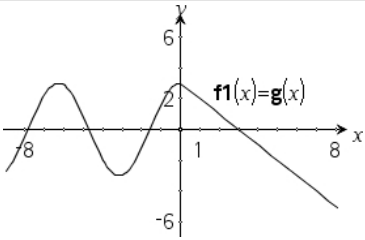
```
Define g(x)=Func                    Done
          If x<0 Then
          Return 3·cos(x)
          Else
          Return 3−x
          EndIf
          EndFunc
```

g(x) 繪圖結果

Func

目錄 >

輸入範例的注意事項：關於輸入多行程式和函數定義的說明，請參閱產品手冊中的「計算工具」章節。



G

gcd()

目錄 >

gcd(*數字1, 數字2***)⇒***運算式*

傳回兩個引數的最大公因數。兩個分數的 **gcd**，是分子的 **gcd** 除以分母的 **lcm**。

在自動或近似值模式中，分數浮點數的 **gcd** 是 1.0。

$\gcd(18,33)$ → 3

gcd(*列表1, 列表2***)⇒***列表*

傳回*列表1*和*列表2*中對應元素的最大公因數。

$\gcd(\{12,14,16\},\{9,7,5\})$ → $\{3,7,1\}$

gcd(*矩陣1, 矩陣2***)⇒***矩陣*

傳回*矩陣1*和*矩陣2*中對應元素的最大公因數。

$\gcd\left(\begin{bmatrix}2 & 4\\6 & 8\end{bmatrix},\begin{bmatrix}4 & 8\\12 & 16\end{bmatrix}\right)$ → $\begin{bmatrix}2 & 4\\6 & 8\end{bmatrix}$

geomCdf()

目錄 >

geomCdf(*p***,***下限***,***上限***)⇒***數字*(若*下限*和*上限*是數字) 或*列表*(若*下限*和*上限*是列表)

geomCdf(*p***,***上限***)**for P(1≤X≤*上限*)⇒*數字*(若*上限*是數字) 或*列表*(若*上限*是列表)

計算從*下限*到*上限*的累積幾何機率(指定成功機率是 *p*) 。

對於 P(X ≤ *上限*)，請設定 *下限*= 1。

geomPdf()

geomPdf(*p,XVal***)⇒***數字*(若 *XVal* 是數字) 或*列表*(若 *XVal* 是列表)

針對離散幾何分布計算 *XVal* 的機率，亦即第一次成功的試驗次數(指定成功機率是 p) 。

Get

Get[*promptString*,]*var*[, *statusVar*]

Get[*promptString*,] *func*(*arg1*, *...argn*) [, *statusVar*]

程式設計指令:從已連接的 TI-Innovator™ Hub 擷取數值，並將該數值指定給變數 *var*。

必須要求數值:

- 預先透過**傳送 "READ ..."** 指令要求。

  —或—

- 嵌入 **"READ ..."** 要求作為選用的 *promptString* 引數。此方法可讓您使用單一指令要求及擷取數值。

執行隱簡化。例如，收到的字串「123」會解譯為數值。如要保留字串，請使用 **GetStr** 而不是 **Get**。

若加入選用引數 *statusVar*，則將操作成功與否指派為其值。零值表示未收到任何資料。

在第二個語法中，*func*() 引數可讓程式把收到的字串儲存成函數定義。這個語法的功能如同程式執行以下指令:

Define *func*(*arg1*, *...argn*) = *收到的字串*

然後，程式就能使用已定義函數*func*()。

附註:您可以在使用者自行定義的程式中使用 **Get** 指令，但無法在函數中使用。

附註:另請參考 **GetStr**, 頁碼: 61 和 **Send**、頁碼: 123.

例如:要求分享器內建亮度等級感應器目前的值。使用 **Get** 擷取數值，並將數值指定給變數 *lightval*。

Send "READ BRIGHTNESS"	*Done*
Get *lightval*	*Done*
lightval	0.347922

在 **Get** 指令中嵌入 READ 要求。

Get "READ BRIGHTNESS",*lightval*	*Done*
lightval	0.378441

getDenom()

getDenom(*分數1***)⇒***值*

將引數轉成有簡化公分母的運算式，然後傳回其分母。

$x:=5: y:=6$	6
$\text{getDenom}\left(\frac{x+2}{y-3}\right)$	3
$\text{getDenom}\left(\frac{2}{7}\right)$	7
$\text{getDenom}\left(\frac{1}{x}+\frac{y^2+y}{y^2}\right)$	30

getKey()

getKey([0|1]) ⇒ returnString

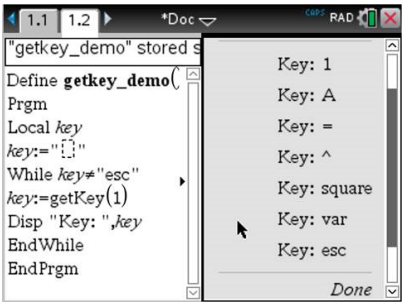
說明 :getKey() 可讓 TI-Basic 程式擷取計算機、桌面及桌面模擬器的鍵盤輸入內容。

範例:

- 若使用者沒有按下任何鍵，keypressed := **getKey()** 會傳回一個鍵或空字串。這個呼叫指令會立即傳回結果。
- keypressed := **getKey(1)** 會等到使用者按下一個鍵，才顯示結果。在使用者按下任何鍵之前，這個呼叫指令都會暫停執行程式。

getKey()

範例:



按鍵處理方式:

計算機/模擬器按鍵	桌面	傳回數值
Esc	Esc	"esc"
觸控板 - 點擊頂部	N/A	"up"
On	N/A	"home"
Scratchapps	N/A	"scratchpad"
觸控板 - 點擊左邊	N/A	"left"
觸控板 - 點擊中央	N/A	"center"
觸控板 - 點擊右邊	N/A	"right"
Doc	N/A	"doc"

計算機/模擬器按鍵	桌面	傳回數值
Tab	Tab	"tab"
觸控板 - 點擊底部	向下鍵	"down"
功能表	N/A	"menu"
Ctrl	Ctrl	無傳回結果
Shift	Shift	無傳回結果
Var	N/A	"var"
Del	N/A	"del"
=	=	"="
trig	N/A	"trig"
0 到 9	0-9	"0" ... "9"
範本	N/A	"template"
目錄	N/A	"cat"
^	^	"^"
X^2	N/A	"square"
/ (除號鍵)	/	"/"
* (乘號鍵)	*	"*"
e^x	N/A	"exp"
10^x	N/A	"10power"
+	+	"+"
-	-	"-"
(	(	"("
)	)	")"
.	.	"."
(-)	N/A	"-" (負號)
Enter	Enter	"enter"
ee	N/A	"E" (科學記號 E)
a - z	a-z	字母 = 按下的字母 (小寫)

計算機/模擬器按鍵	桌面	傳回數值 ("a" - "z")
shift a-z	shift a-z	字母 = 按下的字母 "A" - "Z"
		注意:ctrl-shift 會鎖定大寫
?!	N/A	"?!"
pi	N/A	"pi"
旗幟	N/A	無傳回結果
,	,	","
回傳	N/A	"return"
空格	空格	" " (空格)
無法存取	特殊字元按鍵，例如 @、!、^ 等。	傳回字元
N/A	功能鍵	無傳回字元
N/A	特殊桌面控制鍵	無傳回字元
無法存取	getkey() 等待使用者按下按鍵時，計算工具上沒有的其他桌面按鍵。({, },;, :, ...)	與在筆記 (並非數學方塊) 中取得的字元相同

注意:請務必注意，如在程式中使用 **getKey()**，系統處理特定事件的方式就會改變。以下說明幾個這類案例。

終止程式並處理事件 - 如同使用者按下 **ON** 鍵以中斷程式

下列的 "**Support**" 代表程式繼續執行，系統也如預期中運作。

事件	裝置	桌面 - TI-Nspire™ 學生軟體
簡易投票	終止程式，處理事件	與計算機相同 (僅限 TI-Nspire™ Student Software, TI-Nspire™ Navigator™ NC Teacher Software)
遠端檔案管理 (包括從另一部計算機或桌面計算機傳送「Exit Press 2 Test」檔案)	終止程式，處理事件	與計算機相同 (僅限 TI-Nspire™ Student Software, TI-Nspire™ Navigator™ NC Teacher Software)
結束課程	終止程式，處理事件	支援

事件	裝置	桌面 - TI-Nspire™ 學生軟體
		(僅限 TI-Nspire™ Student Software, TI-Nspire™ Navigator™ NC Teacher Software)

事件	裝置	桌面 - TI-Nspire™ 所有版本
TI-Innovator™ Hub 連線/中斷連線	支援 - 可成功發送指令給 TI-Innovator™ Hub。在您離開程式後，TI-Innovator™ Hub仍會與計算機共同運作。	與計算機相同

getLangInfo()

目錄 > 📖

getLangInfo()⇒*字串*

傳回目前所使用語言簡稱的對應字串。例如您可用於程式或函數中，以決定目前的語言。

英文 =「en」
丹麥文 =「da」
德文 =「de」
芬蘭文 =「fi」
法文 =「fr」
義大利文 =「it」
荷蘭文 =「nl」
比利時荷蘭文 =「nl_BE」
挪威文 =「no」
葡萄牙文 =「pt」
西班牙文 =「es」
瑞典文 =「sv」

getLangInfo()	"en"

getLockInfo()

getLockInfo(*Var***)⇒***值*

傳回 *Var* 目前的狀態是已鎖定還是已解開鎖定。

值 **=0**：*Var* 已解開鎖定或不存在。

值 **=1**：*Var* 已遭鎖定且無法加以修改或刪除。

請參考第 頁碼：79 頁的 **Lock 和第 頁碼：151 頁的 unLock。**

a:=65	65
Lock *a*	*Done*
getLockInfo(*a*)	1
a:=75	"Error: Variable is locked."
DelVar *a*	"Error: Variable is locked."
Unlock *a*	*Done*
a:=75	75
DelVar *a*	*Done*

getMode()

getMode(*模式名稱整數***)⇒***值*

getMode(0)⇒*列表*

getMode(*模式名稱整數*) 所傳回的值，代表*模式名稱整數*模式的目前設定。

getMode(0)：傳回包含成對數值的列表。每對數字由代表該模式整數和設定整數組成。

如需模式列表及模式設定資訊，請參考下表。

若您以 **getMode(0) →** *變數*儲存設定，則可在函數或程式中使用 **setMode(***變數***)**，以便在只執行該函數或程式期間暫時還原設定。請參考 **setMode()**，頁碼：126。

getMode(0)	{1,7,2,1,3,1,4,1,5,1,6,1,7,1}
getMode(1)	7
getMode(7)	1

模式 名稱	模式整數	設定整數
顯示位數	1	**1**=浮點數、**2**=浮點數1、**3**=浮點數2、**4**=浮點數3、**5**=浮點數4、**6**=浮點數5、**7**=浮點數6、**8**=浮點數7、**9**=浮點數8、**10**=浮點數9、**11**=浮點數10、**12**=浮點數11、**13**=浮點數12、**14**=固定0、**15**=固定1、**16**=固定2、**17**=固定3、**18**=固定4、**19**=固定5、**20**=固定6、**21**=固定7、**22**=固定8、**23**=固定9、**24**=固定10、**25**=固定11、**26**=固定12
角度	2	**1**=弧度角、**2**=度數角、**3**=梯度角
指數格式	3	**1**=正常、**2**=科學、**3**=工程
實數或複數	4	**1**=實數、**2**=直角座標、**3**=極座標

模式 名稱	模式整數	設定整數
自動或近似值	5	**1**=自動、**2**=近似值
向量格式	6	**1**=直角座標、**2**=圓柱座標、**3**=球面座標
基底	7	**1**=十進位、**2**=十六進位、**3**=二進位

getNum()

目錄 > 📖

getNum(*分數1***)⇒***值*

將引數轉成有簡化公分母的運算式，然後傳回其分子。

$x:=5: y:=6$	6
$\text{getNum}\left(\frac{x+2}{y-3}\right)$	7
$\text{getNum}\left(\frac{2}{7}\right)$	2
$\text{getNum}\left(\frac{1}{x}+\frac{1}{y}\right)$	11

GetStr

分享器功能表

GetStr[*promptString*,] *var*[, *statusVar*]

GetStr[*promptString*,] *func*(*arg1*, ...*argn*) [, *statusVar*]

程式設計指令:與 **Get** 指令原理相同，不過本指令一律將收到的值解譯為字串。相反地，除非使用者把回答放在引號 ("") 中，否則 **Get** 指令會將回答解譯成運算式。

附註:另請參考 **Get**, 頁碼: 55 和 **Send**、頁碼: 123.

例如，另請參考 **Get**。

getType()

目錄 > 📖

getType(*var***)⇒***字串*

傳回一個字串，說明變數 *var* 的資料類型。

如果未定義 *var*，*則會傳回字串* [NONE]。

$\{1,2,3\}\rightarrow temp$	$\{1,2,3\}$
$\text{getType}(temp)$	"LIST"
$3\cdot i\rightarrow temp$	$3\cdot i$
$\text{getType}(temp)$	"EXPR"
DelVar *temp*	*Done*
$\text{getType}(temp)$	"NONE"

getVarInfo()⇒*矩陣©Œ字串*

getVarInfo(*LibNameString***)**⇒*矩陣©Œ字串*

getVarInfo() 可針對目前問題中定義的所有變數與資料庫物件，傳回相關資訊的矩陣(變數名稱、類型、資料庫存取能力，以及已鎖定/已解開鎖定狀態) 。

若尚未定義變數，則 getVarInfo() 會傳回「無」字串。

getVarInfo(*LibNameString*)可針對 *LibNameString* 資料庫中定義的所有資料庫物件，傳回相關資訊的矩陣。*LibNameString* 必須是字串(以英文引號括住的文字) 或字串變數。

若 *LibNameString* 資料庫不存在，則會出現錯誤。

請注意左例，*vs* 變數的賦值就是 **getVarInfo()** 的結果。若要顯示 *vs* 的第 2 列或第 3 列，則會傳回「列表或矩陣無效」錯誤，因為這些列中至少有一個元素(例如 *b* 變數) 的求值結果是矩陣。

利用 *Ans* 重新對 **getVarInfo()** 結果求值時，也可能出現本錯誤。

方程式組之所以出現以上錯誤，是因為目前版本的軟體，不支援矩陣元素既可是矩陣又能是列表的廣義矩陣結構。

getVarInfo()	"NONE"
Define x=5	*Done*
Lock x	*Done*
Define LibPriv y={1,2,3}	*Done*
Define LibPub $z(x)$=3·x^2−x	*Done*
getVarInfo()	$\begin{bmatrix} x & \text{"NUM"} & \text{"}\square\text{"} & 1 \\ y & \text{"LIST"} & \text{"LibPriv "} & 0 \\ z & \text{"FUNC"} & \text{"LibPub "} & 0 \end{bmatrix}$
getVarInfo(*tmp3*)	"Error: Argument must be a string"
getVarInfo("tmp3")	[*volcyl2* "NONE" "LibPub " 0]

a:=1	1
b:=[1 2]	[1 2]
c:=[1 3 7]	[1 3 7]
vs:=getVarInfo()	$\begin{bmatrix} a & \text{"NUM"} & \text{"}\square\text{"} & 0 \\ b & \text{"MAT"} & \text{"}\square\text{"} & 0 \\ c & \text{"MAT"} & \text{"}\square\text{"} & 0 \end{bmatrix}$
vs[1]	[1 "NUM" "□" 0]
vs[1,1]	1
vs[2]	"Error: Invalid list or matrix"
vs[2,1]	[1 2]

Goto

目錄 >

Goto *labelName*

把控制權轉到 *labelName* 標籤。

您必須用 **Lbl** 指令在同一個函數中定義 labelName。

輸入範例的注意事項：關於輸入多行程式和函數定義的說明，請參閱產品手冊中的「計算工具」章節。

```
Define g()=Func                    Done
        Local temp,i
        0→temp
        1→i
        Lbl top
        temp+i→temp
        If i<10 Then
        i+1→i
        Goto top
        EndIf
        Return temp
        EndFunc
g()                                55
```

▶Grad

目錄 >

運算式1 ▶ **Grad**⇒*運算式*

把 *運算式1* 轉成梯度角測量單位。

附註：如果要從電腦鍵盤插入本運算子，可輸入 `@>Grad`。

度數角模式：

```
(1.5)▶Grad                    (1.66667)g
```

弧度角模式：

```
(1.5)▶Grad                    (95.493)g
```

I

identity()

目錄 >

identity(*整數***)** ⇒ *矩陣*

傳回以*整數*為維度的單位矩陣。

*整數*必須是正整數。

```
identity(4)
```

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

If

目錄 >

If *布林運算式*
　　語句

If *布林運算式* **Then**
　　區段
EndIf

```
Define g(x)=Func                   Done
        If x<0 Then
        Return x^2
        EndIf
        EndFunc
g(-2)                              4
```

If

目錄 > 📖

若*布林運算式*為真，則在繼續執行程式以前，執行單一語句 *Statement* 或整個區段語句 *Block*。

若*布林運算式*為假，則繼續執行程式，但不執行單一語句或整個區段語句。

*區段*可以只是一個語句，也可以是由「:」字元分隔的一系列語句。

輸入 範例的注意事項：關於輸入多行程式和函數定義的說明，請參閱產品手冊中的「計算工具」章節。

If *布林運算式* **Then**
　　區段1
Else
　　區段2
EndIf

若*布林運算式*為真，則執行*區段1*並跳過*區段2*。

若*布林運算式*為假，則跳過*區段1*但執行*區段2*。

*區段1*和*區段2*可以是單一語句。

If *布林運算式1* **Then**
　　區段1
ElseIf *布林運算式2* **Then**
　　區段2
⋮
ElseIf *布林運算式N* **Then**
　　區段N
EndIf

允許有分枝。若*布林運算式1*為真，則執行*區段1*。若*布林運算式1*為假，則計算*布林運算式2*運算式，依此類推。

```
Define g(x)=Func                    Done
              If x<0 Then
              Return -x
              Else
              Return x
              EndIf
              EndFunc
g(12)                                12
g(-12)                               12
```

```
Define 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
                                    Done
g(-4)                                4
g(10)                                3
```

ifFn()

目錄 > 📖

ifFn(*布林運算式*,*若為真的值* [,*若為假的值* [,*若未知的值*]]**)** ⇒ *運算式、列表或矩陣*

```
ifFn({1,2,3}<2.5,{5,6,7},{8,9,10})
                                {5,6,10}
```

ifFn()

目錄 >

對*布林運算式*(或*布林運算式*中的各元素)求值，並根據以下規則求出結果:

- *布林運算式*可檢定個別的值、列表或矩陣。
- 若*布林運算式*的某元素為真，則傳回*若為真的值*對應元素。
- 若*布林運算式*的某元素為假，則傳回*若為偽的值*對應元素。若省略*若為假的值*，則傳回 undef。
- 若*布林運算式*的某元素既非真亦非假，則傳回*若未知的值*對應元素。若乎略*若未知的值*，則傳回未定義 (undef)。
- 若 **ifFn()** 函數的第二、第三或第四個引數是單一運算式，則會將布林檢定套用到*布林運算式*的所有位置。

注意:若化簡的*布林運算式*語句是列表或矩陣，則其他列表或矩陣引數的維數都必須相同，而結果的維數也會相同。

1 的檢定值小於 2.5，所以會將

*若為真的值*對應元素(**5**) 複製到結果列表。

2 的檢定值小於 2.5，所以會將

*若為真的值*對應元素(**6**) 複製到結果列表。

3 的檢定值不小於 2.5，所以會將*若為假的值*對應元素(**10**) 複製到結果列表。

$\text{ifFn}(\{1,2,3\}<2.5,4,\{8,9,10\})$ $\{4,4,10\}$

*若為真的值*是單一值，因此與任何選取位置相同。

$\text{ifFn}(\{1,2,3\}<2.5,\{5,6,7\})$ $\{5,6,\text{undef}\}$

*若為假的值*則尚未指定。此處使用 undef。

$\text{ifFn}(\{2,\text{"a"}\}<2.5,\{6,7\},\{9,10\},\text{"err"})$ $\{6,\text{"err"}\}$

已從*若為真的值*選取一個元素。已從*若未知的值*選取一個元素。

imag()

目錄 >

imag(*值 1***)** ⇒ *值*

傳回引數的虛部。

$\text{imag}(1+2\cdot i)$ 2

imag(*列表 1***)** ⇒ *列表*

傳回元素的虛部列表。

$\text{imag}(\{-3,4-i,i\})$ $\{0,-1,1\}$

imag(*矩陣 1***)** ⇒ *矩陣*

傳回元素的虛部矩陣。

$\text{imag}\left(\begin{bmatrix}1 & 2\\ i\cdot 3 & i\cdot 4\end{bmatrix}\right)$ $\begin{bmatrix}0 & 0\\ 3 & 4\end{bmatrix}$

inString()

目錄 >

inString(*來源字串***,** *分段字串***[,** *開始***])** ⇒ *整數*

傳回*來源字串*中*分段字串*第一次出現的字元開始位置。

開始(若納入指令中) 可用於指定在*來源字串*中開始搜尋的字元位置。預設值 = 1(*來源字串*的第一個字元) 。

若*來源字串*不包含*分段字串*或*開始*值 > *來源字串*長度，則傳回零。

inString("Hello there","the")	7
inString("ABCEFG","D")	0

int()

目錄 >

int(*值***)** ⇒ *整數*
int(*列表 l***)** ⇒ *列表*
int(*矩陣 l***)** ⇒ *矩陣*

傳回小於或等於引數的最大整數。本功能與 **floor()** 相同。

引數可以是實數也可以是複數。

若是列表或矩陣，則傳回各元素的最大整數。

int(-2.5)	-3.
int([-1.234 0 0.37])	[-2. 0 0.]

intDiv()

目錄 >

intDiv(*數字 1***,** *數字 2***)** ⇒ *整數*
intDiv(*列表 1***,** *列表 2***)** ⇒ *列表*
intDiv(*矩陣 1***,** *矩陣 2***)** ⇒ *矩陣*

傳回 (*數字 1* ÷ *數字 2*) 具正負號的整數部份。

若是列表和矩陣，則傳回各成對元素的 (引數1 ÷ 引數2) 具正負號之整數部份。

intDiv(-7,2)	-3
intDiv(4,5)	0
intDiv({12,-14,-16},{5,4,-3})	{2,-3,5}

插入 ()

目錄 > 📖

插入(*x值***,***x列表***,***y列表***,***y質數序列表***)** ⇒ *列表*

此函數會進行下列計算:

針對未知函數 **f**，給定 *x列表*、*y列表*=**f(***x列表***)** 且 *y質數序列表*=**f'(***x列表***)**，使用三次迴歸插值求得函數 **f** 的 *x值*。這是假設 *x列表*是單調遞增或遞減數字的列表，但此函數可能會傳回一個值 (即使其並非如此) 。此函數會搜尋 *x列表*以尋找包含 *x值*的區間 [*x列表*[i], *x列表*[i+1]]。如果此函數找到該區間，則會針對 **f(***x值***)** 傳回一個插值;否則，它會傳回 **。未定義.**

x列表、*y列表*及 *y質數序列表*必須屬於相等的維數 ≥ 2 並包含化簡為數字的運算式。

*x值*可以是數字或數字列表。

微分方程式:
y'=-3•y+6•t+5 且 $y(0)$=5

```
rk:=rk23(-3·y+6·t+5,t,y,{0,10},5,1)
[0.      1.        2.        3.        4.        ▸
 5.  3.19499  5.00394  6.99957  9.00593  10
```

若要看到完整結果，請按 ▲，然後使用 ◀ 與 ▶ 移動游標。

使用 interpolate() 函數以計算 xvaluelist 的函數值:

```
xvaluelist:=seq(i,i,0,10,0.5)
{0,0.5,1.,1.5,2.,2.5,3.,3.5,4.,4.5,5.,5.5,6.,6.5,▸
xlist:=mat▸list(rk[1])
{0.,1.,2.,3.,4.,5.,6.,7.,8.,9.,10.}
ylist:=mat▸list(rk[2])
{5.,3.19499,5.00394,6.99957,9.00593,10.9978▸
yprimelist:=-3·y+6·t+5|y=ylist and t=xlist
{-10.,1.41503,1.98819,2.00129,1.98221,2.006▸
interpolate(xvaluelist,xlist,ylist,yprimelist)
{5.,2.67062,3.19499,4.02782,5.00394,6.00011▸
```

invχ2()

目錄 > 📖

invχ2(*面積***,***df***)**

invChi2(*面積***,***df***)**

針對曲線下已知*面積*的自由度 *df*，計算指定的反累積 χ^2(卡方) 機率函數。

invF()

目錄 > 📖

invF(*面積***,***分子自由度***,***分母自由度***)**

invF(*面積***,***分子自由度***,***分母自由度***)**

針對曲線下已知*面積*的*分子自由度*與*分母自由度*，計算指定的反累積 F 分佈函數。

invBinom()

目錄 >

invBinom(*累積機率,試驗次數,機率,輸出形式***)⇒***純量或矩陣*

反二項式。針對試驗次數 (*NumTrials*) 和各試驗成功的機率 (*Prob*)，此函數會傳回最低成功次數 *k*，使 *k* 值大於或等於特定累積機率 (*CumulativeProb*)。

若*輸出形式*=**0**，則顯示以純量顯示結果(預設) 。

若*輸出形式*=**1**，則以矩陣顯示結果。

範例:Mary 和 Kevin 正在完擲骰子遊戲。Mary 必須猜出 30 次中擲出數字 6 的最高次數。若擲出數字 6 的次數恰巧等於其猜測之次數(或小於該次數) ，則 Mary 獲勝。此外，若她猜測的次數越小，則贏得的獎金越大。若 Mary 希望獲勝的機率高於 77%，則她必須猜測的最低次數為多少？

$\text{invBinom}\left(0.77,30,\frac{1}{6}\right)$	6
$\text{invBinom}\left(0.77,30,\frac{1}{6},1\right)$	$\begin{bmatrix}5 & 0.616447\\ 6 & 0.776537\end{bmatrix}$

invBinomN()

目錄 >

invBinomN(*累積機率,機率,成功次數,輸出形式***)⇒***純量或矩陣*

針對各試驗成功的機率 (*Prob*) 與成功次數 (*NumSuccess*)，此函數會傳回最低試驗次數 *N*，使 *N* 值小於或等於特定累積機率 (*CumulativeProb*)。

若*輸出形式*=**0**，則顯示以純量顯示結果(預設) 。

若*輸出形式*=**1**，則以矩陣顯示結果。

範例:Monique 正在練習籃網球的投籃。根據過去的經驗，她知道自己進籃的機率為 70%。她計劃不斷練習投籃，直到得到 50 分為止。她必續嘗試投籃多少次，才能確保得到至少 50 分的機率大於 0.99？

$\text{invBinomN}(0.01,0.7,49)$	86
$\text{invBinomN}(0.01,0.7,49,1)$	$\begin{bmatrix}85 & 0.010451\\ 86 & 0.00709\end{bmatrix}$

invNorm()

目錄 >

invNorm(*面積***[,μ[,σ]])**

針對 μ 和 *σ* 指定的常態分佈曲線，計算下方已知*面積*的反累積常態分佈函數。

invt()

目錄 >

invt(*面積,df***)**

針對曲線下已知*面積*的自由度 *df*，計算指定的反累積學生-t 機率函數。

iPart()

目錄 >

iPart(*數字***)** ⇒ *整數*
iPart(*列表 1***)** ⇒ *列表*
iPart(*矩陣 1***)** ⇒ *矩陣*

傳回引數的整數部份。

若是列表和矩陣，則傳回各元素的整數部份。

引數可以是實數也可以是複數。

iPart(-1.234)	-1.
iPart({3/2,-2.3,7.003})	{1,-2.,7.}

irr()

目錄 >

irr(*CF0,CF列表* [,*CF頻率*]**)** ⇒ *值*

可計算內部投資報酬率的財務函數。

CF0 是時間為 0 時的初始現金流量，必須是實數。

*CF列表*是初始現金流量 CF0 之後的現金流量金額列表。

*CF頻率*是選用的列表，其中各元素可用於指定群組(相鄰) 現金流量金額 (*CF列表*對應元素) 的出現頻率。預設值是 1;若您輸入值，則必須是 < 10,000 的正整數。

注意:另請參考 **mirr()**，頁碼: 87.

list1:={6000,-8000,2000,-3000}	{6000,-8000,2000,-3000}
list2:={2,2,2,1}	{2,2,2,1}
irr(5000,*list1*,*list2*)	-4.64484

isPrime()

目錄 >

isPrime(*數字***)** ⇒ *布林常數運算式*

傳回真假值，以表示*數字*是否是 ≥ 2 且只能由自己和 1 整除的整數。

若*數字*超過大約 306 位數，而且沒有 ≤1021 的因數，則 **isPrime(***數字***)** 會顯示錯誤訊息。

輸入範例的注意事項: 關於輸入多行程式和函數定義的說明，請參閱產品手冊中的「計算工具」章節。

isPrime(5)	true
isPrime(6)	false

可找出指定數字後下一個質數的函數:

```
Define nextprim(n)=Func                    Done
                  Loop
                  n+1→n
                  If isPrime(n)
                  Return n
                  EndLoop
                  EndFunc
```

nextprim(7)	11

isVoid()

目錄 >

isVoid(*變數***)** ⇒ *布林常數運算式*
isVoid(*運算式***)** ⇒ *布林常數運算式*
isVoid(*列表***)** ⇒ *布林常數運算式列表*

傳回真假值，指出引數是否是「空」資料類型。

如需空元素的詳細資訊，請參考第 頁碼：198.

$a:=_$	$_$
isVoid(a)	true
isVoid({1,_,3})	{false,true,false}

L

Lbl

目錄 >

Lbl *標籤名稱*

在函數中定義名稱是*標籤名稱*的標籤。

您可利用 **Goto** *標籤名稱*指令，將控制權轉給標籤後緊接的指令。

*標籤名稱*必須符合變數名稱的命名規則。

輸入 範例的注意事項：關於輸入多行程式和函數定義的說明，請參閱產品手冊中的「計算工具」章節。

```
Define g()=Func                    Done
        Local temp,i
        0→temp
        1→i
        Lbl top
        temp+i→temp
        If i<10 Then
        i+1→i
        Goto top
        EndIf
        Return temp
        EndFunc
g()                                55
```

lcm()

目錄 >

lcm(*數字1***,** *數字2***)**⇒*運算式*

lcm(*列表1***,** *列表2***)**⇒*列表*

lcm(*矩陣1***,** *矩陣2***)**⇒*矩陣*

傳回兩個引數的最小公倍數。兩個分數的 **lcm**，是分子的 **lcm** 除以分母的 **gcd**。分數浮點數的 **lcm** 就是浮點數的乘積。

若是兩個列表或矩陣，則傳回對應元素的最小公倍數。

lcm(6,9)	18
$\text{lcm}\left(\left\{\frac{1}{3},-14,16\right\},\left\{\frac{2}{15},7,5\right\}\right)$	$\left\{\frac{2}{3},14,80\right\}$

left()

目錄 >

left(來源字串[, *Num*]**)⇒**字串

傳回來源字串中從左邊算來的 *Num* 個字元。

若省略 *Num*，則傳回整個來源字串。

```
left("Hello",2)                    "He"
```

left(列表1[, *Num*]**)⇒**列表

傳回列表1 中從左邊算來的 *Num* 個元素。

若省略 *Num*，則傳回整個列表1。

```
left({1,3,-2,4},3)                 {1,3,-2}
```

left(比較**)⇒**運算式

傳回方程式或不等式左邊的部份。

libShortcut()

目錄 >

libShortcut(*LibNameString*, *ShortcutNameString* [, *LibPrivFlag*]**)⇒**變數列表

在目前問題中建立變數群組。此問題需包含指定資料庫文件 *libNameString* 中所有物件的參照。本功能亦可將群組成員加入 [變數] 功能表。接著即可利用各物件的 *ShortcutNameString* 叫出該物件。

設定 *LibPrivFlag*=**0**，排除自訂資料庫物件(預設值)

設定 *LibPrivFlag*=**1**，納入自訂資料庫物件

若要複製變數群組，請參考 **CopyVar**，頁碼：23。

若要刪除變數群組，請參考 **DelVar**，頁碼：36。

本例假設有已經正常儲存並重新整理的 **linalg2** 資料庫文件，其中包含的已定義物件是 *clearmat*、*gauss1*、*gauss2*。

```
getVarInfo("linalg2")
        [clearmat  "FUNC"  "LibPub "
         gauss1    "PRGM"  "LibPriv "
         gauss2    "FUNC"  "LibPub "]
libShortcut("linalg2","la")
                    {la.clearmat,la.gauss2}
libShortcut("linalg2","la",1)
          {la.clearmat,la.gauss1,la.gauss2}
```

LinRegBx

目錄 >

LinRegBx *X*,*Y*[,[*Freq*][,類別,包含]]

計算線性回歸 Y = A + B•X 在列表X和Y的頻率 *Freq*。其結果概要存儲於 *stat.results* 的變數中。

所有列表的維數都必須相同，*包含*除外。

X、*Y* 是自變數和因變數列表。

Freq 是頻率值列表(非必要) 。*Freq* 的每個元素，可用於指定各 *X*、*Y* 對應資料點的出現頻率。預設值是 1。所有元素都必須是 ≥ 0 的整數。

*類別*是 *X*、*Y* 對應資料的數字或字串類別代號列表。

*包含*是一個以上類別代號的列表。只有本列表中類別代號所對應的資料項目，才會包含在計算作業中。

如要了解列表中有空元素時的影響，請參考°ß空元素°®，頁碼：198。

輸出變數	說明
stat.RegEqn	迴歸方程式：a+b ·x
stat.a、stat.b	迴歸係數
stat.r^2	判定係數
stat.r	相關係數
stat.Resid	迴歸殘差
stat.XReg	實際用於迴歸的已修改 *X 列表*中的資料點列表(根據 *Freq*、*類別列表*、*包含類別的限制)*
stat.YReg	實際用於迴歸的已修改 *Y 列表*中的資料點列表(根據 *Freq*、*類別列表*、*包含類別的限制)*
stat.FreqReg	*stat.XReg* 與 *stat.YReg 對應的頻率列表*

LinRegMx

目錄 > 📖

LinRegMx *X*,*Y*[,[*Freq*][,*類別*,*包含*]]

計算 *X* 列表與 *Y* 列表的 y = m ·x+b 線性迴歸(頻率為 *Freq*) 。*stat.results* 變數會儲存結果摘要(請參閱第 頁碼：134 頁) 。

所有列表的維數都必須相同，*包含*除外。

LinRegMx

目錄 > 📖

X、*Y* 是自變數和因變數列表。

Freq 是頻率值列表(非必要) 。*Freq* 的每個元素，可用於指定各 *X*、*Y* 對應資料點的出現頻率。預設值是 1。所有元素都必須是 ≥ 0 的整數。

*類別*是 *X*、*Y* 對應資料的數字或字串類別代號列表。

*包含*是一個以上類別代號的列表。只有本列表中類別代號所對應的資料項目，才會包含在計算作業中。

如要了解列表中有空元素時的影響，請參考°ß空元素°®，頁碼: 198。

輸出變數	說明
stat.RegEqn	迴歸方程式: y = m ·x+b
stat.m、stat.b	迴歸係數
stat.r^2	判定係數
stat.r	相關係數
stat.Resid	迴歸殘差
stat.XReg	實際用於迴歸的已修改 *X 列表*中的資料點列表(根據 *Freq*、*類別列表*、*包含類別的限制)*
stat.YReg	實際用於迴歸的已修改 *Y 列表*中的資料點列表(根據 *Freq*、*類別列表*、*包含類別的限制)*
stat.FreqReg	*stat.XReg* 與 *stat.YReg 對應的頻率列表*

LinRegtIntervals

目錄 > 📖

LinRegtIntervals *X*,*Y*[,*F*[,**0**[,*CLev*]]]

對於斜率。計算斜率的信賴水準信賴區間。

LinRegtIntervals *X*,*Y*[,*F*[,**1**,*Xval*[,*CLev*]]]

對於反應。計算預測的 y 值、單次觀測的信賴水準預測區間，以及平均反應的信賴水準信賴區間。

stat.results 變數會儲存結果摘要(請參閱第 頁碼: 134 頁) 。

所有列表的維數都必須相同。

X、*Y* 是自變數和因變數列表。

F 是頻率值列表(非必要)。*F* 的每個元素，可用於指定各 *X*、*Y* 對應資料點的出現頻率。預設值是 1。所有元素都必須是 ≥ 0 的整數。

如要了解列表中有空元素時的影響，請參考°ß空元素°®，頁碼：198。

輸出變數	說明
stat.RegEqn	迴歸方程式：a+b ·x
stat.a、stat.b	迴歸係數
stat.df	自由度
stat.r^2	判定係數
stat.r	相關係數
stat.Resid	迴歸殘差

僅用於斜率類型

輸出變數	說明
[stat.CLower, stat.CUpper]	斜率的信賴區間
stat.ME	信賴區間邊際誤差
stat.SESlope	斜率的標準誤差
stat.s	線的標準誤差

僅用於反應類型

輸出變數	說明
[stat.CLower, stat.CUpper]	平均反應的信賴區間
stat.ME	信賴區間邊際誤差
stat.SE	平均反應的標準誤差
[stat.LowerPred, stat.UpperPred]	單次觀測的預測區間
stat.MEPred	預測區間邊際誤差

輸出變數	說明
stat.SEPred	預測的標準誤差
stat.$\hat{y}$	a + b ·XVal

LinRegtTest

目錄 > 📖

LinRegtTest *X*,*Y*[,*Freq*[,*Hypoth*]]

計算 *X* 與 *Y* 列表的線性迴歸、β 斜率值上的 *t* 檢定，以及 *y*=α+βx 方程式的 ρ 相關係數。本指令可根據三個對立假設檢定 H_0:β=0(等同於 ρ=0) 虛無假設。

所有列表的維數都必須相同。

X、*Y* 是自變數和因變數列表。

Freq 是頻率值列表(非必要) 。*Freq* 的每個元素，可用於指定各 *X*、*Y* 對應資料點的出現頻率。預設值是 1。所有元素都必須是 ≥ 0 的整數。

Hypoth 非必要值，用於指定三個對立假設中的其中一個，據以檢定虛無假設 (H_0:β=ρ=0) 。

H_a:β≠0 與 ρ≠0(預設值) 時，設定 *Hypoth*=0

H_a:β<0 與 ρ<0 時，設定 *Hypoth*<0

H_a:β>0 與 ρ>0 時，設定 *Hypoth*>0

stat.results 變數會儲存結果摘要(請參閱第 頁碼：134 頁) 。

如要了解列表中有空元素時的影響，請參考°ß空元素°®，頁碼：198。

輸出變數	說明
stat.RegEqn	迴歸方程式：a+b ·x
stat.t	*t*-統計量(顯著性檢定)
stat.PVal	無效假說被否定之最低意義標準
stat.df	自由度
stat.a、stat.b	迴歸係數

輸出變數	說明
stat.s	線的標準誤差
stat.SESlope	斜率的標準誤差
stat.r^2	判定係數
stat.r	相關係數
stat.Resid	迴歸殘差

linSolve()

目錄 >

linSolve(*SystemOfLinearEqns*, *Var1*, *Var2*, ...**)⇒***列表*

linSolve(*LinearEqn1* **and** *LinearEqn2* **and** ..., *Var1*, *Var2*, ...**)⇒***列表*

linSolve({*LinearEqn1*, *LinearEqn2*, ...**},** *Var1*, *Var2*, ...**) ⇒***列表*

linSolve(*SystemOfLinearEqns*, **{***Var1*, *Var2*, ...**}) ⇒***列表*

linSolve(*LinearEqn1* **and** *LinearEqn2* **and** ..., **{***Var1*, *Var2*, ...**})⇒***列表*

linSolve({*LinearEqn1*, *LinearEgn2*, ...**},** **{***Var1*, *Var2*, ...**}) ⇒***列表*

傳回 *Var1*,*Var2*,.. 變數的求解列表。

第一個引數在求值後必須是線性聯立方程式或單一線性方程式，否則會出現引數錯誤。

例如對 linSolve(x=1 and x=2,x) 求值時，就會產生「引數錯誤」的結果。

$$\text{linSolve}\left(\begin{cases}2\cdot x+4\cdot y=3\\5\cdot x-3\cdot y=7\end{cases},\{x,y\}\right)\qquad\left\{\frac{37}{26},\frac{1}{26}\right\}$$

$$\text{linSolve}\left(\begin{cases}2\cdot x=3\\5\cdot x-3\cdot y=7\end{cases},\{x,y\}\right)\qquad\left\{\frac{3}{2},\frac{1}{6}\right\}$$

$$\text{linSolve}\left(\begin{cases}apple+4\cdot pear=23\\5\cdot apple-pear=17\end{cases},\{apple,pear\}\right)\qquad\left\{\frac{13}{3},\frac{14}{3}\right\}$$

$$\text{linSolve}\left(\begin{cases}apple\cdot 4+\frac{pear}{3}=14\\-apple+pear=6\end{cases},\{apple,pear\}\right)\qquad\left\{\frac{36}{13},\frac{114}{13}\right\}$$

ΔList()

目錄 >

ΔList(*列表1***)⇒***列表*

附註：如果要從鍵盤插入本函數，可輸入 **`deltaList(...)`**。

傳回*列表1*中相鄰元素間的差分表。*列表1*的每個元素都會減掉在*列表1*中的前一個元素。所求出的列表一定會比原始*列表1*少一個元素。

$$\Delta\text{List}(\{20,30,45,70\})\qquad\{10,15,25\}$$

list▸mat()

目錄 >

list▸mat(*列表* [, *每列元素數*])⇒*矩陣*

傳回逐列填入*列表* 元素的矩陣。

elementsPerRow(若加入) 可用於指定每列的元素個數。預設值是*列表*的元素個數(一列) 。

若*列表*無法填滿求出的矩陣，則會補上零。

附註：如果要從電腦鍵盤插入本函數，可輸入 `list@>mat(…)`。

```
list▸mat({1,2,3})                    [1 2 3]
list▸mat({1,2,3,4,5},2)              [1 2]
                                     [3 4]
                                     [5 0]
```

ln()

ctrl e^x 鍵

ln(*值 1***)⇒***值*

ln(*列表 1***)⇒***列表*

傳回引數的自然對數。

若是列表，則傳回各元素的自然對數。

```
ln(2.)                               0.693147
```

若複數格式模式是實數：

```
ln({-3,1.2,5})
                  "Error: Non-real calculation"
```

若複數格式模式是直角座標：

```
ln({-3,1.2,5})
    {1.09861+3.14159·i,0.182322,1.60944}
```

ln(*方陣 1***)⇒***方陣*

傳回*方陣 1* 的矩陣自然對數。這和計算各元素的自然對數不同。若要了解計算方式，請參考 **cos()**。

方陣 1 必需可對角化。結果一定會包含浮點數。

弧度角模式與直角複數格式：

```
ln([1  5  3])
   [4  2  1])
   [6 -2  1])
[ 1.83145+1.73485·i   0.009193−1.49086
 0.448761−0.725533·i  1.06491+0.623491▸
 -0.266891−2.08316·i  1.12436+1.79018·
```

若要看到完整結果，請按 ▲，然後使用 ◀ 與 ▶ 移動游標。

LnReg

目錄 >

LnReg *X*, *Y*[, [*Freq*] [, *類別*, *包含*]]

計算 *X* 列表與 *Y* 列表的 y = a+b ·ln(x) 對數迴歸(頻率為 *Freq*) 。*stat.results* 變數會儲存結果摘要(請參閱第 頁碼: 134 頁) 。

所有列表的維數都必須相同，包含除外。

X、*Y* 是自變數和因變數列表。

Freq 是頻率值列表(非必要) 。*Freq* 的每個元素，可用於指定各 *X*、*Y* 對應資料點的出現頻率。預設值是 1。所有元素都必須是 ≥ 0 的整數。

*類別*是 *X*、*Y* 對應資料的數字或字串類別代號列表。

*包含*是一個以上類別代號的列表。只有本列表中類別代號所對應的資料項目，才會包含在計算作業中。

如要了解列表中有空元素時的影響，請參考°ß空元素°®，頁碼: 198。

輸出變數	說明
stat.RegEqn	迴歸方程式: a+b ·ln(x)
stat.a、stat.b	迴歸係數
stat.r^2	所轉換資料之線性判定的係數
stat.r	所轉換資料 (ln(x), y) 的相關係數
stat.Resid	與對數模型相關的殘差
stat.ResidTrans	與所轉換資料之線性擬合相關的殘差
stat.XReg	實際用於迴歸的已修改 *X 列表*中的資料點列表(根據 *Freq*、*類別列表*、*包含類別的限制)*
stat.YReg	實際用於迴歸的已修改 *Y 列表*中的資料點列表(根據 *Freq*、*類別列表*、*包含類別的限制)*
stat.FreqReg	*stat.XReg* 與 *stat.YReg 對應的頻率列表*

Local

目錄 >

Local *Var1*[, *Var2*] [, *Var3*] ...

