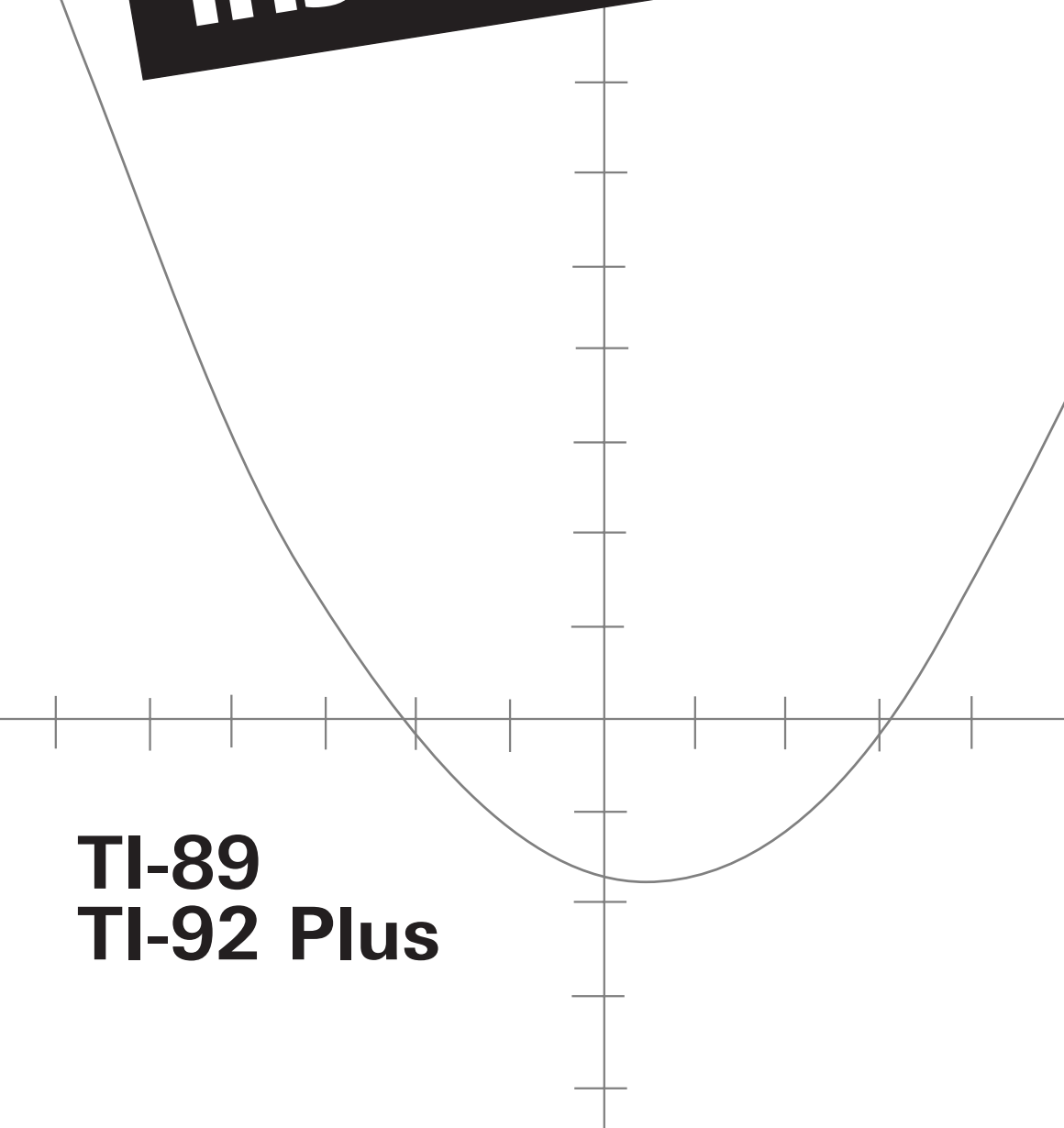
Texas Instruments



TI-89
TI-92 Plus

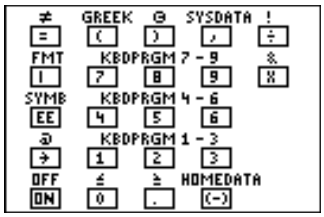
TI-89 Snabbtangenter

Allmänna

◆ APPS	Lista med Flash-applikationer
2nd [⊞]	Växla mellan de två sista programmen eller delade skärmarna
◆ −, ◆ +	Öka/minska kontrasten
◆ ENTER	Beräkna approximativt svar
◆ ⊖, ◆ ⊕	Flytta markören längst upp eller ned (i redigerare)
↑ ⊖, ↑ ⊕	Bläddra stora objekt i historiklistan
↑ ⊖, ↑ ⊕	Markera till vänster eller höger om markören
2nd ⊖, 2nd ⊕	Sida upp och sida ned (i redigerare)
2nd ⊖, 2nd ⊕	Flytta markören längst till vänster eller höger

Tangentbord på skärmen (◆ EE)

Tryck på ESC när du vill stänga det.



På tangentbordet på skärmen visas genvägar som inte är markerade på det vanliga tangentbordet. Tryck på ◆ enligt beskrivningen nedan och sedan på önskad tangent.

◆ =	≠
◆ (	Grekiska tecken (se nästa kolumn)
◆)	◉ (kommentar)
◆ ,	Kopiera grafkoordinater till sysdata
◆ ÷	! (faktorisera)
◆ |	Öppna dialogrutan FORMATS
◆ 1 – ◆ 9	Kör program kbdprgm1() till kbdprgm9()
◆ ×	& (lägg till)
◆ EE	Tangentbordskarta
◆ STO▸	@
◆ ON	Stänger av enheten så att den återgår till aktuellt program nästa gång den startas
◆ 0 (noll)	≤
◆ .	≥
◆ (-)	Kopiera grafkoordinater till grundfönstrets historiklista

Skiftlås

alpha	Skriv in ett gement tecken
↑	Skriv in ett versalt tecken
2nd [a-lock]	Skiftlås för gemener
↑ alpha	Skiftlås för versaler
alpha	Deaktivera skiftlås

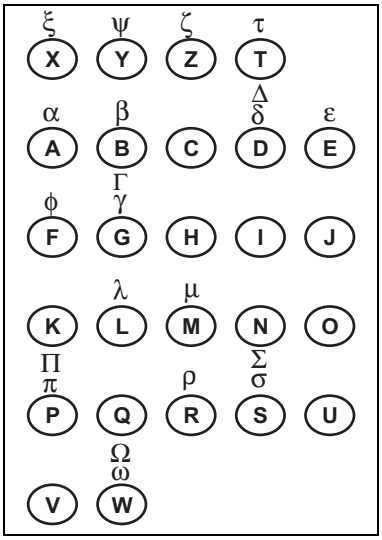
3D-plottning

⊖, ⊕, ⊖, ⊕	Animera graf
+, −	Ändra animeringshastighet
X, Y, Z	Visa längs axlar
0 (noll)	Återgå till orginalvy
|	Ändra stil för graf
×	Expanderad/normal vy

Grekiska tecken

◆ (	Använd grekiska tecken.
◆ (alpha + tecken	Använd gemena grekiska tecken. Exempel: ◆ (alpha [W] ger ω.
◆ (↑ + tecken	Använd versala grekiska tecken. Exempel: ◆ (↑ [W] ger Ω

Om du trycker ned en kombination av tecken som inte motsvarar något grekiskt tecken används det normala tecknet för den tangenten.



Texas Instruments

TI-89
TI-92 Plus
Handbok

för Advanced Mathematics
Software Version 2.0

U.S. Patent No. 4,405,829 Licensierat av RSA Data Security, Inc.

© 1999 by Texas Instruments

Viktigt

Texas Instruments lämnar inga uttryckliga eller underförstådda garantier för något program eller bok. Detta innefattar, men är inte begränsat till, underförstådda garantier om säljbarhet eller lämplighet för ett visst ändamål. Materialet tillhandahålles enbart på “som det är”-basis.

Inte i något fall skall Texas Instruments kunna hållas ansvarigt för speciella eller sekundära skador, skador på grund av olyckor eller följdskador i anslutning till eller härrörande från inköp eller användning av detta material. Det enda betalningsansvar som Texas Instruments påtar sig, oaktat handling, skall inte överstiga inköpspriset för denna utrustning. Dessutom skall inte Texas Instruments ha något betalningsansvar för några krav avseende användning av detta material från annan part.