宣告指定的 *vars* 是區域性變數。這些變數只在函數求值期間存在，一旦函數執行完成就會遭到刪除。

附註：區域性變數可節省記憶體，因為只暫時存在。另外這類變數也不會干擾既有的通用變數值。區域性變數必須用於 **For** 迴圈，以及用於暫時儲存多行函數中的值，因為您無法修改函數中的通用變數。

輸入範例的注意事項：關於輸入多行程式和函數定義的說明，請參閱產品手冊中的「計算工具」章節。

```
Define rollcount()=Func
                   Local i
                   1→i
                   Loop
                   If randInt(1,6)=randInt(1,6)
                   Goto end
                   i+1→i
                   EndLoop
                   Lbl end
                   Return i
                   EndFunc
                                          Done
rollcount()                                 16
rollcount()                                  3
```

Lock

目錄 >

Lock *Var1*[, *Var2*] [, *Var3*] ...

Lock *Var*.

鎖定指定變數或變數群組。您無法修改或刪除已鎖定的變數。

您無法鎖定或解開鎖定 *Ans* 系統變數，也無法鎖定 *stat*.或 *tvm* 系統變數群組。

附註：若將 **Lock** 指令用於已經解開鎖定的變數，則復原/重複功能的歷史記錄會遭到清除。

請參考第 頁碼：151 頁的 **unLock** 和第 頁碼：60 頁的 **getLockInfo()**。

```
a:=65                                       65
Lock a                                    Done
getLockInfo(a)                               1
a:=75                 "Error: Variable is locked."
DelVar a              "Error: Variable is locked."
Unlock a                                  Done
a:=75                                       75
DelVar a                                  Done
```

log()

ctrl 10ˣ 鍵

log(*值 1*[,*值 2*])⇒*值*

log(*列表 1*[,*值 2*])⇒*列表*

傳回第一個引數以 *值 2* 為底的對數值。

附註：另請參考**對數範本**，頁碼：2。

若是列表，則傳回各元素以 *Value2* 為底的對數值。

```
log10(2.)                              0.30103
log4(2.)                                   0.5
log3(10)-log3(5)                       0.63093
```

若複數格式模式是實數：

log()

ctrl 10ˣ 鍵

如果省略第二個引數，則會以 10 為底。

$\log_{10}(\{-3,1.2,5\})$

"Error: Non−real calculation"

若複數格式模式是直角座標：

$\log_{10}(\{-3,1.2,5\})$

$\{0.477121+1.36438\cdot i,0.079181,0.69897\}$

log(*方陣 1*[**,***值*]**)⇒***方陣***

傳回*方陣 1* 以*值*為底的對數矩陣。這和計算各元素以*值*為底的對數不同。若要了解計算方式，請參考 **cos()**。

方陣 1 必需可對角化。結果一定會包含浮點數。

如果省略基底引數，則會以 10 為底。

弧度角模式與直角複數格式：

$\log_{10}\left(\begin{bmatrix}1 & 5 & 3\\ 4 & 2 & 1\\ 6 & -2 & 1\end{bmatrix}\right)$

$\begin{bmatrix}0.795387+0.753438\cdot i & 0.003993-0.6474 \\ 0.194895-0.315095\cdot i & 0.462485+0.2707 \\ -0.115909-0.904706\cdot i & 0.488304+0.7774\end{bmatrix}$

若要看到完整結果，請按 ▲，然後使用 ◀ 與 ▶ 移動游標。

Logistic

目錄 > 📖

Logistic *X*, *Y*[, [*Freq*] [, *類別*, *包含*]]

計算 *X* 列表與 *Y* 列表的 y = (c/(1+a ·e^{-bx})) 羅吉斯迴歸(頻率為 *Freq*) 。*stat.results* 變數會儲存結果摘要(請參閱第 頁碼：134 頁) 。

所有列表的維數都必須相同，*包含*除外。

X、*Y* 是自變數和因變數列表。

Freq 是頻率值列表(非必要) 。*Freq* 的每個元素，可用於指定各 *X*、*Y* 對應資料點的出現頻率。預設值是 1。所有元素都必須是 ≥ 0 的整數。

*類別*是 *X*、*Y* 對應資料的數字或字串類別代號列表。

*包含*是一個以上類別代號的列表。只有本列表中類別代號所對應的資料項目，才會包含在計算作業中。

Logistic

目錄 > 📖

如要了解列表中有空元素時的影響，請參考°ß空元素°®，頁碼：198。

輸出變數	說明
stat.RegEqn	迴歸方程式：$c/(1+a \cdot e^{-bx})$
stat.a、stat.b、stat.c	迴歸係數
stat.Resid	迴歸殘差
stat.XReg	實際用於迴歸的已修改 *X 列表*中的資料點列表(根據 *Freq*、*類別列表*、*包含類別的限制)*
stat.YReg	實際用於迴歸的已修改 *Y 列表*中的資料點列表(根據 *Freq*、*類別列表*、*包含類別的限制)*
stat.FreqReg	*stat.XReg* 與 *stat.YReg 對應的頻率列表*

LogisticD

目錄 > 📖

LogisticD *X*, *Y* [, [*迭代*] , [*Freq*] [, *類別*, *包含*]]

利用指定*迭代*數，計算 *X* 列表與 *Y* 列表的 $y = (c/(1+a \cdot e^{-bx})+d)$ 羅吉斯迴歸(頻率為 *Freq*) 。*stat.results* 變數會儲存結果摘要(請參閱第 頁碼：134 頁) 。

所有列表的維數都必須相同，*包含*除外。

X、*Y* 是自變數和因變數列表。

Freq 是頻率值列表(非必要) 。*Freq* 的每個元素，可用於指定各 *X*、*Y* 對應資料點的出現頻率。預設值是 1。所有元素都必須是 ≥ 0 的整數。

*類別*是 *X*、*Y* 對應資料的數字或字串類別代號列表。

*包含*是一個以上類別代號的列表。只有本列表中類別代號所對應的資料項目，才會包含在計算作業中。

如要了解列表中有空元素時的影響，請參考°ß空元素°®，頁碼：198。

輸出變數	說明
stat.RegEqn	迴歸方程式：$c/(1+a \cdot e^{-bx})+d$)
stat.a、stat.b、stat.c、stat.d	迴歸係數
stat.Resid	迴歸殘差
stat.XReg	實際用於迴歸的已修改 *X 列表*中的資料點列表(根據 *Freq*、*類別列表*、*包含類別的限制)*
stat.YReg	實際用於迴歸的已修改 *Y 列表*中的資料點列表(根據 *Freq*、*類別列表*、*包含類別的限制)*
stat.FreqReg	*stat.XReg* 與 *stat.YReg 對應的頻率列表*

Loop

目錄 > 📖

Loop
區段
EndLoop

反覆執行*區段*中的語句。請注意，迴圈會永久執行，除非在*區段*中執行 **Goto** 或 **Exit** 指令。

*區段*是由「:」字元分隔的一系列語句。

輸入範例的注意事項：關於輸入多行程式和函數定義的說明，請參閱產品手冊中的「計算工具」章節。

```
Define rollcount()=Func
    Local i
    1→i
    Loop
    If randInt(1,6)=randInt(1,6)
    Goto end
    i+1→i
    EndLoop
    Lbl end
    Return i
    EndFunc
                                    Done
rollcount()                          16
rollcount()                           3
```

LU

目錄 >

LU *矩陣*, *l矩陣*, *u矩陣*, *p矩陣*[,*Tol*]

計算實數或複數矩陣的 Doolittle LU 上下分解值。*l矩陣*負責儲存下三角矩陣，*u矩陣*負責儲存上三角矩陣，*p矩陣*負責儲存置換矩陣(描述計算期間執行的列交換情形)。

l矩陣 · *u矩陣* = *p矩陣* · *矩陣*

若任何矩陣元素的絕對值小於 *Tol*(容許值)，則亦可將此元素視為零。只有在矩陣中包含浮點數項目，而且不包含尚未賦值的任何符號變數時，才會使用此容許值。其他時候都會忽略 *Tol*(容許值)。

- 若您使用 ctrl enter 或將**自動或近似值**模式設定成近似值，則系統會利用浮點運算法執行計算作業。
- 若指令省略或未使用 *Tol*，則預設容許值的計算方式如下：
  5E⁻14 ·max(dim(*矩陣*)) ·rowNorm(*矩陣*)

LU 因式分解運算法使用部分軸元法搭配列交換法。

$$\begin{bmatrix}6 & 12 & 18\\5 & 14 & 31\\3 & 8 & 18\end{bmatrix}\rightarrow m1 \qquad \begin{bmatrix}6 & 12 & 18\\5 & 14 & 31\\3 & 8 & 18\end{bmatrix}$$

LU *m1*,*lower*,*upper*,*perm* — *Done*

lower $\begin{bmatrix}1 & 0 & 0\\\frac{5}{6} & 1 & 0\\\frac{1}{2} & \frac{1}{2} & 1\end{bmatrix}$

upper $\begin{bmatrix}6 & 12 & 18\\0 & 4 & 16\\0 & 0 & 1\end{bmatrix}$

perm $\begin{bmatrix}1 & 0 & 0\\0 & 1 & 0\\0 & 0 & 1\end{bmatrix}$

M

mat▸list()

目錄 >

mat▸list(*矩陣***)⇒***列表*

傳回填入*矩陣*元素的列表。系統會從*矩陣*逐列複製元素。

附註：如果要從電腦鍵盤插入本函數，可輸入 **mat@>list(…)**。

mat▸list([1 2 3]) — {1,2,3}

$\begin{bmatrix}1 & 2 & 3\\4 & 5 & 6\end{bmatrix}\rightarrow m1 \qquad \begin{bmatrix}1 & 2 & 3\\4 & 5 & 6\end{bmatrix}$

mat▸list(*m1*) — {1,2,3,4,5,6}

max()

目錄 >

max(*值1*, *值2***)⇒***運算式*

max(*列表1*, *列表2***)⇒***列表*

max(*矩陣1*, *矩陣2***)⇒***矩陣*

max(2.3,1.4) — 2.3

max({1,2},{⁻4,3}) — {1,3}

max()

目錄 >

傳回兩個引數的最大值。若引數是兩個列表或矩陣，則傳回包含每組對應元素最大值的列表或矩陣。

max(*列表***)⇒***運算式*

傳回*列表*的最大元素。

$\max(\{0,1,-7,1.3,0.5\})$ → 1.3

max(*矩陣1***)⇒***矩陣*

傳回包含*矩陣1*中各行最大元素的列向量。

$\max\left(\begin{bmatrix}1 & -3 & 7\\ -4 & 0 & 0.3\end{bmatrix}\right)$ → $\begin{bmatrix}1 & 0 & 7\end{bmatrix}$

空元素會被忽略。如需空元素的詳細資訊，請參考第 頁碼：198 頁。

附註：另請參考 **min()**。

mean()

目錄 >

mean(*列表***[,** *頻率列表***])⇒***運算式*

傳回*列表*元素的平均值。

*頻率列表*的每個元素，代表*列表*中對應元素的連續出現次數。

$\text{mean}(\{0.2,0,1,-0.3,0.4\})$ → 0.26

$\text{mean}(\{1,2,3\},\{3,2,1\})$ → $\frac{5}{3}$

mean(*矩陣1***[,** *頻率矩陣***])⇒***矩陣*

傳回*矩陣1*中所有行之平均數的列向量。

*頻率矩陣*的每個元素，代表*矩陣1*中對應元素的連續出現次數。

空元素會被忽略。如需空元素的詳細資訊，請參考第 頁碼：198 頁。

直角向量格式：

$\text{mean}\left(\begin{bmatrix}0.2 & 0\\ -1 & 3\\ 0.4 & -0.5\end{bmatrix}\right)$ → $\begin{bmatrix}-0.133333 & 0.833333\end{bmatrix}$

$\text{mean}\left(\begin{bmatrix}\frac{1}{5} & 0\\ -1 & 3\\ \frac{2}{5} & \frac{-1}{2}\end{bmatrix}\right)$ → $\begin{bmatrix}\frac{-2}{15} & \frac{5}{6}\end{bmatrix}$

$\text{mean}\left(\begin{bmatrix}1 & 2\\ 3 & 4\\ 5 & 6\end{bmatrix},\begin{bmatrix}5 & 3\\ 4 & 1\\ 6 & 2\end{bmatrix}\right)$ → $\begin{bmatrix}\frac{47}{15} & \frac{11}{3}\end{bmatrix}$

median()

目錄 >

median(*列表***[,** *頻率列表***])⇒***運算式*

傳回*列表*元素的中位數。

*頻率列表*的每個元素，代表*列表*中對應元素的連續出現次數。

$\text{median}(\{0.2,0,1,-0.3,0.4\})$ → 0.2

median()

目錄 > 📖

median(*矩陣1***[,** *頻率矩陣***])⇒***矩陣*

傳回包含*矩陣1*中各行中位數的列向量。

*頻率矩陣*的每個元素，代表*矩陣1*中對應元素的連續出現次數。

$$\text{median}\left(\begin{bmatrix} 0.2 & 0 \\ 1 & -0.3 \\ 0.4 & -0.5 \end{bmatrix}\right) \qquad \begin{bmatrix} 0.4 & -0.3 \end{bmatrix}$$

附註：

- 列表或矩陣中的所有項目都必須化簡成數字。
- 列表或矩陣中的空元素會遭到忽略。如需空元素的詳細資訊，請參考第 頁碼：198 頁。

MedMed

目錄 > 📖

MedMed *X*,*Y* [, *Freq*] [, *類別*, *包含*]]

計算 *X* 列表與 *Y* 列表的y = (m ·x+b)中位數-中位數線(頻率為 *Freq*) 。*stat.results* 變數會儲存結果摘要(請參閱第 頁碼：134 頁) 。

所有列表的維數都必須相同，*包含*除外。

X、*Y* 是自變數和因變數列表。

Freq 是頻率值列表(非必要) 。*Freq* 的每個元素，可用於指定各 *X*、*Y* 對應資料點的出現頻率。預設值是 1。所有元素都必須是 ≥ 0 的整數。

*類別*是 *X*、*Y* 對應資料的數字或字串類別代號列表。

*包含*是一個以上類別代號的列表。只有本列表中類別代號所對應的資料項目，才會包含在計算作業中。

如要了解列表中有空元素時的影響，請參考°ß空元素°®，頁碼：198。

輸出變數	說明
stat.RegEqn	中位數-中位數線方程式：m ·x+b
stat.m、stat.b	模型係數

輸出變數	說明
stat.Resid	中位數-中位數線的殘差
stat.XReg	實際用於迴歸的已修改 *X 列表*中的資料點列表(根據 *Freq*、*類別列表*、*包含類別的限制)*
stat.YReg	實際用於迴歸的已修改 *Y 列表*中的資料點列表(根據 *Freq*、*類別列表*、*包含類別的限制)*
stat.FreqReg	*stat.XReg* 與 *stat.YReg 對應的頻率列表*

mid()

目錄 >

mid(*來源字串***,** *起點***[,** *個數***])⇒***字串*

傳回*來源字串*中的*個數*字元(從*起點*位置的字元算起) 。

若省略*個數*或個數超過*來源字串*長度，則傳回*來源字串*的所有字元(從*起點*位置的字元算起) 。

*個數*必須 ≥ 0。若*個數* = 0，則傳回空字串。

```
mid("Hello there",2)          "ello there"
mid("Hello there",7,3)        "the"
mid("Hello there",1,5)        "Hello"
mid("Hello there",1,0)        ""
```

mid(*來源列表***,** *起點* **[,** *個數***])⇒***列表*

傳回*來源列表*中的*個數*元素(從*起點*位置的元素算起) 。

若省略*個數*或個數超過*來源列表*長度，則傳回*來源列表*的所有元素(從*起點*位置的元素算起) 。

*個數*必須 ≥ 0。若個數 = 0，則傳回空列表。

```
mid({9,8,7,6},3)              {7,6}
mid({9,8,7,6},2,2)            {8,7}
mid({9,8,7,6},1,2)            {9,8}
mid({9,8,7,6},1,0)            {}
```

mid(*來源字串列表***,** *起點***[,** *個數***])⇒***列表*

傳回*來源字串列表*中的*個數*字串(從*起點*位置的元素算起) 。

```
mid({"A","B","C","D"},2,2)
                    {"B","C"}
```

min()

目錄 >

min(*值 1***,** *值 2***)⇒***運算式*

min(*列表 1, 列表 2***)⇒***列表*

min(*矩陣 1, 矩陣 2***)⇒***矩陣*

```
min(2.3,1.4)                  1.4
min({1,2},{-4,3})             {-4,2}
```

min()

目錄 > 📖

傳回兩個引數的最小值。若引數是兩個列表或矩陣，則傳回包含每組對應元素最小值的列表或矩陣。

min(*列表***)⇒***運算式*

傳回*列表*的最小元素。

$\min(\{0,1,-7,1.3,0.5\})$ → -7

min(*矩陣1***)⇒***矩陣*

傳回包含*矩陣1*中各行最小元素的列向量。

$\min\left(\begin{bmatrix}1 & -3 & 7\\ -4 & 0 & 0.3\end{bmatrix}\right)$ → $\begin{bmatrix}-4 & -3 & 0.3\end{bmatrix}$

附註：另請參考 **max()**。

mirr()

目錄 > 📖

mirr(*融資成本***,***現金再投資***,*CF0*,*CFList*[,*CFFreq*])**

傳回修正後內部投資報酬率的財務函數。

*融資成本*是您為現金流量金額支付的利率。

*現金再投資*是現金流量再投資的利率。

CF0 是時間為 0 時的初始現金流量，必須是實數。

CFList 是初始現金流量 CF0 之後的現金流量金額列表。

CFFreq 是選擇性的非必要列表，其中各元素可用於指定分組(相鄰) 現金流量金額(*CFList* 對應元素) 的出現頻率。預設值是 1; 若您輸入值，則必須是 < 10,000 的正整數。

$list1:=\{6000,-8000,2000,-3000\}$ → $\{6000,-8000,2000,-3000\}$

$list2:=\{2,2,2,1\}$ → $\{2,2,2,1\}$

$\text{mirr}(4.65,12,5000,list1,list2)$ → 13.41608607

附註：另請參考 **irr()**，頁碼：69。

mod()

目錄 >

mod(*值1***,** *值2***)⇒***運算式*

mod(*列表1***,** *列表2***)⇒***列表*

mod(*矩陣1***,** *矩陣2***)⇒***矩陣*

依照下列恆等式的定義，傳回第一個引數對第二個引數的模數：

mod(x,0) = x

mod(x,y) = x − y floor(x/y)

若第二個引數不是零，則該引數中的結果會呈週期變化。結果如果不是零，就是與第二個引數有相同符號。

若引數是兩個列表或兩個矩陣，則傳回包含每組對應元素模數的列表或矩陣。

附註：另請參考 **remain()**，頁碼：115

mod(7,0)	7
mod(7,3)	1
mod(⁻7,3)	2
mod(7,⁻3)	⁻2
mod(⁻7,⁻3)	⁻1
mod({12,⁻14,16},{9,7,⁻5})	{3,0,⁻4}

mRow()

目錄 >

mRow(*值***,** *矩陣1***,** *索引***)⇒***矩陣*

傳回*矩陣1*的副本，其中*矩陣1*的*索引*這一列，各元素都要乘以*值*。

$\text{mRow}\left(\frac{-1}{3},\begin{bmatrix}1 & 2\\ 3 & 4\end{bmatrix},2\right)$	$\begin{bmatrix}1 & 2\\ -1 & \frac{-4}{3}\end{bmatrix}$

mRowAdd()

目錄 >

mRowAdd(*值***,** *矩陣1***,** *索引1***,** *索引2***) ⇒** *矩陣*

傳回*矩陣1*的副本，其中*矩陣1*的*索引2*這一列，各元素都要取代成：

值 · 索引1 列 + 索引2 列

$\text{mRowAdd}\left(-3,\begin{bmatrix}1 & 2\\ 3 & 4\end{bmatrix},1,2\right)$	$\begin{bmatrix}1 & 2\\ 0 & -2\end{bmatrix}$

MultReg

目錄 >

MultReg *Y*, *X1*[,*X2*[,*X3*,...[,*X10*]]]

計算 *Y* **列表在** *X1*, *X2*, ..., *X10* **等列表上的多線性迴歸。***stat.results* **變數會儲存結果摘要(請參閱第 頁碼：134 頁) 。**

所有列表的維數都必須相同。

MultReg

目錄 > 📖

如要了解列表中有空元素時的影響，請參考°ß空元素°®，頁碼：198。

輸出變數	說明
stat.RegEqn	迴歸方程式：b0+b1 ·x1+b2 ·x2+ ...
stat.b0、stat.b1...	迴歸係數
stat.R^2	多元判定係數
stat.ŷList	ŷList = b0+b1 ·x1+ ...
stat.Resid	迴歸殘差

MultRegIntervals

目錄 > 📖

MultRegIntervals *Y*, *X1*[,*X2*[,*X3*,... [,*X10*]]],*XValList*[,*CLevel*]

計算預測的 y 值、單次觀測的信賴水準預測區間，以及平均反應的信賴水準信賴區間。

stat.results 變數會儲存結果摘要（請參閱第 頁碼：134 頁）。

所有列表的維數都必須相同。

如要了解列表中有空元素時的影響，請參考°ß空元素°®，頁碼：198。

輸出變數	說明
stat.RegEqn	迴歸方程式：b0+b1 ·x1+b2 ·x2+ ...
stat.ŷ	點估計值：ŷ = b0 + b1 · xl + ...（對於 *XValList*）
stat.dfError	誤差自由度
stat.CLower、stat.CUpper	平均反應的信賴區間
stat.ME	信賴區間邊際誤差
stat.SE	平均反應的標準誤差
stat.LowerPred、 stat.UpperrPred	單次觀測的預測區間
stat.MEPred	預測區間邊際誤差
stat.SEPred	預測的標準誤差

輸出變數	說明
stat.bList	迴歸係數列表，{b0,b1,b2,...}
stat.Resid	迴歸殘差

MultRegTests

目錄 >

MultRegTests *Y*, *X1*[,*X2*[,*X3*,...[,*X10*]]]

多線性迴歸檢定可計算已知資料的多線性迴歸，並提供全域 *F* 檢定統計量和係數的 *t* 檢定統計量。

stat.results 變數會儲存結果摘要(請參閱第 頁碼：134 頁)。

如要了解列表中有空元素時的影響，請參考°ß空元素°®，頁碼：198。

輸出

輸出變數	說明
stat.RegEqn	迴歸方程式：b0+b1 ·x1+b2 ·x2+ ...
stat.F	全域 *F* 檢定統計量
stat.PVal	與全域 *F* 統計量相關的 p 值
stat.R^2	多元判定係數
stat.AdjR^2	調整多元判定係數
stat.s	誤差的標準差
stat.DW	Durbin-Watson 統計量；用於判斷模型中是否有一階自相關現象
stat.dfReg	迴歸自由度
stat.SSReg	迴歸平方和
stat.MSReg	迴歸均方
stat.dfError	誤差自由度
stat.SSError	誤差平方和
stat.MSError	誤差均方
stat.bList	{b0,b1,...}係數表
stat.tList	t 統計量列表；bList 中的每個係數都有一個
stat.PList	每個 t 統計量的 p 值列表

輸出變數	說明
stat.SEList	bList 中的係數標準誤差列表
stat.ŷList	ŷList = b0+b1 ·x1+ …
stat.Resid	迴歸殘差
stat.sResid	標準化殘差；將殘差除以其標準差即可得出
stat.CookDist	庫克距離；根據殘差和槓桿作用測量觀察值的影響力
stat.Leverage	測量自變數的值與其平均值距離

N

nand

ctrl = 鍵

*布林運算式1***nand***布林運算式2* 傳回 *布林運算式*

*布林列表1***nand***布林列表2* 傳回 *布林列表*

*布林矩陣1***nand***布林矩陣2* 傳回 *布林矩陣*

傳回兩個引數的邏輯**and**運算的負值。傳回真偽值或方程式的化簡形式。

若是列表和矩陣，則傳回對應元素逐一比較的結果。

*整數1***nand***整數2⇒整數*

利用 **nand** 功能逐一個別位元比較兩個實際整數。系統內部會把兩個整數轉換成有正負號的 64 位元二進位數字。比較對應的位元時，如果兩個都是 0 則結果為 1；否則結果為 1。傳回的值代表位元結果，是根據基底模式顯示的。

您可以用任何整數做為基底。如果要輸入二進位或十六進位數字，則必須在前面分別加上 0b 或 0h 字首。如果沒有加上字首，就會把整數視為十進位(以 10 為底) 。

3 and 4	0
3 nand 4	-1
{1,2,3} and {3,2,1}	{1,2,1}
{1,2,3} nand {3,2,1}	{-2,-3,-2}

nCr()

目錄 >

nCr(*值1***, *值2***)** ⇒ *運算式*

已知*值1*和*值2*整數且*值1* ≥ *值2* ≥ 0，則 **nCr()** 代表一次取出*值2*件物品時，*值1*的組合數(又稱為二項數係數)。

nCr(z,3)|z=5	10
nCr(z,3)|z=6	20

nCr(*值***, 0)** ⇒ **1**

nCr(*值***, *負整數***)** ⇒ **0**

nCr(*值***, *正整數***)** ⇒ *值* ·**(***值*−1**)...**

(*值*−*正整數***+1) /***正整數***!**

nCr(*值***, *非整數***)** ⇒ *運算式***! / ((***值*−*非整數***)! ·***非整數***!)**

nCr(*列表1***, *列表2***)** ⇒ *列表*

根據兩個列表中的對應成對元素，傳回組合值列表。引數必須是維數大小相同的列表。

nCr({5,4,3},{2,4,2})	{10,1,3}

nCr(*矩陣1***, *矩陣2***)**⇒*矩陣*

根據兩個矩陣中的對應成對元素，傳回組合值矩陣。引數必須是維度相同的矩陣。

$\mathrm{nCr}\left(\begin{bmatrix}6 & 5\\ 4 & 3\end{bmatrix},\begin{bmatrix}2 & 2\\ 2 & 2\end{bmatrix}\right)$	$\begin{bmatrix}15 & 10\\ 6 & 3\end{bmatrix}$

nDerivative()

目錄 >

nDerivative(*運算式1***,***Var***=***值***[,***階數***])**⇒ *值*

nDerivative(*運算式1***,***Var***[,***階數***]) | *Var*=*值***⇒*值***

傳回利用自動微分法計算的數值導數。

指定*值*時，此值會取代之前賦值的所有變數，或目前以「|」取代的所有該變數。

若 *Var* 變數不含數值，則必須提供*值*。

*導數的階數*必須是 **1** 或 **2**。

nDerivative(|x|,x=1)	1
nDerivative(|x|,x)|x=0	undef
nDerivative(√(x−1),x)|x=1	undef

nDerivative()

目錄 >

附註：nDerivative() 運算法有其限制，亦即必須透過未化簡的運算式遞迴執行、計算一階和二階 (視情況) 導數的數值，並且對每個子運算式求值，可能導致意外結果。

請研究右邊的範例。x·(x^2+x)^(1/3) 的一階導數在 x=0 處等於 0。但是由於 (x^2+x)^(1/3) 子運算式的一階導數在 x=0 處尚未定義，而且此值必須用於計算整個運算式的導數，所以 **nDerivative()** 會回報結果是未定義，而且顯示警告訊息。

若您碰到這項限制，請用圖形確認這個解。您也可以嘗試使用 **centralDiff()**。

$$\text{nDerivative}\left(x\cdot\left(x^2+x\right)^{\frac{1}{3}},x,1\right)\Big|x=0 \qquad \text{undef}$$

$$\text{centralDiff}\left(x\cdot\left(x^2+x\right)^{\frac{1}{3}},x\right)\Big|x=0 \qquad 0.000033$$

newList()

目錄 >

newList(*元素數***)⇒***列表*

傳回維數是*元素數*的列表，每個元素都是零。

$$\text{newList}(4) \qquad \{0,0,0,0\}$$

newMat()

目錄 >

newMat(*numRows***, *numColumns***)⇒***矩陣*

傳回以零組成的矩陣，維度是 *numRows* X *numColumns*。

$$\text{newMat}(2,3) \qquad \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

nfMax()

目錄 >

nfMax(*Expr***, *Var***)⇒***值*

nfMax(*Expr***, *Var, 下限***)⇒***值*

nfMax(*Expr***, *Var, 下限***, *上限***)⇒***值*

nfMax(*Expr, Var***) |** *下限*≤*Var*≤*上限*⇒*值*

傳回 *Expr* 出現局部極大值時，*Var* 變數的可能數值。

若您提供*下限*與*上限*，則本函數會在閉區間 [*下限*，*上限*] 之間尋找局部極大值。

$$\text{nfMax}\left(-x^2-2\cdot x-1,x\right) \qquad -1.$$

$$\text{nfMax}\left(0.5\cdot x^3-x-2,x,-5,5\right) \qquad 5.$$

nfMin()

目錄 >

nfMin(*Expr***,** *Var***)⇒***值*

nfMin(*Expr***,** *Var, 下限***)⇒***值*

nfMin(*Expr***,** *Var, 下限***,** *上限***)⇒***值*

nfMin(*Expr, Var***) |** *下限≤Var≤上限⇒值*

傳回 *Expr* 出現局部極小值時，*Var* 變數的可能數值。

若您提供*下限*與*上限*，則本函數會在閉區間 [*下限*，*上限*] 之間尋找最小值。

$\text{nfMin}(x^2+2\cdot x+5,x)$ -1.

$\text{nfMin}(0.5\cdot x^3-x-2,x,-5,5)$ -5.

nInt()

目錄 >

nInt(*運算式1, Var, 下限, 上限***)⇒***運算式*

若 *運算式1* 被積函數只有 *Var* 變數，且若*下限*和*上限*是常數、正向 ∞ 或負向 ∞，則 **nInt()** 會傳回 ∫(*運算式1, Var, 下限, 上限*) 的近似值。這個近似值是 *下限*<*Var*<*上限*區間中，被積函數幾個樣本值的加權平均值。

$\text{nInt}(e^{-x^2},x,-1,1)$ 1.49365

本函數的目標是求出六個有效位數。適應性運算法在可能達到目標，或是更多樣本也不太可能改善結果時，就會停止計算。

$\text{nInt}(\cos(x),x,-\pi,\pi+1.\text{E}-12)$ -1.04144E-12

如果看似無法達到目標，則會顯示警告 (準確性可疑) 。

若要計算多重數值積分，可嵌套 **nInt ()**。積分極限可能受到積分函數外的積分變數影響。

$\text{nInt}\left(\text{nInt}\left(\frac{e^{-x\cdot y}}{\sqrt{x^2-y^2}},y,-x,x\right),x,0,1\right)$ 3.30423

nom()

目錄 >

nom(*實質利率,CpY***)⇒***值*

可將 *effectiveRate* 實際年利率轉成名目利率的財務函數 (已知 *CpY* 是每年複利期數) 。

effectiveRate 必須是實數，*CpY* 必須是 > 0 的實數。

$\text{nom}(5.90398,12)$ 5.75

nom()

目錄 >

附註：另請參考 **eff()**，頁碼：42。

nor

ctrl = 鍵

*布林運算式1***nor***布林運算式2* 傳回 *布林運算式*

*布林列表1***nor***布林列表2* 傳回 *布林列表*

*布林矩陣1***nor***布林矩陣2* 傳回 *布林矩陣*

傳回兩個引數的邏輯**or運算的否定值。傳回真偽值或方程式的化簡形式。**

若是列表和矩陣，則傳回對應元素逐一比較的結果。

*整數1***nor***整數2*⇒*整數*

利用 **nor** 功能逐一個別位元比較兩個實際整數。系統內部會把兩個整數轉換成有正負號的 64 位元二進位數字。比較對應的位元時，如果兩個都是 1 則結果為 1；否則結果為 0。傳回的值代表位元結果，是根據基底模式顯示的。

您可以用任何整數做為基底。如果要輸入二進位或十六進位數字，則必須在前面分別加上 0b 或 0h 字首。如果沒有加上字首，就會把整數視為十進位(以 10 為底) 。

3 or 4	7
3 nor 4	-8
{1,2,3} or {3,2,1}	{3,2,3}
{1,2,3} nor {3,2,1}	{-4,-3,-4}

norm()

目錄 >

norm(*矩陣***)**⇒*運算式*

norm(*向量***)**⇒*運算式*

傳回 Frobenius 範數。

$\text{norm}\left(\begin{bmatrix}1 & 2\\3 & 4\end{bmatrix}\right)$	5.47723
$\text{norm}\left(\begin{bmatrix}1 & 2\end{bmatrix}\right)$	2.23607
$\text{norm}\left(\begin{bmatrix}1\\2\end{bmatrix}\right)$	2.23607

normCdf()

目錄 >

normCdf(*下限***,***上限***[,μ[,σ]])⇒***數字*(若 *下限*和 *上限*是數字) 或*列表*(若 *下限*和 *上限*是列表)

針對指定的 μ (預設值=0) 與 σ (預設值=1)，計算 *下限*與 *上限*間的常態分布機率。

對於 P(X ≤ *上限*)，請設定 *下限* = −9E999。

normPdf()

目錄 >

normPdf(*XVal***[,μ[,σ]])⇒***數字*(若 *XVal* 是數字) 或*列表*(若 *XVal* 是列表)

針對指定 μ 與 σ，計算常態分布在指定 *XVal* 值的機率密度函數(pdf) 。

not (非)

目錄 >

not *布林運算式⇒布林運算式*

傳回真偽值或引數的簡化形式。

not (2≥3)	true
not 0hB0▸Base16	0hFFFFFFFFFFFFFF4F
not not 2	2

not *整數1⇒整數*

傳回實際整數的一補數。系統內部會把 *整數1* 轉換成有正負號的 64 位元二進位數字。每個位元的值都會顛倒成一補數(0 變成 1，反之亦然) 。所顯示的結果會以基底模式為準。

您可以用任何整數做為基底。如果要輸入二進位或十六進位數字，則必須在前面分別加上 0b 或 0h 字首。如果沒有加上字首，就會把整數視為十進位(以 10 為底) 。

如果您輸入的十進位整數太大，超出具正負號 64 位元二進位格式的範圍，系統會以對稱模數運算法使數值落入適當範圍。如需詳細資訊，請參閱 **▸Base2**，頁碼：16。

十六進位基底模式：

重要訊息：數字零，而非英文字母 O。

not 0h7AC36	0hFFFFFFFFFFF853C9

二進位基底模式：

0b100101▸Base10	37
not 0b100101	
0b1111111111111111111111111111111▸	
not 0b100101▸Base10	−38

若要看到完整結果，請按 ▲，然後使用 ◀ 與 ▶ 移動游標。

附註：您可輸入最長 64 位的二進位數字(不含 0b 字首) 。您可輸入最長 16 位的十六進位數字。

nPr()

目錄 >

nPr(*值1***, *值2***)** ⇒ *運算式*

已知*值1*和*值2*整數且*值1* ≥ *值2* ≥ 0，則 **nPr()** 代表一次取出*值2*件物品時，*值1*的排列數。

nPr(z,3)|z=5	60
nPr(z,3)|z=6	120
nPr({5,4,3},{2,4,2})	{20,24,6}
nPr([6 5; 4 3],[2 2; 2 2])	[30 20; 12 6]

nPr(*值***, 0)** ⇒ **1**

nPr(*值***, *負整數***)** ⇒ **1/((***值***+1)** ·**(***值***+2)**… **(***值***−***負整數***))**

nPr(*值***, *正整數***)** ⇒ *值* ·**(***值***−1).(***值***−***正整數***+1)**

nPr(*值***, *非整數***)** ⇒ *值***!/(***值***−***非整數***)!**

nPr(*列表1***, *列表2***)** ⇒ *列表*

根據兩個列表中的對應成對元素，傳回排列值列表。引數必須是維數大小相同的列表。

nPr({5,4,3},{2,4,2})	{20,24,6}

nPr(*矩陣1***, *矩陣2***)** ⇒ *矩陣*

根據兩個矩陣中的對應成對元素，傳回排列值矩陣。引數必須是維度相同的矩陣。

nPr([6 5; 4 3],[2 2; 2 2])	[30 20; 12 6]

npv()

目錄 >

npv(*利率***,***CFO***,***CFList***[,***CFFreq***])**

可計算淨現值、現金流入/流出現值總和的財務函數。npv 計算結果如果是正值，代表投資有獲利。

*利率*是現金流量(資金成本) 一期的貼現率。

CF0 是時間為 0 時的初始現金流量，必須是實數。

CFList 是初始現金流量 *CF0* 之後的現金流量金額列表。

CFFreq 列表各元素可用於指定分組 (相鄰) 現金流量金額(*CFList* 對應元素) 的出現頻率。預設值是 1; 若您輸入值，則必須是 < 10,000 的正整數。

list1:={6000,⁻8000,2000,⁻3000}	{6000,⁻8000,2000,⁻3000}
list2:={2,2,2,1}	{2,2,2,1}
npv(10,5000,*list1*,*list2*)	4769.91

nSolve()

目錄 > 📖

nSolve(*方程式***,***Var***[=***猜測值***])⇒***數字或錯誤字串*

nSolve(*方程式***,***Var***[=***猜測值***],***下限***) ⇒***數字或錯誤字串*

nSolve(*方程式***,***Var***[=***猜測值***],***下限***,***上限***) ⇒***數字或錯誤字串*

nSolve(*方程式***,***Var***[=***猜測值***]) |** *下限* ≤*Var*≤*上限* ⇒*數字或錯誤字串*

反覆搜尋*方程式*其中一個變數的近似實數解。指定變數為：

變數

– 或 –

變數 = *實數*

舉例來說，x 和 x=3 都有效。

nSolve() 會試圖求出殘差是零的一個點，或求出兩個位置較近的點，且點的殘差正負號相反、殘差數值不會太大。若無法用適當數量的樣本點得到以上結果，則傳回「找不到解」字串。

$\text{nSolve}(x^2+5\cdot x-25=9,x)$	3.84429
$\text{nSolve}(x^2=4,x=-1)$	-2.
$\text{nSolve}(x^2=4,x=1)$	2.

附註：若有多個解，則可利用猜測值找出特定解。

$\text{nSolve}(x^2+5\cdot x-25=9,x)\|x<0$	-8.84429
$\text{nSolve}\left(\frac{(1+r)^{24}-1}{r}=26,r\right)\|r>0 \text{ and } r<0.25$	0.006886
$\text{nSolve}(x^2=-1,x)$	"No solution found"

O

OneVar

目錄 > 📖

OneVar [**1,**]*X*[,[*Freq*][,*類別*,*包含*]]

OneVar [*n*,]*X1*,*X2*[*X3*[,...[,*X20*]]]

計算多達 20 個列表的單元統計值。*stat.results* 變數會儲存結果摘要(請參閱第 頁碼: 134 頁) 。

所有列表的維數都必須相同，*包含*除外。

Freq 是頻率值列表(非必要) 。*Freq* 的每個元素，可用於指定各 *X*、*Y* 對應資料點的出現頻率。預設值是 1。所有元素都必須是 ≥ 0 的整數。

OneVar

目錄 > 📖

*類別*是 *X* 對應值的數字類別代號列表。

*包含*是一個以上類別代號的列表。只有本列表中類別代號所對應的資料項目，才會包含在計算作業中。

X、*Freq*、*類別*的任何列表中若有空元素，則所有這些列表的對應元素就會是空元素。*X1* 至 *X20* 的任何列表中若有空元素，則所有這些列表的對應元素就會是空元素。如需空元素的詳細資訊，請參考第 頁碼：198 頁。

輸出變數	說明
stat.$\bar{x}$	x 值的平均值
stat.Σx	x 值的和
stat.Σx^2	x^2 值的和
stat.sx	x 的樣本標準差
stat. x	x 的母群體標準差
stat.n	資料點數量
stat.MinX	x 值的最小值
stat.Q_1X	x 的第一四分位數
stat.MedianX	x 的中位數
stat.Q_3X	x 的第三四分位數
stat.MaxX	x 值的最大值
stat.SSX	x 的離均差平方和

or (或)

目錄 > 📖

布林運算式1 **or** *布林運算式2* 傳回 *布林運算式*

布林列表1 **or** *布林列表2* 傳回 *布林列表*

布林矩陣1 **or** *布林矩陣2* 傳回 *布林矩陣*

傳回真偽值或原始輸入的化簡形式。

```
Define g(x)=Func                    Done
              If x≤0 or x≥5
              Goto end
              Return x·3
              Lbl end
              EndFunc
g(3)                                9
g(0)          A function did not return a value
```

or (或)

目錄 >

若其中一個運算式或兩個運算式可化簡為真，則傳回真。只有在兩個運算式求值後結果皆是偽時，才傳回偽。

附註：請參考 **xor**。

輸入範例的注意事項：關於輸入多行程式和函數定義的說明，請參閱產品手冊中的「計算工具」章節。

整數1 **or** *整數2*⇒*整數*

利用 or 功能個別位元比較兩個實際整數。系統內部會把兩個整數轉換成有正負號的 64 位元二進位數字。比較對應的位元時，如果其中一個位元是 1，則結果是 1；如果兩個位元都是 0，則結果是 0。傳回的值代表位元結果，並會以基底模式為顯示依據。

您可以用任何整數做為基底。如果要輸入二進位或十六進位數字，則必須在前面分別加上 0b 或 0h 字首。如果沒有加上字首，就會把整數視為十進位 (以 10 為底) 。

如果您輸入的十進位整數太大，超出具正負號 64 位元二進位格式的範圍，系統會以對稱模數運算法使數值落入適當範圍。如需詳細資訊，請參閱 **▶Base2**，頁碼：16。

附註：請參考 **xor**。

十六進位基底模式：

0h7AC36 or 0h3D5F	0h7BD7F

重要訊息：數字零，而非英文字母 O。

二進位基底模式：

0b100101 or 0b100	0b100101

附註：您可輸入最長 64 位的二進位數字(不含 0b 字首) 。您可輸入最長 16 位的十六進位數字。

ord()

目錄 >

ord(*字串***)**⇒*整數*

ord(*列表1***)**⇒*列表*

傳回字元*字串*中第一個字元的數字代號，或列表中各元素第一個字元的列表。

ord("hello")	104
char(104)	"h"
ord(char(24))	24
ord({"alpha","beta"})	{97,98}

P

P▶Rx()

目錄 >

P▶Rx(*rExpr***,** θ*Expr***)**⇒*運算式*

弧度角模式：

P▸Rx()

目錄 >

P▸Rx(*r列表*, θ*列表*)⇒*列表*

P▸Rx(*r矩陣*, θ*矩陣*)⇒*矩陣*

傳回 (r, θ) 成對值的等值 x 座標。

附註：系統會根據目前的角度模式，將 θ 引數解譯為度數角、梯度角或弧度角。若引數是運算式，則可使用 °、G 或 r 來暫時覆寫角度模式設定。

附註：如果要從電腦鍵盤插入本函數，可輸入 **P@>Rx(…)**。

$P\blacktriangleright Rx(4,60°)$ 2.

$P\blacktriangleright Rx\left(\{-3,10,1.3\},\left\{\frac{\pi}{3},\frac{-\pi}{4},0\right\}\right)$ $\{-1.5,7.07107,1.3\}$

P▸Ry()

目錄 >

P▸Ry(*r值*, θ*值*)⇒*值*

P▸Ry(*r列表*, θ*列表*)⇒*列表*

P▸Ry(*r矩陣*, θ*矩陣*)⇒*矩陣*

傳回 (r, θ) 成對值的等值 y 座標。

附註：系統會根據目前的角度模式，將 θ 引數解譯為度數角、弧度角或梯度角

附註：如果要從電腦鍵盤插入本函數，可輸入 **P@>Ry(…)**。

弧度角模式：

$P\blacktriangleright Ry(4,60°)$ 3.4641

$P\blacktriangleright Ry\left(\{-3,10,1.3\},\left\{\frac{\pi}{3},\frac{-\pi}{4},0\right\}\right)$ $\{-2.59808,-7.07107,0.\}$

PassErr

目錄 >

PassErr

將錯誤傳遞到下一級。

若 *errCode* 系統變數是零，則 **PassErr** 無作用。

Try…Else…EndTry 段的 **Else** 子句應該使用 **ClrErr** 或 **PassErr**。若要處理或忽略錯誤，請使用 **ClrErr**。若不知如何處理錯誤，請用 **PassErr** 送到下一個錯誤處理程式。若已經沒有待處理的 **Try…Else…EndTry** 錯誤處理程式，則錯誤對話方塊將顯示為正常。

附註：另請參考第 頁碼：22 頁的 **ClrErr** 和第 頁碼：145 頁的 **Try**。

若需 **PassErr** 的參考範例，請參考第 頁碼：145 頁 **Try 指令下的例 2。**

PassErr

輸入範例的注意事項：關於輸入多行程式和函數定義的說明，請參閱產品手冊中的「計算工具」章節。

piecewise()

piecewise(*運算式1* **[,** *Cond1* **[,** *運算式2* **[,** *Cond2* **[, ...]]]])**

以列表形式傳回分段函數的定義。您也可以利用範本定義分段函數。

附註：另請參考**分段函數範本**，頁碼：2。

Define $p(x)=\begin{cases} x, & x>0 \\ \text{undef}, & x\le 0 \end{cases}$	*Done*
$p(1)$	1
$p(-1)$	undef

poissCdf()

poissCdf(λ,*下限***,***上限***)⇒***數字*(若*下限*和*上限*是數字)或*列表*(若*下限*和*上限*是列表)

poissCdf(λ,*上限***)**for P(0≤X≤*上限*)⇒*數字*(若*上限*是數字)或列表(若*上限*是列表)

以 λ 指定平均值計算卜松離散分布的累積機率。

對於 P(X ≤ *上限*)，請設定 *下限*=0

poissPdf()

poissPdf(λ,*XVal***)⇒***數字*(若 *XVal* 是數字)或*列表*(若 *XVal* 是列表)

以 λ 指定平均值計算卜松離散散分布的機率。

▸Polar

向量 **▸Polar**

附註：如果要從電腦鍵盤插入本運算子，可輸入 **@>Polar**。

[1 3.]▸Polar	[3.16228 ∠71.5651]

▶Polar

目錄 > 📖

以極座標形式 [r∠θ] 顯示*向量*。向量必須是 2 維，而且可以是列向量也可以是行向量。

附註：▶Polar 是顯示格式指令，並非轉換函數。本指令只能用於輸入線末尾，而且無法更新 *ans*。

附註：另請參考 **▶Rect**，頁碼：112。

複數值 **▶Polar**

以極座標形式顯示*複數向量*。

- 若是角度模式則傳回 (r∠θ)。
- 若是弧度角模式則傳回 re$^{i\theta}$。

*複數值*可以是任何複數形式，但輸入 re$^{i\theta}$ 時會導致度數角模式錯誤。

附註：您必須使用括弧才能輸入 (r∠θ) 極座標。

弧度角模式：

$(3+4\cdot i)$▶Polar	$e^{0.927295\cdot i}\cdot 5$
$\left(\left(4\angle\frac{\pi}{3}\right)\right)$▶Polar	$e^{1.0472\cdot i}\cdot 4.$

梯度角模式：

$(4\cdot i)$▶Polar	$(4\angle\ 100.)$

度數角模式：

$(3+4\cdot i)$▶Polar	$(5\angle\ 53.1301)$

polyEval()

目錄 > 📖

polyEval(*列表1, 運算式1***)⇒***運算式*

polyEval(*列表1, 列表2***)⇒***運算式*

將第一個引數解譯為降次多項式的係數，並傳回針對第二個引數值進行求值的多項式。

polyEval({1,2,3,4},2)	26
polyEval({1,2,3,4},{2,-7})	{26,-262}

polyRoots()

目錄 > 📖

polyRoots(*Poly*,*Var***) ⇒***列表*

polyRoots(*係數表***) ⇒***列表*

第一個語法 polyRoots(*Poly*,*Var*) 可傳回 *Poly* 多項式的實根列表(對 *Var* 變數)。若無實根，則傳回空白列表。{ }。

polyRoots(y^3+1,y)	{-1}
cPolyRoots(y^3+1,y)	$\left\{-1,\frac{1}{2}-\frac{\sqrt{3}}{2}\cdot i,\frac{1}{2}+\frac{\sqrt{3}}{2}\cdot i\right\}$
polyRoots($x^2+2\cdot x+1,x$)	{-1,-1}

polyRoots()

目錄 >

Poly 必須是一個變數中的展開多項式。請勿使用未展開的形式，例如 $y^2 \cdot y+1$ 或 $x \cdot x+2 \cdot x+1$

第二個語法 **polyRoots(***係數表***)** 可傳回係數表中係數的實根列表。

附註：另請參考 **cPolyRoots()**，頁碼：29。

$\text{polyRoots}(y^3+1,y)$ $\{-1\}$

$\text{cPolyRoots}(y^3+1,y)$ $\{-1,0.5-0.866025 \cdot i,0.5+0.866025 \cdot i\}$

$\text{polyRoots}(x^2+2 \cdot x+1,x)$ $\{-1,-1\}$

$\text{polyRoots}(\{1,2,1\})$ $\{-1,-1\}$

PowerReg

目錄 >

PowerReg *X*,*Y* [, *Freq*] [, *類別*, *包含*]]

計算 *X* 列表與 *Y* 列表的 $y = (a \cdot (x)^b)$乘冪迴歸(頻率為 *Freq*) 。*stat.results* 變數會儲存結果摘要(請參閱第 頁碼：134 頁) 。

所有列表的維數都必須相同，*包含*除外。

X、*Y* 是自變數和因變數列表。

Freq 是頻率值列表(非必要) 。*Freq* 的每個元素，可用於指定各 *X*、*Y* 對應資料點的出現頻率。預設值是 1。所有元素都必須是 ≥ 0 的整數。

*類別*是 *X*、*Y* 對應資料的數字或字串類別代號列表。

*包含*是一個以上類別代號的列表。只有本列表中類別代號所對應的資料項目，才會包含在計算作業中。

如要了解列表中有空元素時的影響，請參考°ß空元素°®，頁碼：198。

輸出變數	說明
stat.RegEqn	迴歸方程式：$a \cdot (x)^b$
stat.a、stat.b	迴歸係數
stat.r^2	所轉換資料之線性判定的係數
stat.r	所轉換資料 (ln(x), ln(y)) 的相關係數
stat.Resid	與乘冪模型相關的殘差

輸出變數	說明
stat.ResidTrans	與所轉換資料之線性擬合相關的殘差
stat.XReg	實際用於迴歸的已修改 *X 列表*中的資料點列表(根據 *Freq*、*類別列表*、*包含類別的限制)*
stat.YReg	實際用於迴歸的已修改 *Y 列表*中的資料點列表(根據 *Freq*、*類別列表*、*包含類別的限制)*
stat.FreqReg	*stat.XReg* 與 *stat.YReg 對應的頻率列表*

Prgm

目錄 >

Prgm
區段
EndPrgm

製作使用者自行定義程式的範本。必須搭配 **Define**、**Define LibPub** 或 **Define LibPriv** 指令。

*區段*可以只是一個語句、由「:」字元分隔的一連串語句，也可以是分成多行的一連串語句。

輸入範例的注意事項：關於輸入多行程式和函數定義的說明，請參閱產品手冊中的「計算工具」章節。

計算最大公因數並顯示中間結果。

```
Define proggcd(a,b)=Prgm
                    Local d
                    While b≠0
                    d:=mod(a,b)
                    a:=b
                    b:=d
                    Disp a," ",b
                    EndWhile
                    Disp "GCD=",a
                    EndPrgm
                                Done
```

```
proggcd(4560,450)
                                450 60
                                60 30
                                30 0
                                GCD=30
                                Done
```

prodSeq()

請參考 Π()，頁碼：173。

Product (PI)

請參考 Π()，頁碼：173。

product()

目錄 >

product(*列表***[,** *起點***[,** *終點***]])⇒***運算式*

傳回*列表*元素的乘積。*起點*和*終點*可選擇性輸入，用於指定元素範圍。

$\text{product}(\{1,2,3,4\})$ → 24

$\text{product}(\{4,5,8,9\},2,3)$ → 40

product(*矩陣1***[,** *起點***[,** *終點***]])⇒***矩陣*

傳回包含*矩陣1*中各行元素乘積的列向量。*起點*和*終點*可選擇性輸入，用於指定列範圍。

$\text{product}\left(\begin{bmatrix}1&2&3\\4&5&6\\7&8&9\end{bmatrix}\right)$ → $\begin{bmatrix}28&80&162\end{bmatrix}$

$\text{product}\left(\begin{bmatrix}1&2&3\\4&5&6\\7&8&9\end{bmatrix},1,2\right)$ → $\begin{bmatrix}4&10&18\end{bmatrix}$

空元素會被忽略。如需空元素的詳細資訊，請參考第 頁碼: 198 頁。

propFrac()

目錄 >

propFrac(*值1***[,** *Var***])⇒***值*

propFrac(*有理數***):** 傳回*有理數*形式的整數與分數總和，兩者正負號必須相同，且分母比分子大。

$\text{propFrac}\left(\frac{4}{3}\right)$ → $1+\frac{1}{3}$

$\text{propFrac}\left(\frac{-4}{3}\right)$ → $-1-\frac{1}{3}$

propFrac(*有理式***,***Var***):** 傳回適當比例和多項式的總和(對 *Var* 變數) 。在每個適當比例中，分母的 *Var* 次數必須超過分子的 *Var* 次數。系統將匯集 *Var* 的類似乘冪。各項及其因式以 *Var* 為主變數進行排序。

若省略 *Var*，則會根據主變數展開真分數。接著先對多項式部份之係數的主變數，將係數轉為真分數，然後依此類推。

您可用 **propFrac()** 函數表示帶分數，並說明帶分數的加減運算。

$\text{propFrac}\left(\frac{11}{7}\right)$ → $1+\frac{4}{7}$

$\text{propFrac}\left(3+\frac{1}{11}+5+\frac{3}{4}\right)$ → $8+\frac{37}{44}$

$\text{propFrac}\left(3+\frac{1}{11}-\left(5+\frac{3}{4}\right)\right)$ → $-2-\frac{29}{44}$

Q

QR

目錄 >

QR *矩陣***,** *q矩陣***,** *r矩陣***[,** *Tol***]**

m1 中的浮點數(9.) 會使得系統以浮點數形式計算結果。

QR

目錄 > 📖

計算實數或複數矩陣的 Householder QR 分解值。系統會將求出的 Q、R 矩陣儲存到指定*矩陣*中。Q 矩陣是單一矩陣。R 矩陣是上三角矩陣。

若任何矩陣元素的絕對值小於 *Tol*(容許值) ，則亦可將此元素視為零。只有在矩陣中包含浮點數項目，而且不包含尚未賦值的任何符號變數時，才會使用此容許值。其他時候都會忽略 *Tol*(容許值) 。

- 若您使用 ctrl enter 或將**自動或近似值**模式設定成近似值，則系統會利用浮點運算法執行計算作業。
- 若指令省略或未使用 *Tol*，則預設容許值的計算方式如下：
  5E⁻14 ·max(dim(*矩陣*)) ·rowNorm(*矩陣*)

進行 QR 分解的數值計算時，會進行 Householder 轉換。計算符號解時會採用 Gram-Schmidt。*qMatName* 的行向量是單範正交基底向量，跨越*矩陣*定義的空間。

$$\begin{bmatrix}1&2&3\\4&5&6\\7&8&9.\end{bmatrix}\rightarrow m1 \qquad \begin{bmatrix}1&2&3\\4&5&6\\7&8&9.\end{bmatrix}$$

QR *m1,qm,rm* — *Done*

$$qm \quad \begin{bmatrix}0.123091&0.904534&0.408248\\0.492366&0.301511&-0.816497\\0.86164&-0.301511&0.408248\end{bmatrix}$$

$$rm \quad \begin{bmatrix}8.12404&9.60114&11.0782\\0.&0.904534&1.80907\\0.&0.&0.\end{bmatrix}$$

QuadReg

目錄 > 📖

QuadReg *X*,*Y* [, *Freq*] [, *類別*, *包含*]]

計算 *X* 列表與 *Y* 列表的y = a ·x²+b ·x+c二次多項式迴歸(頻率為 *Freq*) 。*stat.results* 變數會儲存結果摘要(請參閱第 頁碼: 134 頁) 。

所有列表的維數都必須相同，*包含*除外。

X、*Y* 是自變數和因變數列表。

Freq 是頻率值列表(非必要) 。*Freq* 的每個元素，可用於指定各 *X*、*Y* 對應資料點的出現頻率。預設值是 1。所有元素都必須是 ≥ 0 的整數。

*類別*是 *X*、*Y* 對應資料的數字或字串類別代號列表。

*包含*是一個以上類別代號的列表。只有本列表中類別代號所對應的資料項目，才會包含在計算作業中。

QuadReg

如要了解列表中有空元素時的影響，請參考°ß空元素°®，頁碼：198。

輸出變數	說明
stat.RegEqn	迴歸方程式：a ·x^2+b ·x+c
stat.a、stat.b、stat.c	迴歸係數
stat.R^2	判定係數
stat.Resid	迴歸殘差
stat.XReg	實際用於迴歸的已修改 *X 列表*中的資料點列表(根據 *Freq*、*類別列表*、*包含類別的限制)*
stat.YReg	實際用於迴歸的已修改 *Y 列表*中的資料點列表(根據 *Freq*、*類別列表*、*包含類別的限制)*
stat.FreqReg	*stat.XReg* 與 *stat.YReg 對應的頻率列表*

QuartReg

QuartReg *X*,*Y* [, *Freq*] [, *類別*, *包含*]]

計算 *X* 列表與 *Y* 列表的 y = a ·x^4+b ·x^3+c ·x^2+d ·x+e四次多項式迴歸(頻率為 *Freq*) 。*stat.results* 變數會儲存結果摘要(請參閱第 頁碼：134 頁) 。

所有列表的維數都必須相同，*包含*除外。

X、*Y* 是自變數和因變數列表。

Freq 是頻率值列表(非必要) 。*Freq* 的每個元素，可用於指定各 *X*、*Y* 對應資料點的出現頻率。預設值是 1。所有元素都必須是 ≥ 0 的整數。

*類別*是 *X*、*Y* 對應資料的數字或字串類別代號列表。

*包含*是一個以上類別代號的列表。只有本列表中類別代號所對應的資料項目，才會包含在計算作業中。

如要了解列表中有空元素時的影響，請參考°ß空元素°®，頁碼：198。

輸出變數	說明
stat.RegEqn	迴歸方程式：a ·x^4+b ·x^3+c ·x^2+d ·x+e
stat.a、stat.b、stat.c、stat.d、stat.e	迴歸係數
stat.R^2	判定係數
stat.Resid	迴歸殘差
stat.XReg	實際用於迴歸的已修改 *X 列表*中的資料點列表(根據 *Freq*、*類別列表*、*包含類別的限制)*
stat.YReg	實際用於迴歸的已修改 *Y 列表*中的資料點列表(根據 *Freq*、*類別列表*、*包含類別的限制)*
stat.FreqReg	*stat.XReg* 與 *stat.YReg 對應的頻率列表*

R

R▶Pθ()

目錄 >

R▶Pθ (*x值***,** *y值***) ⇒** *值*
R▶Pθ (*x列表***,** *y列表***) ⇒** *列表*
R▶Pθ (*x矩陣***,** *y矩陣***) ⇒** *矩陣*

傳回 (x,y) 成對引數的相等 *θ* 座標。

注意:系統會根據目前的角度模式設定，將結果傳回為度數角、梯度角或弧度角。

注意:如果要使用電腦鍵盤插入本函數，可輸入 **R@>Ptheta(…)**。

度數角模式:

R▶Pθ(2,2) 45.

梯度角模式:

R▶Pθ(2,2) 50.

弧度角模式:

R▶Pθ(3,2) 0.588003

R▶Pθ([3 -4 2],[0 π/4 1.5]) [0. 2.94771 0.643501]

R▶Pr()

目錄 >

R▶Pr (*x值***,** *y值***) ⇒** *值*
R▶Pr (*x列表***,** *y列表***) ⇒** *列表*
R▶Pr (*x矩陣***,** *y矩陣***) ⇒** *矩陣*

傳回 (*x*,*y*) 成對引數的等值 r 座標。

弧度角模式:

R▶Pr(3,2) 3.60555

R▶Pr([3 -4 2],[0 π/4 1.5]) [3 4.07638 5/2]

R▶Pr()

目錄 >

注意:如果要使用電腦鍵盤插入本函數，可輸入 **R@>Pr(...)**。

▶Rad

目錄 >

值1▶*Rad* ⇒ *值*

把引數轉成弧度角測量單位。

注意:如果要使用電腦鍵盤插入本運算子，可輸入 **@>Rad**。

度數角模式:

(1.5)▶Rad	(0.02618)r

梯度角模式:

(1.5)▶Rad	(0.023562)r

rand()

目錄 >

rand() ⇒ *運算式*
rand(*試驗次數***)** ⇒ *列表*

rand() 會傳回 0 與 1 之間的隨機值。

rand(*試驗次數***)** 會傳回包含 0 與 1 之間*試驗次數*個隨機值的列表。

設定亂數種子。

RandSeed 1147	*Done*
rand(2)	{0.158206,0.717917}

randBin()

目錄 >

randBin(*n***, *p*)** ⇒ *運算式*
randBin(*n***, *p*, *試驗次數*)** ⇒ *列表*

randBin(*n***, *p*)** 會從指定的二項式分佈傳回隨機實數。

randBin(*n***, *p*, *試驗次數*)** 會從指定的二項分佈，傳回包含*試驗次數*個隨機實數的列表。

randBin(80,0.5)	46.
randBin(80,0.5,3)	{43.,39.,41.}

randInt()

目錄 >

randInt(*下限***,***上限***)** ⇒ *運算式*
randInt(*下限***,***上限***，***試驗次數***)** ⇒ *列表*

randInt(3,10)	3.
randInt(3,10,4)	{9.,3.,4.,7.}

randInt()

目錄 >

randInt(*下限*,*上限***)** 會傳回 *下限*和 *上限*整數邊界指定範圍內的隨機整數。

randInt(*下限*,*上限*, *試驗次數***)** 會傳回指定範圍內包含*試驗次數*個隨機整數的列表。

randMat()

目錄 >

randMat(*列數*, *欄數***)** ⇒ *矩陣*

傳回指定維度的矩陣，其中元素值是介於 -9 至 9 之間的整數。

兩個引數都必須化簡成整數。

RandSeed 1147	*Done*
randMat(3,3)	$\begin{bmatrix} 8 & -3 & 6 \\ -2 & 3 & -6 \\ 0 & 4 & -6 \end{bmatrix}$

注意:每按一次 enter，本矩陣中的數值就會變換。

randNorm()

目錄 >

randNorm(μ, σ) ⇒ *運算式*
randNorm(μ, σ, *試驗次數***)** ⇒ *列表*

randNorm(μ, σ) 會從指定的常態分佈傳回十進位數字。結果可以是任何實數，但會集中在 [μ−3•σ, μ+3•σ] 區間範圍內。

randNorm(μ, σ, *試驗次數***)** 會從指定的常態分佈，傳回包含*試驗次數*個十進位數字的列表。

RandSeed 1147	*Done*
randNorm(0,1)	0.492541
randNorm(3,4.5)	-3.54356

randPoly()

目錄 >

randPoly(*變數*, *階數***)** ⇒ *運算式*

傳回對*變數*的指定*階數*多項式，其係數是介於 −9 至 9 範圍內的隨機整數。首項係數不得為零。

*階數*必須是 0 至 99。

RandSeed 1147	*Done*
randPoly(x,5)	$-2\cdot x^5+3\cdot x^4-6\cdot x^3+4\cdot x-6$

randSamp()

目錄 >

randSamp(*列表*,*試驗次數*[,*不放回抽樣*]) ⇒ *列表*

傳回列表，其中元素包含*列表*中以*試驗次數*為個數所試驗的隨機樣本，且可選擇替換樣本 (*不放回抽樣*=0) 或不替換樣本 (*不放回抽樣*=1)。預設是替換樣本。

Define *list3*={1,2,3,4,5}	*Done*
Define *list4*=randSamp(*list3*,6)	*Done*
list4	{1.,3.,3.,1.,3.,1.}

RandSeed

目錄 >

RandSeed *數字*

若*數字* = 0，則以亂數產生程式的出廠預設值為種子。若*數字* ≠ 0，則用於產生兩個亂數種子，並儲存在系統變數種子1 和種子2 中。

RandSeed 1147	*Done*
rand()	0.158206

real()

目錄 >

real(*值 l***)** ⇒ *值*

傳回引數的實部。

real(*列表 l***)** ⇒ *列表*

傳回所有元素的實部。

real(*矩陣 l***)** ⇒ *矩陣*

傳回所有元素的實部。

real(2+3·*i*)	2
real({1+3·*i*,3,*i*})	{1,3,0}
real([1+3·*i* 3; 2 *i*])	[1 3; 2 0]

▶Rect

目錄 >

向量 **▶Rect**

注意:如果要使用電腦鍵盤插入本運算子，可輸入 `@>Rect`。

以直角座標形式 [x, y, z] 顯示*向量*。向量必須是 2 維或 3 維，而且可以是列向量或行向量。

注意:▶Rect是顯示格式指令，並非轉換函數。本指令只能用於輸入列末尾，而且無法更新 *ans*。

注意:另請參考此處的 **▶Polar** 頁碼: 102.

$\left(\left[3 \quad \angle\frac{\pi}{4} \quad \angle\frac{\pi}{6}\right]\right)$▶Rect	[1.06066 1.06066 2.59808]

▶Rect　　目錄 >

複數值 ▶**Rect**

以直角座標形式 a+bi 顯示*複數值*。*複數值*可以是任何複數形式，但輸入 re$^{i\theta}$ 時會導致度數角模式錯誤。

注意:您必須使用括弧才能輸入 (r∠θ) 極座標。

弧度角模式:

$$\left(4\cdot e^{\frac{\pi}{3}}\right)\blacktriangleright\text{Rect} \qquad 11.3986$$

$$\left(\left(4\angle\frac{\pi}{3}\right)\right)\blacktriangleright\text{Rect} \qquad 2.+3.4641\cdot i$$

梯度角模式:

$$\left(\left(1\angle 100\right)\right)\blacktriangleright\text{Rect} \qquad i$$

度數角模式:

$$\left(\left(4\angle 60\right)\right)\blacktriangleright\text{Rect} \qquad 2.+3.4641\cdot i$$

注意:若要輸入 ∠，請從「目錄」的符號表中選取。

ref()　　目錄 >

ref(*矩陣1*[, *容許值*]) ⇒ *矩陣*

傳回*矩陣1* 的列梯形。

若任何矩陣元素的絕對值小於*容許值*，則亦可將此元素視為零。只有在矩陣中包含浮點數項目，而且不包含尚未賦值的任何符號變數時，才會使用此容許值。其他時候都會忽略*容許值*。

- 若您使用 ctrl enter 或將**自動或近似值**模式設定成近似值，則系統會利用浮點運算法執行計算作業。
- 若指令省略或未使用*容許值*，則預設容許值的計算方式如下:
  5E−14 •max(dim(*矩陣1*)) •rowNorm(*矩陣1*)

請避免在*矩陣1* 中加入未定義的元素，否則可能出現意外結果。

例如若未定義以下運算式中的 *a*，則會出現警告訊息，並顯示以下結果:

$$\text{ref}\left(\begin{bmatrix}-2 & -2 & 0 & -6\\ 1 & -1 & 9 & -9\\ -5 & 2 & 4 & -4\end{bmatrix}\right) \qquad \begin{bmatrix}1 & \frac{-2}{5} & \frac{-4}{5} & \frac{4}{5}\\ 0 & 1 & \frac{4}{7} & \frac{11}{7}\\ 0 & 0 & 1 & \frac{-62}{71}\end{bmatrix}$$

ref()

$$\text{ref}\left(\begin{bmatrix} a & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}\right) \qquad \begin{bmatrix} 1 & \frac{1}{a} & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

之所以出現警告，是因為若 a=0，則 1/a 廣義元素會無效。

若要避免這個情形，可先儲存 a 的值，或用下例所示的 (「|」) 取代值。

$$\text{ref}\left(\begin{bmatrix} a & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}\right)|a=0 \qquad \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}$$

注意:另請參考此處的 **rref()** 頁碼: 122.

RefreshProbeVars

RefreshProbeVars

可讓您透過 TI-Basic 程式存取所有連接感應器探針的感應器數據。

狀態變數值	狀態
狀態變數 =0	正常(程式繼續運作)
狀態變數 =1	Vernier DataQuest™ 應用程式使用數據收集模式。 **注意:**Vernier DataQuest™ 應用程式必須使用儀表模式，才可使此指令運作。
狀態變數 =2	Vernier DataQuest™ 應用程式尚未啟動。
狀態變數 =3	Vernier DataQuest™ 應用程式已啟動，但尚未連接任何探針。

範例

```
Define temp()=
Prgm
© Check if system is ready
RefreshProbeVars status
If status=0 Then
Disp "ready"
For n,1,50
RefreshProbeVars status
temperature:=meter.temperature
Disp "溫度: ",temperature
If temperature>30 Then
Disp "Too hot"
EndIf
© Wait for 1 second between
samples
Wait 1
```

RefreshProbeVars


```
EndFor

Else

Disp "尚未準備好，請稍後再試！"

EndIf

EndPrgm
```

注意:此指令碼可搭配 TI-Innovator™ 分享器使用。

remain()

remain(*值1***,** *值2***)** ⇒ *值*
remain(*列表1***,** *列表2***)** ⇒ *列表*
remain(*矩陣1***,** *矩陣2***)** ⇒ *矩陣*

依照下列恆等式的定義，傳回第一個引數對第二個引數的餘:

remain(x,0) x
remain(x,y) x−y•iPart(x/y)

請注意到 **remain(**−x,y**)** − **remain(**x,y**)** 這個結果。結果如果不是零，就是與第一個引數有相同符號。

注意:另請參考此處的 **mod()** 頁碼: 88.

remain(7,0)	7
remain(7,3)	1
remain(⁻7,3)	⁻1
remain(7,⁻3)	1
remain(⁻7,⁻3)	⁻1
remain({12,⁻14,16},{9,7,⁻5})	{3,0,1}

$$\text{remain}\left(\begin{bmatrix} 9 & -7 \\ 6 & 4 \end{bmatrix},\begin{bmatrix} 4 & 3 \\ 4 & -3 \end{bmatrix}\right) \qquad \begin{bmatrix} 1 & -1 \\ 2 & 1 \end{bmatrix}$$

Request

Request *提示字串*, *變數*[, *顯示旗標* [, *狀態變數*]]

Request *提示字串*, *func*(*引數1*, ...*引數*) [, *顯示旗標* [, *狀態變數*]]

程式設計指令:程式暫停執行並顯示包含*提示字串*訊息的對話方塊，以及讓使用者輸入回覆的方塊。

使用者輸入回覆並按一下 **[確定]** 時，輸入方塊的內容就會指派為變數 *var* 的值。

寫出以下程式:

```
Define request_demo()=Prgm
    Request "半徑:",r
    Disp "面積 = ",pi*r2
EndPrgm
```

執行程式並輸入回覆:

```
request_demo()
```

若使用者按一下 **[取消]**，則程式會繼續進行，且不會接受任何輸入。如果*變數*已完成定義，則程式會使用先前*變數*的值

選用的*顯示旗標*引數可以是任何運算式。

- 若省略*顯示旗標*或其求值是 **1**，則在計算工具歷史記錄中顯示提示訊息和使用者的回覆。
- 若*顯示旗標*的求值是 **0**，則不在歷史記錄中顯示提示訊息和回覆。

選用的*狀態變數*引數可使程式判斷使用者取消對話方塊的方式。請注意，*狀態變數*需要*顯示旗標*引數。

- 如果使用者按一下 **[確定]** 或按 **Enter** 或 **Ctrl+Enter**，則*狀態變數*的值就會設定為 **1**。
- 否則，*狀態變數*的值就會設定為 **0**。

func() 引數可讓程式把使用者的回覆儲存成函數定義。這個語法的功能，就像使用者執行以下指令:

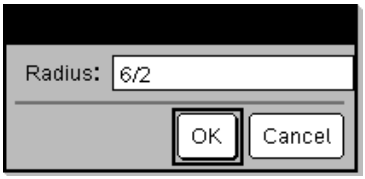
Define *func*(*引數1, ...引數*) = *使用者的回覆*

然後，程式就能使用已定義函數 *func*()。*提示字串*應該引導使用者輸入適當的*使用者的回覆*，以便完成整個函數定義。

注意:您可以在使用者自行定義的程式中使用 Request 指令，但不能用在函數中。

若要停止包含在無限迴圈內 **Request** 指令的程式，請執行下列動作:

- **計算機:** 按住 on 鍵並重複按 · 鍵。
- **Windows®:** 按住 **F12** 鍵並重複按 **Enter** 鍵。
- **Macintosh®:** 按住 **F5** 鍵並重複按 **Enter** 鍵。



選取 **[確定]** 後，計算出結果:

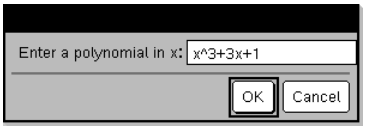
```
半徑: 6/2
面積= 28.2743
```

寫出以下程式:

```
Define polynomial()=Prgm
    Request "輸入一個x的多項式:",p
(x)
    Disp "其實根為:",polyRoots(p
(x),x)
EndPrgm
```

執行程式並輸入回覆:

```
polynomial()
```



輸入 x^3+3x+1 並選取 **[確定]** 後，計算出結果:

```
其實根為:{-0.322185}
```

Request

目錄 > 📖

- **iPad®**：應用程式顯示提示。您可以繼續等待或取消。

注意:另請參考 **此處的**, 頁碼: 117.

RequestStr

目錄 > 📖

RequestStr *提式字串*, *變數*[, *顯示旗標*]

程式設計指令:與 **Request** 指令的第一個語法原理相同，不過本指令一律將使用者的回覆視為字串。相反地，除非使用者把回覆放在引號(“”) 中，否則 **Request** 指令會將回覆解譯成運算式。

注意:您可以在使用者自行定義的程式中使用 **RequestStr** 指令，但不能用在函數中。

若要停止包含在無限迴圈內 **RequestStr** 指令的程式，請執行下列動作:

- **計算機**：按住 on 鍵並重複按 · 鍵。
- **Windows®**：按住 **F12** 鍵並重複按 **Enter** 鍵。
- **Macintosh®**：按住 **F5** 鍵並重複按 **Enter** 鍵。
- **iPad®**：應用程式顯示提示。您可以繼續等待或取消。

注意:另請參考 **此處的**, 頁碼: 115.