Innehållsförteckning

I denna handbok beskrivs hur du använder din TI-89 / TI-92 Plus. I innehållsförteckningen hittar du snabbt information för att "komma igång" med din TI-89 / TI-92 Plus. Appendix A är ett enkelt sätt att hitta detaljer om varje funktion och instruktion i din TI-89 / TI-92 Plus.

Flash-applikation x
Tangentskillnader xii
Vad är nytt? xiv

Kapitel 1: Komma igång

Innan du börjar använda TI-89 2
Innan du börjar använda TI-92 Plus 3
Ställa in kontrast och välja språk 4
Utföra beräkningar 8
Plotta en funktion 11

Kapitel 2: Använda räknaren

Slå på och stänga av TI-89 / TI-92 Plus 14
Justera kontrasten i teckenfönstret 15
Tangentbordet TI-89 16
Tangentbordet TI-92 Plus 17
Funktionstangenter 18
Mata in alfabetiska tecken 21
Grundfönstret 23
Skriva tal 25
Mata in uttryck och instruktioner 26
Format på visade resultat 29
Redigera ett uttryck på inmatningsraden 32
Menyer 34
Använda menyn Custom 37
Välja ett applikationer 38
Ställa in lägen 40
Använda menyn Clean Up för att påbörja ett nytt problem 43
Använda dialogrutan Catalog 44
Spara och hämta variabelvärden 47
Återanvända en tidigare inmatning eller det senaste resultatet 49
Automatiskt klistra in en inmatning eller ett resultat från historiklistan 52
Indikatorer för statusfältet i fönstret 53
Visa applikationerversion och ID-nummer 55

Kapitel 3: Algebra

Översikt över algebra ... 58
Använda definierade eller odefinierade variabler ... 59
Använda Exact-, Approximate- och Auto-lägena ... 61
Automatisk förenkling ... 64
Fördröjd förenkling för vissa inbyggda funktioner ... 66
Substituera värden och ange restriktioner ... 67
Översikt över menyn Algebra ... 70
Vanliga algebraiska operationer ... 72
Översikt över menyn Calc ... 75
Vanliga beräkningsoperationer ... 76
Användardefinierade funktioner och symbolisk hantering ... 77
Felmeddelande om att minnet håller på att ta slut ... 79
Specialkonstanter som används i algebra ... 80

Kapitel 4: Konstanter och Måttenheter

Konstanter och måttenheter ... 82
Ange konstanter och måttenheter ... 83
Konvertera mellan måttenheter ... 85
Ange förvalda måttenheter för visningsläge ... 87
Skapa egna måttenheter ... 88
Fördefinierade konstanter och måttenheter ... 89

Kapitel 5: Ytterligare funktioner i grundfönstret

Spara inmatningar i grundfönstret som ett Text Editor-skript ... 94
Klippa ut, kopiera och klistra in information ... 95
Skapa och beräkna användardefinierade funktioner ... 97
Använda mappar för att spara oberoende uppsättningar variabler ... 100
Om storleken på en inmatning eller ett resultat är för stor ... 103

Kapitel 6: Grundläggande funktion splottning

Översikt över grundläggande funktionsplottning ... 106
Översikt över stegen i funktionsplottning ... 107
Ställa in grafläget ... 108
Definiera funktioner som ska plottas ... 109
Välja vilka funktioner som ska plottas ... 111
Ange visningsformat för funktioner ... 112
Definiera visningsfönstret ... 113
Ändra grafformatet ... 114
Plotta de markerade funktionerna ... 115
Visa koordinater med den rörliga markören ... 116
Spåra en funktion ... 117
Använda zoomning för att utforska en graf ... 119
Använda matematikverktyg för att analysera funktioner ... 122

Kapitel 7: Plotta parameterkurvor

Förhandsvisning över plottning av kurvor i parameterform ... 128
Översikt över plottning av parameterekvationer ... 129
Skillnader mellan parameter- och funktionsplottning ... 130

Kapitel 8: Plotta polära ekvationer

Förhandsvisning över polär plottning ... 134
Översikt över stegen i att plotta polära ekvationer ... 135
Skillnader mellan polär- och funktionsplottning ... 136

Kapitel 9: Plotta talföljder

Översikt över plottning av talföljder ... 140
Översikt över stegen i plottning av talföljder ... 141
Skillnader mellan plottning av talföljder och funktioner ... 142
Ställa in plottning med axlar för Time, Web eller Custom ... 146
Använda plottning med WEB-axlar ... 147
Använda plottningar med Custom-axlar ... 150
Använda talföljder för att generera tabeller ... 151

Kapitel 10: 3D-plottning

Nya funktioner för plottning av 3D-grafer ... 154
Översikt över stegen i plottning av 3D-ekvationer ... 156
Skillnader mellan 3D- och funktionsplottning ... 157
Flytta markören i 3D ... 160
Rotera en 3D-graf med variabeln eyeψ ... 162
Animera en 3D-graf interaktivt ... 164
Ändra koordinataxlarna och visningsformatet ... 165
Konturplottning ... 167
Exempel: Konturer hos en komplex modulär yta ... 170
Implicit plottning ... 171
Exempel: Implicit plottning av en mer komplicerad ekvation ... 173

Kapitel 11: Plottning av differentialekvationer

Plottning av differentialekvationer - översikt ... 176
Plottning av lösningar till differentialekvationer - steg för steg ... 178
Skillnader mellan funktionsplottning och plottning av lösningar till differentialekvationer ... 179
Ange begynnelsevillkor ... 184
Definiera ett system för ekvationer av högre ordning ... 186
Exempel på en ekvation av andra ordningen ... 187
Exemple på en ekvation av tredje ordningen ... 189
Ställa in axlar till Time eller Custom ... 190
Exempel på axlar (Time och Custom) ... 191
Jämförelse av RK och Euler ... 193
Exempel på funktionen deSolve( ) ... 196
Felsökning av grafformat ... 197

Kapitel 12: Ytterligare graffunktioner

Översikt över ytterligare grafverktyg 202
Spara datapunkter från en graf 203
Plotta en funktion som är definierad i grundfönstret 204
Plotta en funktion som består av flera delar 206
Plotta en familj av kurvor 208
Använda tvågrafsläget 209
Rita en funktion eller en invers på en graf 212
Rita en linje, cirkel eller textetikett på en graf 213
Spara och öppna en bild av en graf 217
Animera en serie grafbilder 219
Spara och öppna en grafdatabas 220

Kapitel 13: Tabeller

Översikt över tabeller 222
Översikt över stegen i tabellgenerering 223
Mata in tabellparametrar 224
Visa en automatisk tabell 226
Skapa en manuell tabell 229

Kapitel 14: Delade fönster

Förhandsvisning av funktioner för delade fönster 232
Aktivera och avaktivera läget för delat fönster 233
Markera vilken applikation som ska vara aktiv 235

Kapitel 15: Data/Matrix Editor

Förhandsvisning av funktioner i Data/Matrix Editor 238
Översikt över list-, data- och matrisvariabler 239
Starta en Data/Matrix Editor-session 241
Mata in och visa cellvärden 243
Infoga och ta bort rader, kolumner och celler 246
Definiera ett kolumnhuvud med ett uttryck 248
Använda funktionerna Shift och CumSum i ett kolumnhuvud 250
Sortera kolumner 251
Spara en kopia av en list-, data- eller matrisvariabel 252

Kapitel 16: Plotta statistiska data

Förhandsvisning över plottning av statistiska data 254
Översikt över stegen i statistisk analys 259
Utföra en statistisk beräkning 260
Statistiska beräkningstyper 262
Statistiska variabler 264
Definiera plottning av statistiska data 265
Statistiska plottningstyper 267
Använda Y= Editor med statistiska plottningar 269
Plotta och spåra en definierad statistisk plottning 270
Använda frekvenser och kategorier 271
Om du har tillvalsprogrammet CBL och CBR 273

Kapitel 17: Programmering

Förhandsvisning av programmeringsfunktioner 276
Köra befintliga program 278
Starta en session i Program Editor 280
Översikt över inmatning av program 282
Översikt över inmatning av funktioner 285
Anropa ett program från ett annat 287
Använda variabler i ett program 288
Använda lokala variabler i funktioner och program 290
Strängoperationer 292
Villkorliga tester 294
Använda If, Lbl och Goto för att styra programflödet 295
Använda slingor för att repetera en grupp kommandon 297
Konfigurera TI-89 / TI-92 Plus 300
Låta användaren göra en inmatning och sedan visa utmatningen 301
Skapa en anpassad meny 303
Skapa en tabell eller graf 305
Rita i graffönstret 307
Använda en annan TI-89 / TI-92 Plus, en CBL eller en CBR 309
Felsöka program och hantera fel 310
Exempel: Använda alternativa metoder 311
Assemblerprogram 313