寫出以下程式:

```
Define requestStr_demo()=Prgm
    RequestStr “你的姓名:”,name,0
    Disp “回應具有 “,dim(name),” 字元。.”
EndPrgm
```

執行程式並輸入回覆:

```
requestStr_demo()
```

選取 **[確定]** 後計算出結果(請注意，*顯示旗標*引數如果是 **0**，則歷史記錄中會省略提示訊息和回覆) :

```
requestStr_demo()
                    回應具有 5 個字元。
```

Return

目錄 > 📖

Return [*運算式*]

傳回*運算式*作為函數結果。用於 **Func**...**EndFunc** 區段。

注意:若要結束程式，請在 **Prgm**...**EndPrgm** 區段中使用 **Return** 並且不加引數。

輸入 範例的注意事項：關於輸入多行程式和函數定義的說明，請參閱產品手冊中的「計算工具」章節。

```
Define factorial (nn)=
Func
Local answer,counter
1→answer
For counter,1,nn
answer·counter→answer
EndFor
Return answer
EndFunc
```

```
factorial (3)                    6
```

right()

目錄 >

right(*列表1***[, *數字*]) ⇒ *列表***

傳回*列表1*中從右邊算來的*數字*個元素。

若省略*數字*，則傳回整個*列表1*。

```
right({1,3,-2,4},3)                    {3,-2,4}
```

right(*來源字串***[, *數字*]) ⇒ *字串***

傳回*來源字串*中從右邊算來的*數字*個字元。

若省略*數字*，則傳回整個*來源字串*。

```
right("Hello",2)                       "lo"
```

right(*比較***) ⇒ *運算式***

傳回方程式或不等式右邊的部份。

rk23 ()

目錄 >

rk23(*運算式*, *變數*, *因變數*, {*變數0*, *最大變數*}, *因變數0*, *變數步階* [, *diftol*]) ⇒ *矩陣***

rk23(*運算式系統*, *變數*, *因變序列表*, {*變數0*, *最大變數*}, *因變序列表0*, *變數步階*[, *diftol*]) ⇒ *矩陣***

rk23(*運算式列表*, *變數*, *因變序列表*, {*變數0*, *最大變數*}, *因變序列表0*, *變數步階*[, *diftol*]) ⇒ *矩陣***

使用隆巨—庫塔方法求解方程式組

$$\frac{d\,depVar}{d\,Var} = Expr(Var, depVar)$$

其中，*depVar*(*變數0*)=*因變數0*，區間為[*變數0*,*最大變數*]。傳回一個矩陣，其中第一列定義如*變數步階*所定義的*變數*輸出值。第二列會定義對應*變數*值的第一個求解元素的值，依此類推。

位於右邊的*運算式*用於定義常微分方程式 (ODE)。

位於右邊的*運算式系統*，用於定義 ODE 方程式組 (對應於*因變數序列表*中的因變數順序) 。

位於右邊的*運算式列表*，用於定義 ODE 方程式組 (對應於*因變數序列表*中的因變數順序) 。

微分方程式:

y'=0.001*y*(100-y) 與 y(0)=10

```
rk23(0.001·y·(100−y),t,y,{0,100},10,1)
[0.    1.        2.        3.        4.
 10.   10.9367   11.9493   13.042    14.2
```

若要看到完整結果，請按 ▲，然後使用 ◀ 與 ▶ 移動游標。

與 *diftol* 設為 1.E−6 時相同的方程式

```
rk23(0.001·y·(100−y),t,y,{0,100},10,1,1.E-6)
[0.    1.        2.        3.        4.
 10.   10.9367   11.9495   13.0423   14.2189
```

方程式組:

$$\begin{cases} y1' = -y1 + 0.1 \cdot y1 \cdot y2 \\ y2' = 3 \cdot y2 - y1 \cdot y2 \end{cases}$$

其中，*y1*(0)=2 及 *y2*(0)=5

```
rk23({-y1+0.1·y1·y2, 3·y2−y1·y2},t,{y1,y2},{0,5},{2,5},1)
[0.    1.        2.        3.        4.
 2.    1.94103   4.78694   3.25253   1.82848
 5.    16.8311   12.3133   3.51112   6.27245
```

rk23 ()

目錄 > 📖

*變數*是自變數。

*因變數序列表*是顯示因變數的列表。

{*變數0, 最大變數*} 是一個雙元素列表，表示函數從*變數0* 積分至*最大變數*。

因變數序列表0 是因變數初始值的列表。

如果*變數步階*計算出非零數字:sign(*變數步階*) = sign(*最大變數-變數0*)，且傳回 *變數0*+i**變數步階*對所有的 i=0,1,2,... 的解，使 *變數0*+i**變數步階*位於 [*變數0,最大變數*](在*最大變數*中可能沒有解) 。

如果*變數步階*計算出零，會傳回在「隆巨—庫塔」之*變數*值的解。

diftol 是容錯(預設值為 0.001) 。

root()

目錄 > 📖

root(*值***) ⇒** *根*
root(*值 1, 值 2***) ⇒** *根*

root(*值*) 會傳回*值*的平方根。

root(*值 1, 值 2***)** 會傳回*值 1* 的*值 2* 次方根。*值 1* 可以是實數浮點常數、複數浮點常數、整數有理常數或複數有理常數。

注意:另請參考此處的 **N 次方根範本** 頁碼: 1.

$\sqrt[3]{8}$	2
$\sqrt[3]{3}$	1.44225

rotate()

目錄 > 📖

rotate(*整數 1***[,***互換位數***]) ⇒** *整數*

互換二進位整數中的位元。您可以用任何數基輸入*整數 1*;系統會自動轉換成有正負號的 64 位元二進位形式。若*整數 1* 太大超出該格式範圍，則系統會以對稱模數運算法使其落入範圍。如需詳細資訊，請參考此處的 **▶Base2** 頁碼: 16.

二進位基底模式:

rotate(0b1111111111111111111111111111111)	0b10000000000000000000000000000001
rotate(256,1)	0b1000000000

若要看到完整結果，請按 ▲，然後使用 ◀ 與 ▶ 移動游標。

rotate()

若*互換位數*是正值，則向左輪替。若*互換位數*是負值，則向右輪替。預設值是 −1(向右移一位) 。

例如向右互換時:

每個位元向右互換。

0b00000000000000111101011000011010 1

最右邊的位元移到最左邊。

求出:

0b10000000000000011110101100001101 0

所顯示的結果會以基底模式為準。

十六進位基底模式:

rotate(0h78E)	0h3C7
rotate(0h78E,-2)	0h80000000000001E3
rotate(0h78E,2)	0h1E38

重要資訊:若要輸入二進位或十六進位數字，前面一定要有 0b 或 0h 字首(數字零，而非英文字母 O) 。

rotate(*列表 1*[,*互換位數*]**)** ⇒ *列表*

傳回*列表 1* 左右移動*互換位數*個元素後的副本。請勿更改*列表 1*。

若*互換位數*是正值，則向左輪替。若*互換位數*是負值，則向右輪替。預設值是 −1(向右互換一個元素) 。

十進位基底模式:

rotate({1,2,3,4})	{4,1,2,3}
rotate({1,2,3,4},-2)	{3,4,1,2}
rotate({1,2,3,4},1)	{2,3,4,1}

rotate(*字串 1*[,*互換位數*]**)** ⇒ *字串*

傳回*字串 1* 左右移動*互換位數*個字元後的副本。請勿更改*字串 1*。

若*互換位數*是正值，則向左輪替。若*互換位數*是負值，則向右輪替。預設值是 −1(向右互換一個字元) 。

rotate("abcd")	"dabc"
rotate("abcd",-2)	"cdab"
rotate("abcd",1)	"bcda"

round()

round(*值 1*[, *位數*]**)** ⇒ *值*

傳回四捨五入到小數點後指定位數的引數。

*位數*必須是在 0 至 12 範圍內的整數。若省略*位數*，則傳回四捨五入到 12 個有效位數的引數。

注意:顯示位數模式可能影響顯示方式。

round(1.234567,3)	1.235

round(*列表 1*[, *位數*]**)** ⇒ *列表*

round({π,√2,ln(2)},4)	{3.1416,1.4142,0.6931}

round()

目錄 >

傳回其中元素四捨五入到指定位數的列表。

round(*矩陣1***[, *位數*]) ⇒** *矩陣*

傳回其中元素四捨五入到指定位數的矩陣。

$$\text{round}\left(\begin{bmatrix}\ln(5) & \ln(3)\\ \pi & e^1\end{bmatrix},1\right) \qquad \begin{bmatrix}1.6 & 1.1\\ 3.1 & 2.7\end{bmatrix}$$

rowAdd()

目錄 >

rowAdd(*矩陣1***, *rIndex1*, *rIndex2*) ⇒** *矩陣*

傳回*矩陣1*的副本，其中 *rIndex1* 與 *rIndex2* 兩列的總和取代 *rIndex2* 列。

$$\text{rowAdd}\left(\begin{bmatrix}3 & 4\\ -3 & -2\end{bmatrix},1,2\right) \qquad \begin{bmatrix}3 & 4\\ 0 & 2\end{bmatrix}$$

rowDim()

目錄 >

rowDim(*矩陣***) ⇒** *運算式*

傳回*矩陣*列數。

注意:另請參考此處的 **colDim()** 頁碼: 22.

$$\begin{bmatrix}1 & 2\\ 3 & 4\\ 5 & 6\end{bmatrix}\rightarrow m1 \qquad \begin{bmatrix}1 & 2\\ 3 & 4\\ 5 & 6\end{bmatrix}$$

$$\text{rowDim}(m1) \qquad 3$$

rowNorm()

目錄 >

rowNorm(*矩陣***) ⇒** *運算式*

傳回*矩陣*中各列元素絕對值的總和最大值。

注意:矩陣的所有元素都必須化簡成數字。另請參考此處的 **colNorm()** 頁碼: 22.

$$\text{rowNorm}\left(\begin{bmatrix}-5 & 6 & -7\\ 3 & 4 & 9\\ 9 & -9 & -7\end{bmatrix}\right) \qquad 25$$

rowSwap()

目錄 >

rowSwap(*矩陣1***, *rIndex1*, *rIndex2*) ⇒** *矩陣*

傳回 *rIndex1* 與 *rIndex2* 兩列互換的*矩陣1*。

$$\begin{bmatrix}1 & 2\\ 3 & 4\\ 5 & 6\end{bmatrix}\rightarrow mat \qquad \begin{bmatrix}1 & 2\\ 3 & 4\\ 5 & 6\end{bmatrix}$$

$$\text{rowSwap}(mat,1,3) \qquad \begin{bmatrix}5 & 6\\ 3 & 4\\ 1 & 2\end{bmatrix}$$

rref()

目錄 >

rref(*矩陣1***[, ***容許值***]) ⇒ ***矩陣***

傳回*矩陣1*的列簡化梯形矩陣。

$$\text{rref}\left(\begin{bmatrix} -2 & -2 & 0 & -6 \\ 1 & -1 & 9 & -9 \\ -5 & 2 & 4 & -4 \end{bmatrix}\right) \qquad \begin{bmatrix} 1 & 0 & 0 & \frac{66}{71} \\ 0 & 1 & 0 & \frac{147}{71} \\ 0 & 0 & 1 & \frac{-62}{71} \end{bmatrix}$$

若任何矩陣元素的絕對值小於*容許值*，則亦可將此元素視為零。只有在矩陣中包含浮點數項目，而且不包含尚未賦值的任何符號變數時，才會使用此容許值。其他時候都會忽略*容許值*。

- 若您使用 ctrl enter 或將**自動或近似值**模式設定成近似值，則系統會利用浮點運算法執行計算作業。
- 若指令省略或未使用*容許值*，則預設容許值的計算方式如下：
  5E−14 •max(dim(*矩陣1*)) •rowNorm(*矩陣1*)

注意：另請參考此處的 **ref()** 頁碼：113.

S

sec()

trig 鍵

sec(*值1***) ⇒ ***值***

sec(*列表1***) ⇒ ***列表***

傳回*值1*的正割值，或傳回包含*列表1*中各元素之正割的列表。

附註：系統會根據目前的角度模式設定，將引數解譯為度數角、梯度角或弧度角。您可使用 °、G 或 r 來暫時覆寫角度模式。

度數角模式：

sec(45)	1.41421
sec({1,2.3,4})	{1.00015,1.00081,1.00244}

sec⁻¹()

trig 鍵

sec⁻¹(*值1***) ⇒ ***值***

sec⁻¹(*列表1***) ⇒ ***列表***

傳回正割值是*值1*的角度，或傳回包含*列表1*中各元素之反正割值的列表。

度數角模式：

$\sec^{-1}(1)$	0.

sec⁻¹() 鍵

附註：系統會根據目前的角度模式設定，將結果傳回為度數角、梯度角或弧度角。

附註：如果要從鍵盤插入本函數，可輸入 `arcsec(...)`。

梯度角模式：

$\sec^{-1}(\sqrt{2})$ 50.

弧度角模式：

$\sec^{-1}(\{1,2,5\})$ $\{0,1.0472,1.36944\}$

sech() 目錄 >

sech(*值 1***)** ⇒ *值*

sech(*列表 1***)** ⇒ *列表*

傳回*值 1* 的雙曲正割值，或傳回包含*列表 1* 各元素之雙曲正割值的列表。

$\text{sech}(3)$ 0.099328

$\text{sech}(\{1,2.3,4\})$ $\{0.648054,0.198522,0.036619\}$

sech⁻¹() 目錄 >

sech⁻¹(*值 1***)** ⇒ *值*

sec⁻¹(*列表 1***)** ⇒ *列表*

傳回*值 1* 的反雙曲正割值，或傳回包含*列表 1* 中各元素之反雙曲正割值的列表。

附註：如果要從鍵盤插入本函數，可輸入 `arcsech(...)`。

弧度角與直角複數模式：

$\text{sech}^{-1}(1)$ 0

$\text{sech}^{-1}(\{1,-2,2.1\})$ $\{0,2.0944\cdot i,8.E-15+1.07448\cdot i\}$

Send 分享器功能表

Send *exprOrString1*[, *exprOrString2*] ...

程式設計指令:傳送一個或多個 TI-Innovator™ Hub 指令到已連接的分享器。

exprOrString 必須是有效的 TI-Innovator™ Hub 指令。*exprOrString* 通常包含 **"SET ..."** 指令以控制裝置，或包含 **"READ ..."** 指令以要求資料。

引數會連續傳送到分享器。

例如:將內建 RGB LED 的藍色元素開啟 0.5 秒。

```
Send "SET COLOR.BLUE ON TIME .5"
                                Done
```

例如:要求分享器內建亮度等級感應器目前的值。**Get** 指令會擷取數值，並將數值指定給變數 *lightval*。

Send 分享器功能表

附註:您可以在使用者自行定義的程式中使用 **Send** 指令，但無法在函數中使用。

附註:另請參考 **Get** (頁碼: 55)、**GetStr** (頁碼: 61) 和 **eval()** (頁碼: 45).

Send "READ BRIGHTNESS"	*Done*
Get *lightval*	*Done*
lightval	0.347922

例如:將完成計算的頻率傳送到分享器內建的喇叭。使用特殊變數 *iostr.SendAns* 可顯示分享器指令，並將求得的運算式包含在內。

n:=50	50
m:=4	4
Send "SET SOUND eval(m·n)"	*Done*
iostr.SendAns	"SET SOUND 200"

seq() 目錄 >

seq(*Expr***,** *Var***,** *低***,** *高***[,** *步階***])⇒***列表*

從 Low 到 High 增加 Var 值 (以 Step 為增量級距)，對 Expr 求值，並以列表將結果傳回。**seq() 執行完成後，Var 的原始內容仍然存在。**

Step 的預設值 = 1。

$\text{seq}\left(n^2,n,1,6\right)$ $\{1,4,9,16,25,36\}$

$\text{seq}\left(\frac{1}{n},n,1,10,2\right)$ $\left\{1,\frac{1}{3},\frac{1}{5},\frac{1}{7},\frac{1}{9}\right\}$

$\text{sum}\left(\text{seq}\left(\frac{1}{n^2},n,1,10,1\right)\right)$ $\frac{1968329}{1270080}$

附註: 強迫表示結果近似值，

計算機: 按 ctrl enter。
Windows®: 按 **Ctrl+Enter**。
Macintosh®: 按 ⌘+**Enter**。
iPad®: 按住 **enter** 然後選擇 ≈。

$\text{sum}\left(\text{seq}\left(\frac{1}{n^2},n,1,10,1\right)\right)$ 1.54977

seqGen() 目錄 >

seqGen(*Expr***,** *Var***,** *depVar***, {***Var0***,** *VarMax***}[,** *ListOfInitTerms* **[,** *VarStep* **[,** *CeilingValue***]]]) ⇒***列表*

產生序列 $u(n) = u(n\text{-}1)^2/2$ 的前 5 項，其中 $u(1)=2$ 且 $VarStep=1$。

seqGen()

目錄 >

針對序列 *depVar*(*Var*)=*Expr* 產生項列表，如下所示：從 *Var0* 到 *VarMax* 增加自變數 *Var* 的值 (以 *VarStep* 為增量級距)，使用 Expr 公式及 ListOfInitTerms 求 depVar(Var)對應於 Var 的值，並將結果傳回為列表。

$$\text{seqGen}\left(\frac{(u(n-1))^2}{n},n,u,\{1,5\},\{2\}\right)$$

$$\left\{2,2,\frac{4}{3},\frac{4}{9},\frac{16}{405}\right\}$$

seqGen(*ListOrSystemOfExpr*, *Var*, *ListOfDepVars*, {*Var0*, *VarMax*} [, *MatrixOfInitTerms* [, *VarStep* [, *CeilingValue*]]]) ⇒*矩陣*

針對序列 *ListOfDepVars* (*Var*)=*ListOrSystemOfExpr* 產生方程組 (或列表) 的項矩陣，如下所示：從 Var0 到 VarMax 增加自變數 *Var* 的值 (以 *VarStep* 為增量級距)，使用 *ListOrSystemOfExpr* 公式及 *MatrixOfInitTerms* 求 *ListOfDepVars (Var)*對應於 *Var* 的值，並將結果傳回為矩陣。

Var0=2 的範例：

$$\text{seqGen}\left(\frac{u(n-1)+1}{n},n,u,\{2,5\},\{3\}\right)$$

$$\left\{3,\frac{4}{3},\frac{7}{12},\frac{19}{60}\right\}$$

seqGen() 執行完成後，*Var* 的原始內容不會更改。

VarStep 的預設值 = 1。

兩個序列的方程組：

$$\text{seqGen}\left(\left\{\frac{1}{n},\frac{u2(n-1)}{2}+u1(n-1)\right\},n,\{u1,u2\},\{1,5\},\left[\begin{matrix}_\\2\end{matrix}\right]\right)$$

$$\begin{bmatrix}1 & \frac{1}{2} & \frac{1}{3} & \frac{1}{4} & \frac{1}{5}\\ 2 & 2 & \frac{3}{2} & \frac{13}{12} & \frac{19}{24}\end{bmatrix}$$

附註：以上初始項矩陣中的 Void (_) 是用來指明 u1(n) 的初始項是使用顯序列公式 u1(n)=1/n 所計算的。

seqn()

目錄 >

seqn(*Expr*(*u*, *n* [, *ListOfInitTerms*[, *nMax* [, *CeilingValue*]]])⇒*列表*

針對序列 u(n)=Expr(u, n) 產生項列表，如下所示：從 1 到 *nMax* 增加 n 的值 (以 1 為增量級距)，使用 Expr(u, n) 公式及 *ListOfInitTerms* 對 n 的對應值求 u(n) 的值，並將結果傳回為列表。

產生序列 u(n) = u(n-1)/2 的前 6 項，其中 *u*(1)=**2**。

$$\text{seqn}\left(\frac{u(n-1)}{n},\{2\},6\right)$$

$$\left\{2,1,\frac{1}{3},\frac{1}{12},\frac{1}{60},\frac{1}{360}\right\}$$

seqn(*Expr*(*n* [, *nMax* [, *CeilingValue*]])⇒*列表*

針對非遞迴序列 u(n)=Expr(n) 產生項列表，如下所示：從 1 到 *nMax* 增加 *n* 的值 (以 1 為增量級距)，使用 *Expr(n)* 公式求 u(n) 對應於 n 的值，並將結果傳回為列表。

$$\text{seqn}\left(\frac{1}{n^2},6\right) \qquad \left\{1,\frac{1}{4},\frac{1}{9},\frac{1}{16},\frac{1}{25},\frac{1}{36}\right\}$$

seqn()

如果沒有 *nMax*，設定 *nMax* 為 2500

如果 *nMax*=0，則設定 *nMax* 為 2500

附註：seqn() 會呼叫 **seqGen( )**，其中 *n0*=**1** 且 *nstep* =**1**

setMode()

setMode(*模式名稱整數***,** *設定整數***)** ⇒ *整數*

setMode(*列表***)** ⇒ *整數列表*

只在函數或程式中才有效。

setMode(*模式名稱整數*, *設定整數*) 可暫時將*模式名稱整數*的模式設成*設定整數*這個新設定，並傳回該模式原始設定的對應整數。這項變動受限於程式/函數的執行時間。

*模式名稱整數*用於指定您要的模式，必須設定成下表列出的模式整數。

*設定整數*用於指定該模式的新設定，必須是下列欲設定模式的其中一個設定整數。

setMode(*列表***)** 可讓您更改多項設定。*列表*內容是成對的模式整數和設定整數。**setMode(***列表***)**：傳回類似的表，但內容是代表原始模式和設定的成對整數。

若您以 **getMode(0)** → *var* 儲存所有的模式設定，則可用 **setMode(***var***)** 還原這些設定，直到函數或程式存在為止。請參考 **getMode()**，頁碼：60。

附註：系統會將目前的模式設定，傳遞給所呼叫的副程式。若任何副程式更動了模式設定，則控制權回到發出呼叫的程式時，所更改的模式就會消失。

輸入範例的注意事項：關於輸入多行程式和函數定義的說明，請參閱產品手冊中的「計算工具」章節。

利用「顯示位數」的預設值，顯示 π 的近似值，然後以 Fix2 的設定顯示 π。執行程式後，請檢查預設值是否還原。

```
Define prog1()=Prgm                    Done
               Disp π
               setMode(1,16)
               Disp π
               EndPrgm
prog1()
                                   3.14159
                                   3.14
                                    Done
```

模式 名稱	模式整數	設定整數
顯示位數	1	**1**=浮點數、**2**=浮點數1、**3**=浮點數2、**4**=浮點數3、**5**=浮點數4、**6**=浮點數5、**7**=浮點數6、**8**=浮點數7、**9**=浮點數8、**10**=浮點數9、**11**=浮點數10、**12**=浮點數11、**13**=浮點數12、**14**=固定0、**15**=固定1、**16**=固定2、**17**=固定3、**18**=固定4、**19**=固定5、**20**=固定6、**21**=固定7、**22**=固定8、**23**=固定9、**24**=固定10、**25**=固定11、**26**=固定12
角度	2	**1**=弧度角、**2**=度數角、**3**=梯度角
指數格式	3	**1**=正常、**2**=科學、**3**=工程
實數或複數	4	**1**=實數、**2**=直角座標、**3**=極座標
自動或近似值	5	**1**=自動、**2**=近似值
向量格式	6	**1**=直角座標、**2**=圓柱座標、**3**=球面座標
基底	7	**1**=十進位、**2**=十六進位、**3**=二進位

shift()

目錄 >

shift(*整數1*[,*移位位數*])⇒*整數*

移動二進位整數中的位元。您可以用任何數基輸入*整數1*；系統會自動轉換成有正負號的 64 位元二進位形式。若*整數1* 太大，超出這種格式的範圍，系統會以對稱模數運算法使其落入範圍。如需詳細資訊，請參閱 ▶**Base2**，頁碼：16。

若*移位位數*是正值，則向左移位。若*移位位數*是負值，則向右移位。預設值是 -1(向右移一位) 。

向右移位時會捨棄最右邊的位元並補上 0 或 1，以符合最左邊的位元。向左移位時會捨棄最左邊的位元，並補上 0 當作最右邊的位元。

例如向右移位時：

每個位元向右移位。

0b00000000000000111101011000011010

若最左邊的位元是 0，則補上 0；

若最左邊的位元是 1，則補上 1。

二進位基底模式：

shift(0b1111010110000110101)	0b1111010110000011010
shift(256,1)	0b1000000000

十六進位基底模式：

shift(0h78E)	0h3C7
shift(0h78E,-2)	0h1E3
shift(0h78E,2)	0h1E38

重要訊息：若要輸入二進位或十六進位數字，前面一定要有 0b 或 0h 字首 (數字零，而非英文字母 O) 。

shift()

目錄 >

求出：

0b00000000000000011110101100001101 0

所顯示的結果會以基底模式為準。系統不會顯示最前面的零。

shift(*列表 1* [**,***移位位數*]**)⇒***列表*

傳回*列表 1* 左右移動*移位位數*個元素後的副本。請勿更改*列表 1*。

若*移位位數*是正值，則向左移位。若*移位位數*是負值，則向右移位。預設值是 -1(向右移動一個元素) 。

因移位而在*列表*開頭或結尾加入的元素，系統會將其設定為「undef」符號。

十進位基底模式：

shift({1,2,3,4})	{undef,1,2,3}
shift({1,2,3,4},-2)	{undef,undef,1,2}
shift({1,2,3,4},2)	{3,4,undef,undef}

shift(*字串 1* [**,***移位位數*]**)⇒***字串*

傳回*字串 1* 左右移動*移位位數*個字元後的副本。請勿更改*字串 1*。

若*移位位數*是正值，則向左移位。若*移位位數*是負值，則向右移位。預設值是 -1(向右移動一個字元) 。

因移位而在*字串*開頭或結尾加入的字元，系統會將其設定為空格。

shift("abcd")	" abc"
shift("abcd",-2)	" ab"
shift("abcd",1)	"bcd "

sign()

目錄 >

sign(*值 1***)⇒***值*

sign(*列表 1***)⇒***列表*

sign(*矩陣 1***)⇒***矩陣*

若是實數或複數*值 1*，則*值 1*≠ 0 時傳回*值 1* / **abs(***值 1***)**。

若*值 1* 是正值，則傳回 1。

若*值 1* 是負值，則傳回 -1。

sign(0) 傳回 ±1 的前提是複數格式模式為實數，否則會傳回自己的值。

sign(0) 代表複數值域中的單位圓。

若是列表或矩陣，則傳回各元素的正負號。

sign(-3.2)	-1
sign({2,3,4,-5})	{1,1,1,-1}

若複數格式模式是實數：

sign([-3 0 3])	[-1 undef 1]

simult(*coeffMatrix***, *constVector*[, *Tol*])⇒** *矩陣*

傳回行向量，內容包含線性聯立方程式的解。

附註：另請參考 **linSolve()**，頁碼：76。

coeffMatrix 必須是包含方程式係數的方陣。

constVector 的列數必須和 *coeffMatrix* 相同(維數相同)，而且包含常數。

若任何矩陣元素的絕對值小於 *Tol*(容許值)，則亦可將此元素視為零。只有在矩陣中包含浮點數項目，而且不包含尚未賦值的任何符號變數時，才會使用此容許值。其他時候都會忽略 *Tol* (容許值) 。

- 若您將**自動或近似值**模式設定成近似值，則系統會利用浮點運算法執行計算作業。
- 若指令省略或未使用 *Tol*，則預設容許值的計算方式如下：
  5E⁻14 ·max(dim(*coeffMatrix*))
  ·rowNorm(*coeffMatrix*)

求 x、y 的值：

x + 2y = 1

3x + 4y = ⁻1

$$\text{simult}\left(\begin{bmatrix}1 & 2\\3 & 4\end{bmatrix},\begin{bmatrix}1\\-1\end{bmatrix}\right) \qquad \begin{bmatrix}-3\\2\end{bmatrix}$$

解答是 x=⁻3、y=2。

simult(*coeffMatrix***, *constMatrix*[, *Tol*])⇒** *矩陣*

求解多個線性聯立方程式，每個聯立方程式的方程式係數都相同，但常數不同。

constMatrix 各行必須包含聯立方程式的常數。所求得矩陣中的各行，都包含對應聯立方程式的解。

求解：

ax + by = 1

cx + dy = 2

$$\begin{bmatrix}1 & 2\\3 & 4\end{bmatrix}\rightarrow matx1 \qquad \begin{bmatrix}1 & 2\\3 & 4\end{bmatrix}$$

$$\text{simult}\left(matx1,\begin{bmatrix}1\\2\end{bmatrix}\right) \qquad \begin{bmatrix}0\\\frac{1}{2}\end{bmatrix}$$

求解：

x + 2y = 1

3x + 4y = ⁻1

x + 2y = 2

3x + 4y = ⁻3

$$\text{simult}\left(\begin{bmatrix}1 & 2\\3 & 4\end{bmatrix},\begin{bmatrix}1 & 2\\-1 & -3\end{bmatrix}\right) \qquad \begin{bmatrix}-3 & -7\\2 & \frac{9}{2}\end{bmatrix}$$

第一個聯立方程式是 x=⁻3、y=2。第二個聯立方程式是 x=⁻7、y=9/2。

sin()

trig 鍵

sin(*值 1***)⇒***值*

度數角模式：

sin()

trig 鍵

sin(*列表 l***)⇒***列表*

sin(*值 l***):** 傳回引數的正弦值。

sin(*列表 l***):** 傳回*列表 l* 中各元素之正弦值的列表。

附註：系統會根據目前的角度模式，將引數解譯為度數角、梯度角或弧度角。您可使用 °、G 或 r 來暫時覆寫角度模式設定。

$\sin\left(\left(\frac{\pi}{4}\right)r\right)$	0.707107
$\sin(45)$	0.707107
$\sin(\{0,60,90\})$	$\{0.,0.866025,1.\}$

梯度角模式：

$\sin(50)$	0.707107

弧度角模式：

$\sin\left(\frac{\pi}{4}\right)$	0.707107
$\sin(45°)$	0.707107

sin(*方陣 l***)⇒***方陣*

傳回*方陣 l* 的矩陣正弦。這和計算各元素的正弦不同。若要了解計算方式，請參考 **cos()**。

方陣 l 必需可對角化。結果一定會包含浮點數。

弧度角模式：

$$\sin\left(\begin{bmatrix}1 & 5 & 3\\ 4 & 2 & 1\\ 6 & -2 & 1\end{bmatrix}\right)$$

$$\begin{bmatrix}0.9424 & -0.04542 & -0.031999\\ -0.045492 & 0.949254 & -0.020274\\ -0.048739 & -0.00523 & 0.961051\end{bmatrix}$$

sin⁻¹()

trig 鍵

sin⁻¹(*值 l***)⇒***值*

sin⁻¹(*列表 l***)⇒***列表*

sin⁻¹(*值 l***):** 傳回正弦值是值 1 的角度。

sin⁻¹(*列表 l***):** 傳回*列表 l* 各元素之反正弦值的列表。

附註：系統會根據目前的角度模式設定，將結果傳回為度數角、梯度角或弧度角。

附註：如果要從鍵盤插入本函數，可輸入 `arcsin(…)`。

sin⁻¹(*方陣 l***)⇒***方陣*

度數角模式：

$\sin^{-1}(1)$	90.

梯度角模式：

$\sin^{-1}(1)$	100.

弧度角模式：

$\sin^{-1}(\{0,0.2,0.5\})$	$\{0.,0.201358,0.523599\}$

弧度角模式與直角複數格式模式：

$\sin^{-1}()$

trig 鍵

傳回*方陣1*的矩陣反正弦。這和計算各元素的反正弦不同。若要了解計算方式，請參考 **cos()**。

*方陣1*必需可對角化。結果一定會包含浮點數。

$$\sin^{-1}\left(\begin{bmatrix}1 & 5\\ 4 & 2\end{bmatrix}\right)$$

$$\begin{bmatrix}-0.174533-0.12198\cdot i & 1.74533-2.35591\cdot i\\ 1.39626-1.88473\cdot i & 0.174533-0.593162\cdot i\end{bmatrix}$$

sinh()

目錄 >

sinh(*值1***)⇒***值*

sinh(*列表1***)⇒***列表*

sinh (*值1***):** 傳回引數的雙曲正弦值。

sinh (*列表1***):** 傳回*列表1*各元素之雙曲正弦值的列表。

$$\sinh(1.2) \qquad 1.50946$$

$$\sinh(\{0,1.2,3.\}) \qquad \{0,1.50946,10.0179\}$$

sinh (*方陣1***)⇒***方陣*

傳回*方陣1*的矩陣雙曲正弦。這和計算各元素的雙曲正弦不同。若要了解計算方式，請參考 **cos()**。

*方陣1*必需可對角化。結果一定會包含浮點數。

弧度角模式：

$$\sinh\left(\begin{bmatrix}1 & 5 & 3\\ 4 & 2 & 1\\ 6 & -2 & 1\end{bmatrix}\right)$$

$$\begin{bmatrix}360.954 & 305.708 & 239.604\\ 352.912 & 233.495 & 193.564\\ 298.632 & 154.599 & 140.251\end{bmatrix}$$

$\sinh^{-1}()$

目錄 >

sinh⁻¹(*值1***)⇒***值*

sinh⁻¹(*列表1***)⇒***列表*

sinh⁻¹(*值1***):** 傳回引數的反雙曲正弦值。

sinh⁻¹(*列表1***):** 傳回*列表1*各元素之反雙曲正弦值的列表。

附註：如果要從鍵盤插入本函數，可輸入 `arcsinh(…)`。

$$\sinh^{-1}(0) \qquad 0$$

$$\sinh^{-1}(\{0,2.1,3\}) \qquad \{0,1.48748,1.81845\}$$

sinh⁻¹(*方陣1***)⇒***方陣*

傳回*方陣1*的矩陣反雙曲正弦。這和計算各元素的反雙曲正弦不同。若要了解計算方式，請參考 **cos()**。

*方陣1*必需可對角化。結果一定會包含浮點數。

弧度角模式：

$$\sinh^{-1}\left(\begin{bmatrix}1 & 5 & 3\\ 4 & 2 & 1\\ 6 & -2 & 1\end{bmatrix}\right)$$

$$\begin{bmatrix}0.041751 & 2.15557 & 1.1582\\ 1.46382 & 0.926568 & 0.112557\\ 2.75079 & -1.5283 & 0.57268\end{bmatrix}$$

SinReg *X*, *Y* [, [*迭代*] ,[*週期*] [, *類別*, *包含*]]

計算 *X* 列表與 *Y* 列表的正弦迴歸。*stat.results* 變數會儲存結果摘要(請參閱第 頁碼: 134 頁) 。

所有列表的維數都必須相同，*包含*除外。

X、*Y* 是自變數和因變數列表。

*迭代*值用於指定求解次數的上限(1 至 16) 。若省略，則會採用 8。通常值越大精度越高，但執行時間越久，反之亦然。

*週期*用於指定預估週期。若省略，則 *X* 中各值的差應該相等，而且依照順序。若您指定*週期*，則 x 中各值的差應該不同。

*類別*是 *X*、*Y* 對應資料的數字或字串類別代號列表。

*包含*是一個以上類別代號的列表。只有本列表中類別代號所對應的資料項目，才會包含在計算作業中。

無論使用何種角度模式設定，**SinReg** 輸出都一定是採用弧度角形式。

如要了解列表中有空元素時的影響，請參考°ß空元素°®，頁碼: 198。

輸出變數	說明
stat.RegEqn	迴歸方程式: a ·sin(bx+c)+d
stat.a、stat.b、stat.c、stat.d	迴歸係數
stat.Resid	迴歸殘差
stat.XReg	實際用於迴歸的已修改 *X 列表*中的資料點列表(根據 *Freq*、*類別列表*、*包含類別的限制)*
stat.YReg	實際用於迴歸的已修改 *Y 列表*中的資料點列表(根據 *Freq*、*類別列表*、*包含類別的限制)*
stat.FreqReg	*stat.XReg* 與 *stat.YReg 對應的頻率列表*

SortA

目錄 >

SortA *列表1*[, *列表2*] [, *列表3*] ...

SortA *向量1*[, *向量2*] [, *向量3*] ...

以升冪順序排列第一個引數的元素。

若您加入其他引數，則各引數中的元素會重新排列，以便對應第一個引數中各元素的新位置。

所有引數都必須是列表或向量名稱，而且維數必須相同。

系統會把第一個引數中的空元素移到末尾。如需空元素的詳細資訊，請參考第 頁碼：198 頁。

$\{2,1,4,3\} \rightarrow list1$	$\{2,1,4,3\}$
SortA *list1*	*Done*
list1	$\{1,2,3,4\}$
$\{4,3,2,1\} \rightarrow list2$	$\{4,3,2,1\}$
SortA *list2*,*list1*	*Done*
list2	$\{1,2,3,4\}$
list1	$\{4,3,2,1\}$

SortD

目錄 >

SortD *列表1*[, *列表2*] [, *列表3*] ...

SortD *向量1*[, *向量2*] [,*向量3*] ...

與 **SortA** 相同，但 **SortD** 會以降冪順序排列各元素。

系統會把第一個引數中的空元素移到末尾。如需空元素的詳細資訊，請參考第 頁碼：198 頁。

$\{2,1,4,3\} \rightarrow list1$	$\{2,1,4,3\}$
$\{1,2,3,4\} \rightarrow list2$	$\{1,2,3,4\}$
SortD *list1*,*list2*	*Done*
list1	$\{4,3,2,1\}$
list2	$\{3,4,1,2\}$

▸Sphere

目錄 >

向量 **▸Sphere**

附註：如果要從電腦鍵盤插入本運算子，可輸入 **@>Sphere**。

以球面座標形式 [ρ ∠θ ∠ϕ] 顯示列向量或行向量。

*向量*必須是 3 維，而且可以是列向量也可以是行向量。

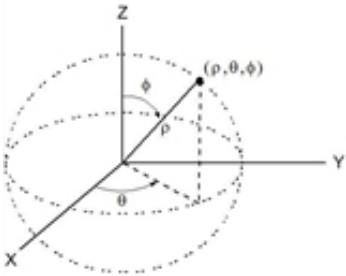
附註：▸Sphere 是顯示格式指令，並非轉換函數。本指令只能用於輸入線的末尾。

$$[1 \quad 2 \quad 3] \blacktriangleright \text{Sphere}$$
$$[3.74166 \quad \angle 1.10715 \quad \angle 0.640522]$$

$$\left(\begin{bmatrix} 2 & \angle\frac{\pi}{4} & 3 \end{bmatrix}\right) \blacktriangleright \text{Sphere}$$
$$[3.60555 \quad \angle 0.785398 \quad \angle 0.588003]$$

▸Sphere

目錄 > 📖



sqrt()

目錄 > 📖

sqrt(*值 l***)⇒***值*

sqrt(*列表 l***)⇒***列表*

傳回引數的平方根。

若是列表，則傳回*列表 l* 中各元素的平方根。

附註：另請參考**平方根範本**，頁碼：1。

$\sqrt{4}$	2
$\sqrt{\{9,2,4\}}$	$\{3,1.41421,2\}$

stat.results

目錄 > 📖

stat.results

顯示統計計算結果。

系統會將結果顯示為一組成對的「名稱/值」。所顯示的特定名稱，要看最近求值的統計函數或指令而定。

您可複製名稱或值，然後貼到其他位置。

附註：定義變數時，請避免讓名稱和統計分析變數相同，否則有時可能出現錯誤。下表列出統計分析時所用的變數名稱。

xlist:={1,2,3,4,5}	{1,2,3,4,5}
ylist:={4,8,11,14,17}	{4,8,11,14,17}

LinRegMx *xlist*,*ylist*,1: *stat.results*

"Title"	"Linear Regression (mx+b)"
"RegEqn"	"m*x+b"
"m"	3.2
"b"	1.2
"r²"	0.996109
"r"	0.998053
"Resid"	"{...}"

stat.values

"Linear Regression (mx+b)"
"m*x+b"
3.2
1.2
0.996109
0.998053
"{-0.4,0.4,0.2,0.,-0.2}"

stat.a	stat.dfDenom	stat.MedianY	stat.Q3X	stat.SSBlock
stat.AdjR²	stat.dfBlock	stat.MEPred	stat.Q3Y	stat.SSCol
stat.b	stat.dfCol	stat.MinX	stat.r	stat.SSX
stat.b0	stat.dfError	stat.MinY	stat.r²	stat.SSY
stat.b1	stat.dfInteract	stat.MS	stat.RegEqn	stat.SSError
stat.b2	stat.dfReg	stat.MSBlock	stat.Resid	stat.SSInteract
stat.b3	stat.dfNumer	stat.MSCol	stat.ResidTrans	stat.SSReg
stat.b4	stat.dfRow	stat.MSError	stat.σx	stat.SSRow
stat.b5	stat.DW	stat.MSInteract	stat.σy	stat.tList
stat.b6	stat.e	stat.MSReg	stat.σx1	stat.UpperPred
stat.b7	stat.ExpMatrix	stat.MSRow	stat.σx2	stat.UpperVal
stat.b8	stat.F	stat.n	stat.Σx	stat.x̄
stat.b9	stat.FBlock	stat.p̂	stat.Σx²	stat.x̄1
stat.b10	stat.Fcol	stat.p̂1	stat.Σxy	stat.x̄ 2
stat.bList	stat.FInteract	stat.p̂2	stat.Σy	stat.x̄ Diff
stat.χ²	stat.FreqReg	stat.p̂Diff	stat.Σy²	stat.x̄List
stat.c	stat.Frow	stat.PList	stat.s	stat.XReg
stat.CLower	stat.Leverage	stat.PVal	stat.SE	stat.XVal
stat.CLowerList	stat.LowerPred	stat.PValBlock	stat.SEList	stat.XValList
stat.CompList	stat.LowerVal	stat.PValCol	stat.SEPred	stat.ȳ
stat.CompMatrix	stat.m	stat.PValInteract	stat.sResid	stat.ȳ
stat.CookDist	stat.MaxX	stat.PValRow	stat.SEslope	stat.ȳList
stat.CUpper	stat.MaxY	stat.Q1X	stat.sp	stat.YReg
stat.CUpperList	stat.ME	stat.Q1Y	stat.SS	
stat.d	stat.MedianX			

附註：Lists & Spreadsheet 應用程式每次計算統計結果時，就會將「stat.」群組變數複製到「stat#.」群組。# 代表自動增量值。本功能讓您保留以前的結果，同時執行多次計算。

stat.values

目錄 >

stat.values

請參考 **stat.results** 範例。

stat.values

目錄 > 📖

針對最近求值的統計函數或指令，顯示計算出來的數值矩陣。

stat.values 會省略數值的對應名稱，這一點與 **stat.results** 不同。

您可複製數值，然後貼到其他位置。

stDevPop()

目錄 > 📖

stDevPop(*列表***[,** *頻率列表***])⇒***運算式*

傳回*列表*元素的母群體標準差。

*頻率列表*的每個元素，代表*列表*中對應元素的連續出現次數。

附註：*列表*至少必須有兩個元素。空元素會被忽略。如需空元素的詳細資訊，請參考第 頁碼：198 頁。

弧度角與自動模式：

stDevPop({1,2,5,-6,3,-2})	3.59398
stDevPop({1.3,2.5,-6.4},{3,2,5})	4.11107

stDevPop(*矩陣1***[,** *頻率矩陣***])⇒***矩陣*

傳回*矩陣1* 中各行母群體標準差的列向量。

*頻率矩陣*的每個元素，代表*矩陣1* 中對應元素的連續出現次數。

附註：*矩陣1* 至少必須有兩列。空元素會被忽略。如需空元素的詳細資訊，請參考第 頁碼：198 頁。

$$\text{stDevPop}\left(\begin{bmatrix} 1 & 2 & 5 \\ -3 & 0 & 1 \\ 5 & 7 & 3 \end{bmatrix}\right)$$

$$\begin{bmatrix} 3.26599 & 2.94392 & 1.63299 \end{bmatrix}$$

$$\text{stDevPop}\left(\begin{bmatrix} -1.2 & 5.3 \\ 2.5 & 7.3 \\ 6 & -4 \end{bmatrix},\begin{bmatrix} 4 & 2 \\ 3 & 3 \\ 1 & 7 \end{bmatrix}\right)$$

$$\begin{bmatrix} 2.52608 & 5.21506 \end{bmatrix}$$

stDevSamp()

目錄 > 📖

stDevSamp(*列表***[,** *頻率列表***])⇒***運算式*

傳回*列表*元素的樣本標準差。

*頻率列表*的每個元素，代表*列表*中對應元素的連續出現次數。

附註：*列表*至少必須有兩個元素。空元素會被忽略。如需空元素的詳細資訊，請參考第 頁碼：198 頁。

stDevSamp({1,2,5,-6,3,-2})	3.937
stDevSamp({1.3,2.5,-6.4},{3,2,5})	4.33345

stDevSamp()

目錄 >

stDevSamp(*矩陣1*[, *頻率矩陣*])⇒*矩陣*

傳回*矩陣1* 中各行樣本標準差的列向量。

*頻率矩陣*的每個元素，代表*矩陣1* 中對應元素的連續出現次數。

附註：*矩陣1* 至少必須有兩列。空元素會被忽略。如需空元素的詳細資訊，請參考第 頁碼：198 頁。

$$\text{stDevSamp}\left(\begin{bmatrix}1 & 2 & 5\\ -3 & 0 & 1\\ 5 & 7 & 3\end{bmatrix}\right)$$

$$\begin{bmatrix}4. & 3.60555 & 2.\end{bmatrix}$$

$$\text{stDevSamp}\left(\begin{bmatrix}-1.2 & 5.3\\ 2.5 & 7.3\\ 6 & -4\end{bmatrix},\begin{bmatrix}4 & 2\\ 3 & 3\\ 1 & 7\end{bmatrix}\right)$$

$$\begin{bmatrix}2.7005 & 5.44695\end{bmatrix}$$

Stop

目錄 >

Stop

程式設計指令：中止程式。

Stop 不得用於函數中。

輸入 範例的注意事項：關於輸入多行程式和函數定義的說明，請參閱產品手冊中的「計算工具」章節。

```
i:=0                                  0
Define prog1()=Prgm                   Done
                For i,1,10,1
                If i=5
                Stop
                EndFor
                EndPrgm
prog1()                               Done
i                                     5
```

Store

請參考 → (store)，頁碼：180。

string()

目錄 >

string(*Expr***)**⇒*字串*

化簡 *Expr* 並將結果傳回為字元字串。

```
string(1.2345)                        "1.2345"
string(1+2)                           "3"
```

subMat()

目錄 >

subMat(*矩陣1*[, *起始列*] [, *起始行*] [, *結束列*] [, *結束行*]) ⇒*矩陣*

傳回*矩陣1* 的指定子矩陣。

預設值：*起始列*=1、*起始行*=1、*結束列*=最後一列、*結束行*=最後一行。

$$\begin{bmatrix}1 & 2 & 3\\ 4 & 5 & 6\\ 7 & 8 & 9\end{bmatrix}\rightarrow m1 \qquad \begin{bmatrix}1 & 2 & 3\\ 4 & 5 & 6\\ 7 & 8 & 9\end{bmatrix}$$

$$\text{subMat}(m1,2,1,3,2) \qquad \begin{bmatrix}4 & 5\\ 7 & 8\end{bmatrix}$$

$$\text{subMat}(m1,2,2) \qquad \begin{bmatrix}5 & 6\\ 8 & 9\end{bmatrix}$$

Sum (Sigma)

請參考 Σ()，頁碼：174。

sum()

目錄 >

sum(*列表***[,** *起點***[,** *終點***]])⇒***運算式*

傳回*列表*中各元素的和。

*起點*和*終點*可選擇性輸入，用於指定元素範圍。

若有任何空引數，都會求出空結果。*列表*中的空元素會遭到忽略。如需空元素的詳細資訊，請參考第 頁碼：198 頁。

$\text{sum}(\{1,2,3,4,5\})$ 15

$\text{sum}(\{a,2\cdot a,3\cdot a\})$ "Error: Variable is not defined"

$\text{sum}(\text{seq}(n,n,1,10))$ 55

$\text{sum}(\{1,3,5,7,9\},3)$ 21

sum(*矩陣 1***[,** *起點***[,** *終點***]])⇒***矩陣*

傳回包含*矩陣 1* 中各行所有元素和的列向量。

*起點*和*終點*可選擇性輸入，用於指定列範圍。

若有任何空引數，都會求出空結果。*矩陣 1* 中的空元素會遭到忽略。如需空元素的詳細資訊，請參考第 頁碼：198 頁。

$\text{sum}\left(\begin{bmatrix}1&2&3\\4&5&6\end{bmatrix}\right)$ $\begin{bmatrix}5&7&9\end{bmatrix}$

$\text{sum}\left(\begin{bmatrix}1&2&3\\4&5&6\\7&8&9\end{bmatrix}\right)$ $\begin{bmatrix}12&15&18\end{bmatrix}$

$\text{sum}\left(\begin{bmatrix}1&2&3\\4&5&6\\7&8&9\end{bmatrix},2,3\right)$ $\begin{bmatrix}11&13&15\end{bmatrix}$

sumIf()

目錄 >

sumIf(*列表***,***準則***[,** *總和表***])⇒***值*

傳回*列表*中符合指定*準則*之所有元素的累積總和。您也可以指定另一個列表(*總和表*) ，以提供要累加的元素。

*列表*可以是運算式、列表或矩陣。*總和表*(若有指定) 與*列表*的維數必須相同。

*準則*可以有以下形式：

- 值、運算式、字串。例如 **34** 代表只累加*列表*中可化簡成 34 這個值的元素。
- 包含 **?**符號當作各元素預留位置的布林運算式。例如 **?<10** 代表只累加*列表*中小於 10 的元素。

$\text{sumIf}(\{1,2,\mathbf{e},3,\pi,4,5,6\},2.5<?<4.5)$ 12.859874482

$\text{sumIf}(\{1,2,3,4\},2<?<5,\{10,20,30,40\})$ 70

sumIf()

目錄 >

*列表*中的某元素若符合*準則*，就會將該元素計入累積總和。若您加入*總和表*，則會改而將*總和表*中的對應元素計入總和。

您可在 Lists & Spreadsheet 應用程式中用儲存格範圍取代*列表*與*總和表*。

空元素會被忽略。如需空元素的詳細資訊，請參考第 頁碼：198 頁。

附註：另請參考 **countIf()**，頁碼：28。

sumSeq()

請參考 Σ()，頁碼：174。

system()

目錄 >

system(*值 1* [, *值 2* [, *值 3* [, ...]]])

傳回列表格式的聯立方程式。您也可以利用範本建立聯立方程式。

T

T (transpose)

目錄 >

*矩陣 1***T**⇒*矩陣*

傳回*矩陣 1* 的複共軛轉置矩陣。

附註：如果要從電腦鍵盤插入本運算子，可輸入 `@t`。

$$\begin{bmatrix}1&2&3\\4&5&6\\7&8&9\end{bmatrix}^T \qquad \begin{bmatrix}1&4&7\\2&5&8\\3&6&9\end{bmatrix}$$

tan()

trig 鍵

tan(*值 1***)**⇒*值*

tan(*列表 1***)**⇒*列表*

tan(*值 1***):** 傳回引數的正切值。

tan(*列表 1***):** 傳回*列表 1* 中各元素之正切值的列表。

度數角模式：

$$\tan\left(\left(\frac{\pi}{4}\right)^r\right) \qquad 1.$$

$$\tan(45) \qquad 1.$$

$$\tan(\{0,60,90\}) \qquad \{0.,1.73205,\text{undef}\}$$

梯度角模式：

tan()

trig 鍵

附註：系統會根據目前的角度模式，將引數解譯為度數角、梯度角或弧度角。您可使用 °、G 或 r 來暫時覆寫角度模式設定。

$\tan\left(\left(\frac{\pi}{4}\right)r\right)$ 1.

$\tan(50)$ 1.

$\tan(\{0,50,100\})$ $\{0.,1.,\text{undef}\}$

弧度角模式：

$\tan\left(\frac{\pi}{4}\right)$ 1.

$\tan(45°)$ 1.

$\tan\left(\left\{\pi,\frac{\pi}{3},\pi,\frac{\pi}{4}\right\}\right)$ $\{0.,1.73205,0.,1.\}$

tan(*方陣 1***)⇒***方陣*

傳回*方陣 1* 的矩陣正切。這和計算各元素的正切不同。若要了解計算方式，請參考 **cos()**。

方陣 1 必需可對角化。結果一定會包含浮點數。

弧度角模式：

$$\tan\left(\begin{bmatrix}1 & 5 & 3\\ 4 & 2 & 1\\ 6 & -2 & 1\end{bmatrix}\right)$$

$$\begin{bmatrix}-28.2912 & 26.0887 & 11.1142\\ 12.1171 & -7.83536 & -5.48138\\ 36.8181 & -32.8063 & -10.4594\end{bmatrix}$$

$\tan^{-1}()$

trig 鍵

tan⁻¹(*值 1***)⇒***值*

tan⁻¹(*列表 1***)⇒***列表*

tan⁻¹(*值 1***):** 傳回正切值是值 1 的角度。

tan⁻¹(*列表 1***):** 傳回*列表 1* 各元素之反正切值的列表。

附註：系統會根據目前的角度模式設定，將結果傳回為度數角、梯度角或弧度角。

附註：如果要從鍵盤插入本函數，可輸入 `arctan(...)`。

tan⁻¹(*方陣 1***)⇒***方陣*

傳回*方陣 1* 的矩陣反正切。這和計算各元素的反正切不同。若要了解計算方式，請參考 **cos()**。

度數角模式：

$\tan^{-1}(1)$ 45

梯度角模式：

$\tan^{-1}(1)$ 50

弧度角模式：

$\tan^{-1}(\{0,0.2,0.5\})$ $\{0,0.197396,0.463648\}$

弧度角模式：

tan⁻¹() 鍵

方陣1 必需可對角化。結果一定會包含浮點數。

$$\tan^{-1}\left(\begin{bmatrix}1 & 5 & 3\\ 4 & 2 & 1\\ 6 & -2 & 1\end{bmatrix}\right)$$

$$\begin{bmatrix}-0.083658 & 1.26629 & 0.62263\\ 0.748539 & 0.630015 & -0.070012\\ 1.68608 & -1.18244 & 0.455126\end{bmatrix}$$

tanh()

目錄 >

tanh(*值 1***)⇒***值*

tanh(*列表 1***)⇒***列表*

tanh(*值 1***):** 傳回引數的雙曲正切值。

tanh(*列表 1***):** 傳回*列表 1* 各元素之雙曲正切值的列表。

$$\tanh(1.2) \qquad 0.833655$$

$$\tanh(\{0,1\}) \qquad \{0.,0.761594\}$$

tanh(*方陣 1***)⇒***方陣*

傳回*方陣 1* 的矩陣雙曲正切。這和計算各元素的雙曲正切不同。若要了解計算方式，請參考 **cos()**。

方陣 1 必需可對角化。結果一定會包含浮點數。

弧度角模式：

$$\tanh\left(\begin{bmatrix}1 & 5 & 3\\ 4 & 2 & 1\\ 6 & -2 & 1\end{bmatrix}\right)$$

$$\begin{bmatrix}-0.097966 & 0.933436 & 0.425972\\ 0.488147 & 0.538881 & -0.129382\\ 1.28295 & -1.03425 & 0.428817\end{bmatrix}$$

tanh⁻¹()

目錄 >

tanh⁻¹(*值 1***)⇒***值*

tanh⁻¹(*列表 1***)⇒***列表*

tanh⁻¹(*值 1***):** 傳回引數的反雙曲正切值。

tanh⁻¹(*列表 1***):** 傳回*列表 1* 各元素之反雙曲正切值的列表。

附註：如果要從鍵盤插入本函數，可輸入 `arctanh(…)`。

tanh⁻¹(*方陣 1***)⇒***方陣*

傳回*方陣 1* 的矩陣反雙曲正切。這和計算各元素的反雙曲正切不同。若要了解計算方式，請參考 **cos()**。

方陣 1 必需可對角化。結果一定會包含浮點數。

直角複數格式：

$$\tanh^{-1}(0) \qquad 0.$$

$$\tanh^{-1}(\{1,2.1,3\})$$

$$\{\text{undef},0.518046-1.5708\cdot i,0.346574-1.570$$

若要看到完整結果，請按 ▲，然後使用 ◀ 與 ▶ 移動游標。

弧度角模式與直角複數格式：

tanh⁻¹() 目錄 >

$$\tanh^{-1}\left(\begin{bmatrix}1 & 5 & 3\\ 4 & 2 & 1\\ 6 & -2 & 1\end{bmatrix}\right)$$

$$\begin{bmatrix}-0.099353+0.164058\cdot i & 0.267834-1.4908\\ -0.087596-0.725533\cdot i & 0.479679-0.9473\\ 0.511463-2.08316\cdot i & -0.878563+1.7901\end{bmatrix}$$

若要看到完整結果，請按▲，然後使用◀與▶移動游標。

tCdf() 目錄 >

tCdf(*下限*,*上限*,*df*)⇒*數字*(若*下限*和*上限*是數字) 或*列表*(若*下限*和*上限*是列表)

針對指定自由度 *df* 的*下限*和*上限*之間，計算 Student-*t* 分布機率。

對於 P(X ≤ *上限*)，請設定 *下限* = −9E999。

Text 目錄 >

Text*提示字串*[, *顯示旗標*]

程式設計指令：程式暫停執行，並在對話方塊中顯示包含*提示字串*的字元字串。

使用者選取 **[確定]** 時，程式就會繼續執行。若選取 **[取消]** 可停止執行程式。

*旗標*選擇性(非必要) 引數可以是任何運算式。

- 若省略*顯示旗標*或其求值為 **1**，則在 Calculator 歷史記錄中加入文字訊息。
- 若*顯示旗標*求值為 **0**，則不在歷史記錄中加入文字訊息。

若程式需要使用者輸入回答，請參考 **Request(頁碼：115) 或 RequestStr(頁碼：117)** 。

附註：您可以在使用者自行定義的程式中使用本指令，但不能用在函數中。

定義讓程式暫停，並在對話方塊中顯示五個隨機數字。

在 Prgm...EndPrgm 範本的各行結尾處按 ↵，不要按 enter。按住電腦鍵盤的 Alt 然後按 Enter。

```
Define text_demo()=Prgm
  For i,1,5
    strinfo:=”Random number “ &
string(rand(i))
    Text strinfo
  EndFor
EndPrgm
```

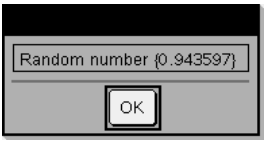
執行程式：

```
text_demo()
```

Text

目錄 > 📖

對話方塊一例：



Then

請參考 If，頁碼：63。

tInterval

目錄 > 📖

tInterval *列表*[,*Freq*[,*CLevel*]]

(輸入資料列表)

tInterval $\bar{x}$,*sx*,*n*[,*CLevel*]

(輸入統計摘要)

計算 *t* 信賴區間。*stat.results* 變數會儲存結果摘要(請參閱第 頁碼：134 頁) 。

如要了解列表中有空元素時的影響，請參考°ß空元素°®，頁碼：198。

輸出變數	說明
stat.CLower、stat.CUpper	不明母群體平均值的信賴區間
stat.$\bar{x}$	常態隨機分布的資料序列平均值樣本
stat.ME	邊際誤差
stat.df	自由度
stat.σx	樣本標準差
stat.n	資料序列的長度與平均值樣本

tInterval_2Samp

目錄 > 📖

tInterval_2Samp *列表1*,*列表2*[,*Freq1*[,*Freq2*[,*CLevel*[,*合併*]]]]

(輸入資料列表)

tInterval_2Samp

目錄 > 📖

tInterval_2Samp $\bar{x}1,sx1,n1,\bar{x}2,sx2,n2$ [,*CLevel*[,*合併*]]

(輸入統計摘要)

計算二樣本 *t* 信賴區間。*stat.results* 變數會儲存結果摘要(請參閱第 頁碼: 134 頁) 。

合併=**1** 合併變異數; *合併*=**0** 不合併變異數。

如要了解列表中有空元素時的影響，請參考°ß空元素°®，頁碼: 198。

輸出變數	說明
stat.CLower、stat.CUpper	包含信心水準分布機率的信賴區間
stat.$\bar{x}$1-$\bar{x}$2	常態隨機分布的資料序列平均值樣本
stat.ME	邊際誤差
stat.df	自由度
stat.$\bar{x}$1, stat.$\bar{x}$2	常態隨機分布的資料序列平均值樣本
stat.σx1、stat.σx2	*列表 1* 和*列表 2 的標準差樣本*
stat.n1、stat.n2	資料序列的樣本數
stat.sp	合併標準差。*合併* = YES 時才予計算

tPdf()

目錄 > 📖

tPdf(*XVal***,***df***)⇒***數字*(若 *XVal* 是數字) 或 *列表*(若 *XVal* 是列表)

針對指定自由度 *df*，計算 Student-*t* 分布在指定 *x* 值的機率密度函數(pdf) 。

trace()

目錄 > 📖

trace(*方陣***)⇒***值*

傳回*方陣*的描跡(主對角線上的所有元素總和)。

$$\text{trace}\left(\begin{bmatrix}1 & 2 & 3\\ 4 & 5 & 6\\ 7 & 8 & 9\end{bmatrix}\right) \qquad 15$$

$$a:=12 \qquad 12$$

$$\text{trace}\left(\begin{bmatrix}a & 0\\ 1 & a\end{bmatrix}\right) \qquad 24$$

Try
區段1
Else
區段2
EndTry

執行*區段1*，除非出現錯誤。若*區段1*出現錯誤，則程式改為執行*區段2*。*errCode* 系統變數含有錯誤代碼，可讓程式執行錯誤復原作業。若需錯誤代碼一覽表，請參考「*錯誤代碼與錯誤訊息*」，頁碼：208。

區段1 與*區段2* 可以只是一個語句，也可以是由「:」字元分隔的一連串語句。

輸入範例的注意事項：關於輸入多行程式和函數定義的說明，請參閱產品手冊中的「計算工具」章節。

例 2

若要觀察 **Try**、**ClrErr** 與 **PassErr** 指令的運作方式，請輸入如右所示的 eigenvals() 程式。請執行以下各個運算式，以便執行程式。

$$eigenvals\left(\begin{bmatrix}-3\\-41\\5\end{bmatrix},\begin{bmatrix}-1 & 2 & -3.1\end{bmatrix}\right)$$

附註：另請參考第 頁碼：22 頁的 **ClrErr** 和第 頁碼：101 頁的 **PassErr**。

```
Define prog1()=Prgm
    Try
    z:=z+1
    Disp "z incremented."
    Else
    Disp "Sorry, z undefined."
    EndTry
    EndPrgm
                                    Done
z:=1:prog1()
                                z incremented.
                                    Done
DelVar z:prog1()
                                Sorry, z undefined.
                                    Done
```

```
Define eigenvals(a,b)=Prgm
c Program eigenvals(A,B) displays
eigenvalues of A·B
Try
  Disp "A= ",a
  Disp "B= ",b
  Disp " "
  Disp "Eigenvalues of A·B are:",eigVl(a*b)
Else
  If errCode=230 Then
    Disp "Error:Product of A·B must be a
square matrix"
    ClrErr
  Else
    PassErr
  EndIf
EndTry
EndPrgm
```

tTest

目錄 > 📖

tTest μ*0*,*列表*[,*Freq*[,*Hypoth*]]

(輸入資料列表)

tTest μ*0*,x̄,*sx*,*n*,[*Hypoth*]

(輸入統計摘要)

執行單個不明母群體平均值 μ 的假設檢定，且母群體標準差 σ 不明。*stat.results* 變數會儲存結果摘要(請參閱第 頁碼：134 頁) 。

以 H_0:μ = μ0 檢定以下項目：

H_a:μ < μ0 時，設定 *Hypoth*<0

H_a:μ ≠ μ0(預設值) 時，設定 *Hypoth*=0

H_a:μ > μ0 時，設定 *Hypoth*>0

如要了解列表中有空元素時的影響，請參考°ß空元素°®，頁碼：198。

輸出變數	說明
stat.t	(x̄ − μ0) / (stdev / sqrt(n))
stat.PVal	無效假說被否定之最低意義標準
stat.df	自由度
stat.x̄	*列表中資料序列的平均值樣本*
stat.sx	資料序列的樣本標準差
stat.n	樣本數

tTest_2Samp

目錄 > 📖

tTest_2Samp *列表1*,*列表2*[,*Freq1*[,*Freq2*[,*Hypoth*[,*合併*]]]]

(輸入資料列表)

tTest_2Samp x̄*1*,*sx1*,*n1*,x̄*2*,*sx2*,*n2*[,*Hypoth*[,*合併*]]

(輸入統計摘要)

tTest_2Samp

計算二樣本 *t* 檢定。*stat.results* 變數會儲存結果摘要(請參閱第 頁碼：134 頁)。

以 H_0:μ1 = μ2 檢定以下項目：

H_a:μ1< μ2 時，設定 *Hypoth*<0

H_a:μ1≠ μ2(預設值)時，設定 *Hypoth*=0

H_a:μ1> μ2 時，設定 *Hypoth*>0

合併=**1** 合併變異數

合併=**0** 不合併變異數

如要了解列表中有空元素時的影響，請參考°ß空元素°®，頁碼：198。

輸出變數	說明
stat.t	平均值差的標準常態計算值
stat.PVal	無效假說被否定之最低意義標準
stat.df	t-統計量自由度
stat.$\bar{x}$1、stat.$\bar{x}$2	*列表 1* 和 *列表 2 中資料序列的平均值樣本*
stat.sx1、stat.sx2	*列表 1* 和 *列表 2 中資料序列的標準差樣本*
stat.n1、stat.n2	樣本大小
stat.sp	合併標準差。*合併*=1 時才予計算。

tvmFV()

tvmFV(*N*,*I*,*PV*,*Pmt*,[*PpY*],[*CpY*],[*PmtAt*]**)**⇒*值*

```
tvmFV(120,5,0,-500,12,12)        77641.1
```

可計算貨幣未來價值的財務函數。

附註：貨幣時間價值引數表說明了貨幣時間價值函數使用的引數(第 頁碼：148 頁)。另請參考 **amortTbl()**，頁碼：7。

tvmI()

tvmI(*N*,*PV*,*Pmt*,*FV*,[*PpY*],[*CpY*],[*PmtAt*]**)**⇒*值*

```
tvmI(240,100000,-1000,0,12,12)        10.5241
```

tvmI()

目錄 >

可計算每年利率的財務函數。

附註：貨幣時間價值引數表說明了貨幣時間價值函數使用的引數(第 頁碼：148 頁)。另請參考 **amortTbl()**，頁碼：7。

tvmN()

目錄 >

tvmN(*I*,*PV*,*Pmt*,*FV*,[*PpY*],[*CpY*],[*PmtAt*]**)**⇒*值*

可計算付款期數的財務函數。

附註：貨幣時間價值引數表說明了貨幣時間價值函數使用的引數(第 頁碼：148 頁)。另請參考 **amortTbl()**，頁碼：7。

```
tvmN(5,0,-500,77641,12,12)          120.
```

tvmPmt()

目錄 >

tvmPmt(*N*,*I*,*PV*,*FV*,[*PpY*],[*CpY*],[*PmtAt*]**)**⇒*值*

可計算每期付款金額的財務函數。

附註：貨幣時間價值引數表說明了貨幣時間價值函數使用的引數(第 頁碼：148 頁)。另請參考 **amortTbl()**，頁碼：7。

```
tvmPmt(60,4,30000,0,12,12)          -552.496
```

tvmPV()

目錄 >

tvmPV(*N*,*I*,*Pmt*,*FV*,[*PpY*],[*CpY*],[*PmtAt*]**)**⇒*值*

可計算現值的財務函數。

附註：貨幣時間價值引數表說明了貨幣時間價值函數使用的引數(第 頁碼：148 頁)。另請參考 **amortTbl()**，頁碼：7。

```
tvmPV(48,4,-500,30000,12,12)        -3426.7
```

貨幣時間價值引數*	說明	資料類型
N	付款期數	實數

貨幣時間價值引數*	說明	資料類型
I	年利率	實數
PV	現值	實數
Pmt	付款金額	實數
FV	未來值	實數
PpY	每年付款期數，預設值=1	> 0 的整數
CpY	每年複利期數，預設值=1	> 0 的整數
PmtAt	每期付款時間是期末還是期初，預設值=期末	整數(0=期末，1=期初)

* 這些貨幣時間價值引數名稱，與 Calculator 應用程式財務解題系統所用的貨幣時間價值變數名稱(例如 **tvm.pv** 與 **tvm.pmt**) 類似。但是財務函數不會把引數值或計算結果儲存到 TVM 變數。

TwoVar

目錄 > 📖

TwoVar *X*, *Y*[, [*Freq*] [, *類別*, *包含*]]

計算 TwoVar 統計。*stat.results* 變數會儲存結果摘要(請參閱第 頁碼：134 頁) 。

所有列表的維數都必須相同，*包含*除外。

X、*Y* 是自變數和因變數列表。

Freq 是頻率值列表(非必要) 。*Freq* 的每個元素，可用於指定各 *X*、*Y* 對應資料點的出現頻率。預設值是 1。所有元素都必須是 ≥ 0 的整數。

*類別*是 *X*、*Y* 對應資料的數字類別代號列表。

*包含*是一個以上類別代號的列表。只有本列表中類別代號所對應的資料項目，才會包含在計算作業中。

X、*Freq*、*類別*的任何列表中若有空元素，則所有這些列表的對應元素就會是空元素。*X1* 至 *X20* 的任何列表中若有空元素，則所有這些列表的對應元素就會是空元素。如需空元素的詳細資訊，請參考第 頁碼：198 頁。

輸出變數	說明
stat.$\bar{x}$	x 值的平均值
stat. x	x 值的和
stat. x2	x2 值的和
stat.sx	x 的樣本標準差
stat. x	x 的母群體標準差
stat.n	資料點數量
stat.$\bar{y}$	y 值的平均值
stat. y	y 值的和
stat. y^2	y2 值的和
stat.sy	y 的樣本標準差
stat. y	y 的母群體標準差
stat. xy	x ·y 值的和
stat.r	相關係數
stat.MinX	x 值的最小值
stat.Q_1X	x 的第一四分位數
stat.MedianX	x 的中位數
stat.Q_3X	x 的第三四分位數
stat.MaxX	x 值的最大值
stat.MinY	y 值的最小值
stat.Q_1Y	y 的第一四分位數
stat.MedY	y 的中位數
stat.Q_3Y	y 的第三四分位數
stat.MaxY	y 值的最大值
stat. $(x-)^2$	x 的離均差平方和
stat. $(y-)^2$	y 的離均差平方和

U

unitV()

目錄 >

unitV(*向量1***)⇒***向量*

依照*向量1*的形式傳回列單位向量或行單位向量。

*向量1*必須是單列矩陣或單行矩陣。

若要看到完整結果，請按▲，然後使用◀與▶移動游標。

unitV([1 2 1])	[0.408248 0.816497 0.408248]
unitV([1; 2; 3])	[0.267261; 0.534522; 0.801784]

unLock

目錄 >

unLock *Var1*[, *Var2*] [, *Var3*] ...

unLock *Var*.

解開鎖定指定變數或變數群組。您無法修改或刪除已鎖定的變數。

請參考第 頁碼：79 頁的 **Lock** 和第 頁碼：60 頁的 **getLockInfo()**。

a:=65	65
Lock *a*	*Done*
getLockInfo(*a*)	1
a:=75	"Error: Variable is locked."
DelVar *a*	"Error: Variable is locked."
Unlock *a*	*Done*
a:=75	75
DelVar *a*	*Done*

V

varPop()

目錄 >

varPop(*列表***[,** *頻率列表***])⇒***運算式*

傳回*列表*的母群體變異數。

*頻率列表*的每個元素，代表*列表*中對應元素的連續出現次數。

附註：*列表*至少必須包含兩個元素。

若任何列表中有空元素，則系統會忽略該元素和另一列表中的對應元素。如需空元素的詳細資訊，請參考第 頁碼：198 頁。

varPop({5,10,15,20,25,30})	72.9167

varSamp()

目錄 >

varSamp(*列表***[,** *頻率列表***])⇒***運算式*

傳回*列表*的樣本變異數。

*頻率列表*的每個元素，代表*列表*中對應元素的連續出現次數。

附註：*列表*至少必須包含兩個元素。

若任何列表中有空元素，則系統會忽略該元素和另一列表中的對應元素。如需空元素的詳細資訊，請參考第 頁碼：198 頁。

$$\text{varSamp}(\{1,2,5,-6,3,-2\}) \qquad \frac{31}{2}$$

$$\text{varSamp}(\{1,3,5\},\{4,6,2\}) \qquad \frac{68}{33}$$

varSamp(*矩陣1***[,** *頻率矩陣***])⇒***矩陣*

傳回包含*矩陣1*中各行樣本變異數的列向量。

*頻率矩陣*的每個元素，代表*矩陣1*中對應元素的連續出現次數。

若任何矩陣中有空元素，則系統會忽略該元素和另一矩陣中的對應元素。如需空元素的詳細資訊，請參考第 頁碼：198 頁。

附註：*矩陣1*至少必須包含兩列。

$$\text{varSamp}\left(\begin{bmatrix} 1 & 2 & 5 \\ -3 & 0 & 1 \\ .5 & .7 & 3 \end{bmatrix}\right) \qquad \begin{bmatrix} 4.75 & 1.03 & 4 \end{bmatrix}$$

$$\text{varSamp}\left(\begin{bmatrix} -1.1 & 2.2 \\ 3.4 & 5.1 \\ -2.3 & 4.3 \end{bmatrix},\begin{bmatrix} 6 & 3 \\ 2 & 4 \\ 5 & 1 \end{bmatrix}\right) \qquad \begin{bmatrix} 3.91731 & 2.08411 \end{bmatrix}$$

W

Wait

目錄 >

Wait *timeInSeconds*

停止執行 *timeInSeconds* 秒。

Wait 特別適合需要短時間延遲以等待請求的資料的程式。

引數 *timeInSeconds* 必須包含一個簡化為 0 到 100 範圍內十進制值的表達式。此命令將此值四捨五入至到最接近的 0.1 秒。

如要取消進行中的 **Wait**，

- **計算機：**按住 on 鍵並重複按 · 鍵。
- **Windows®：**按住 **F12** 鍵並重複按 **Enter** 鍵。
- **Macintosh®：**按住 **F5** 鍵並重複按

要等待 4 秒:

Wait 4

要等待 1/2 秒:

Wait 0.5

要使用變量 *seccount* 等待 1.3 秒:

seccount:=1.3
Wait seccount

此範例將開啟一個綠色 LED 0.5 秒，然後關閉。

Send "SET GREEN 1 ON"
Wait 0.5
Send "SET GREEN 1 OFF"

Wait

Enter 鍵。

- **iPad®：**應用程式顯示提示。您可以繼續等待或取消。

附註:可以在使用者自行定義的程式中使用 **Wait** 指令，而非在函數中。

warnCodes()

warnCodes(*Expr1*, *StatusVar***)⇒***運算式*

計算運算式 *Expr1*，傳回結果，並將任何所產生警告的代碼儲存在 *StatusVar* 列表變數中。如果未產生警告，則此函數會指定空的列表給 StatusVar。

Expr1 可以是任何有效的 TI-Nspire™ 或 TI-Nspire™ CAS 數學運算式。無法對 *Expr1* 使用指令或指定工作。

StatusVar 必須是有效的變數名稱。

如需警告代碼與相關訊息的列表，請參考第 頁碼：216 頁。

warnCodes(det([1.23456E-999]),*warn*)	1.23456E-999
warn	{10029}

when()

when(*條件*, *真結果* [, *偽結果*][, *不明結果*]**)** ⇒*運算式*

根據*條件*是真、偽還是不明，傳回*真結果*、*偽結果*或*不明結果*。若引數太少，不足以求出適當結果，則傳回輸入值。

若省略*偽結果*和*不明結果*，則可得出定義僅限於*條件*是真的範圍內運算式。

用 **undef** *偽結果*可定義出只在某區間範圍內繪圖的運算式。

when(*x*<0,*x*+3)|*x*=5	undef

when() 可用於定義遞迴函數。

when(*n*>0,*n*·*factoral*(*n*−1),1)→*factoral*(*n*)	*Done*
factoral(3)	6
3!	6

While

目錄 >

While *條件*
　區段
EndWhile

只要*條件*是真，就執行*區段*中的語句。

*區段*可以只是一個語句，也可以是由「:」字元分隔的一系列語句。

輸入 範例的注意事項：關於輸入多行程式和函數定義的說明，請參閱產品手冊中的「計算工具」章節。

```
Define sum_of_recip(n)=Func
                Local i,tempsum
                1→i
                0→tempsum
                While i≤n
                tempsum+1/i→tempsum
                i+1→i
                EndWhile
                Return tempsum
                EndFunc
                                Done
sum_of_recip(3)                 11/6
```

X

xor (互斥)

目錄 >

布林運算式1 **xor** *布林運算式2* 傳回 *布林運算式*

布林列表1 **xor** *布林列表2* 傳回 *布林列表*

布林矩陣1 **xor** *布林矩陣2* 傳回 *布林矩陣*

若*布林運算式1*是真而*布林運算式2*是偽則傳回真，反之亦然。

若兩個引數同時是真或同時是偽則傳回偽。若其中一個引數無法解出真偽值，則傳回化簡的布林運算式。

附註：請參考 **or，頁碼：99。**

```
true xor true                   false
5>3 xor 3>5                     true
```

整數1 **xor** *整數2*⇒*整數*

利用 **xor** 功能個別位元比較兩個實際整數。系統內部會把兩個整數轉換成有正負號的 64 位元二進位數字。比較對應的位元時，如果只有一個位元是 1，則結果是 1；如果兩個位元都是 0 或都是 1，則結果是 0。傳回的值代表位元結果，並會以基底模式為顯示依據。

十六進位基底模式：

請注意：數字零，而非英文字母 O。

```
0h7AC36 xor 0h3D5F              0h79169
```

二進位基底模式：

```
0b100101 xor 0b100              0b100001
```

xor (互斥)

目錄 > 📖

您可以用任何整數做為基底。如果要輸入二進位或十六進位數字，則必須在前面分別加上 0b 或 0h 字首。如果沒有加上字首，就會把整數視為十進位(以 10 為底) 。

如果您輸入的十進位整數太大，超出具正負號 64 位元二進位格式的範圍，系統會以對稱模數運算法使數值落入適當範圍。如需詳細資訊，請參閱 ▶**Base2**，頁碼：16。

附註：請參考 **or**，頁碼：99。

附註：您可輸入最長 64 位的二進位數字(不含 0b 字首) 。您可輸入最長 16 位的十六進位數字。

Z

zInterval

目錄 > 📖

zInterval σ,*列表*[,*Freq*[,*CLevel*]]

(輸入資料列表)

zInterval σ,x̄,*n* [,*CLevel*]

(輸入統計摘要)

計算 *z* 信賴區間。*stat.results* 變數會儲存結果摘要(請參閱第 頁碼：134 頁) 。

如要了解列表中有空元素時的影響，請參考°ß空元素°®，頁碼：198。

輸出變數	說明
stat.CLower、stat.CUpper	不明母群體平均值的信賴區間
stat.x̄	常態隨機分布的資料序列平均值樣本
stat.ME	邊際誤差
stat.sx	樣本標準差
stat.n	資料序列的長度與平均值樣本
stat.σ	*列表資料序列的已知母群體標準差*

zInterval_1Prop

目錄 > 📖

zInterval_1Prop *x*,*n* [,*CLevel*]

zInterval_1Prop

目錄 > 📖

計算單一母群體比例 *z* 信賴區間。*stat.results* 變數會儲存結果摘要(請參閱第 頁碼: 134 頁)。

x 是不為負的整數。

如要了解列表中有空元素時的影響，請參考°ß空元素°®，頁碼: 198。

輸出變數	說明
stat.CLower、stat.CUpper	包含信心水準分布機率的信賴區間
stat.$\hat{p}$	計算出的成功比例
stat.ME	邊際誤差
stat.n	資料序列的樣本數

zInterval_2Prop

目錄 > 📖

zInterval_2Prop *x1*,*n1*,*x2*,*n2*[,*CLevel*]