Kapitel 18: Text Editor

Förhandsvisning av textfunktioner 316
Starta en session i Text Editor 317
Skriva och redigera text 319
Skriva specialtecken 324
Mata in och köra ett kommandoskript 328
Skapa en labbrapport 330

Kapitel 19: Ekvationslösaren

Förhandsvisning av den numeriska ekvationslösaren 334
Öppna Solver och skriv in en ekvation 335
Definiera kända variabler 337
Lösa för den okända variabeln 339
Plotta lösningen 340

Kapitel 20: Talsystem

Talsystem 344
Skriva in och konvertera talsystem 345
Beräkningar med binära och hexadecimala tal 346
Jämföra eller manipulera enskilda bitar 347

Kapitel 21: Minnes- och variabelhantering

Översikt över minnes- och variabelhantering ... 350
Kontrollera och nollställa minnet ... 354
Visa VAR-LINK-fönstret ... 356
Hantera variabler och mappar med VAR-LINK ... 358
Klistra in ett variabelnamn i ett program ... 360
Spara och hämta variabler ... 361
Om ett meddelande om Garbage Collection visas ... 363
Minnesfel vid läsning av sparad variabel ... 365

Kapitel 22: Överföring av variabler och uppgradering

Länka två enheter ... 366
Överföra variabler, Flash-program och mappar ... 367
Överföra variabler under från ett program ... 371
Uppgradera programkod (baskod) ... 373
Samla in och överföra ID-listor ... 378
Kompatibilitet mellan TI-89, TI-92 Plus, och TI-92 ... 380

Kapitel 23: Övningar

Analysera problemet med en påle som ska runt ett hörn ... 384
Härledning av formeln för andragradsekvationens rötter ... 386
Utforska en matris ... 388
Utforska cos(x) = sin(x) ... 389
Hitta den minsta ytarean på en parallellepiped ... 390
Köra ett självstudieskript med Text Editor ... 392
Dela upp en rationell funktion ... 394
Studera statistik: Filtrera data med kategorier ... 396
Ett CBL program för TI-89 / TI-92 Plus ... 399
Analysera färden för en ivägslagen boll ... 400
Visualisera komplexa rötter till ett tredjegradspolynom ... 402
Lösa ett allmänt sparproblem ... 404
Beräkna betalning ... 405
Hitta rationella, reella och komplexa faktorer ... 406
Simulering av dragning utan återläggning ... 407

Appendix A: Funktioner och instruktioner

Snabbguide ... 410
Alfabetisk lista över operationer ... 414

Appendix B: Referensinformation

TI-89 / TI-92 Plus felmeddelanden ... 538
Lägen ... 546
TI-89 / TI-92 Plus teckenkoder ... 551
TI-89 Tangentkoder ... 552
TI-92 Plus tangentkoder ... 555
Skriva Komplexa tal ... 559
Information om noggrannhet ... 562
Systemvariabler och reserverade namn ... 563
Hierarkin EOS (Equation Operating System) ... 564
Regressionsformler ... 566
Konturnivåer och algoritmer för implicit plottning ... 568
Runge-Kuttas metod ... 569

Appendix C: Information om service och garanti

Batterierna 572
Om du får problem 575
TI produktservice och garantiinformation 576

Appendix D: Programmerings-handbok

setModus() och lesModus() 578
setGraf() 581
setTab() 583

Innhållsförteckning 585

TI-89 Snabbtangenter ..... omslagets främre insida
TI-92 Plus Snabbtangenter ..... omslagets bakre insida

Flash-applikation

Program

Med Flash-tekniken kan du ladda ner olika applikation till din TI-89 / TI-92 Plus från den medföljande CD-skivan, TI:s hemsidor eller från andra räknare.

Innan du laddar ner applikationer till din TI-89 / TI-92 Plus måste du läsa igenom och godkänna licensavtalet på applikation-CD:n för din TI-89 / TI-92 Plus.

Hårdvaru-/ applikation-varukrav

Innan du kan installera Flash-applikation behöver du:

- En dator med en CD-enhet och en serieport.
- TI-GRAPH LINK™, säljs separat och inkluderar applikationvara samt en kabel som kopplar samman datorn och räknaren. Om du behöver TI-GRAPH LINK eller en kabel hittar du mer information på TI:s hemsidor: **http://www.ti.com/calc/docs/link.htm**

Inställningar på datorn

Gör så här:

1. Sätt i den lilla kontakten på TI-GRAPH LINK-kabeln i porten på undersidan av din TI-89 eller överst på en TI-92 Plus.
2. Anslut den andra änden av kabeln till serieporten på din dator med en adapter (om nödvändigt, 25-9 stift).

Installera Flash-applikation från CD-skivan

Obs! *Mer information om hur du överför data mellan en dator och en räknare finns i handboken för TI-GRAPH LINK.*

När du vill installera ett applikation:

1. Sätt i applikation-CD:n för din TI-89 / TI-92 Plus i datorns CD-spelare.
2. Starta applikationmet TI-GRAPH LINK på din dator.
3. Välj Send Flash Software ▶ Applications and Certificates på menyn Link.
4. Leta rätt på önskad Flash-applikation på CD-skivan och dubbelklicka på symbolen. Flash-applikationmet kopieras till räknaren.

Köra Flash-applikation

När du vill köra en applikation:

1. On the , Tryck på [◆] [APPS] på din TI-89 / TI-92 Plus för att öppna menyn FLASH APPLICATIONS.
2. Använd piltangenterna ⊕⊖ för att markera önskad applikation och tryck sedan på [ENTER].

Överföra ett Flash-applikation från en annan TI–89 / TI–92 Plus

Försök aldrig att överföra applikation om ett meddelande om låg batterispänning visas på någon av räknarna.

1. Koppla samman räknarna med kabeln för anslutning av två enheter som medföljde din TI–89 / TI–92 Plus.
2. På den sändande räknaren:
   a. Tryck på [2nd] [VAR-LINK]
   b. Tryck på:
      TI-89: [2nd] [F7]
      TI-92 Plus: [F7]
   c. Markera önskat Flash-applikation och tryck på [F4] (tecknet ✓ visas till vänster om markerat alternativ)

Obs! *I denna handbok används skärmbilder från en TI–89.*

3. På den mottagande räknaren:
   a. Tryck på [2nd] [VAR-LINK]
   b. Tryck på [F3]
   c. Välj: 2:Receive
   d. Tryck på [ENTER]

4. På den sändande räknaren:
   a. Tryck på [F3]
   b. Välj: 1:Send to TI-89/92 Plus
   c. Tryck på [ENTER]

Säkerhetskopiera Flash-applikation

När du vill säkerhetskopiera ett applikation till en dator:

1. På din TI–89 / TI–92 Plus, tryck på:
   TI-89: [HOME]
   TI-92 Plus: [♦] [HOME]
2. Starta applikationmet TI-GRAPH LINK på din dator
3. Välj Receive Flash Software på menyn Link
4. Markera ett eller flera Flash-applikation och klicka på Lägg till
5. Klicka på OK
6. Spara applikationmet på datorn och gör en notering så att du hittar igen det senare.

Obs! *Mer information om hur du överför data mellan en dator och en räknare finns i handboken för TI-GRAPH LINK.*

Ta bort Flash-applikation

När du vill ta bort ett applikation från räknaren:

1. Tryck på [2nd] [VAR-LINK] för att öppna fönstret VAR-LINK
2. Tryck på:
   TI-89: [2nd] [F7]
   TI-92 Plus: [F7]
3. Markera önskat Flash-applikation och tryck på [F4] (tecknet ✓ visas till vänster om markerat alternativ)
4. Tryck på [F1] och välj 1:Delete
   — eller —
   Tryck på [←] (ett meddelande visas)
5. Tryck på [ENTER] för att bekräfta borttagningen.

Obs! *Om du vill markera alla Flash-applikation kan du välja alternativet [F5] All.*

Tangentskillnader

Det finns vissa skillnader på tangenterna mellan TI-89 / TI-92 Plus för olika operationer. I följande tabell hittar du tangentsekvenserna för de vanligaste kommandona på de två räknarna.