計算二母群體比例 *z* 信賴區間。*stat.results* 變數會儲存結果摘要(請參閱第 頁碼: 134 頁)。

x1 和 *x2* 是不為負的整數。

如要了解列表中有空元素時的影響，請參考°ß空元素°®，頁碼: 198。

輸出變數	說明
stat.CLower、stat.CUpper	包含信心水準分布機率的信賴區間
stat.$\hat{p}$Diff	計算出的兩個比例差值
stat.ME	邊際誤差
stat.$\hat{p}$1	第一個樣本比例估計值
stat.$\hat{p}$2	第二個樣本比例估計值
stat.n1	資料序列一的樣本數
stat.n2	資料序列二的樣本數

zInterval_2Samp

目錄 > 📖

zInterval_2Samp σ_1,σ_2 ,*列表1*,*列表2*[,*Freq1*[,*Freq2*,[*CLevel*]]]

zInterval_2Samp

目錄 > 📖

(輸入資料列表)

zInterval_2Samp $\sigma_1,\sigma_2,\bar{x}1,n1,\bar{x}2,n2$ [,*CLevel*]

(輸入統計摘要)

計算二樣本 *z* 信賴區間。*stat.results* 變數會儲存結果摘要(請參閱第 頁碼：134 頁)。

如要了解列表中有空元素時的影響，請參考°ß空元素°®，頁碼：198。

輸出變數	說明
stat.CLower、stat.CUpper	包含信心水準分布機率的信賴區間
stat.$\bar{x}$1-$\bar{x}$2	常態隨機分布的資料序列平均值樣本
stat.ME	邊際誤差
stat.$\bar{x}$1、stat.$\bar{x}$2	常態隨機分布的資料序列平均值樣本
stat.σx1、stat.σx2	*列表 1* 和*列表 2 的標準差樣本*
stat.n1、stat.n2	資料序列的樣本數
stat.r1、stat.r2	*列表 1* 和*列表 2 資料序列的已知母群體標準差*

zTest

目錄 > 📖

zTest $\mu 0,\sigma,$*列表*,[*Freq*[,*Hypoth*]]

(輸入資料列表)

zTest $\mu 0,\sigma,\bar{x},n$[,*Hypoth*]

(輸入統計摘要)

執行 *z* 檢定(*頻率列表*為頻率)。*stat.results* 變數會儲存結果摘要(請參閱第 頁碼：134 頁)。

以 H_0:μ = μ0 檢定以下項目：

H_a:μ < μ0 時，設定 *Hypoth*<0

H_a:μ ≠ μ0(預設值)時，設定 *Hypoth*=0

H_a:μ > μ0 時，設定 *Hypoth*>0

zTest

目錄 > 📖

如要了解列表中有空元素時的影響，請參考°ß空元素°®，頁碼：198。

輸出變數	說明
stat.z	$(\bar{x} - \mu 0) / (\sigma / \text{sqrt}(n))$
stat.P 值	無效假說被否定之最低機率
stat.$\bar{x}$	*列表中資料序列的平均值樣本*
stat.sx	資料序列的樣本標準差。只在輸入*資料*時才傳回結果。
stat.n	樣本數

zTest_1Prop

目錄 > 📖

zTest_1Prop *p0,x,n*[,*Hypoth*]

計算單一母群體比例 *z* 檢定。*stat.results* 變數會儲存結果摘要(請參閱第 頁碼：134 頁) 。

x 是不為負的整數。

以 H_0:$p = p0$ 檢定以下項目：

H_a:$p > p0$ 時，設定 *Hypoth*>0

H_a:$p \neq p0$(*預設值)* 時，設定 *Hypoth*=0

H_a:$p < p0$ 時，設定 *Hypoth*<0

如要了解列表中有空元素時的影響，請參考°ß空元素°®，頁碼：198。

輸出變數	說明
stat.p0	假設的母群體比例
stat.z	比例的標準常態計算值
stat.PVal	無效假說被否定之最低意義標準
stat.$\hat{p}$	樣本比例估計值
stat.n	樣本數

zTest_2Prop

目錄 > 📖

zTest_2Prop *x1,n1,x2,n2*[,*Hypoth*]

zTest_2Prop

目錄 >

計算二母群體比例 *z* 檢定。*stat.results* 變數會儲存結果摘要(請參閱第 頁碼：134 頁)。

x1 和 *x2* 是不為負的整數。

以 H_0:*p1* = *p2* 檢定以下項目：

H_a:*p1* > *p2* 時，設定 *Hypoth*>0

H_a:*p1* ≠ *p2*(預設值)時，設定 *Hypoth*=0

H_a:*p* < *p0* 時，設定 *Hypoth*<0

如要了解列表中有空元素時的影響，請參考°ß空元素°®，頁碼：198。

輸出變數	說明
stat.z	比例差的標準常態計算值
stat.PVal	無效假說被否定之最低意義標準
stat.$\hat{p}$1	第一個樣本比例估計值
stat.$\hat{p}$2	第二個樣本比例估計值
stat.$\hat{p}$	合併樣本比例估計值
stat.n1、stat.n2	試驗 1 和試驗 2 採集的樣本數

zTest_2Samp

目錄 >

zTest_2Samp σ_1,σ_2 ,*列表1*,*列表2*[,*Freq1*[,*Freq2*[,*Hypoth*]]]

(輸入資料列表)

zTest_2Samp σ_1,σ_2,$\bar{x}$*1*,*n1*,$\bar{x}$*2*,*n2*[,*Hypoth*]

(輸入統計摘要)

計算二樣本 *z* 檢定。*stat.results* 變數會儲存結果摘要(請參閱第 頁碼：134 頁)。

以 H_0:μ1 = μ2 檢定以下項目：

H_a:μ1 < μ2 時，設定 *Hypoth*<0

H_a:μ1 ≠ μ2(預設值)時，設定 *Hypoth*=0

H_a:μ1 > μ2 時，*Hypoth*>0

如要了解列表中有空元素時的影響，請
參考°ß空元素°®，頁碼：198。

輸出變數	說明
stat.z	平均值差的標準常態計算值
stat.PVal	無效假說被否定之最低意義標準
stat.x̄1, stat.x̄2	*列表1* 和 *列表2 中資料序列的平均值樣本*
stat.sx1、stat.sx2	*列表1* 和 *列表2 中資料序列的標準差樣本*
stat.n1、stat.n2	樣本大小

符號

+(加)　　⊞ 鍵

值1 + 值2⇒值

傳回兩個引數的和。

56	56
56+4	60
60+4	64
64+4	68
68+4	72

列表1 + *列表2⇒列表*

矩陣1 + *矩陣2⇒矩陣*

傳回包含*列表1*和*列表2*(或*矩陣1*和*矩陣2*) 對應元素相加之和的列表(或矩陣) 。

引數的維數必須相同。

$\left\{22,\pi,\frac{\pi}{2}\right\}\rightarrow l1$	$\{22,3.14159,1.5708\}$
$\left\{10,5,\frac{\pi}{2}\right\}\rightarrow l2$	$\{10,5,1.5708\}$
l1+l2	$\{32,8.14159,3.14159\}$

傳回列表，其元素包含*列表1*中各元素加上 *Expr* 的和。

值 + 列表1⇒列表

列表1 + *值⇒列表*

傳回列表，其元素包含*列表1*中各元素加上*值*的和。

$15+\{10,15,20\}$	$\{25,30,35\}$
$\{10,15,20\}+15$	$\{25,30,35\}$

傳回矩陣，其元素是*矩陣1*中對角線上各元素加上 *Expr* 的和。*矩陣1* 必需是方陣。

值 + 矩陣1⇒矩陣

矩陣1 + *值⇒矩陣*

傳回矩陣，其元素是*矩陣1*中對角線上各元素加上*值*的和。*矩陣1* 必需是方陣。

$20+\begin{bmatrix}1 & 2\\3 & 4\end{bmatrix}$	$\begin{bmatrix}21 & 2\\3 & 24\end{bmatrix}$

附註：請用 **.+**(點加) 讓各元素加上運算式。

−(減) 鍵

值1 − 值2⇒值

傳回*值1* 減*值2* 的差。

$6-2$	4
$\pi-\frac{\pi}{6}$	2.61799

列表1 − 列表2⇒列表

矩陣1 − 矩陣2⇒矩陣

使*列表1*(或*矩陣1*) 的各元素減掉*列表2*(或*矩陣2*) 的對應元素，然後傳回結果。

引數的維數必須相同。

$\left\{22,\pi,\frac{\pi}{2}\right\}-\left\{10,5,\frac{\pi}{2}\right\}$	$\{12,-1.85841,0.\}$
$[3\ \ 4]-[1\ \ 2]$	$[2\ \ 2]$

使 *Expr* 減掉*列表1* 的各元素，或使*列表1* 的各元素減掉 *Expr*，然後傳回結果列表。

$15-\{10,15,20\}$	$\{5,0,-5\}$
$\{10,15,20\}-15$	$\{-5,0,5\}$

值 − 列表1⇒列表

列表1 − 值⇒列表

使*值*減掉*列表1* 的各元素，或使*列表1* 的各元素減掉*值*，然後傳回結果列表。

值 − 矩陣1⇒矩陣

矩陣1 − 值⇒矩陣

值− 矩陣1 ：傳回*值*乘以單位矩陣後減掉*矩陣1* 的矩陣。 *矩陣1* 必須是方陣。

矩陣1 − 值：傳回*矩陣1* 減掉*值*和單位矩陣相乘結果的矩陣。 *矩陣1* 必須是方陣。

$20-\begin{bmatrix}1 & 2\\ 3 & 4\end{bmatrix}$	$\begin{bmatrix}19 & -2\\ -3 & 16\end{bmatrix}$

附註：請用 **.−**(點減) 使各元素減掉運算式。

·(乘) 鍵

值1 ·值2⇒值

傳回兩個引數的乘積。

$2\cdot 3.45$	6.9

列表1·列表2⇒列表

傳回列表，其元素包含*列表1* 和*列表2* 中對應元素的乘積。

列表的維數必須相同。

$\{1.,2,3\}\cdot\{4,5,6\}$	$\{4,10,18\}$

·(乘)　　× 鍵

矩陣1 ·*矩陣2*⇒*矩陣*

傳回*矩陣1* 和*矩陣2* 的矩陣乘積。

矩陣1 的行數必須等於*矩陣2* 的列數。

$$\begin{bmatrix}1 & 2 & 3\\4 & 5 & 6\end{bmatrix}.\begin{bmatrix}7 & 8\\7 & 8\\7 & 8\end{bmatrix} \qquad \begin{bmatrix}42 & 48\\105 & 120\end{bmatrix}$$

傳回列表，其元素包含 *Expr* 乘以*列表1* 各元素的乘積。

$$\pi\cdot\{4,5,6\} \qquad \{12.5664,15.708,18.8496\}$$

Value ·*列表1*⇒*列表*

列表1 ·*值*⇒*列表*

傳回列表，其元素包含*值*乘以*列表1* 各元素的乘積。

傳回矩陣，其元素包含 *Expr* 乘以*矩陣1* 各元素的乘積。

$$\begin{bmatrix}1 & 2\\3 & 4\end{bmatrix}\cdot 0.01 \qquad \begin{bmatrix}0.01 & 0.02\\0.03 & 0.04\end{bmatrix}$$

值 ·*矩陣1*⇒*矩陣*

矩陣1 ·*值*⇒*矩陣*

傳回矩陣，其元素包含*值*乘以*矩陣1* 各元素的乘積。

$$6\cdot\text{identity}(3) \qquad \begin{bmatrix}6 & 0 & 0\\0 & 6 & 0\\0 & 0 & 6\end{bmatrix}$$

附註：請用 .·(點乘) 讓運算式乘以各元素。

/(除)　　÷ 鍵

值1 / *值2*⇒*值*

傳回*值1* 除以*值2* 的商。

$$\frac{2}{3.45} \qquad 0.57971$$

附註：另請參考**分數範本**，頁碼：1。

列表1 / *列表2*⇒*列表*

傳回列表，其元素包含*列表1* 除以*列表2* 的商。

$$\frac{\{1.,2,3\}}{\{4,5,6\}} \qquad \left\{0.25,\frac{2}{5},\frac{1}{2}\right\}$$

列表的維數必須相同。

傳回列表，其元素包含 *Expr* 除以*列表1* 或*列表1* 除以 *Expr* 的商。

$$\frac{6}{\{3,6,\sqrt{6}\}} \qquad \{2,1,2.44949\}$$

值 / *列表1* ⇒ *列表*

列表1 / *值* ⇒ *列表*

傳回列表，其元素包含*值*除以*列表1* 或*列表1* 除以*值*的商。

$$\frac{\{7,9,2\}}{7\cdot 9\cdot 2} \qquad \left\{\frac{1}{18},\frac{1}{14},\frac{1}{63}\right\}$$

/ (除)

÷ 鍵

值 / 矩陣1⇒ 矩陣

矩陣1 / 值 ⇒ 矩陣

傳回矩陣，其元素包含*矩陣1/值*的商。

附註：請用 **./**(點除) 使運算式除以各元素。

$$\frac{[7\ \ 9\ \ 2]}{7\cdot 9\cdot 2} \qquad \begin{bmatrix} \frac{1}{18} & \frac{1}{14} & \frac{1}{63} \end{bmatrix}$$

^(乘冪)

^ 鍵

值1 ^ 值2 ⇒ 值

列表1 ^ 列表2 ⇒ 列表

傳回以第一個引數為底，第二個引數為冪的指數值。

附註：另請參考**指數範本**，頁碼：1。

若是列表，則傳回以*列表1* 中各元素為底，*列表2* 對應元素為冪的指數值。

在實數領域中，有奇分母的簡化指數之分數冪會使用實數分枝，複數模式則使用主枝。

$$4^2 \qquad 16$$

$$\{2,4,6\}^{\{1,2,3\}} \qquad \{2,16,216\}$$

值 ^ 列表1 ⇒ 列表

傳回以*值*為底，*列表1* 各元素為冪的指數值。

$$\pi^{\{1,2,-3\}} \qquad \{3.14159,9.8696,0.032252\}$$

列表1 ^ 值 ⇒ 列表

傳回以*列表1* 中各元素為底，*值*為冪的指數值。

$$\{1,2,3,4\}^{-2} \qquad \left\{1,\frac{1}{4},\frac{1}{9},\frac{1}{16}\right\}$$

方陣1 ^ 整數 ⇒ 矩陣

傳回以 *方陣1* 為底，*整數*為冪的指數值。

方陣1 必須是方陣。

若 *整數* = -1，則計算反矩陣。

若 *整數* < -1，則以適當的正冪次方計算反矩陣。

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}^2 \qquad \begin{bmatrix} 7 & 10 \\ 15 & 22 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}^{-1} \qquad \begin{bmatrix} -2 & 1 \\ \frac{3}{2} & \frac{-1}{2} \end{bmatrix}$$

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}^{-2} \qquad \begin{bmatrix} \frac{11}{2} & \frac{-5}{2} \\ \frac{-15}{4} & \frac{7}{4} \end{bmatrix}$$

x²(平方)

x² 鍵

值$^2 \Rightarrow$ *值*

傳回引數的平方值。

列表1$^2 \Rightarrow$ *列表*

傳回*列表1*中各元素平方值的列表。

方陣1$^2 \Rightarrow$ *矩陣*

傳回*方陣1*的矩陣平方值。這和計算各元素的平方值不同。請用 .^2 計算各元素的平方值。

4^2 → 16

$\{2,4,6\}^2$ → $\{4,16,36\}$

$\begin{bmatrix} 2 & 4 & 6 \\ 3 & 5 & 7 \\ 4 & 6 & 8 \end{bmatrix}^2$ → $\begin{bmatrix} 40 & 64 & 88 \\ 49 & 79 & 109 \\ 58 & 94 & 130 \end{bmatrix}$

$\begin{bmatrix} 2 & 4 & 6 \\ 3 & 5 & 7 \\ 4 & 6 & 8 \end{bmatrix} .^\wedge 2$ → $\begin{bmatrix} 4 & 16 & 36 \\ 9 & 25 & 49 \\ 16 & 36 & 64 \end{bmatrix}$

.+(點和)

. + 鍵

矩陣1 **.+** *矩陣2* $\Rightarrow$ *矩陣*

值 .+ *矩陣1* $\Rightarrow$ *矩陣*

矩陣1 **.+** *矩陣2*: 傳回矩陣，其元素是*矩陣1*和*矩陣2*中每組成對對應元素的和。

值 **.+**矩陣1: 傳回矩陣，其元素是值和矩陣1各元素的和。

$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} .+ \begin{bmatrix} 10 & 30 \\ 20 & 40 \end{bmatrix}$ → $\begin{bmatrix} 11 & 32 \\ 23 & 44 \end{bmatrix}$

$5 .+ \begin{bmatrix} 10 & 30 \\ 20 & 40 \end{bmatrix}$ → $\begin{bmatrix} 15 & 35 \\ 25 & 45 \end{bmatrix}$

.-(點差)

. − 鍵

矩陣1 **.−** *矩陣2* $\Rightarrow$ *矩陣*

值 .−*矩陣1* $\Rightarrow$ *矩陣*

矩陣1 **.−***矩陣2*: 傳回矩陣，其元素是*矩陣1*和*矩陣2*中每組成對對應元素的差。

值 **.−**矩陣1: 傳回矩陣，其元素是值和矩陣1各元素的差。

$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} .- \begin{bmatrix} 10 & 20 \\ 30 & 40 \end{bmatrix}$ → $\begin{bmatrix} -9 & -18 \\ -27 & -36 \end{bmatrix}$

$5 .- \begin{bmatrix} 10 & 20 \\ 30 & 40 \end{bmatrix}$ → $\begin{bmatrix} -5 & -15 \\ -25 & -35 \end{bmatrix}$

.·(點乘)

[.][×] 鍵

矩陣1 .· *矩陣2* ⇒ *矩陣*

值 .· *矩陣1* ⇒ *矩陣*

矩陣1 .· *矩陣2*：傳回矩陣，其元素是*矩陣1* 和*矩陣2* 中每組成對對應元素的乘積。

值 .· 矩陣1：傳回矩陣，其元素包含值乘以矩陣1 各元素的乘積。

$$\begin{bmatrix}1 & 2\\3 & 4\end{bmatrix} .\cdot \begin{bmatrix}10 & 20\\30 & 40\end{bmatrix} \qquad \begin{bmatrix}10 & 40\\90 & 160\end{bmatrix}$$

$$5 .\cdot \begin{bmatrix}10 & 20\\30 & 40\end{bmatrix} \qquad \begin{bmatrix}50 & 100\\150 & 200\end{bmatrix}$$

./(點除)

[.][÷] 鍵

矩陣1 ./ *矩陣2* ⇒ *矩陣*

值 ./ *矩陣1* ⇒ *矩陣*

矩陣1 ./ *矩陣2*：傳回矩陣，其元素是*矩陣1* 和*矩陣2* 中每組成對對應元素的商。

值 ./矩陣1：傳回矩陣，其元素是值乘以矩陣1 各元素的商。

$$\begin{bmatrix}1 & 2\\3 & 4\end{bmatrix} ./ \left(\begin{bmatrix}10 & 20\\30 & 40\end{bmatrix}\right) \qquad \begin{bmatrix}\frac{1}{10} & \frac{1}{10}\\\frac{1}{10} & \frac{1}{10}\end{bmatrix}$$

$$5 ./ \left(\begin{bmatrix}10 & 20\\30 & 40\end{bmatrix}\right) \qquad \begin{bmatrix}\frac{1}{2} & \frac{1}{4}\\\frac{1}{6} & \frac{1}{8}\end{bmatrix}$$

.^(點冪)

[.][^] 鍵

矩陣1 .^ *矩陣2* ⇒ *矩陣*

值 .^ *矩陣1* ⇒ *矩陣*

矩陣1 .^ *矩陣2*：傳回矩陣，使*矩陣2* 的各元素是*矩陣1* 對應元素的指數。

值 .^ *矩陣1*：傳回矩陣，使矩陣1 的各元素是值的指數。

$$\begin{bmatrix}1 & 2\\3 & 4\end{bmatrix} .^\wedge \begin{bmatrix}0 & 2\\3 & -1\end{bmatrix} \qquad \begin{bmatrix}1 & 4\\27 & \frac{1}{4}\end{bmatrix}$$

$$5 .^\wedge \begin{bmatrix}0 & 2\\3 & -1\end{bmatrix} \qquad \begin{bmatrix}1 & 25\\125 & \frac{1}{5}\end{bmatrix}$$

-(負值)

[(−)] 鍵

-*值1* ⇒ *值*

-*列表1* ⇒ *列表*

-*矩陣1* ⇒ *矩陣*

傳回引數的負值。

若是列表或矩陣，則傳回所有元素的負值。

-2.43	-2.43
-{-1,0.4,1.2E19}	{1.,-0.4,-1.2E19}

二進位基底模式：

重要訊息：數字零，而非英文字母 O

⁻(負值)

(−) 鍵

若引數是二進位或十六進位整數，則負值會提供二的補數。

```
-0b100101
0b11111111111111111111111111111111▸
```

若要看到完整結果，請按 ▲，然後使用 ◀ 與 ▶ 移動游標。

%(百分比)

ctrl 鍵

值1 % ⇒ *值*

列表1 % ⇒ *列表*

矩陣1 % ⇒ *矩陣*

	argument
傳回	100

若是列表或矩陣，則傳回各元素都除以 100 的列表或矩陣。

附註：強迫表示結果近似值，

計算機：按 ctrl enter。
Windows®：按 **Ctrl+Enter**。
Macintosh®：按 **⌘+Enter**。
iPad®：按住 **enter** 然後選擇 ≈。

13%	0.13
{{1,10,100}}%	{0.01,0.1,1.}

=(等於)

= 鍵

運算式1 = *運算式2*⇒*布林運算式*

列表1 = *列表2*⇒*布林列表*

矩陣1 = *矩陣2*⇒*布林矩陣*

若確定 *運算式1* 等於 *運算式2*，則傳回真。

若確定 *運算式1* 不等於 *運算式2*，則傳回偽。

若是其他情形，則傳回方程式的化簡形式。

若是列表和矩陣，則傳回對應元素逐一比較的結果。

輸入 範例的注意事項：關於輸入多行程式和函數定義的說明，請參閱產品手冊中的「計算工具」章節。

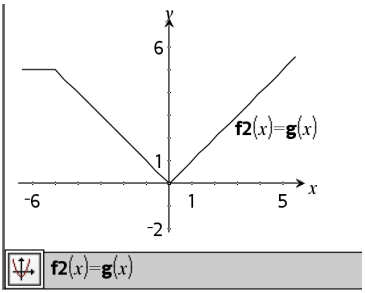
使用以下數學檢定符號的函數範例：=、≠、<、≤、>、≥

```
Define 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
                                Done
```

g(x) 繪圖結果

=(等於)

[=] 鍵



≠(不等於)

[ctrl] [=] 鍵

運算式1 ≠ *運算式2* ⇒ *布林運算式*

列表1 ≠ *列表2* ⇒ *布林列表*

矩陣1 ≠ *矩陣2* ⇒ *布林矩陣*

若確定 *運算式1* 不等於 *運算式2*，則傳回真。

若確定 *運算式1* 等於 *運算式2*，則傳回偽。

若是其他情形，則傳回方程式的化簡形式。

若是列表和矩陣，則傳回對應元素逐一比較的結果。

附註：如果要從鍵盤插入本運算子，可輸入 **/=**

請參考「=」(等於) 的範例。

< (小於)

[ctrl] [=] 鍵

運算式1 < *運算式2* ⇒ *布林運算式*

列表1 < *列表2* ⇒ *布林列表*

矩陣1 < *矩陣2* ⇒ *布林矩陣*

若確定 *運算式1* 小於 *運算式2*，則傳回真。

若確定 *運算式1* 大於或等於 *運算式2*，則傳回偽。

請參考「=」(等於) 的範例。

< (小於)

ctrl = 鍵

若是其他情形，則傳回方程式的化簡形式。

若是列表和矩陣，則傳回對應元素逐一比較的結果。

≤(小於或等於)

ctrl = 鍵

運算式1 ≤ *運算式2* ⇒ *布林運算式*

列表1 ≤ *列表2* ⇒ *布林列表*

矩陣1 ≤ *矩陣2* ⇒ *布林矩陣*

若確定 *運算式1* 小於或等於 *運算式2*，則傳回真。

若確定 *運算式1* 大於 *運算式2*，則傳回偽。

若是其他情形，則傳回方程式的化簡形式。

若是列表和矩陣，則傳回對應元素逐一比較的結果。

附註：如果要從鍵盤插入本運算子，可輸入 **<=**

請參考「=」(等於) 的範例。

> (大於)

ctrl = 鍵

運算式1 > *運算式2* ⇒ *布林運算式*

列表1 > *列表2* ⇒ *布林列表*

矩陣1 > *矩陣2* ⇒ *布林矩陣*

若確定 *運算式1* 大於 *運算式2*，則傳回真。

若確定 *運算式1* 小於或等於 *運算式2*，則傳回偽。

若是其他情形，則傳回方程式的化簡形式。

若是列表和矩陣，則傳回對應元素逐一比較的結果。

請參考「=」(等於) 的範例。

≥(大於或等於)

ctrl = 鍵

運算式1 ≥ *運算式2* ⇒ *布林運算式*

列表1 ≥ *列表2* ⇒ *布林列表*

矩陣1 ≥ *矩陣2* ⇒ *布林矩陣*

若確定 *運算式1* 大於或等於 *運算式2*，則傳回真。

若確定 *運算式1* 小於 *運算式2*，則傳回偽。

若是其他情形，則傳回方程式的化簡形式。

若是列表和矩陣，則傳回對應元素逐一比較的結果。

附註：如果要從鍵盤插入本運算子，可輸入 **>=**

請參考「=」(等於) 的範例。

⇒(邏輯隱含)

ctrl = 鍵

布林運算式1 ⇒ *布林運算式2* 傳回 *布林運算式*

布林列表1 ⇒ *布林列表2* 傳回 *布林列表*

布林矩陣1 ⇒ *布林矩陣2* 傳回 *布林矩陣*

整數1 ⇒ *整數2* 傳回 *整數*

計算運算式 **not** <引數1> **or** <引數2> 並傳回真偽值或方程式的化簡形式。

若是列表和矩陣，則傳回對應元素逐一比較的結果。

附註：如果要從鍵盤插入本運算子，可輸入 **=>**

5>3 or 3>5	true
5>3 ⇒ 3>5	false
3 or 4	7
3 ⇒ 4	-4
{1,2,3} or {3,2,1}	{3,2,3}
{1,2,3} ⇒ {3,2,1}	{-1,-1,-3}

⇔(邏輯雙隱含，XNOR)

ctrl = 鍵

布林運算式1 ⇔ *布林運算式2* 傳回 *布林運算式*

布林列表1 ⇔ *布林列表2* 傳回 *布林列表*

布林矩陣1 ⇔ *布林矩陣2* 傳回 *布林矩陣*

整數1 ⇔ *整數2* 傳回 *整數*

5>3 xor 3>5	true
5>3 ⇔ 3>5	false
3 xor 4	7
3 ⇔ 4	-8
{1,2,3} xor {3,2,1}	{2,0,2}
{1,2,3} ⇔ {3,2,1}	{-3,-1,-3}

傳回兩個引數 **XOR** 布林運算的負值。傳回真偽值或方程式的化簡形式。

若是列表和矩陣，則傳回對應元素逐一比較的結果。

附註：如果要從鍵盤插入本運算子，可輸入**<=>**

!(階乘)

?!▸ 鍵

值1!⇒ *值*

列表1!⇒ *列表*

矩陣1!⇒ *矩陣*

傳回引數的階乘。

若是列表或矩陣，則傳回由各元素階乘所組成的列表或矩陣。

5!	120
{{5,4,3}}!	{120,24,6}
$\left(\begin{bmatrix}1 & 2\\3 & 4\end{bmatrix}\right)!$	$\begin{bmatrix}1 & 2\\6 & 24\end{bmatrix}$

&(接上)

ctrl 鍵

字串1 **&** *字串2* ⇒ *字串*

傳回將*字串2*接上*字串1*的文字字串。

"Hello "&"Nick"	"Hello Nick"

d()(導數)

目錄 >

d(*運算式1*, *Var*[, *階數*]) | *Var*=*值*⇒*值*

d(*運算式1*, *Var*[, *階數*])⇒*值*

d(*列表1*, *Var*[, *階數*])⇒*列表*

d(*矩陣1*, *Var*[, *階數*])⇒*矩陣*

除非使用第一個語法，否則必須先在 *Var* 變數中儲存數值，才能對 **d()** 求值。請參考範例。

d() 可利用自動微分法計算某個點的一階和二階數值導數。

階數(若加入) 必須等於 **1** 或 **2**。預設值是 **1**。

附註：如果要從鍵盤插入本函數，可輸入 **`derivative(…)`**。

附註：另請參考第 頁碼：5 頁的一階導數或第 頁碼：5 頁的二階導數。

$$\frac{d}{dx}(|x|)|x=0 \qquad \text{undef}$$

$$x:=0:\frac{d}{dx}(|x|) \qquad \text{undef}$$

$$x:=3:\frac{d}{dx}\left(\left\{x^2,x^3,x^4\right\}\right) \qquad \{6,27,108\}$$

附註：d() 運算法有其限制，亦即必須透過未化簡的運算式遞迴執行、計算一階和二階(視情況) 導數的數值，並且對每個子運算式求值，可能導致意外結果。

請研究右邊的範例。x·(x^2+x)^(1/3) 的一階導數在 x=0 處等於 0。但是由於 (x^2+x)^(1/3) 子運算式的一階導數在 x=0 處尚未定義，而且此值必須用於計算整個運算式的導數，所以 **d()** 會回報結果是未定義，而且顯示警告訊息。

若您碰到這項限制，請用圖形確認這個解。您也可以嘗試使用 **centralDiff()**。

$$\frac{d}{dx}\left(x\cdot\left(x^2+x\right)^{\frac{1}{3}}\right)|x=0 \qquad \text{undef}$$

$$\text{centralDiff}\left(x\cdot\left(x^2+x\right)^{\frac{1}{3}},x\right)|x=0 \qquad 0.000033$$

∫()(積分)

目錄 >

∫(*運算式1*, *Var*, *下限*, *上限*) ⇒ *值*

傳回 *運算式1* 對於 *Var* 變數從 *下限* 到 *上限* 的積分。可利用與 nInt() 相同的方法，執行定積分的數值計算作業。

附註：如果要從鍵盤插入本函數，可輸入 **`integral(…)`**。

$$\int_0^1 x^2\,dx \qquad 0.333333$$

∫()(積分)

目錄 > 📖

附註：另請參考第 頁碼：94 頁的 **nInt()** 和第 頁碼：6 頁的**定積分範本**。

√()(平方根)

ctrl x^2 鍵

√ (*值 1*)⇒*值*

√ (*列表 1*)⇒*列表*

傳回引數的平方根。

若是列表，則傳回*列表*1 中各元素的平方根。

附註：如果要從鍵盤插入本函數，可輸入 `sqrt(...)`

附註：另請參考**平方根範本**，頁碼：1。

$\sqrt{4}$ → 2

$\sqrt{\{9,2,4\}}$ → $\{3,1.41421,2\}$

Π() (prodSeq)

目錄 > 📖

Π(*運算式 1*, *Var*, *低*, *高*)⇒*運算式*

附註：如果要從鍵盤插入本函數，可輸入 `prodSeq(...)`。

計算 *運算式 1* 在 *Var* 從*低*到*高*取值時的結果，並傳回結果的乘積。

附註：另請參考**乘積範本 (Π)**，頁碼：5。

$\prod_{n=1}^{5}\left(\frac{1}{n}\right)$ → $\frac{1}{120}$

$\prod_{n=1}^{5}\left(\left\{\frac{1}{n},n,2\right\}\right)$ → $\left\{\frac{1}{120},120,32\right\}$

Π(*運算式 1*, *Var*, *低*, *低*−1)⇒1

Π(*運算式 1*, *Var*, *低*, *高*) ⇒1/Π(*運算式 1*, *Var*, *高*+*1*, *低*−*1*) if *高* < *低*−*1*

$\prod_{k=4}^{3}(k)$ → 1

所用的乘積公式乃由以下參考資料導出：

Ronald L. Graham, Donald E. Knuth, and Oren Patashnik.Concrete Mathematics:A Foundation for Computer Science.Reading, Massachusetts:Addison-Wesley, 1994.

$\prod_{k=4}^{1}\left(\frac{1}{k}\right)$ → 6

$\prod_{k=4}^{1}\left(\frac{1}{k}\right)\cdot\prod_{k=2}^{4}\left(\frac{1}{k}\right)$ → $\frac{1}{4}$

Σ() (sumSeq)

目錄 > 📖

Σ(*運算式1*, *Var*, *低*, *高***)⇒***運算式*

附註：如果要從鍵盤插入本函數，可輸入 **sumSeq(…)**。

計算 *運算式1* 在 *Var* 從 *低* 到 *高* 取值時的結果，並傳回結果的總和。

附註：另請參考**總和範本**，頁碼：5。

$$\sum_{n=1}^{5}\left(\frac{1}{n}\right) \qquad \frac{137}{60}$$

Σ(*運算式1*, *Var*, *低*, *低−0***)⇒1**

Σ(*運算式1*, *Var*, *低*, *高***) ⇒-Σ(***運算式1*, *Var*, *高+1*, *低−***1) if** *高* < *低−1*

$$\sum_{k=4}^{3}(k) \qquad 0$$

所用的加總法公式乃由以下參考資料導出：

Ronald L. Graham, Donald E. Knuth, and Oren Patashnik.Concrete Mathematics:A Foundation for Computer Science.Reading, Massachusetts:Addison-Wesley, 1994.

$$\sum_{k=4}^{1}(k) \qquad -5$$

$$\sum_{k=4}^{1}(k)+\sum_{k=2}^{4}(k) \qquad 4$$

ΣInt()

目錄 > 📖

ΣInt(*NPmt1*, *NPmt2*, *N*, *I*, *PV* ,[*Pmt*], [*FV*], [*PpY*], [*CpY*], [*PmtAt*], [*四捨五入值*])⇒*值*

ΣInt(1,3,12,4.75,20000,,12,12) -213.48

ΣInt(*NPmt1*,*NPmt2*,*amortTable***)⇒***值*

針對指定付款期數範圍，計算利息總和的攤銷函數。

NPmt1 與 *NPmt2* 用於界定付款期數範圍的起點和終點。

N,*I*,*PV*,*Pmt*,*FV*,*PpY*,*CpY*,*PmtAt* 的說明位於貨幣時間價值引數表(第頁碼：148頁)。

- 如省略 *Pmt*，則會以 *Pmt*=**tvmPmt(***N*,*I*,*PV*,*FV*,*PpY*,*CpY*,*PmtAt***)** 為預設值。
- 如省略 *FV*，則會以 *FV*=0 為預設值。
- *PpY*、*CpY*、*PmtAt* 的預設值和貨幣

tbl:=amortTbl(12,12,4.75,20000,,12,12)

0	0.	0.	20000.
1	-77.49	-1632.43	18367.6
2	-71.17	-1638.75	16728.8
3	-64.82	-1645.1	15083.7
4	-58.44	-1651.48	13432.2
5	-52.05	-1657.87	11774.4
6	-45.62	-1664.3	10110.1
7	-39.17	-1670.75	8439.32
8	-32.7	-1677.22	6762.1
9	-26.2	-1683.72	5078.38
10	-19.68	-1690.24	3388.14
11	-13.13	-1696.79	1691.35
12	-6.55	-1703.37	-12.02

ΣInt(1,3,*tbl*) -213.48

ΣInt()

時間價值函數相同。

*四捨五入值*用於指定四捨五入的小數位數。預設值 = 2。

ΣInt(*NPmt1,NPmt2,amortTable***)** 可根據 *amortTable* 攤銷表計算利息總和。*amortTable* 引數必須是 **amortTbl()** 一節所描述的矩陣形式(第 頁碼: 7 頁)。

附註: 另請參考下一節 ΣPrn() 和 **Bal()** (頁碼: 15)。

ΣPrn()

ΣPrn(*NPmt1*, *NPmt2*, *N*, *I*, *PV*, [*Pmt*], [*FV*], [*PpY*], [*CpY*], [*PmtAt*], [*四捨五入*])⇒*值*

ΣPrn(*NPmt1,NPmt2,amortTable***)**⇒*值*

針對指定付款期數範圍，計算本金總和的攤銷函數。

NPmt1 與 *NPmt2* 用於界定付款期數範圍的起點和終點。

N,I,PV,Pmt,FV,PpY,CpY,PmtAt 的說明位於貨幣時間價值引數表(第 頁碼: 148 頁)。

- 如省略 *Pmt*，則會以 *Pmt*=**tvmPmt (***N,I,PV,FV,PpY,CpY,PmtAt***)** 為預設值。
- 如省略 *FV*，則會以 *FV*=0 為預設值。
- *PpY*、*CpY*、*PmtAt* 的預設值和貨幣時間價值函數相同。

*四捨五入值*用於指定四捨五入的小數位數。預設值 = 2。

ΣPrn(*NPmt1,NPmt2,amortTable***)** 可根據 *amortTable* 攤銷表計算已付本金總和。*amortTable* 引數必須是 **amortTbl()** 一節所描述的矩陣形式(第 頁碼: 7 頁)。

附註: 另請參考上一節 ΣInt() 和 **Bal()** (頁碼: 15)。

ΣPrn(1,3,12,4.75,20000,,12,12) -4916.28

tbl:=amortTbl(12,12,4.75,20000,,12,12)

0	0.	0.	20000.
1	-77.49	-1632.43	18367.57
2	-71.17	-1638.75	16728.82
3	-64.82	-1645.1	15083.72
4	-58.44	-1651.48	13432.24
5	-52.05	-1657.87	11774.37
6	-45.62	-1664.3	10110.07
7	-39.17	-1670.75	8439.32
8	-32.7	-1677.22	6762.1
9	-26.2	-1683.72	5078.38
10	-19.68	-1690.24	3388.14
11	-13.13	-1696.79	1691.35
12	-6.55	-1703.37	-12.02

ΣPrn(1,3,*tbl*) -4916.28

#(間接)

ctrl 鍵

*var名稱字串*

參照名稱是 *var名稱字串*的變數。本功能讓您在函數中用字串建立變數名稱。

xyz:=12	12
#("x"&"y"&"z")	12

建立或參照 xyz 變數。

10→*r*	10
"r"→*s1*	"r"
#*s1*	10

傳回名稱儲存在 s1 變數中的變數(r) 值。

E(科學記號)

EE 鍵

*尾數***E***指數*

以科學記號輸入數字。系統會將數字解譯為*尾數* × 10指數。

提示：若要輸入 10 的次方卻不要結果是十進位值，請用 10^*整數*。

附註：如果要從電腦鍵盤插入本運算子，可輸入 **@E**。例如輸入 **2.3@E4** 即可得到 2.3E4。

23000.	23000.
2300000000.+4.1E15	4.1E15
$3\cdot10^4$	30000

g(梯度)

1 鍵

運算式 1g⇒運算式

*列表 1***g***⇒列表*

*矩陣 1***g***⇒矩陣*

本函數讓您在度數角或弧度角模式下指定梯度角。

若是弧度角模式，則將 *運算式1* 乘以 π/200。

若是度數角模式，則將 *運算式1* 乘以 g/100。

若是梯度角模式，則傳回 *運算式1* 不變。

度數角、梯度角或弧度角模式：

cos(50g)	0.707107
cos({0,100g,200g})	{1,0.,-1.}

g(梯度) 1 鍵

附註：如果要從電腦鍵盤插入本符號，可輸入 **@g**。

ʳ(弧度) 1 鍵

值 1ʳ⇒值

列表 1ʳ⇒列表

矩陣 1ʳ⇒矩陣

本函數讓您在度數角或梯度角模式下指定弧度角。

若是度數角模式，則將引數乘以 180/π。

若是弧度角模式，則傳回引數不變。

若是梯度角模式，則將引數乘以 200/π。

提示：若無論使用函數時所處模式為何，都要在函數定義中強迫採用弧度角，請用 ʳ。

附註：如果要從電腦鍵盤插入本符號，可輸入 **@r**。

度數角、梯度角或弧度角模式：

$\cos\left(\frac{\pi}{4^r}\right)$ 0.707107

$\cos\left(\left\{0^r,\left(\frac{\pi}{12}\right)^r,-(\pi)^r\right\}\right)$ $\{1,0.965926,-1.\}$

° (度數) 1 鍵

值 1°⇒值

列表 1°⇒列表

矩陣 1°⇒矩陣

本函數讓您在梯度角或弧度角模式下指定度數角。

若是弧度角模式，則將引數乘以 π/180。

若是度數角模式，則傳回引數不變。

若是梯度角模式，則將引數乘以 10/9。

附註：如果要從電腦鍵盤插入本符號，可輸入 **@d**。

度數角、梯度角或弧度角模式：

$\cos(45°)$ 0.707107

弧度角模式：

$\cos\left(\left\{0,\frac{\pi}{4},90°,30.12°\right\}\right)$

$\{1.,0.707107,0.,0.864976\}$

°、'、"(度/分/秒)

ctrl 鍵

dd°*mm*'*ss.ss*''⇒*運算式*

*dd*正數或負數

*mm*不可是負數

*ss.ss*不可是負數

傳回 *dd*+(*mm*/60)+(*ss.ss*/3600)。

這個 -60 進位輸入格式有以下功能：

- 可輸入「度/分/秒」格式的角度，無須理會目前的角度模式。
- 可輸入「時/分/秒」格式的時間。

附註：ss.ss 後面接的是兩個單引號 ('') ，而非雙引號(") 。

度數角模式：

25°13'17.5"	25.2215
25°30'	$\frac{51}{2}$

∠(角度)

ctrl 鍵

[*弧度*,∠θ_*角度*]⇒*向量*

(極座標輸入)

[*弧度*,∠θ_*角度*,*Z_座標*]⇒*向量*

(圓柱座標輸入)

[*弧度*,∠θ_*角度*,∠θ_*角度*]⇒*向量*

(球面座標輸入)

根據直角、圓柱、球面的向量格式模式，傳回向量形式的座標。

附註：如果要從電腦鍵盤插入本符號，可輸入 @<。

在弧度角模式將向量格式設定成：

直角

$[5\ \angle 60°\ \angle 45°]$

$[1.76777\quad 3.06186\quad 3.53553]$

圓柱

$[5\ \angle 60°\ \angle 45°]$

$[3.53553\quad \angle 1.0472\quad 3.53553]$

球面

$[5\ \angle 60°\ \angle 45°]$

$[5.\quad \angle 1.0472\quad \angle 0.785398]$

(*量* ∠ *角度*)⇒*複數值*

(極座標輸入)

以 (r∠θ) 極座標形式輸入複數值。系統會以目前的角度模式設定解譯*角度*。

弧度角模式與直角複數格式：

$5+3\cdot i-\left(10\ \angle\frac{\pi}{4}\right)$ $\quad -2.07107-4.07107\cdot i$

_(代表空元素的底線)

請參考 °ß空元素°®，頁碼：198。

10^()

目錄 >

10^ (*值 1***)⇒***值*

10^ (*列表 1***)⇒***列表*

傳回以 10 為底，以引數為冪的指數值。

若是列表，則傳回以 10 為底，以*列表 1* 中各元素為冪的指數值。

$10^{1.5}$ → 31.6228

10^(*方陣 1***)⇒***方陣*

傳回以 10 為底，以*方陣 1* 為冪的指數值。這和計算以 10 為底、以各元素為冪的指數值不同。若要了解計算方式，請參考 **cos()**。

方陣 1 必需可對角化。結果一定會包含浮點數。

$$10^{\begin{bmatrix}1&5&3\\4&2&1\\6&-2&1\end{bmatrix}}$$

$$\begin{bmatrix}1.14336\text{E}7 & 8.17155\text{E}6 & 6.67589\text{E}6\\9.95651\text{E}6 & 7.11587\text{E}6 & 5.81342\text{E}6\\7.65298\text{E}6 & 5.46952\text{E}6 & 4.46845\text{E}6\end{bmatrix}$$

^-1(倒數)

目錄 >

值 1 **^-1⇒***值*

列表 1 **^-1⇒***列表*

傳回引數的倒數。

若是列表，則傳回*列表 1* 中各元素的倒數。

$(3.1)^{-1}$ → 0.322581

方陣 1 **^-1⇒***方陣*

傳回*方陣 1* 的反矩陣。

方陣 1 不得是奇異方陣。

$$\begin{bmatrix}1&2\\3&4\end{bmatrix}^{-1} \rightarrow \begin{bmatrix}-2&1\\\frac{3}{2}&\frac{-1}{2}\end{bmatrix}$$

|(約束運算子)

ctrl 鍵

運算式 **|** *布林運算式 1* [**and***布林運算式 2*]...

運算式 **|** *布林運算式 1* [**or***布林運算式 2*]...

$x+1|x=3$ → 4

$x+55|x=\sin(55)$ → 54.0002

|(約束運算子)

ctrl 鍵

(「|」) 符號可充當二元運算子。| 左邊的運算元是運算式。| 右邊的運算元可用於指定一個以上的關係，目的是決定運算式的化簡方法。| 後的多種關係必須由「**and**」or「**or**」邏輯連接。

約束運算子提供三種基本功能：

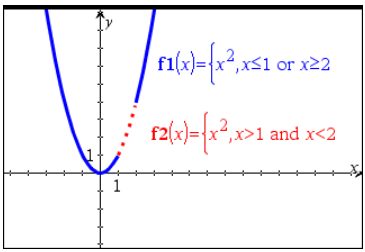
- 代換
- 區間限制
- 排除

代換的形式是等式，例如 x=3 或 y=sin (x)。為提高效果，左側應該是簡單變數。*Expr* | *變數* = *值*：將*值*代入 *Expr* 中的每個*變數*。

$x^3-2\cdot x+7\rightarrow f(x)$	*Done*
$f(x)\|x=\sqrt{3}$	8.73205

區間約束的形式為「**and**」or「**or**」邏輯運算子連接的一個以上不等式。用其他方式可能化簡無效或無法計算時，區間約束法也能加以化簡。

$\text{nSolve}(x^3+2\cdot x^2-15\cdot x=0,x)$	0.
$\text{nSolve}(x^3+2\cdot x^2-15\cdot x=0,x)\|x>0 \text{ and } x<5$	3.



排除功能會以「不等於」(/= 或 ≠) 關係運算子，在計算期間排除特定值。

→(儲存)

ctrl var 鍵

值 → *Var*

列表 → *Var*

矩陣 → *Var*

Expr → *函數*(*Param1*,...)

列表 → *函數*(*Param1*,...)

矩陣 → *函數*(*Param1*,...)

$\frac{\pi}{4}\rightarrow myvar$	0.785398
$2\cdot\cos(x)\rightarrow y1(x)$	*Done*
$\{1,2,3,4\}\rightarrow lst5$	$\{1,2,3,4\}$
$\begin{bmatrix}1&2&3\\4&5&6\end{bmatrix}\rightarrow matg$	$\begin{bmatrix}1&2&3\\4&5&6\end{bmatrix}$
"Hello" → *str1*	"Hello"

→(儲存)

ctrl var 鍵

若 *Var* 變數不存在，則自動建立並賦予初值，以成為*值*、*列表*或*矩陣*。

若 *Var* 變數已經存在，而且並未遭到鎖定或保護，則以*值*、*列表*或*矩陣*取代其值。

附註：如果要從鍵盤插入本運算子，可輸入 **=:**當作快速鍵。例如輸入 **pi/4 =: myvar**。

:=(賦值)

ctrl 鍵

Var := *值*

Var **:=** *列表*

Var **:=** *矩陣*

函數(*Param1*,...)**:=** *Expr*

函數(*Param1*,...)**:=** *列表*

函數(*Param1*,...)**:=** *矩陣*

若 *Var* 變數不存在，則自動建立 *Var* 並賦予初值，以成為*值*、*列表*或*矩陣*。

若 *Var* 已經存在，而且並未遭到鎖定或保護，則以*值*、*列表*或*矩陣*取代其值。

$myvar:=\frac{\pi}{4}$	.785398
$y1(x):=2\cdot\cos(x)$	*Done*
$lst5:=\{1,2,3,4\}$	$\{1,2,3,4\}$
$matg:=\begin{bmatrix}1&2&3\\4&5&6\end{bmatrix}$	$\begin{bmatrix}1&2&3\\4&5&6\end{bmatrix}$
str1:="Hello"	"Hello"

©(註解)

ctrl 鍵

© [文字]

© 符號可將文字視為註釋行處理，讓您為自己建立的函數和程式加上註釋。

© 可放在指令行開頭或任何位置。c 符號右邊到行尾的所有內容都是註解。

輸入 範例的注意事項：關於輸入多行程式和函數定義的說明，請參閱產品手冊中的「計算工具」章節。

```
Define g(n)=Func
    © Declare variables
    Local i,result
    result:=0
    For i,1,n,1 ©Loop n times
    result:=result+i^2
    EndFor
    Return result
    EndFunc
                                Done
g(3)                            14
```

0b *二進位數字*

0h *十六進位數字*

分別代表二進位和十六進位數字。若要輸入二進位或十六進位數字，無論使用哪種基底模式，前面一定要輸入 0b 或 0h 字首。如果沒有加上字首，就會把數字視為十進位(以 10 為底) 。

所顯示的結果會以基底模式為準。

十進位基底模式：

0b10+0hF+10	27

二進位基底模式：

0b10+0hF+10	0b11011

十六進位基底模式：

0b10+0hF+10	0h1B

TI-Nspire™ CX II - 繪圖指令

此為《TI-Nspire™ 參考手冊》和《TI-Nspire™ CAS 參考手冊》的補充文件。所有 TI-Nspire™CX II 指令都將納入並在《TI-Nspire™ 參考手冊》5.1 版和《TI-Nspire™ CAS 參考手冊》中發佈。

函數繪圖程式設計

新指令已新增至 TI-Nspire™ CX II 計算機和 TI-Nspire™ 桌面應用程式，專用於函數繪圖程式設計。

TI-Nspire™ CX II 計算機將在執行函數繪圖指令時切換到此函數繪圖模式，並會在程式完成後切換回程序執行內容。

執行程式時，畫面會在頂端列上顯示「執行中...」。程式完成後，則顯示「完成」。按下任一按鍵均可使系統退出函數繪圖模式。

- 若在執行 TI-Basic 程式期間遇到其中一個「繪圖」(函數繪圖) 指令，系統便會自動觸發函數繪圖模式轉換功能。
- 只有在從計算工具，在速寫板的文件或計算工具中執行程式時，才會發生模式轉換。
- 程式終止後，即會退出函數繪圖模式。
- 只有 TI-Nspire™ CX II 計算機和桌面 TI-Nspire™ CX II 計算機檢視才可使用函數繪圖模式。這表示桌面或 iOS 上的電腦文件檢視或 PublishView (.tnsp) 均無法使用此模式。
  - 如果從錯誤的內容執行 TI-Basic 程式時遇到函數繪圖指令，系統將會顯示錯誤訊息並終止 TI-Basic 程式。

函數繪圖畫面

函數繪圖畫面頂端將會包含一個標頭，且該標頭無法由函數繪圖指令寫入。

初始化函數繪圖畫面時，系統將會清除函數繪圖畫面的繪製區域(色彩 = 255,255,255) 。

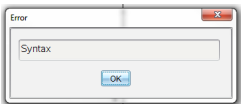
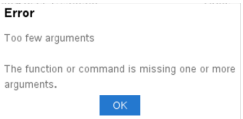
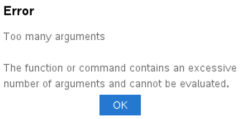
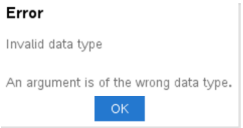
函數繪圖畫面	預設值
高度	212
寬度	318
顏色	白色:255,255,255

預設檢視與設定

- 執行函數繪圖程式時，系統不會顯示頂端列上的狀態圖示(電池狀態、隨按即試狀態，以及網路指示條等) 。
- 預設繪製色彩:黑色 (0,0,0)
- 預設畫筆樣式 - 標準、平滑
  - 粗細:1(細) 、2(標準) 、3(最粗)
  - 樣式:1(平滑) 、2(點線) 、3(虛線)
- 所有繪製指令均會使用目前的色彩和畫筆設定;對於預設值或透過 TI-Basic 指令設定的值也是如此。
- 文字字型已固定，無法變更。
- 函數繪圖畫面上的任何輸出都將在裁剪視窗內繪製，裁剪視窗大小與函數繪圖畫面的繪製區域大小相同。對於超出此裁剪函數繪圖畫面繪製區域以外的任何繪製輸出，系統均不會進行繪製。同時亦不會顯示錯誤訊息。
- 對於為繪製指令指定的所有 x,y 座標，系統均會作出定義，因此 0,0 將位於函數繪圖畫面繪製區域的左上角。
  - **例外情況:**
    - **DrawText** 使用座標作為文字邊界框的左下角。
    - **SetWindow** 使用畫面的左下角
- 指令的所有參數均能以運算式提供，得出數字求值後，系統會將其四捨五入到最接近的整數。

函數繪圖畫面錯誤訊息

若驗證失敗，系統會顯示錯誤訊息。

錯誤訊息	說明	檢視
錯誤 語法	如果語法檢查器發現任何語法錯誤，會顯示一個錯誤訊息，並嘗試將游標放在第一個錯誤附近，讓您能夠進行更正。	
錯誤 引數太少	函數或指令缺少一個或多個引數	
錯誤 引數太多	函數或指令包含過多引數，無法求值。	
錯誤 資料類型無效	引數的資料類型錯誤。	

函數繪圖模式中的無效指令

當程式切換到函數繪圖模式時，某些指令將無法使用。如果在函數繪圖模式中遇到這些指令，系統將會顯示錯誤訊息並終止程式。

不允許的指令	錯誤訊息
Request	Request 不可以在函數繪圖模式中執行
RequestStr	RequestStr 不可以在函數繪圖模式中執行
文字	Text 不可以在函數繪圖模式中執行

用於將文字列印到計算工具的指令(**disp** 和 **dispAt**) 將在函數繪圖內容中受到支援。系統會將這些指令中的文字傳送到計算工具畫面(而非函數繪圖) ，並在程式結束以及系統切換回計算工具應用程式後予以顯示

清除

清除 *x, y, 寬度, 高度*

若未指定參數，系統將清除整個畫面。

若已指定 *x*、*y*、*寬度*和*高度*，系統將會清除由參數定義的矩形區域。

清除

清除整個畫面

清除 10,10,100,50

清除左上角 (10, 10) 以及寬度為 100、高度為 50 的矩形區域

DrawArc

目錄 > 📖
CXII

DrawArc *x, y, 寬度, 高度, 起始角度, 弧角*

使用所提供的起始角度和弧角，在定義的週框內繪製圓弧。

x, y:週框的左上角座標

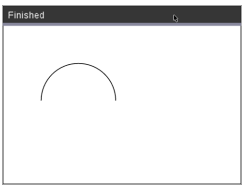
寬度, 高度:週框的維度

「弧角」用於定義圓弧的範圍。

這些參數均能以運算式提供，得出數字求值後，系統會將其四捨五入到最接近的整數。

```
DrawArc 20,20,100,100,0,90
```

```
DrawArc 50,50,100,100,0,180
```



另請參考: FillArc

DrawCircle

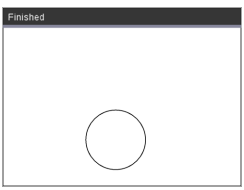
目錄 > 📖
CXII

DrawCircle *x, y, 半徑*

x, y:中心座標

半徑:圓的半徑

```
DrawCircle 150,150,40
```



另請參考: FillCircle

DrawLine

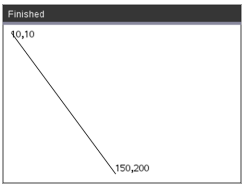
DrawLine *x1, y1, x2, y2*

從 *x1, y1, x2, y2* 繪製一條線條。

對運算式得出數字求值後，系統會將其四捨五入到最接近的整數。

畫面界限:如果指定的座標導致任何線條部分在繪製時超出函數繪圖畫面，則系統將會剪切超出的部分，而不會顯示錯誤訊息。

```
DrawLine 10,10,150,200
```



DrawPoly

指令分為以下兩種:

DrawPoly *xlist, ylist*

或

DrawPoly *x1, y1, x2, y2, x3, y3...xn, yn*

注意: DrawPoly *xlist, ylist*
作圖時，系統會將 *x1, y1* 連接至 *x2, y2*，並將 *x2, y2* 連接至 *x3, y3*，依此類推。

注意: DrawPoly *x1, y1, x2, y2, x3, y3...xn, yn*
xn, yn 將**不會**自動連接至 *x1, y1*。

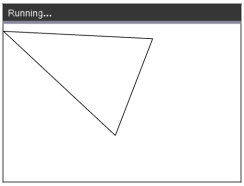
求值為系列實數浮點數的運算式 *xlist, ylist*

求值為單一實數浮點數的運算式 *x1, y1...xn, yn* = 多邊形頂點的座標

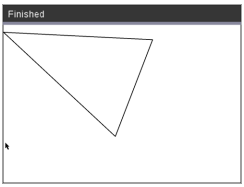
注意:DrawPoly:相對於繪製線條的輸入大小維度(寬度/高度)。
系統會在圈定指定座標和維度的邊界框中繪製線條，如此一來，所繪製的多邊形的實際大小便會大於已有的寬高值。

另請參考: FillPoly

```
xlist:={0,200,150,0}
ylist:={10,20,150,10}
DrawPoly xlist,ylist
```




```
DrawPoly 0,10,200,20,150,150,0,10
```



DrawRect

DrawRect *x, y, 寬度, 高度*

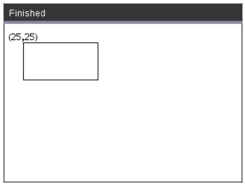
x, y:矩形的左上角座標

寬度, *高度*:矩形的寬度與高度(從起始座標往右下方繪製的矩形)。

注意:系統會在圈定指定座標和維度的邊界框中繪製線條，如此一來，所繪製的矩形的實際大小便會大於指定的寬度和高度。

另請參考: FillRect

```
DrawRect 25,25,100,50
```



DrawText

DrawText *x, y, exprOrString1 [,exprOrString2]...*

x, *y*:文字輸出的座標

在指定 *x*, *y* 座標位置的 *exprOrString* 內繪製文字。

exprOrString 與 **Disp** 的適用規則相同 – **DrawText** 可使用多個引數。

```
DrawText 50,50,"Hello World"
```

FillArc

目錄 > CXII

FillArc *x, y, 寬度, 高度, 起始角度, 弧角*

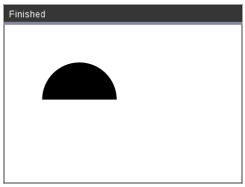
x, y:週框的左上角座標

使用所提供的起始角度和弧角，在定義的週框內繪製並填滿圓弧。

預設填滿色彩為黑色。可透過 SetColor 指令設定填滿色彩

「弧角」用於定義圓弧的範圍

```
FillArc 50,50,100,100,0,180
```



FillCircle

目錄 > CXII

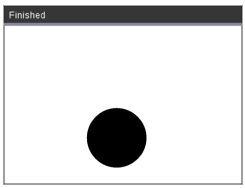
FillCircle *x, y, 半徑*

x, y:中心座標

在指定中心以指定半徑繪製並填滿圓。

預設填滿色彩為黑色。可透過 SetColor 指令設定填滿色彩。

```
FillCircle 150,150,40
```




```
Here!
```

FillPoly

目錄 > CXII

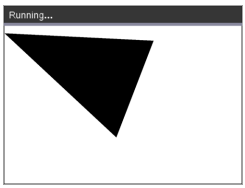
FillPoly *xlist, ylist*

或

FillPoly *x1, y1, x2, y2, x3, y3...xn, yn*

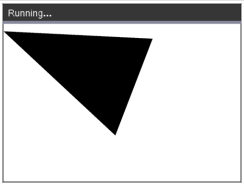
注意:線條與色彩由 SetColor 和 SetPen 指定

```
xlist:={0,200,150,0}
ylist:={10,20,150,10}
FillPoly xlist,ylist
```




```
FillPoly 0,10,200,20,150,150,0,10
```

FillPoly



FillRect

FillRect *x, y, 寬度, 高度*

x, y:矩形的左上角座標

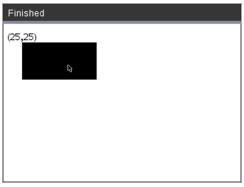
寬度, 高度:矩形的寬度與高度

在 (*x, y*) 指定的座標處繪製並填滿左上角的矩形

預設填滿色彩為黑色。可透過 SetColor 指令設定填滿色彩

注意:線條與色彩由 SetColor 和 SetPen 指定

```
FillRect 25,25,100,50
```



getPlatform()

getPlatform()

```
getPlatform()                    "dt"
```

傳回結果如下:
桌面軟體應用程式:"dt"
TI-Nspire™ CX 計算機:"hh"
TI-Nspire™ CX iPad® 應用程式:"ios"

PaintBuffer

PaintBuffer

在螢幕上繪製函數繪圖緩衝區

此指令與 UseBuffer 搭配使用，可在程式產生多個圖形物件時提高螢幕的顯示速度。

```
UseBuffer
For n,1,10
x:=randInt(0,300)
y:=randInt(0,200)
radius:=randInt(10,50)
Wait 0.5
DrawCircle x, y, 半徑
EndFor
PaintBuffer
```

此程式將一次顯示全部結果，即 10 個圓。

如果移除「UseBuffer」指令，則系統會在繪製每個圓時即時顯示。

另請參考: UseBuffer

PlotXY *x, y, 形狀*

x, y:用於繪製形狀的座標

形狀:介於 1 和 13 之間的數字，用於指定形狀

1 - 填滿的圓

2 - 空心圓

3 - 填滿的方形

4 - 空心方形

5 - 交叉

6 - 加號

7 - 細線

8 - 中型實心點

9 - 中型空心點

10 - 大實心點

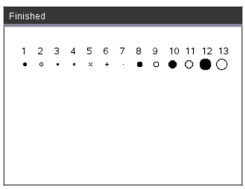
11 - 大空心點

12 - 最大實心點

13 - 最大空心點

```
PlotXY 100,100,1
```

```
針對 n,1,13
DrawText 1+22*n,40,n
PlotXY 5+22*n,50,n
EndFor
```



SetColor

目錄 > CXII

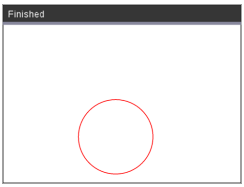
SetColor

紅色值、綠色值、藍色值

紅、綠、藍的有效值必須介於 0 和 255 之間

設定後續「繪圖」指令的顏色

```
SetColor 255,0,0
DrawCircle 150,150,100
```



SetPen

目錄 > CXII

SetPen

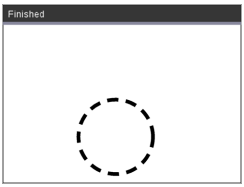
粗細、樣式

粗細:1 <= 粗細 <= 3|1 最細、3 最粗

樣式:1 = 平滑、2 = 點線、3 = 虛線

設定後續「繪圖」指令的畫筆樣式

```
SetPen 3,3
DrawCircle 150,150,50
```



SetWindow

目錄 > CXII

SetWindow

xMin、xMax、yMin、yMax

建立一個與函數繪圖區域相對應的邏輯視窗。所有參數皆必填。

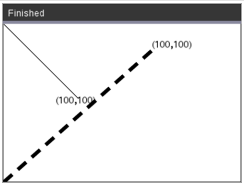
如果繪製物件的一部分位於視窗之外，則系統將會剪切輸出(未顯示)，而不會顯示錯誤訊息。

```
SetWindow 0,160,0,120
```

如此一來，輸出視窗將設定為左下角的 0,0，且寬度和高度會分別設定為 160 和 120

```
DrawLine 0,0,100,100
SetWindow 0,160,0,120
SetPen 3,3
DrawLine 0,0,100,100
```

如果 xmin 大於或等於 xmax，或者 ymin 大於或等於 ymax，則系統會顯示錯誤訊息。



在新設定中，系統不會重新繪製在執行 SetWindow 指令之前所繪製的任何物件。

若要將視窗參數重設為預設值，請使用:

SetWindow 0,0,0,0

UseBuffer

UseBuffer

繪製至螢幕外函數繪圖緩衝區而非螢幕(以增加效能)

此指令與 PaintBuffer 搭配使用，可在程式產生多個圖形物件時提高螢幕的顯示速度。

若使用 UseBuffer，只有在執行下一個 PaintBuffer 指令後才會顯示所有圖形。

您只需在程式中呼叫一次 UseBuffer，亦即每次使用 PaintBuffer 時都不需要相應的 UseBuffer

```
UseBuffer
For n,1,10
x:=randInt(0,300)
y:=randInt(0,200)
radius:=randInt(10,50)
Wait 0.5
DrawCircle x, y, 半徑
EndFor
PaintBuffer
```

此程式將一次顯示全部結果，即 10 個圓。

如果移除「UseBuffer」指令，則系統會在繪製每個圓時即時顯示。

另請參考: PaintBuffer

空元素

分析現實世界的資料時，不一定每次都能取得完整的資料。TI-Nspire™ 軟體能容許空資料元素，讓您根據幾乎完整的資料進行運算，而不用重頭來過或捨棄不完整的案例。

Lists & Spreadsheet 一章的＜*試算表資料繪圖*＞一節列出空元素的資料範例。

delVoid() 函數可讓您刪除列表中的空元素。**isVoid()** 函數可讓您檢定空元素。詳情請參考第 頁碼：37 頁的 **delVoid()** 和第 頁碼：70 頁的 **isVoid()**。

附註：若要在數學式中手動輸入空元素，請輸入「_」或 **void** 關鍵字。系統在對運算式求值時，會自動把 **void** 關鍵字轉成「_」符號。若要在計算機上輸入「_」，請按 [ctrl] [␣]。

涵蓋空元素的計算作業

計算內容涵蓋空元素時，大部分情況下都會求出空結果。請見以下特殊案例。

$\lvert_\rvert$	_
gcd(100,_)	_
3+_	_
{5,_,10}−{3,6,9}	{2,_,1}

列出包含空元素的引數

以下函數和指令會忽略 (跳過) 列表引數中的空元素。

count、**countIf**、**cumulativeSum**、**freqTable▸list**、**frequency**、**max**、**mean**、**median**、**product**、**stDevPop** 、**stDevSamp**、**sum**、**sumIf**、**varPop**、**varSamp**、迴歸計算式、**OneVar**、**TwoVar**、**FiveNumSummary** 統計、信賴區間、統計檢定

sum({2,_,3,5,6.6})	16.6
median({1,2,_,_,_,3})	2
cumulativeSum({1,2,_,4,5})	{1,3,_,7,12}
cumulativeSum$\left(\begin{bmatrix} 1 & 2 \\ 3 & _ \\ 5 & 6 \end{bmatrix}\right)$	$\begin{bmatrix} 1 & 2 \\ 4 & _ \\ 9 & 8 \end{bmatrix}$

SortA 與 **SortD** 會把第一個引數中的所有空元素移到末尾。

{5,4,3,_,1}→*list1*	{5,4,3,_,1}
{5,4,3,2,1}→*list2*	{5,4,3,2,1}
SortA *list1*,*list2*	*Done*
list1	{1,3,4,5,_}
list2	{1,3,4,5,2}

列出包含空元素的引數

{1,2,3,_,5} → *list1*	{1,2,3,_,5}
{1,2,3,4,5} → *list2*	{1,2,3,4,5}
SortD *list1*,*list2*	*Done*
list1	{5,3,2,1,_}
list2	{5,3,2,1,4}

以迴歸來說，X、Y 列表中若有空元素，則殘差的對應元素就會是空元素。

l1:={1,2,3,4,5}: *l2*:={2,_,3,5,6.6}	{2,_,3,5,6.6}
LinRegMx *l1*,*l2*	*Done*
stat.Resid	{0.434286,_,-0.862857,-0.011429,0.44}
stat.XReg	{1.,_,3.,4.,5.}
stat.YReg	{2.,_,3.,5.,6.6}
stat.FreqReg	{1.,_,1.,1.,1.}

若在迴歸中省略類別，則殘差的對應元素就會是空元素。

l1:={1,3,4,5}: *l2*:={2,3,5,6.6}	{2,3,5,6.6}
cat:={"M","M","F","F"}: *incl*:={"F"	{"F"}
LinRegMx *l1*,*l2*,1,*cat*,*incl*	*Done*
stat.Resid	{_,_,0.,0.}
stat.XReg	{_,_,4.,5.}
stat.YReg	{_,_,5.,6.6}
stat.FreqReg	{_,_,1.,1.}

若迴歸的頻率是 0，則殘差的對應元素就會是空元素。

l1:={1,3,4,5}: *l2*:={2,3,5,6.6}	{2,3,5,6.6}
LinRegMx *l1*,*l2*,{1,0,1,1}	*Done*
stat.Resid	{0.069231,_,-0.276923,0.207692}
stat.XReg	{1.,_,4.,5.}
stat.YReg	{2.,_,5.,6.6}
stat.FreqReg	{1.,_,1.,1.}

輸入數學式的快速鍵

快速鍵讓您自行輸入數學式的元素，而不需使用「目錄」或「符號調色盤」。例如若要輸入 √6 運算式，可在輸入線上輸入 **sqrt(6)**。當您按下 enter 時，**sqrt(6)** 運算式就會變成 √6。有的快速鍵可同時用在計算機和電腦鍵盤上。有的快速鍵則主要用在電腦鍵盤上。

計算機和電腦鍵盤快速鍵

若要輸入這個符號：	請輸入這個快速鍵：
π	**pi**
θ	**theta**
∞	**infinity**
≤	**<=**
≥	**>=**
≠	**/=**
⇒ (邏輯隱含)	=>
⇔ (邏輯雙隱含，**XNOR**)	<=>
➞ (儲存運算子)	**=:**
| |(絕對值)	**abs(...)**
√()	**sqrt(...)**
Σ()(總和範本)	**sumSeq(...)**
Π()(乘積範本)	**prodSeq(...)**
$\sin^{-1}$()、$\cos^{-1}$().....	**arcsin(...)**、**arccos(...)**.....
ΔList()	**deltaList(...)**

電腦鍵盤快速鍵

若要輸入這個符號：	請輸入這個快速鍵：
i(虛常數)	**@i**
e(以 e 為底的自然對數)	**@e**
E(科學記號)	**@E**
T(轉置)	**@t**
r(弧度)	**@r**
° (度數)	**@d**

若要輸入這個符號：	請輸入這個快速鍵：
g(梯度)	**@g**
∠(角度)	**@<**
▶(轉換)	**@>**
▶Decimal、**▶approxFraction ()**……	**@>Decimal**、**@>approxFraction()**……

EOS™(方程式作業系統)階層

i的章節描述TI-Nspire™數學與科學學習技術所使用的Operating System (EOS™)。數學、變數和函數以簡單、直接的序列輸入。EOS™軟體使用包含在括號內的群組並根據以下描述的優先順序來計算運算式和方程式。

計算順序

等級	運算子
1	小括弧 ( )、中括弧 [ ]、大括弧 { }
2	間接 (#)
3	函數呼叫
4	後置運算子：度-分-秒(°、'、") 、分數(!) 、百分比(%) 、弧度角(ʳ) 、下標([ ]) 、轉置(ᵀ)
5	指數化、乘冪(^)
6	負值(⁻)
7	字串連接(&)
8	乘法(•) 、除法(/)
9	加法(+) 、減法(-)
10	等式關係：等於(=) 、不等於(≠ 或 /=) 、 小於(<) 、小於或等於(≤ 或 <=) 、大於(>) 、大於或等於(≥ 或 >=)
11	邏輯 **not**
12	邏輯 **and**
13	邏輯 **or**
14	**xor**、**nor**、**nand**
15	邏輯隱含 (⇒)
16	邏輯雙隱含，XNOR (⇔)
17	約束運算子 (「|」)
18	儲存 (→)

小括弧、中括弧、大括弧

它將先計算包含在圓括號、括弧或大括號內的所有計算。例如，在4(1+2)運算式中，EOS™軟體將先計算圓括號內的運算式1+2，然後將結果乘以4。.

運算式或方程式中的左右小括弧、中括弧和大括弧數量必須相同，否則會顯示錯誤訊息，指出缺少元素。例如 (1+2)/(3+4 的式子會導致錯誤訊息「缺少)」產生。

附註：因為TI-Nspire™軟體可以讓您定義自己的函數，所以變數名稱和後面的包含於圓括號內的運算式，將作為「函數調用」而不是隱式的乘法。例如，a(b+c)是b+c運算的a函數。若要將b+c運算式乘以a，請使用明顯的乘法：a*(b+c)。

Indirection

間接運算子(#) 可將字串轉成變數或函數名稱。例如 #(“x”&”y”&”z”) 這個式子可建立 xyz 這個變數名稱。間接運算子也可讓您在程式內建立、修改變數。舉例來說，若 10→r 且 “r”→s1，則 #s1=10。

後置運算子

後置運算子是引數後面緊接的運算子，例如 5!、25% 或 60°15' 45"。後置運算子前面的引數，擁有第四級的求值優先順序。例如 4^3!運算式的 3!會先進行取值。接著計算結果 6 就會成為 4 的指數，得出 4096 的值。

指數化

指數化(^) 和各元素指數化(.^) 的求值順序是從右到左。例如 2^3^2 運算式的求值結果，就和 2^(3^2) 一樣是 512。(2^3)^2 就不同了，結果是 64。

負值

若要輸入負數，請按 (−) 然後輸入數字。後置計算和指數化的執行順序都比負值早。例如 -x2 的結果是負數，$-9^2 = -81$。如果用小括弧將負數括起，例如 $(-9)^2$，結果就是 81。

約束 (「|」)

(「|」) 運算子後面的引數是一組約束條件，影響運算子前面的引數求值結果。

TI-Nspire CX II - TI-Basic 程式設計功能

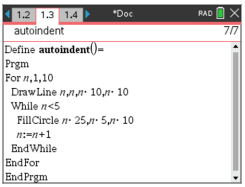
程式設計編輯器中的自動縮排

TI-Nspire™ 程式編輯器現在會將區塊指令中的陳述式自動縮排。

區塊指令為 If/EndIf、For/EndFor、While/EndWhile、Loop/EndLoop，以及 Try/EndTry

編輯器將在區塊指令中自動為程式指令增加空格。區塊的結尾指令將對齊開始指令。

以下範例顯示了巢狀區塊指令的自動縮排格式。



複製和貼上的程式碼片段將保留原始縮排格式。

開啟以較早軟體版本建立的程式將保留原始縮排格式。

改進 TI-Basic 的錯誤訊息

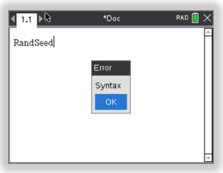
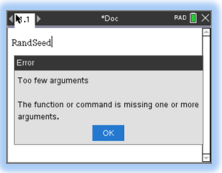
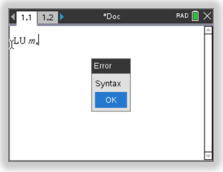
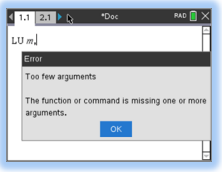
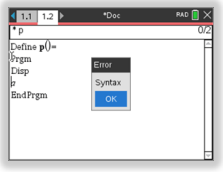
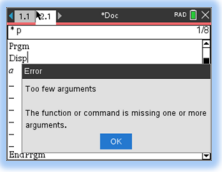
錯誤

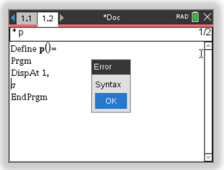
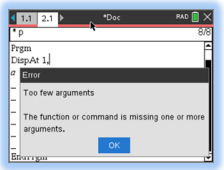
錯誤狀況	新訊息
條件陳述式中的錯誤 (**If/While**)	條件陳述式並未解析為 **TRUE** 或 **FALSE** **注意:**透過將游標改為放在發生錯誤的行上，我們將無需指定錯誤是位於 "**If**" 陳述式還是 "**While**" 陳述式。
缺少 **EndIf**	預期為 **EndIf**，但卻找到不同的 End 陳述式
缺少 **EndFor**	預期為 **EndFor**，但卻找到不同的 End 陳述式
缺少 **EndWhile**	預期為 **EndWhile**，但卻找到不同的 End 陳述式
缺少 **EndLoop**	預期為 **EndLoop**，但卻找到不同的 End 陳述式
缺少 **EndTry**	預期為 **EndTry**，但卻找到不同的 End 陳述式
If <condition>後的 "**Then**" 遭到省略	缺少 **If..Then**

錯誤狀況	新訊息
ElseIf <condition>後的 "**Then**" 遭到省略	以下區塊缺少 **Then**:**ElseIf**。
當在控制區塊外遇到 "**Then**"、"**Else**" 和 "**ElseIf**" 時	以下區塊外部的 **Else** 無效:**If..Then..EndIf** 或 **Try..EndTry**
"**ElseIf**" 出現在 "**If..Then..EndIf**" 區塊外部	以下區塊外部的 **ElseIf** 無效:**If..Then..EndIf**
"**Then**" 出現在 "**If....EndIf**" 區塊外部	以下區塊外部的 **Then** 無效:**If..EndIf**

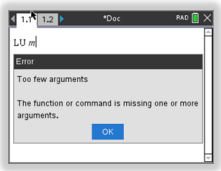
語法錯誤

如果使用不完整的引數列表來呼叫期望一個或多個引數的指令，則將發出「**引數過少錯誤**」，而非「**語法**」錯誤

目前行為	新 CX II 行為
	
	
	

目前行為	新 CX II 行為
	

注意:如果引數的不完整列表之後未接逗號，則將顯示如下錯誤訊息:「引數過少」。此與前版相同。



常數和數值

以下表格列出執行單位換算時可用的常數及其數值。您可手動輸入數值，或從 **[公用程式] > [單位換算]** 中的**常數**清單中選擇 (計算機:按 📖 **3**)。

常數	名稱	值
_c	光速	299792458 _m/_s
_Cc	庫倫常數	8987551787.3682 _m/_F
_Fc	法拉第常數	96485.33289 _coul/_mol
_g	重力加速度	9.80665 _m/_s^2
_Gc	重力常數	6.67408E-11 _m^3/_kg/_s^2
_h	普朗克常數	6.626070040E-34 _J _s
_k	波茲曼常數	1.38064852E-23 _J/_°K
_µ0	真空磁導率	1.2566370614359E-6 _N/_A^2
_µb	波耳磁元	9.274009994E-24 _J _m^2/_Wb
_Me	電子靜質量	9.10938356E-31 _kg
_Mµ	介子質量	1.883531594E-28 _kg
_Mn	中子靜質量	1.674927471E-27 _kg
_Mp	質子靜質量	1.672621898E-27 _kg
_Na	亞佛加厥常數	6.022140857E23 /_mol
_q	電子電荷	1.6021766208E-19 _coul
_Rb	波耳半徑	5.2917721067E-11 _m
_Rc	莫耳氣體常數	8.3144598 _J/_mol/_°K
_Rdb	芮得柏常數	10973731.568508/_m
_Re	電子半徑	2.8179403227E-15 _m
_u	原子質量	1.660539040E-27 _kg
_Vm	莫耳體積	2.2413962E-2 _m^3/_mol
_ε0	真空介電係數	8.8541878176204E-12 _F/_m
_σ	史特凡波茲曼常數	5.670367E-8 _W/_m^2/_°K^4
_ϕ0	磁通量量子	2.067833831E-15 _Wb