FUNKTION	TI-89	TI-92 Plus
BOKSTÄVER		
Ett gement tecken (a-s, u, v, w)	[alpha] A-S, U-W	A-S, U-W
Ett gement tecken (t, x, y, z)	T, X, Y, Z	T, X, Y, Z
Flera gemena tecken	[2nd] [a-lock]	
Avbryt läge för gemena tecken	[alpha]	
Flera versala tecken	[↑] [a-lock]	[2nd] [CAPS]
Avbryt läge för versala tecken	[alpha]	[2nd] [CAPS]
FUNKTIONSTANGENTER		
F6	[2nd] [F6]	[F6]
F7	[2nd] [F7]	[F7]
F8	[2nd] [F8]	[F8]
NAVIGERING		
Bläddra objekt uppåt och nedåt i historiklistan	[↑] ⊕, [↑] ⊖	[⊡] ⊕, [⊡] ⊖
Flytta markören längst till vänster/höger	[2nd] ⊖, [2nd] ⊕	[2nd] ⊖, [2nd] ⊕
Diagonal rörelse	⊕ och ⊖ ⊕ och ⊕ ⊖ och ⊖ ⊖ och ⊕	⊖ ⊕ ⊖ ⊕
FUNKTIONER		
Öppna grundfönstret	[HOME]	[◆] [HOME]
Klipp ut	[◆] [CUT]	[◆] X
Kopiera	[◆] [COPY]	[◆] C
Klistra in	[◆] [PASTE]	[◆] V
Katalog	[CATALOG]	[2nd] [CATALOG]
Öppna dialogrutan Enheter	[2nd] [UNITS]	[◆] [UNITS]
Sinus	[2nd] [SIN]	[SIN]
Cosinus	[2nd] [COS]	[COS]
Tangent	[2nd] [TAN]	[TAN]
LN	[2nd] [LN]	[LN]
e^x	[◆] [e^x]	[2nd] [e^x]
EE	[EE]	[2nd] [EE]

FUNKTION	TI-89	TI-92 Plus
SYMBOLER		
▶ (Omvandlingstriangel)	[2nd] [▶]	[2nd] [▶]
_ (Understrykning)	[◆] [_]	[2nd] [_]
θ (Theta)	[◆] [θ]	[θ]
| ("With")	[|]	[2nd] [|]
' (Prim)	[2nd] [′]	[2nd] [′]
° (Grader)	[2nd] [°]	[2nd] [°]
∠ (Vinkel)	[2nd] [∠]	[2nd] [∠]
Σ (Sigma)	[CATALOG] Σ (	[2nd] [Σ]
x^{-1} (Reciprok)	[CATALOG] ^-1	[2nd] [x^{-1}]
Blanksteg	[alpha] [␣]	Blanksteg
GENVÄGAR		
Spara data i variabeln sysdata	[◆] [,]	[◆] D
Grekiska tecken	[◆] [(] [alpha] eller [◆] [(] [↑]	[◆] G eller [◆] G [↑]
Tangentbordskarta	[◆] [EE]	[◆] [KEY]
Spara data i historiklistan	[◆] [(-)]	[◆] H
Grave (à, è, ì, ò, ù)	[2nd] [CHAR] 5	[2nd] A a, e, i, o, u
Cedilla (ç)	[2nd] [CHAR] 5 6	[2nd] C c
Acute (á, é, í, ó, ú, ý)	[2nd] [CHAR] 5	[2nd] E a, e, i, o, u, y
Tilde (ã, ñ, õ)	[2nd] [CHAR] 5 6	[2nd] N a, n, o
Caret (â, ê, î, ô, û)	[2nd] [CHAR] 5	[2nd] O a, e, i, o, u
Umlaut (ä, ë, ï, ö, ü, ÿ)	[2nd] [CHAR] 5	[2nd] U a, e, i, o, u, y
? (Frågetecken)	[2nd] [CHAR] 3	[2nd] Q
β (Beta)	[2nd] [CHAR] 5 6	[2nd] S
# (Indirection)	[2nd] [CHAR] 3	[2nd] T
& (Lägg till)	[◆] [×] (gånger)	[2nd] H
@ (Arbitrary)	[◆] [STO▶]	[2nd] R
≠ (Inte lika med)	[◆] [=]	[2nd] V
! (Faktorisering)	[◆] [÷]	[2nd] W
Kommentar (Cirkel-C)	[◆] [)] ◉	[2nd] X ◉
Nytt	[F1] 3	[◆] N
Öppna	[F1] 1	[◆] O
Spara kopia som	[F1] 2	[◆] S
Dialogrutan Format	[◆] [|]	[◆] F

Vad är nytt?

Vi introducerar Advanced Mathematics Software Version 2.0

TI har utvecklat Advanced Mathematics Software Version 2.0 för att möjliggöra nedladdningsbara program för TI-89 och TI-92 Plus.

Mer information:
kapitel 21 och 22

Advanced Mathematics Software Version 2.0 är en förbättrad version av nuvarande Advanced Mathematics Software Version 1.xx. Programmet innehåller alla funktioner som finns i version 1.xx. Den nya, utökade, versionen tillåter fler nedladdningsbara applikationer och språkanpassning. Den förbättrade versionen gör också att begränsningen hos TI-89 / TI-92 Plus vad gäller flyttning av mer än 702 Kb Flash-minne till och från användarminnet och räknaren är löst.

Alla tidigare TI-89 och TI-92 Plus Moduler kan uppgraderas till version 2.0. På en del TI-89 och alla TI-92 Plus Moduler kan dock användarminnet bara använda 384-KB av minnet som är större än det 702-KB Flash-minne som delas med räknarens programvaror.

Du kan ladda ner Advanced Mathematics Software Version 2.0 till din dator från TI:s hemsidor: **http://www.ti.com/calc/flash**, och sedan överföra det till din TI-89 / TI-92 Plus med programmet TI-GRAPH LINK™ och en speciell kabel (sälj separat). Du kan även överföra programmet mellan två TI-89 / TI-92 Plus med hjälp av kabel som används för att koppla samman två räknare. Programvaran Advanced Mathematics Software kan hämtas gratis på **http://www.ti.com/calc/flash**

Språkanpassning

Din TI-89 / TI-92 Plus kan anpassas till flera olika språk. Dessa gratisprogram översätter promptar, felmeddelanden och de flesta funktioner till ett av flera möjliga språk.

Mer information:
kapitel 1

Förbättrat användargränssnitt

Det förbättrade användargränssnittet låter dig öppna/stänga mappar samt utökar menyn CATALOG till att omfatta även programfunktioner och användardefinierade funktioner.

Uppgradera med Flash-ROM

Flash-tekniken i TI-89 / TI-92 Plus låter dig uppgradera din räknare med nya applikationer utan att du behöver köpa en ny räknare.

Mer information:
kapitel 22

När nya funktioner blir tillgängliga kan du uppgradera din TI-89 / TI-92 Plus elektroniskt. Framtida programversioner omfattar underhållsuppgraderingar som är gratis liksom nya program och större uppdateringar som kan köpas från TI:s hemsidor.

När du vill ladda ner uppdateringar från TI måste du ha en Internet-ansluten dator, programvaran TI-GRAPH LINK™ och en kabel för att koppla samman din dator och din räknare (säljs separat). Du kan även överföra programvaror (baskod) och Flash-applikationer från en TI-89 / TI-92 Plus till en annan med kabeln för att koppla samman två räknare. En förutsättning är dock att den mottagande räknare har ett certifikat för programvaran.

Egna menyer

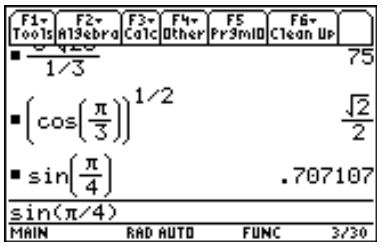
Nytt på TI-89 / TI-92 Plus är en funktion som låter dig skapa egna menyer. Dina menyer kan innehålla alla funktioner, instruktioner och teckenuppsättningar. Din TI-92 Plus har en förvald meny som du kan anpassa och definiera om.

Komma igång

1

Innan du börjar använda TI-89 2
Innan du börjar använda TI-92 Plus 3
Ställa in kontrast och välja språk 4
Utföra beräkningar 8
Plotte en funktion 11

I det här kapitlet får du hjälp med att snabbt komma igång med att använda TI-89 / TI-92 Plus. Kapitlet innehåller flera exempel som visar några av de vanligaste funktionerna och graffunktionerna.



När du har installerat TI-89 / TI-92 Plus och utfört exemplen som följer, bör du läsa kapitel 2: Använda räknaren. Därefter är du redo att gå vidare till den mer detaljerade informationen som finns i de återstående kapitlen i denna handbok.

TI-89 levereras med fyra AAA-batterier. I kapitlet beskrivs även hur du slår på enheten första gången, hur du ställer in kontrasten, väljer språk och visar grundfönstret på både TI-89 och TI-92 Plus.

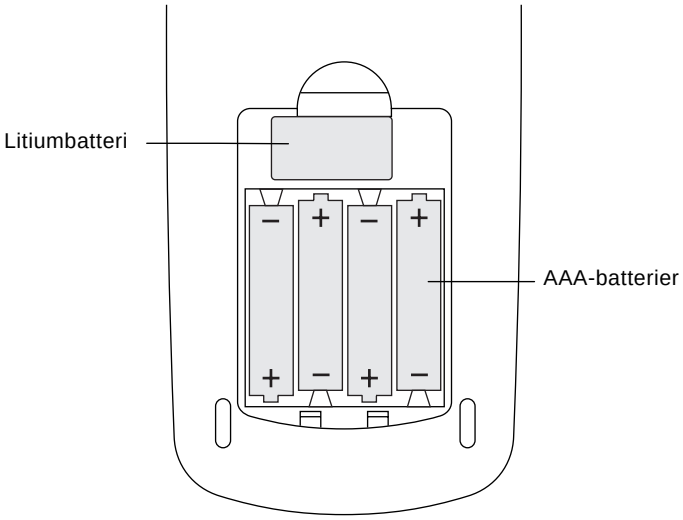
Installera batterier av typ AAA

Gör så här när du vill installera de fyra batterierna (storlek AAA):

1. Lägg din TI-89 på en mjuk duk med framsidan nedåt.
2. Tryck ned låsningen för batteriluckan. Lyft upp och ta bort luckan.
3. Ta ur batterierna ur förpackningen och lägg ned dem i batterifacket. Kontrollera att du sätter i batterierna åt rätt håll enligt bilden (+ och −) i batterifacket.
4. Sätt tillbaks batteriluckan genom att först passa in de två pinnarna i hålen i räknaren. Skjut sedan in luckan tills den låses i läge med snäppfästet.

Viktigt! *När du ska byta batterier, se till att räknaren är avstängd genom att trycka på* [2nd] [OFF].

I appendix C finns information om hur du kan byta batterier utan att förlora någon information som finns lagrat i minnet.



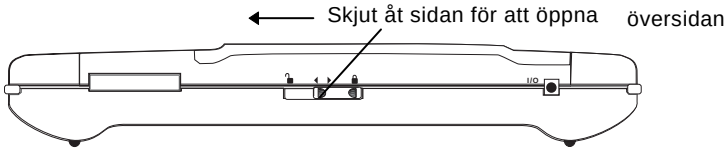
TI-92 Plus levereras med fyra AA-batterier. I kapitlet beskrivs även hur du slår på enheten första gången, hur du ställer in kontrasten, väljer språk och visar grundfönstret på både TI-92 Plus och TI-89.

Installera batterier av typ AA

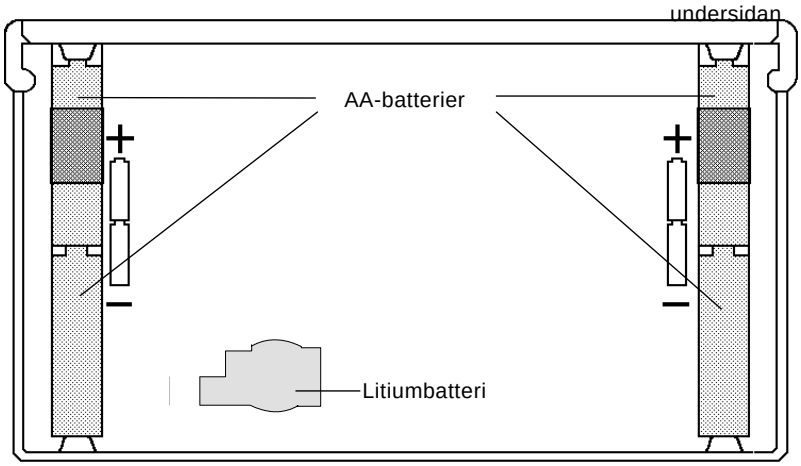
Så här sätter du i de fyra alkaliska AA-batterierna:

1. Håll din TI-92 Plus upprätt och skjut låsningen överst på enheten till vänster, skjut locket överst på enheten till höger (olåst position) med fingret och ta bort luckan på baksidan av räknaren.

Viktigt! *När du ska byta batterier, se till att räknaren är avstängd genom att trycka på* 2nd [OFF].



2. Lägg räknaren upp och ned på en mjuk trasa, för att förhindra att teckenfönstret blir repat.
3. Sätt i de fyra AA-batterierna. Placera batterierna enligt schemat inuti räknaren. Plussidan (+) på varje batteri ska peka mot ovansidan av räknaren.



4. Sätt tillbaks locket och skjut låsningen till höger tills locket låses i rätt läge.

Ställa in kontrast och välja språk

Slå på enheten och justera kontrasten

När du har satt i batterierna i din TI-89 / TI-92 Plus, tryck på [ON]. Det kan inträffa att bildskärmens kontrast är för ljus eller mörk för att du ska kunna se någonting på skärmen.

Justera bildskärmens kontrast genom att hålla ned [◆] (diamant med en grön ram) och tryck på [−] (minus) om du vill göra bilden ljusare. Håll ned [◆] och tryck på [+] (plus) om du vill göra bilden mörkare.

När kontrasten är korrekt inställd visas en lista med flera olika språk. Listan med språk kan se annorlunda ut på din räknare.

Språk på TI-89 / TI-92 Plus

Andra språk än engelska finns tillgängliga som Flash-applikationer. Engelska är en del av programkoden (baskoden). Du kan ha hur många, språk du vill på din räknare (begränsas endast av tillgängligt minne) och enkelt växla mellan dem vid behov. Under processen nedan kommer du att kunna välja vilka språk du vill använda och vilka du vill ta bort. Du kan också lägga till och ta bort olika språk i fönstret VAR-LINK.

Viktig infomation om språkanpassning

TI-89 / TI-92 Plus kan anpassas till flera olika språk. Språkanpassning innebär att alla menyer, dialogrutor, felmeddelanden, etc visas på det språk du valt.

TI-89 / TI-92 Plus kan bara anpassas till ett språk i taget, men du kan ha flera språk installerade på räknaren och växla mellan dem när du vill.

Den inledande språkanpassningen av TI-89 / TI-92 Plus sker i tre steg:

- **Steg I -** Välj vilket språk du vill anpassa din TI-89 / TI-92 Plus till. Installationsbeskrivningen visas sedan på valt språk.
- **Steg II -** Läs instruktionerna som visas på skärmen (på det språk du valde under steg I).
- **Steg III -** Din TI-89 / TI-92 Plus anpassas till det språk du valde under steg I. Du kan nu välja ett eller flera ytterligare språk som du också vill ha på din räknare (om du vill växla till ett annat språk senare). Du kan alltid ladda in fler språk senare om nödvändigt. Räknaren kommer då automatiskt att ta bort språk som inte är markerade (förutom engelska).

Obs! *Engelska kan inte tas bort och finns alltid tillgänglig i programmets baskod.*

Ändra språk på TI-89 / TI-92 Plus

1. Tryck på piltangenterna (⊖ eller ⊕) och flytta markören till det språk du vill använda på din TI-89 / TI-92 Plus. (Listan med språk kan se annorlunda ut på din räknare.)

Obs! *Om du inte väljer språk kommer denna dialogruta att visas varje gång du slår på räknaren.*

2. Tryck på [ENTER] för att ställa in din TI-89 / TI-92 Plus till önskat språk. (Om du trycker på [ESC] avbryts språkanpassningen och grundfönstret öppnas.)

3. Läs meddelandet som visas och tryck sedan på [ENTER].

Detta meddelande visas på det språk du tidigare valt.

4. Tryck på piltangenterna (⊖ eller ⊕) för att flytta markören och tryck på [F1] för att markera varje extra språk du vill behålla.
— eller —
Tryck på [F2] för att markera och behålla *alla* språken.

Du kan inte avmarkera engelska eller det språk du valde i steg 1.

Tryck på [F1] för att växla ✓ av och på.

5. Tryck på [ENTER] för att slutföra språkanpassningen. De extra språk du valt, om du nu valt några behålls i minnet och de andra språken tas bort för att frigöra Flash-minne. (Om du trycker på [ESC] avbryts språkanpassningen och grundfönstret öppnas.)