錯誤代碼和訊息

出現錯誤時，errCode 變數會有指定代碼。使用者自行定義的程式和函數可檢查 errCode，以判斷錯誤原因。若需 errCode 的使用參考範例，請參考第 頁碼：145 頁 Try 指令下的例 2。

附註：有的錯誤狀況只適用於 TI-Nspire™ CAS *產品，有的只適用於* TI-Nspire™ *產品。*

錯誤代碼	說明
10	函數並未傳回值
20	檢定並未解出 TRUE 或 FALSE。 系統通常無法比較未定義的變數。例如若執行 If 語句時，a 或 b 其中之一屬於未定義，則 If a<b 檢定就會導致這項錯誤。
30	引數不得是資料夾名稱。
40	引數錯誤
50	引數不符 兩個以上的引數必須屬於相同類型。
60	引數必須是布林運算式或整數
70	引數必須是十進位數
90	引數必須是列表
100	引數必須是矩陣
130	引數必須是字串
140	引數必須是變數名稱。 名稱的條件： • 開頭不得是數字 • 不可包含空格或特殊字元 • 不得以無效方式使用底線或句號 • 不可超過長度限制 詳情請參考說明文件中的＜Calculator＞一節。
160	引數必須是運算式
165	電池電力不足，無法收發資料 請裝入新電池再收發資料。
170	界限 下限必須小於上限，才能界定搜尋區間。

錯誤代碼	說明
180	中斷 在冗長計算或執行程式期間按 退出 或 开机 鍵。
190	循環定義 在化簡期間無盡期取代變數值時，為避免用完記憶體而顯示此訊息。例如在 a+1->a 中，a 是未定義的變數，因此會導致本錯誤。
200	約束運算式無效 例如 solve(3x^2-4=0,x) | x<0 or x>5 就會產生這個錯誤訊息，因為約束條件是以「or」分隔，而非「and」。
210	資料類型無效 引數的資料類型錯誤。
220	相依限制
230	維度 列表或矩陣指數無效。例如若將 list {1,2,3,4} 儲存在 L1，則 L1[5] 會導致維度錯誤，因為 L1 只包含四個元素。
235	維度錯誤。列表中的元素不足。
240	維度不符 兩個以上引數的維數必須相同。例如 [1,2]+[1,2,3] 的維數就不符，因為兩個矩陣的元素數目不同。
250	分母是零
260	值域錯誤 引數必須在指定值域。例如 **rand(0)** 就無效。
270	變數名稱重複
280	Else 和 ElseIf 在 If...EndIf 段以外時無效
290	EndTry 缺少相對的 Else 語句
295	迭代過多
300	預期的列表或矩陣是 2 元或 3 元
310	nSolve 的第一個引數必須是只有一個變數的方程式。其中不能包含需注意變數以外的未取值變數。
320	solve 或 cSolve 的第一個引數必須是方程式或不等式 例如 solve(3x^2-4,x) 無效，因為第一個引數不是方程式。
345	單位不一致

錯誤代碼	說明
350	指數在範圍外
360	間接字串不是有效的變數名稱
380	未定義 Ans 如果不是前一項計算作業沒有產生 Ans，就是根本沒有輸入前一項計算作業。
390	指定無效
400	指定值無效
410	指令無效
430	無法用於目前的模式設定
435	猜測值無效
440	隱式乘法無效 例如 x(x+1) 無效，但 x*(x+1) 是正確語法。這是為了避免隱式乘法和函數呼叫有所混淆。
450	無法用於函數或目前的運算式 只有特定指令可用於使用者自行定義的函數。
490	無法用於 Try..EndTry 段
510	列表或矩陣無效
550	不可在函數或程式以外 幾個指令不可用在函數或程式以外。例如不可使用 Local，除非在函數或程式中。
560	不可在 Loop..EndLoop、For..EndFor、While..EndWhile 段以外 例如 Exit 指令只有在這些迴圈段落中才有效。
565	不可用在程式以外
570	路徑名稱無效 例如 \var 就無效。
575	複極無效
580	程式參照無效 不得在 1+p(x) 這類函數或運算式中參照程式。p 代表程式。
600	表格無效
605	單位用法無效

錯誤代碼	說明
610	Local 語句中的變數名稱無效
620	變數或函數名稱無效
630	變數參照無效
640	向量語法無效
650	連結傳輸 兩台計算機無法傳輸資料。請檢查兩端的連接線是否已經接牢。
665	矩陣無法對角化
670	記憶體不足 1.刪除本文件的某些資料 2.儲存本文件然後關閉 如果 1、2 失敗，請取出電池然後重新裝入
672	資源耗盡
673	資源耗盡
680	缺少 (
690	缺少)
700	缺少“
710	缺少]
720	缺少 }
730	缺少段落語法的開頭或結尾
740	缺少 If..EndIf 段中的 Then
750	名稱不是函數或程式
765	尚未選取函數
780	找不到解
800	非實數結果 例如若軟體使用「實數」設定，√(-1) 就無效。 如果要進行複數計算，請把「實數或複數」模式設定改成 RECTANGULAR 或 POLAR。
830	溢位
850	找不到程式

錯誤代碼	說明
	執行期間無法在所提供的路徑中，找到其他程式內部的程式參照項目。
855	繪圖時不可使用 Rand 類型函數
860	遞迴太深
870	保留名稱或系統變數
900	引數錯誤 無法將中位數-中位數模型應用到資料集。
910	語法錯誤
920	找不到文字
930	引數太少 函數或指令缺少一個以上的引數。
940	引數太多 運算式或方程式包含過多引數，無法求值。
950	下標太多
955	未定義的變數太多
960	尚未定義變數 尚未指定變數值。請使用以下其中一個指令： • sto ➞ • := • **Define** 以便指定變數值。
965	作業系統無使用授權
970	正在使用該變數，因此無法參照或更改
980	變數已經受到保護
990	變數名稱無效 名稱絕不可超過長度限制
1000	Window 變數值域
1010	縮放
1020	內部錯誤
1030	受保護的記憶體違規

錯誤代碼	說明
1040	不支援此函數。本函數需要「電腦代數系統」。請嘗試 TI-Nspire™ CAS。
1045	不支援此運算子。本運算子需要「電腦代數系統」。請嘗試 TI-Nspire™ CAS。
1050	不支援此功能。本運算子需要「電腦代數系統」。請嘗試 TI-Nspire™ CAS。
1060	輸入引數必須是數字。只允許包含數值的輸入。
1070	三角函數引數太大，無法精確約化
1080	不支援使用 Ans。此應用程式不支援 Ans。
1090	尚未定義函數。請使用以下其中一個指令： • **Define** • **:=** • sto →
1100	非實數計算 例如若軟體使用「實數」設定，√(-1) 就無效。 如果要進行複數計算，請把「實數或複數」模式設定改成 RECTANGULAR 或 POLAR。
1110	邊界無效
1120	不得更改正負號
1130	引數不得是列表或矩陣
1140	引數錯誤 第一個引數必須是第二個引數中的多項式。如果省略第二個引數，軟體會試圖選取預設值。
1150	引數錯誤 前二個引數必須是第三個引數中的多項式。如果省略第三個引數，軟體會試圖選取預設值。
1160	資料庫路徑名稱無效 路徑名稱的格式必須是 *xxx**yyy*： • *xxx* 部份可包含 1 至 16 個字元。 • *yyy* 部份可包含 1 至 15 個字元。 詳情請參考說明文件中的＜資料庫＞一節。
1170	資料庫路徑名稱用法無效 • 不可使用 **Define**、**:=** 或 sto →指定路徑名稱的值。 • 不可將路徑名稱宣告為區域性變數，或將該名稱用在函數或程式定義中當作參數。
1180	資料庫變數名稱無效。

錯誤代碼	說明
	名稱的條件： • 不包含句點 • 開頭不得是底線 • 長度不可超過 15 個字元 詳情請參考說明文件中的＜資料庫＞一節。
1190	找不到資料庫文件： • 確認資料庫是否在 MyLib 資料夾中。 • 重新整理資料庫。 詳情請參考說明文件中的＜資料庫＞一節。
1200	找不到資料庫變數： • 確認資料庫變數是否在資料庫的第一個問題中。 • 請務必將資料庫變數界定為 LibPub 或 LibPriv。 • 重新整理資料庫。 詳情請參考說明文件中的＜資料庫＞一節。
1210	資料庫捷徑名稱無效。 名稱的條件： • 不包含句點 • 開頭不得是底線 • 長度不可超過 16 個字元 • 不得是保留名稱 詳情請參考說明文件中的＜資料庫＞一節。
1220	值域錯誤： tangentLine 與 normalLine 函數只支援實值函數。
1230	值域錯誤。 度數角或梯度角模式不支援三角轉換運算子。
1250	引數錯誤 請使用線性方程式。 變數是 x 和 y 的兩個線性聯立方程式舉例如下： 3x+7y=5 2y-5x=-1
1260	引數錯誤： nfMin 或 nfMax 的第一個引數必須是只有一個變數的運算式。其中不能包含需注意變數以外的未取值變數。

錯誤代碼	說明
1270	引數錯誤 導數階數必須等於 1 或 2。
1280	引數錯誤 請使用只有一個變數的展開多項式。
1290	引數錯誤 請使用只有一個變數的多項式。
1300	引數錯誤 多項式的係數在求值後必須是數值。
1310	引數錯誤： 無法對函數的一個以上引數求值。
1380	引數錯誤： 不允許巢狀調用 domain() 函數。

警告代碼和訊息

可使用 warnCodes() 函數以儲存由計算運算式所產生的警告代碼。下表列出每一個數值警告代碼及其相關訊息。

如需儲存警告代碼的範例，請參考第 頁碼：153 頁的 warnCodes()。

警告代碼	訊息
10000	作業可能產生錯誤的解。
10001	對方程式作微分可能產生錯誤的方程式。
10002	有問題的解
10003	有問題的準確性
10004	作業可能遺失解。
10005	cSolve 可能指定更多的零。
10006	Solve 可能指定更多的零。
10007	可能存在更多的解。
10007	可能存在其他解決方案。請指定適當的下限或上限及/或猜測值。 solve() 的使用範例： • solve(方程式，變數=估計值) |下限<變數<上限 • solve(方程式，變數) |下限<}變數<上限 • solve(方程式，變數=估計值)
10008	結果的值域可能比輸入的值域小。
10009	結果的值域可能比輸入的值域大。
10012	非實數計算
10013	1 取代 ∞^0 或 undef^0
10014	1 取代 undef^0
10015	1 取代 1^∞ 或 1^undef
10016	1 取代 1^undef
10017	∞ 或 ∞ 取代 溢位
10018	作業需要並傳回 64 位元的值
10019	記憶體用盡，化簡可能不完整。
10020	三角函數引數太大，無法精確約化。
10021	輸入包含未定義的參數。 結果不一定對所有可能的參數值有效。

警告代碼	訊息
10022	指定適當的下限和上限會產生解。
10023	單位矩陣已乘以純量。
10024	使用適當的近似值運算法取得結果。
10025	無法在 [精確] 模式中驗證等值。
10026	可能忽略約束。由形式 "\" '變數 MathTestSymbol 約束' 或結合這些形式以指定約束，例如 'x<3 and x>-12'

一般資訊

線上說明

education.ti.com/eguide

選擇您的國家/地區以取得更多產品資訊。

連絡 TI 技術支援部門

education.ti.com/ti-cares

選擇您的國家/地區以取得技術和其他支援資源。

服務與保固資訊

education.ti.com/warranty

選擇您的國家 / 地區有關的長度和有關產品或服務的保修條款。

有限保證。這保證不會影響您的法定權利。

Texas Instruments Incorporated

12500 TI Blvd.

Dallas, TX 75243

索引

–

–次方根
範本 1
–, 減[*] 162

!

!, 階乘 171

"

", 秒符號 178

#

#, 間接 176
#, 間接運算子 203

%

%, 百分比 167

&

&, 接上 171

*****

*, 乘 162

,

, 分符號 178

.

.-, 點差 165
.*, 點乘 166
./, 點除 166
.^, 點冪 166
.+, 點和 165

:

:=, 賦值 181

^

$^{-1}$, 倒數 179
^, 乘冪 164

|

|, 約束運算子 179

+

+, 加 161

/

/, 除[*] 163

=

≠, 不等於[*] 168
=, 等於 167

>

>, 大於 169

∏

∏, 乘積[*] 173

∑

∑( ), 總和[*] 174
∑Int( ) 174
∑Prn( ) 175

√

√, 平方根[*] 173

∫

∫, 積分[*] 172

≤

≤, 大於或等於 170
≤, 小於或等於 169

▸

▸, 轉成梯度角 [Grad] 63
▸approxFraction( ) 12
▸Base10, 以十進位整數顯示 [Base10] 17
▸Base16, 以十六進位顯示 [Base16] 17
▸Base2, 以二進位顯示 [Base2] ... 16
▸Cylind, 以圓柱座標形式顯示向量 [Cylind] 33
▸DD, 以十進位角顯示 [DD] 34
▸Decimal, 以十進位顯示結果 [Decimal] 34
▸DMS, 以度/分/秒顯示 [DMS] ... 40
▸Polar, 以極座標形式顯示向量 [Polar] 102
▸Rad，轉換成弧度角 110
▸Rect，顯示為直角向量 112
▸Sphere, 以球面座標形式顯示向量 [Sphere] 133

→

→, 儲存 180

⇒

⇒，邏輯隱含 [*] 170
⇒，邏輯隱含 [*] 200

⇔

⇔，邏輯雙隱含 [*] 171

©

©, 註解 181

°

°, 度/分/秒 [*] 178
°, 度數符號 [*] 177

0

0b, 二進位指示 182
0h, 十六進位指示 182

1

10 的次方, 10^( ) 179
10^( ), 10 的次方 179

A

abs( ), 絕對值 7
amortTbl( ), 攤銷表 7, 15
and, 布林運算子 8
angle( ), 角度 8
ANOVA, 單因子變異數分析 9
ANOVA2way, 雙因子變異數分析 10
Ans, 最近結果 12
answer (last), Ans 12
approx( ), 近似值 12
approxRational( ) 13
arccos() 13
arccosh() 13
arccot() 13
arccoth() 13
arccsc() 13
arccsch() 13
arcsec() 13
arcsech() 13
arcsin() 14
arcsinh() 14
arctan() 14
arctanh() 14
augment( ), 擴展/串連 14
avgRC( ), 平均變化率 14

B

binomCdf( ) 18, 68
binomPdf( ) 18

C

Cdf( ) 48
ceiling( ), 無條件進入法 19
centralDiff( ) 19
char( ), 字元字串 20
$\chi^2$2way 20
χ^2Cdf( ) 20

χ²GOF 21
χ²Pdf( ) 21
ClearAZ 21
ClrErr, 清除錯誤 22
colAugment 22
colDim( ), 矩陣行維數 22
colNorm( ), 矩陣行範數 22
conj( ), 共軛複數 23
constructMat( ), 建立矩陣 23
corrMat( ), 相關矩陣 24
cos⁻¹, 反餘弦 25
cos( ), 餘弦 24
cosh⁻¹( ), 反雙曲餘弦 26
cosh( ), 雙曲餘弦 26
cot⁻¹( ), 反餘切 27
cot( ), 餘切 27
coth⁻¹( ), 反雙曲餘切 28
coth( ), 雙曲餘切 28
count( ), 計算列表中的項目數 .. 28
countif( ), 有條件計算列表中的
項目數 28
cPolyRoots() 29
crossP( ), 外積 29
csc⁻¹( ), 反餘割 30
csc( ), 餘割 30
csch⁻¹( ), 反雙曲餘割 31
csch( ), 雙曲餘割 31
CubicReg, 三次迴歸 31
cumulativeSum( ), 累積總和 32
Cycle, 迴圈 32

D

d ( ), 一階導數 172
dbd( ), 日期之間的天數 33
Define 34
Define LibPriv 35
Define LibPub 36
Define, 定義 34
deltaList() 36
DelVar, 刪除變數 36
delVoid( ), 移除空元素 37
det( ), 矩陣行列式 37
diag( ), 矩陣對角線 38
dim( ), 維數 38
Disp, 顯示資料 38
DispAt 39
dotP( ), 點積 41

E

e 指數
範本 2
E, 科學記號 176
e^( ), 以 e 為底乘冪 41
eff( ), 名目利率轉成實際年利率 . 42
eigVc( ), 特徵向量 42
eigVl( ), 特徵值 42
else if, ElseIf 43
ElseIf, else if 43
end
for, EndFor 50
while, EndWhile 154
end while, EndWhile 154
EndTry, 結束嘗試 145
EndWhile, end while 154
EOS (方程式作業系統) 202
euler( ), 尤拉函數 44
Exit, 結束 46
exp( ), 以 e 為底乘冪 46
expr( ), 字串至運算式 46
ExpReg, 指數迴歸 47

F

factor( ), 因式 48
Fill, 矩陣填充 49
FiveNumSummary 49
floor( ), 無條件捨去法 50
For 50
for, For 50
For, for 50
format( ), 格式字串 50
fpart( ), 函數部分 51
freqTable( ) 51
frequency( ) 52
Frobenius 範數, norm( ) 95

Func, 函數 53
Func, 程式函數 53

G

g, 梯度 176
gcd( ), 最大公因數 54
geomCdf( ) 54
geomPdf( ) 55
Get 55, 192
getDenom( ), 取得/傳回分母 56
getKey() 56
getLangInfo( ), 取得/傳回語言資訊 59
getLangInfo( ), 取得/傳回變數資訊 62
getLockInfo( ), 檢驗變數或變數群組的鎖定狀態 60
getMode( ), 取得模式設定 60
getNum( ), 取得/傳回數字 61
GetStr 61
getType( ), 取得變數類型 61
Goto, 轉到 63

I

identity( ), 單位矩陣 63
If, 如果 63
ifFn( ) 64
imag( ), 虛部 65
inString( ), 在字串內 66
int( ), 整數 66
intDiv( ), 整數除法 66
interpolate( ), 插入 67
invF( ) 67
invNorm( ), 逆累積常態分佈) ... 68
invt( ) 68
Invχ²( ) 67
iPart( ), 整數部份 69
irr( ), 內部投資報酬率
內部投資報酬率, irr( ) 69
isPrime( ), 質數檢定 69
isVoid( ), 空值檢定 70

L

Lbl, 標籤 70
lcm, 最小公倍數 70
left( ), 左邊 71
LibPriv 35
LibPub 36
libShortcut( ), 建立資料庫群組快速鍵 71
LinRegBx, 線性迴歸 71
LinRegMx, 線性迴歸 72
LinRegtIntervals, 線性迴歸 73
LinRegtTest 75
linSolve() 76
Δlist( ), 列表差異 76
list▸mat( ), 列表至矩陣 77
ln( ), 自然對數 77
LnReg, 對數迴歸 77
Local, 區域性變數 79
Lock, 鎖定變數或變數群組 79
Logistic, 羅吉斯迴歸 80
LogisticD, 羅吉斯迴歸 81
Loop, 迴圈 82
LU, 矩陣上下分解值 83

M

mat▸list( ), 矩陣至列表 83
max( ), 最大值 83
mean( ), 平均值 84
median( ), 中位數 84
MedMed, 中位數-中位數線性迴歸 85
mid( ), 字串中 86
min( ), 最小值 86
mirr( ), 修正後內部投資報酬率 . 87
mod( ), 模數 88
mRow( ), 矩陣行運算 88
mRowAdd( ), 矩陣行乘法與加法 88
MultReg 88
MultRegIntervals( ) 89
MultRegTests( ) 90

N

nand，布林運算子 91
nCr( ), 組合 92
nDerivative( ), 數值導數 92
newList( ), 新列表 93
newMat( ), 新矩陣 93
nfMax( ), 數值函數上限 93
nfMin( ), 數值函數下限 94
nInt( ), 數值積分 94
nom), 實際年利率轉成名目利率 94
nor，布林運算子 95
norm( ), Frobenius 範數 95
normCdf( ) 96
normPdf( ) 96
not，布林運算子 96
nPr( ), 排列 97
npv( ), 淨現值 97
nSolve( ), 數值解 98

O

OneVar, 單變數統計 98
or(布林) , 或 99
or，布林運算子 99
ord( ), 數字代號 100

P

P▸Rx( ), 成對值的等值 x 座標 ... 100
P▸Ry( ), 成對值的等值 y 座標 ... 101
PassErr, 傳遞錯誤 101
Pdf( ) 51
piecewise( ) 102
poissCdf( ) 102
poissPdf( ) 102
polyEval( ), 多項式求值 103
PolyRoots() 103
PowerReg, 乘冪迴歸 104
Prgm, 定義程式 105
prodSeq() 105
product( ), 乘積 106
propFrac, 真分數 106

Q

QR 分解值, QR 106
QR, QR 分解值 106
QuadReg, 二次迴歸 107
QuartReg, 四次迴歸 108

R

R, 弧度 177
R▸Pr( )，極座標 109
R▸Pθ( )，極座標 109
rand( )，亂數 110
randBin，亂數 110
randInt( )，隨機整數 110
randMat( )，隨機矩陣 111
randNorm( )，隨機範數 111
randPoly( )，隨機多項式 111
randSamp( ) 112
RandSeed，亂數種子 112
real( )，實部 112
ref( )，列梯形 113
RefreshProbeVars 114
remain( )，餘 115
RequestStr 117
Return，傳回 117
right( )，從右取字串 118
rk23( )，Runge Kutta 函數 118
rotate( )，互換 119
round( )，四捨五入法 120
rowAdd( )，矩陣列加法 121
rowDim( )，矩陣的列數 121
rowNorm( )，矩陣的列範數 121
rowSwap( )，矩陣列交換 121
rref( )，列簡化梯形 122

S

sec^{-1}( ), 反正割 122
sec( ), 正割 122
$sech^{-1}$( ), 反雙曲正割 123
sech( ), 雙曲正割 123
seq( ), 序列 124
seqGen( ) 124

seqn( ) 125
setMode( ), 設定模式 126
shift( ), 移位 127
sign( ), 符號 128
simult( ), 聯立方程式 129
$\sin^{-1}$( ), 反正弦 130
sin( ), 正弦 129
$\sinh^{-1}$( ), 反雙曲正弦 131
sinh( ), 雙曲正弦 131
SinReg, 正弦迴歸 132
SortA, 升冪排列 133
SortD, 降冪排列 133
sqrt( ), 平方根 134
stat.results 134
stat.values 135
stdDevPop( ), 母群體標準差 136
stdDevSamp( ), 樣本標準差 136
string( ), 運算式至字串 137
student-t 分布機率, tCdf( ) 142
student-t 機率密度, tPdf( ) 144
subMat( ), 子矩陣 137, 139
sum( ), 求和 138
sumIf( ) 138
sumSeq() 139

T

t 檢定, tTest 146
T, 轉置 139
$\tan^{-1}$( ), 反正切 140
tan( ), 正切 139
$\tanh^{-1}$( ), 反雙曲正切 141
tanh( ), 雙曲正切 141
tCdf( ), studentt 分布機率 142
Test_2S, 二樣本 F檢定 52
tInterval, t 信賴區間 143
tInterval_2Samp, 二 t 信賴區間 ... 143
tPdf( ), studentt 機率密度 144
trace( ) 144
Try, 錯誤處理指令 145
tTest, t 檢定 146
tTest_2Samp, 二樣本 t 檢定 146
tvmFV( ) 147
tvmI( ) 147
tvmN( ) 148
tvmPmt( ) 148
tvmPV( ) 148
TwoVar, 二變數結果 149

U

unitV( ), 單位向量 151
unLock, 解開鎖定變數或變數群組 151

V

varPop( ) 151
varSamp( ), 樣本變異數 152

W

Wait 指令 152
warnCodes( ), 警告代碼 153
when( ), when 153
when, when( ) 153
while, While 154
While, while 154

X

x^2, 平方 165
XNOR 171
xor，布林互斥的「或」..... 154

Z

zInterval, z 信賴區間 155
zInterval_1Prop, 單一母群體比例 z 信賴區間 155
zInterval_2Prop, 二母群體比例 z 信賴區間 156
zInterval_2Samp, 二樣本 z 信賴區間 156
zTest 157
zTest_1Prop, 單一母群體比例 z 檢定 158
zTest_2Prop, 二母群體比例 z 檢定 158
zTest_2Samp, 二樣本 z 檢定 159

一

一階導數
範本 5

二

二次迴歸, QuadReg 107
二進位
指示, 0b 182
顯示, ▸Base2 16
二階導數
範本 5
二樣本 F檢定 52
二變數結果, TwoVar 149

十

十六進位
指示, 0h 182
顯示, ▸Base16 17
十進位
角度顯示, ▸DD 34
整數顯示, ▸Base10 17

三

三次迴歸, CubicReg 31

大

大於, > 169
大於或等於, | 170

子

子矩陣, subMat( ) 137, 139

小

小於, ... 168
小於或等於, { 169

不

不等於, ≠ 168

中

中位數-中位數線性迴歸,
MedMed 85
中位數, median( ) 84

互

互換, rotate( ) 119

分

分布函數
normCdf( ) 96
normPdf( ) 96
poissCdf( ) 102
poissPdf( ) 102
tCdf( ) 142
tPdf( ) 144
$\chi^2$2way( ) 20
χ^2Cdf( ) 20
χ^2GOF( ) 21
χ^2Pdf( ) 21
分佈函數
binomCdf( ) 18, 68
binomPdf( ) 18
invNorm( ) 68
invt( ) 68
Invχ^2( ) 67
分段函數(2段)
範本 2
分段函數(N段)
範本 2
分符號, ... 178
分數
propFrac 106
範本 1

反

反正切, $\tan^{-1}$( ) 140
反正弦, $\sin^{-1}$( ) 130
反向, ^-¹ 179
反餘弦, $\cos^{-1}$( ) 25
反雙曲
正切, $\tanh^{-1}$( ) 141

正弦, sinh⁻¹( ) 131
餘弦, cosh⁻¹( ) 26

文

文字命令 142

方

方陣
QR 分解值, QR 106
上下分解值, LU 83
子矩陣, subMat( ) 137, 139
列表至矩陣, list▸mat( ) 77
行列式, det( ) 37
行乘法與加法, mRowAdd( ) . 88
行運算, mRow( ) 88
行維數, colDim( ) 22
行範數, colNorm( ) 22
求和, sum( ) 138
乘積, product( ) 106
特徵向量, eigVc( ) 42
特徵值, eigVl( ) 42
矩陣至列表, mat▸list( ) 83
累積總和, cumulativeSum( ) .. 32
最大值, max( ) 83
最小值, min( ) 86
填充, Fill 49
新, newMat( ) 93
對角線, diag( ) 38
維數, dim( ) 38
點和, .+ 165
點乘, .* 166
點差N 165
點除, ./ 166
點冪, .^ 166
擴展/串連, augment( ) 14
轉置, T 139
方程式作業系統 (EOS) 202

日

日期之間的天數, dbd( ) 33

以

以 e 為底乘冪, e^( ) 41, 46

加

加, + ... 161

右

右邊, right( ) 44, 153

四

四次迴歸, QuartReg 108
四捨五入法，round( ) 120

外

外積, crossP( ) 29

左

左邊, left( ) 71

布

布林運算子
⇒ 170
⇔ 171
and 8
nand 91
nor 95
Þ 200
互斥 154
或 99
非 96

平

平方根
範本 1
平方根, ‡( ) 134, 173
平均值, mean( ) 84
平均變化率, avgRC( ) 14

本

本金總和 175

正

正切, tan( ) 139
正弦, sin( ) 129
正弦迴歸, SinReg 132

用

用「|」運算子代換 179
用「|」運算子排除 179

列

列表
升冪排列, SortA 133
外積, crossP( ) 29
列表中的差異, Δlist( ) 76
列表至矩陣, list▸mat( ) 77
字串中, mid( ) 86
求和, sum( ) 138
空元素, 在 198
降冪排列, SortD 133
乘積, product( ) 106
差異, Δlist( ) 76
矩陣至列表, mat▸list( ) 83
累積總和, cumulativeSum( ) .. 32
最大值, max( ) 83
最小值, min( ) 86
新, newList( ) 93
點積, dotP( ) 41
擴展/串連, augment( ) 14
列表, 有條件項目計數 28
列表, 項目計數 28
列表至矩陣, list▸mat( ) 77
列梯形，ref( ) 113
列簡化梯形，rref( ) 122

名

名目利率, nom( ) 94

向

向量
外積, crossP( ) 29
單位, unitV( ) 151
圓柱座標形式顯示, ▸Cylind . 33
點積, dotP( ) 41

因

因式, factor( ) 48

在

在字串內，inString( ) 66

多

多項式
求值, polyEval( ) 103
隨機，randPoly( ) 111
多項式求值, polyEval( ) 103
多線性迴歸 t 檢定 90

如

如果，If 63
如果結束，EndIf 63

字

字元
字串, char( ) 20
數值代號, ord( ) 100
字元字串, char( ) 20
字串
互換，rotate( ) 119
右邊, right( ) 44, 153
左邊, left( ) 71
用來建立變數名稱 203
在字串內，inString 66
字元字串, char( ) 20
字串中, mid( ) 86
字串至運算式, expr( ) 46
長度 38
格式 50
格式, format( ) 50
從右取字串，right( ) 67, 118
接上, & 171
移位, shift( ) 127
間接, # 176

運算式至字串, string( ) 137
維數, dim( ) 38
數字代號, ord( ) 100
字串中, mid( ) 86
字串的長度 38

成

成對值的等值 x 座標, P▸Rx( ) ... 100
成對值的等值 y 座標, P▸Ry( ) ... 101

有

有條件計算列表中的項目數,
countif( ) 28

百

百分比, % 167

自

自然對數, ln( ) 77

利

利息總和 174

刪

刪除
列表中的空元素 37
變數, DelVar 36

否

否則，Else 63

序

序列, seq( ) 124-125

求

求和, sum( ) 138
求值, 順序 202

角

角度, angle( ) 8

使

使用者定義的函數 34
使用者定義的函數和程式 35-36

函

函數
使用者定義的 34
部分, fpart( ) 51
程式函數, Func 53
函數和變數
複製 23

取

取得/傳回
分母, getDenom( ) 56
數字, getNum( ) 61
變數資訊, getVarInfo( ) 59, 62

和

和，| 179

定

定義
公用的函數或程式 36
自訂的函數或程式 35
定義, Define 34
定積分
範本 6

弧

弧度, R 177

物

物件
建立資料庫快速鍵 71

直

直角向量顯示，▸Rect 112

空

空元素 198
空元素,移除 37
空值，檢定 70
空值檢定，isVoid() 70

近

近似值,approx() 12

度

度/分/秒符號 178
度/分/秒顯示,▸DMS 40
度數符號,- 177

建

建立矩陣,constructMat() 23

指

指數
範本 1
指數迴歸,ExpReg 47

相

相關矩陣,corrMat() 24

科

科學記號,E 176

秒

秒符號," 178

約

約束運算子「|」..... 179
約束運算子，計算順序 202

要

要求 115

計

計算日期之間的天數,dbd() 33
計算列表中的項目數,count() .. 28

負

負值,輸入負數 203

乘

乘,* 162
乘冪,^ 164
乘冪迴歸,PowerReg 103-104, 142
乘冪迴歸，PowerReg 115, 117
乘積(Π)
範本 5
乘積,product() 106
乘積,Π() 173

修

修正後內部投資報酬率,mirr() . 87

倒

倒數,^⁻¹ 179

格

格式字串,format() 50

特

特徵向量,eigVc() 42
特徵值,eigVl() 42

真

真分數,propFrac 106

矩

矩陣
列加法，rowAdd() 121
列交換，rowSwap() 121
列梯形，ref() 113
列數，rowDim() 121
列範數，rowNorm() 121

列簡化梯形，rref( ) 122
單位，identity( ) 63
隨機，randMat( ) 111
矩陣(1×2)
範本 4
矩陣(2×1)
範本 4
矩陣(2×2)
範本 4
矩陣(m×n)
範本 4
矩陣至列表, mat▸list( ) 83

財

財務函數, tvmFV( ) 147
財務函數, tvmI( ) 147
財務函數, tvmN( ) 148
財務函數, tvmPmt( ) 148
財務函數, tvmPV( ) 148

迴

迴圈, Cycle 32
迴圈, Loop 82
迴歸
MultReg 88
二次, QuadReg 107
三次, CubicReg 31
中位數-中位數線性,
MedMed 85
四次, QuartReg 108
正弦, SinReg 132
指數, ExpReg 47
乘冪迴歸, PowerReg ... 103-104, 142
乘冪迴歸，PowerReg 115, 117
對數, LnReg 77
線性迴歸, LinRegAx 72
線性迴歸, LinRegBx 71, 73
羅吉斯 80
羅吉斯, Logistic 81

逆

逆累積常態分佈 (invNorm( ) 68

除

除, / 163

停

停止指令 137

區

區域性, Local 79
區域性變數, Local 79

帶

帶分數, 使用 propFrac(› with 106

常

常態分布機率, normCdf( ) 96

從

從右取字串，right( ) 67, 118

排

排列
升冪, SortA 133
降冪, SortD 133
排列, nPr( ) 97

接

接上, & 171

梯

梯度, g 176

淨

淨現值, npv( ) 97

清

清除 186
錯誤, ClrErr 22

球

球面座標形式向量顯示 ▸Sphere 133

移

移位, shift( ) 127
移除
列表中的空元素 37

符

符號, sign( ) 128

累

累積總和, cumulativeSum( ) 32

組

組合, nCr( ) 92

統

統計
二變數結果, TwoVar 149
中位數, median( ) 84
平均值, mean( ) 84
排列, nPr( ) 97
組合, nCr( ) 92
單變數統計, OneVar 98
階乘, ! 171
亂數種子，RandSeed 112
標準差, stdDev( ) 136, 151
隨機範數，randNorm( ) 111
變異數, variance( ) 152

設

設定
模式, setMode( ) 126
設定, 取得目前的 60

貨

貨幣時間價值, 付款期數 148
貨幣時間價值, 未來價值 147
貨幣時間價值, 利率 147
貨幣時間價值, 現值 148
貨幣時間價值引數 148
貨幣時間價值函數中的引數 ... 148

單

單位向量, unitV( ) 151
單位矩陣，identity( ) 63
單變數統計, OneVar 98

換

換算
▸Rad 110

最

最大公因數, gcd( ) 54
最大值, max( ) 83
最小公倍數, lcm 70
最小值, min( ) 86

減

減, - 162

無

無條件捨去, floor( ) 50
無條件進入法, ceiling( ) 19, 29

程

程式
定義公用的資料庫 36
定義自訂的資料庫 35
程式設計
顯示資料, 顯示 123
程式與程式設計
顯示 I/O 螢幕, 顯示 123
程式與編程
清除錯誤, ClrErr 22
結束嘗試, EndTry 145
嘗試, Try 145
顯示 I/O 畫面, Disp 38

等

等於, = ... 167

結

結束
　如果，EndIf 63
　函數, EndFunc 53
　迴圈, EndLoop 82
　嘗試, EndTry 145
結束, Exit 46
結束函數, EndFunc 53
結束迴圈, EndLoop 82
結果, 統計 134
結果值, 統計 135

絕

絕對值
　範本 3

虛

虛部，imag( ) 65

註

註解, © 181

間

間接, # 176
間接運算子 (#) 203

階

階乘, ! 171

亂

亂
　數種子，RandSeed 112

傳

傳回，Return 117
傳遞錯誤, PassErr 101

圓

圓柱座標形式向量顯示, ▸Cylind 33

填

填滿 190-191

新

新
　列表, newList( ) 93
　矩陣, newMat( ) 93

極

極
　座標，R▸Pr( ) 109
　座標，R▸Pθ( ) 109
極座標形式
　顯示向量, ▸Polar 102

群

群組, 檢驗鎖定狀態 60
群組, 鎖定和解開鎖定 79, 151

解

解開鎖定變數和變數群組 151

資

資料庫
　建立物件快速鍵 71

運

運算子
　求值順序 202
運算式
　字串至運算式, expr( ) 46

實

實部，real( ) 112
實際年利率, eff( ) 42

對

對數 77
範本 2
對數迴歸, LnReg 77

維

維數, dim( ) 38

語

語言
取得語言資訊 59

數

數值
解, nSolve( ) 98
導數, nDeriv( ) 93-94
導數, nDerivative( ) 92
積分, nInt( ) 94

標

標準差, stdDev( ) 136, 151
標籤, Lbl 70

模

模式
設定, setMode( ) 126
模式設定 , getMode( ) 60
模數, mod( ) 88

範

範本
e 指數 2
N 次方根 1
一階導數 5
二階導數 5
分段函數(2 段) 2
分段函數(N 段) 2
分數 1
平方根 1
定積分 6
指數 1
乘積(P) 5
矩陣(1×2) 4
矩陣(2×1) 4
矩陣(2×2) 4
矩陣(m×n) 4
絕對值 3
對數 2
總和(Σ) 5
聯立方程式(2 個方程式) .. 3
聯立方程式(N 個方程式) .. 3

線

線性迴歸, LinRegAx 72
線性迴歸, LinRegBx 71, 73

編

編程
定義程式, Prgm 105
傳遞錯誤, PassErr 101
顯示資料, Disp 38

複

複製變數或函數, CopyVar 23
複數
共軛, conj( ) 23

質

質數檢定，isPrime( ) 69

餘

餘，remain( ) 115
餘切, cot( ) 27
餘弦, cos( ) 24

導

導數
一階導數, d( ) 172
數值導數, nDeriv( ) 93-94
數值導數, nDerivative( ) 92

整

整數，int( ) 66
整數除法，intDiv( ) 66
整數部份，iPart( ) 69

機

機率密度, normPdf( ) 96

積

積分, ∫ 172

錯

錯誤代碼和訊息 208
錯誤與疑難排除
清除錯誤, ClrErr 22
傳遞錯誤, PassErr 101

隨

隨機
多項式，randPoly( ) 111
矩陣，randMat( ) 111
範數，randNorm( ) 111
隨機樣本 112

儲

儲存
符號, & 180-181

總

總和(Σ)
範本 5
總和, Σ( ) 174

聯

聯立方程式(2 個方程式)
範本 3
聯立方程式(N 個方程式)
範本 3
聯立方程式, simult( ) 129

點

點
和, .+ 165
乘, .* 166
差, .- 165
除, ./ 166
冪, .^ 166
積, dotP( ) 41

擴

擴展/串連, augment( ) 14

轉

轉成
4Grad 63
轉到, Goto 63
轉置, T 139

鎖

鎖定變數和變數群組 79

雙

雙曲
正切, tanh( ) 141
正弦, sinh( ) 131
餘弦, cosh( ) 26

繪

繪圖 187-189

羅

羅吉斯迴歸, Logistic 80
羅吉斯迴歸, LogisticD 81

警

警告代碼和訊息 216

攤

攤銷表, amortTbl( ) 7, 15

變

變異數, variance( ) 152
變數
　刪除, DelVar 36
　區域性, Local 79
　從字元字串建立名稱 203
　清除所有單字元 21
變數, 鎖定和解開鎖定 60, 79, 151
變數和函數
　複製 23

邏

邏輯隱含，⇒ 170, 200
邏輯雙隱含，⇔ 171

顯

顯示
　極座標形式向量, ▸Polar 102
顯示, 顯示資料 123
顯示為
　二進位, ▸Base2 16
　十六進位, ▸Base16 17
　十進位角, ▸DD 34
　十進位整數 , ▸Base10 17
　直角向量，▸Rect 112
　度/分/秒, ▸DMS 40
　球面座標形式向量, ▸Sphere 133
　圓柱座標形式向量, ▸Cylind . 33
顯示資料, Disp 38
顯示資料, 顯示 123