Om du behöll några extra språk på din TI-89 / TI-92 Plus kan du ändra språk på Page 3 ([F3]) i dialogrutan Mode. Se “Lägesinställningar” i kapitel 2 för mer information om hur du använder dialogrutan Mode. Du kan senare ta bort och lägga till språk och andra Flash-applikationer i fönstret VAR-LINK. Se “Överföra variabler, Flash-applikationer och mappar” i kapitel 22.

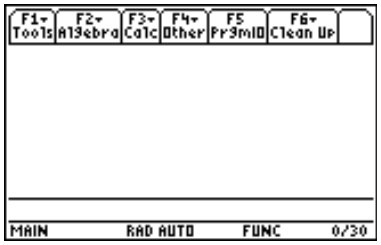
Olika språkapplikationer finns på den medföljande CD-skivan och på Texas Instruments hemsidor. Om du vill läsa den senaste informationen om Flash-applikationer och extra språk kan du gå till Texas Instruments hemsidor:

http://www.ti.com/calc

Om grundfönstret

När du först slår på räknaren visas ett tomt grundfönster.

I grundfönstret utför du instruktioner, beräknar uttryck och visar resultat.

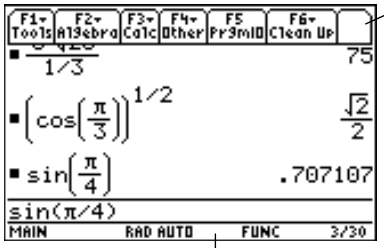


I följande exempel visas nyligen inskriven information och en beskrivning av huvuddelarna i grundfönstret. Inmatningar och resultat i historiklistan visas i pretty print-läget. Med "Pretty Print" skrivs uttryck ut, på samma sätt som vid beräkningar på svarta tavlan eller i läroböcker.

Historiklista
Visar par av inmatningar och resultat som du beräknat. Paren flyttas uppåt i fönstret när du matar in nya beräkningar.

Senaste inmatning
Den sista inmatningen.

Inmatningsrad
Här matar du in uttryck och instruktioner.



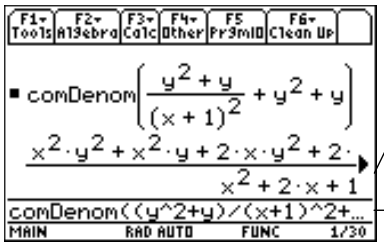
Verktygsfält
Visar menyer med kommandon som är anpassade till grundfönstret. Tryck på F1, F2, osv för att visa en meny.

Senaste resultat
Resultatet av den sista inmatningen. Observera att resultaten inte visas på inmatningsraden.

Statusfält
Visar räknarens nuvarande läge.

Följande exempel visar ett svar som inte är på samma rad som det inmatade uttrycket uttrycket. Observera att svaret är längre än skärmens bredd. Pilen (▶) anger att svaret fortsätter utanför skärmbilden. På inmatningsraden visas tre punkter (...). Detta visar att inmatningen är större än skärmens bredd.

Senaste inmatning
”Pretty Print” är aktiverat. Exponenter, rötter, bråktal etc, visas i samma format som de traditionellt skrivs.



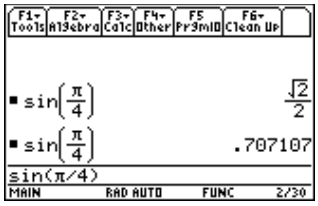
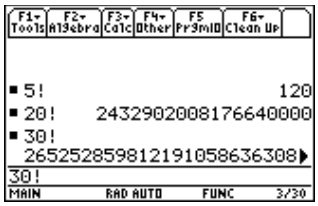
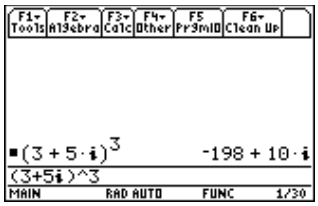
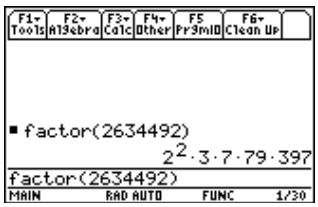
Svaret fortsätter
Markera svaret och tryck på ⊙ när du vill bläddra till höger och se resten av svaret. Observera att svaret inte är på samma rad som uttrycket.

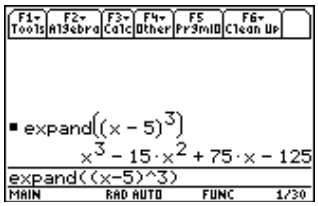
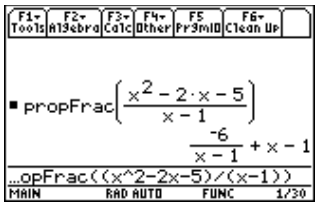
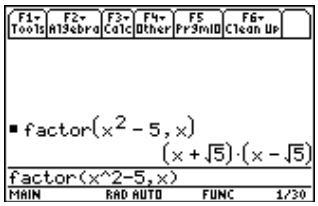
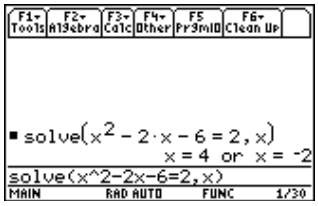
Uttrycket fortsätter
Tryck på ⊙ för att bläddra till höger och se resterande inmatning. Tryck på [2nd] ⊙ eller [2nd] ⊙ när du vill gå till uttryckets början eller slut.

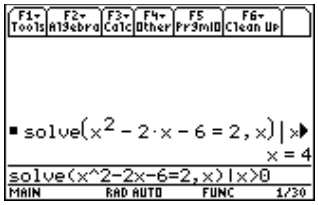
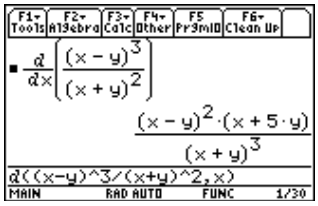
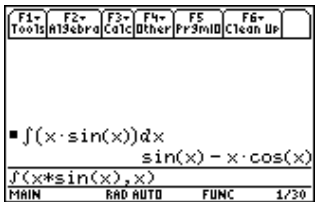
Stänga av din TI-89 / TI-92 Plus

När du vill stänga av din TI-89 / TI-92 Plus trycker du på [2nd] [OFF]. (Obs: [OFF] är andrafunktionen av tangenten [ON].)

I det här avsnittet finns flera exempel som du kan göra. Dessa exempel som visar några av beräkningsfunktionerna i räknaren. Historiklistan i varje fönsterbild rensades genom att trycka på [F1] och välja 8:Clear Home, innan varje exempel utfördes. Detta för att endast illustrera resultatet av tangentnedtryckningarna i exemplet.

Steg	TI-89 Tangenter	TI-92 Plus Tangenter	Teckenfönster
Visa beräkningar			
1. Beräkna sin(π/4) och visa resultatet i symboliskt och numeriskt format. *Du raderar historiklistan för tidigare beräkningar genom att trycka på [F1] och väja* 8:Clear Home.	[2nd] [SIN] [2nd] [π] [÷] 4 [)] [ENTER] [◆] [ENTER]	[SIN] [2nd] [π] [÷] 4 [)] [ENTER] [◆] [ENTER]	
Ta fram värdet för ett givet tal skrivet som en fakaltet			
1. Beräkna fakulteten av flera tal för att se hur TI-89 / TI-92 Plus hanterar mycket stora heltal *Du får fram fakultetsoperatorn (!) genom att trycka på [2nd] [MATH], välja* 7:Probability *och därefter välja* 1:!.	5 [2nd] [MATH] 7 1 [ENTER] 2 0 [2nd] [MATH] 7 1 [ENTER] 3 0 [2nd] [MATH] 7 1 [ENTER]	5 [2nd] W [ENTER] 2 0 [2nd] W [ENTER] 3 0 [2nd] W [ENTER]	
Utveckla komplexa tal			
1. Beräkna $(3+5i)^3$ för att se hur TI-89 / TI-92 Plus hanterar beräkningar som innehåller komplexa tal.	[(] 3 [+] 5 [2nd] [i] [)] [^] 3 [ENTER]	[(] 3 [+] 5 [2nd] [i] [)] [^] 3 [ENTER]	
Faktoruppdela i primtalsfaktorer			
1. Faktoruppdela det rationella talet 2634492. *Du kan mata in "factor" på inmatningsraden genom att skriva* FACTOR *på tangentbordet eller genom att trycka på [F2] och välja* 2:factor(. 2. (*Valfritt*) Skriv andra tal.	[F2] 2 2 6 3 4 4 9 2 [)] [ENTER]	[F2] 2 2 6 3 4 4 9 2 [)] [ENTER]	

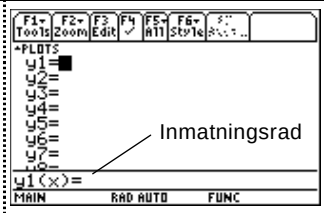
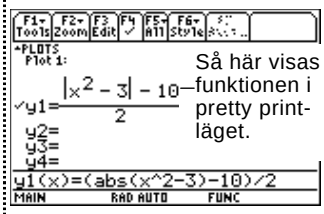
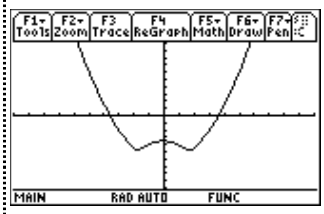
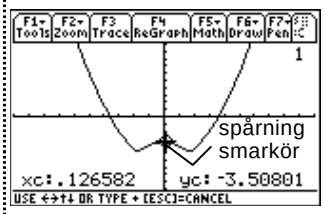
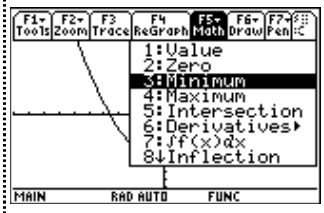
Steg	TI-89 Tangenter	TI-92 Plus Tangenter	Teckenfönster
Utveckla uttryck			
1. Utveckla uttrycket $(x-5)^3$. *Du kan mata in "expand" på inmatningsraden genom att skriva* EXPAND *på tangentbordet eller genom att trycka på* F2 *och välja* 3:expand(.	F2 3 (X − 5) ^ 3) ENTER	F2 3 (X − 5) ^ 3) ENTER	
2. (*Valfritt*) Skriv egna uttryck.			
Reducera uttryck			
1. Reducera uttrycket $(x^2-2x-5)/(x-1)$ så långt det går. *Du kan mata in "propFrac" på inmatningsraden genom att skriva* PROPFRAC *på tangentbordet eller genom att trycka på* F2 *och välja* 7:propFrac(.	F2 7 (X ^ 2 − 2 X − 5) ÷ (X − 1)) ENTER	F2 7 (X ^ 2 − 2 X − 5) ÷ (X − 1)) ENTER	
Faktoruppdela polynom			
1. Faktoruppdela polynomet (x^2-5) med avseende på x. *Du kan mata in "factor" på inmatningsraden genom att skriva* FACTOR *på tangentbordet eller genom att trycka på* F2 *och välja* 2:factor(.	F2 2 X ^ 2 − 5 , X) ENTER	F2 2 X ^ 2 − 5 , X) ENTER	
Lösa ekvationer			
1. Lös ekvationen $x^2-2x-6=2$ med avseende på x. *Du kan mata in "solve(" på inmatningsraden genom att välja* "solve(" *på menyn Catalog, genom att skriva* SOLVE(*på tangentbordet eller genom att trycka på* F2 *och välja* 1:solve(. *I statusfältet visas syntaxen för objektet som är markerat på menyn Catalog.*	F2 1 X ^ 2 − 2 X − 6 = 2 , X) ENTER	F2 1 X ^ 2 − 2 X − 6 = 2 , X) ENTER	 

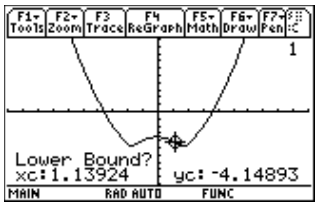
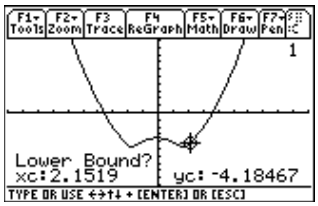
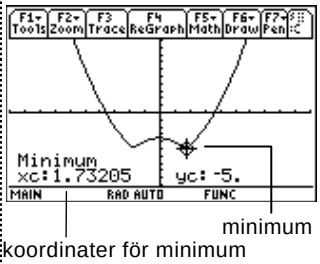
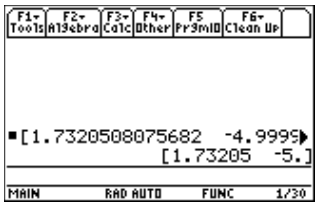
Steg	TI-89 Tangenter	TI-92 Plus Tangenter	Teckenfönster
Lösa ekvationer med en angiven definitionsmängd			
1. Lös ekvationen $x^2 - 2x - 6 = 2$ med avseende på x, där x är större än noll. *Operatorn "with" (I) ger domänbegränsningar.* ***TI-89:*** [I] ***TI-92 Plus:*** [2nd] [|]	[F2] 1 X [^] 2 [−] 2 X [−] 6 [=] 2 [,] X [)] [I] X [2nd] [>] 0 [ENTER]	[F2] 1 X [^] 2 [−] 2 X [−] 6 [=] 2 [,] X [)] [2nd] [|] X [2nd] [>] 0 [ENTER]	
Derivera funktioner			
1. Derivera $(x-y)^3/(x+y)^2$ med avseende på x. *Detta exempel visar hur du använder deriveringsfunktionen och hur funktionen visas i pretty print-läget i historiklistan.*	[2nd] [d] [(] X [−] Y [)] [^] 3 [÷] [(] X [+] Y [)] [^] 2 [,] X [)] [ENTER]	[2nd] [d] [(] X [−] Y [)] [^] 3 [÷] [(] X [+] Y [)] [^] 2 [,] X [)] [ENTER]	
Integrera funktioner			
1. Integrera $x * \sin(x)$ med avseende på x. *Detta exempel visar hur du använder integreringsfunktionen.*	[2nd] [∫] X [×] [2nd] [SIN] X [)] [,] X [)] [ENTER]	[2nd] [∫] X [×] [SIN] X [)] [,] X [)] [ENTER]	

Plotta en funktion

Exemplet i det här avsnittet visas några av graffunktionerna i TI-89 / TI-92 Plus.
I det här exemplet visas hur du plottar en funktion (visar en funktion som en graf) med hjälp av Y= Editor. Du lär dig mata in en funktion, visa funktionen som en graf, spåra en kurva, hitta ett minimum och överföra koordinaterna för minimum till grundfönstret.

Utforska graffunktionerna i TI-89 / TI-92 Plus genom att plotta funktionen $y=(|x^2-3|-10)/2$.

Steg	TI-89 Tangenter	TI-92 Plus Tangenter	Teckenfönster
1. Visa Y= Editor.	◆ [Y=]	◆ [Y=]	
2. Mata in funktionen (abs(x^2− 3)− 10)/2.	(CATALOG A ENTER X ^ 2 − 3) − 1 0) ÷ 2 ENTER	(2nd [CATALOG] A ENTER X ^ 2 − 3) − 1 0) ÷ 2 ENTER	
3. Visa grafen av funktionen. *Välj* 6:ZoomStd *genom att trycka på* 6 *eller genom att flytta markören till* 6:ZoomStd *och trycka på* ENTER.	F2 6	F2 6	
4. Slå på Trace. *Spårningsmarkören och koordinaterna x och y visas.*	F3	F3	
5. Öppna menyn MATH och välj 3:Minimum.	F5 ⊖ ⊖ ENTER	F5 ⌄ ⌄ ENTER	

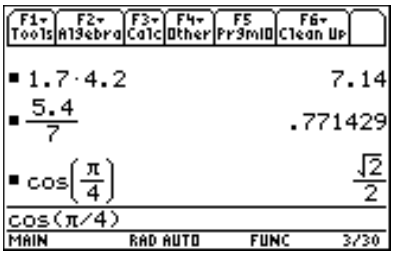
Steg	TI-89 Tangenter	TI-92 Plus Tangenter	Teckenfönster
6. Ange undre begränsning. *Tryck på ▷ (höger markör) för att flytta spårningsmarkören tills den befinner sig precis till vänster om minimipunkten innan du trycker på ENTER en andra gång.*	▷ . . . ▷ ENTER	▷ . . . ▷ ENTER	
7. Ange övre begränsning. *Tryck på ▷ (höger markör) för att flytta spårningsmarkören tills den befinner sig precis till höger om minimipunkten.*	▷ . . . ▷	▷ . . . ▷	
8. Sök efter minimum på kurvan inom det angivna intervallet.	ENTER	ENTER	
9. Överför resultatet till grundfönstret och visa därefter grundfönstret. *Genvägar för att kopiera grafkoordinater till historiklistan:* ***TI-89:*** ◆ (-) ***TI-92 Plus:*** ◆ H	◆ (-) HOME	◆ H ◆ [HOME]	

Använda räknaren

2

Slå på och stänga av TI-89 / TI-92 Plus 14
Justera kontrasten i teckenfönstret 15
Tangentbordet TI-89 16
Tangentbordet TI-92 Plus 17
Funktionstangenter 18
Mata in alfabetiska tecken 21
Grundfönstret 23
Skriva tal 25
Mata in uttryck och instruktioner 26
Format på visade resultat 29
Redigera ett uttryck på inmatningsraden 32
Menyer 34
Använda menyn Custom 37
Välja ett applikationer 38
Ställa in lägen 40
Använda menyn Clean Up för att påbörja ett nytt problem 43
Använda dialogrutan Catalog 44
Spara och hämta variabelvärden 47
Återanvända en tidigare inmatning eller det senaste resultatet 49
Automatiskt klistra in en inmatning eller ett resultat från historiklistan 52
Indikatorer för statusfältet i fönstret 53
Visa applikationerversion och ID-nummer 55

Det här kapitlet innehåller en allmän översikt över TI-89 / TI-92 Plus och beskriver de grundläggande funktionerna. Genom att bekanta dig med informationen i det här kapitlet kan du använda TI-89 / TI-92 Plus för att lösa problem mer effektivt.



Grundfönstret är det vanligaste applikationermet i TI-89 / TI-92 Plus och det du använder oftast. Du kan använda grundfönstret för att utföra många olika typer av matematiska funktioner.

Slå på och stänga av TI-89 / TI-92 Plus

Du kan slå på och stänga av räknaren manuellt genom att använda tangenterna [ON] och [2nd][OFF] (eller tangenterna [◆][OFF]). Med hjälp av funktionen APD™ (Automatic Power Down™) som stänger av TI-89 / TI-92 Plus automatiskt, förlängs batteriernas livslängd.

Slå på TI-89 / TI-92 Plus

Tryck på [ON].

- Om du stängde av räknaren genom att trycka på [2nd] [OFF], grundfönstret på din TI-89 / TI-92 Plus öppnas.
- Om du stängde av räknaren genom att trycka på [◆] [OFF], eller om enheten stängde av sig själv via APD, kommer din TI-89 / TI-92 Plus att återgå till det applikationer du senast använde.

Stänga av TI-89 / TI-92 Plus

Du kan stänga av räknaren genom att använda någon av följande tangenter.

Obs! [OFF] *är sekundärfunktionen på tangenten* [ON].

Tryck på:	Beskrivning
[2nd] [OFF] (tryck på [2nd] och sedan på [OFF])	Inställningar och innehåll i minnet behålls tack vare funktionen Constant Memory™. Kom dock ihåg följande: • Du kan inte använda [2nd][OFF] om ett felmeddelande visas. • När du slår på räknaren igen visas alltid grundfönstret (oavsett vilket applikationer du använde senast).
[◆] [OFF] (tryck på [◆] och sedan på [OFF])	På samma sätt som med [2nd][OFF] förutom: • Du kan använda [◆][OFF] även om ett felmeddelande visas. • När du slår på räknaren igen är den precis så som den var när du senast använde den.

APD (Automatic Power Down)

Efter flera minuter utan någon aktivitet stänger räknaren av sig själv automatiskt. Denna funktion kallas APD.

När du trycker på [ON] är räknaren så som den var när du senast använde den.

- Teckenfönstret, markören och eventuella felförhållanden är som när du senast använde räknaren.
- Alla inställningar och innehållet i minnet bevaras.

APD aktiveras inte om en beräkning eller ett applikationer pågår, såvida applikationermet inte är inställt på paus.

Batterier

I TI-89 används fyra alkaliska AAA-batterier och ett extra litiumbatteri. TI-92 Plus använder fyra batterier (AA) och ett litiumbatteri för reservström. Information om hur du byter batterier på dessa räknare utan att förlora lagrad information finns i appendix C.

Justera kontrasten i teckenfönstret

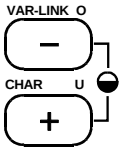
Ljusstyrkan och kontrasten i teckenfönstret är beroende av ljuset i rummet, batteriernas ålder, visningsvinkel och justering av kontrasten. Kontrastinställningen behålls i minnet när du stänger av räknaren.

Justera kontrasten

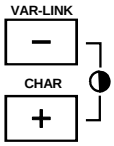
Du kan justera teckenfönstrets kontrast så att det passar din visningsvinkel och de ljusförhållanden som gäller.

Om du vill:	Trycker du på och håller ned:
Minska (göra ljusare) kontrasten	◆ och −
Öka (göra mörkare) kontrasten	◆ och +

TI-89 Kontrasttangenter



TI-92 Plus Kontrasttangenter



Om du trycker på och håller ned ◆ + eller ◆ − för länge, kan teckenfönstret bli helt tomt eller helt svart. Du finjusterar genom att hålla ned ◆ och sedan trycka på + eller −.

När ska batterierna bytas ut?

__Tips!__ När du har bytt batterier kan teckenfönstret vara mycket mörkt. Använd ◆− för att göra fönstret ljusare.

När batterierna börjar ta slut börjar teckenfönstret bli suddigt (särskilt under beräkningar) och du måste öka kontrasten. Om du ofta behöver öka kontrasten bör du byta ut de fyra alkaliska batterierna.

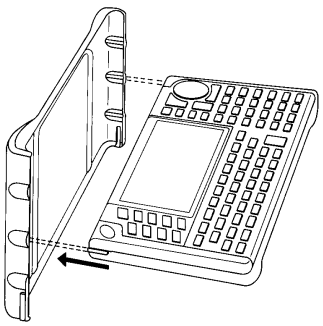
I statusfältet nederst i teckenfönstret visas också information om batterierna.

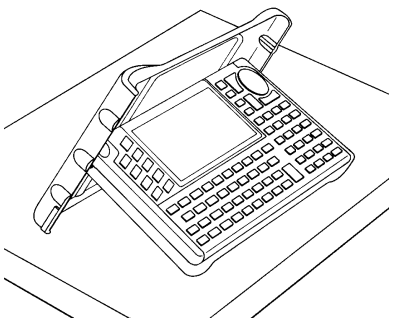
Indikator i statusfältet	Beskrivning
BATT	Batterierna börjar ta slut.
BATT	Byt ut batterierna så snart som möjligt.

Använda locket för TI-92 Plus som ett ställ

__Obs!__ Skjut in flikarna överst på enheten i springorna som finns i locket.

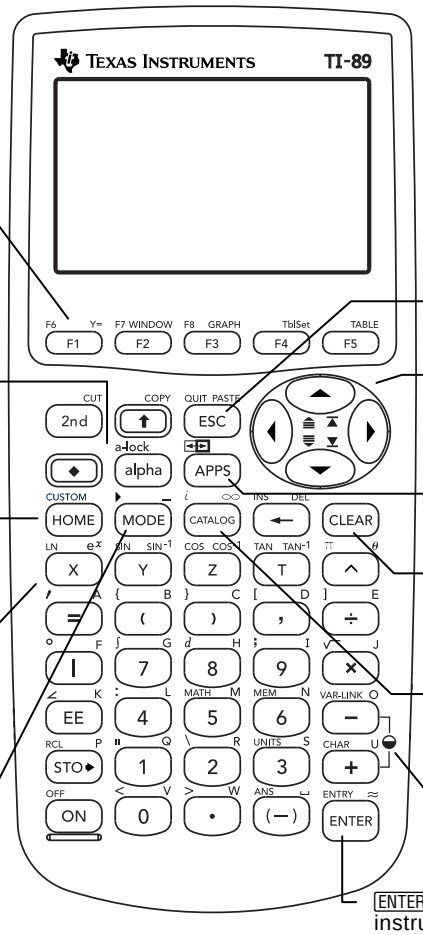
När du använder TI-92 Plus på ett bord kan du använda locket för att stötta enheten i tre olika vinklar. Detta stöd kan göra det enklare att visa teckenfönstret under olika ljusförhållanden.





Tack vare TI-89:s greppvänliga form och tangentbordslayouten kan du snabbt få tillgång till vilket område av tangentbordet som helst, även när du håller enheten med båda händer.

Områden på tangentbordet



Med funktionstangenterna F1 till 2nd F8 kan du välja menyer i verktygsfältet. Tillsammans med ◆ kan du även välja applikationer (sidan 39).

2nd, ◆, ⬆ och alpha ändrar funktionen hos andra tangenter (sidan 18).

HOME öppnar grundfönstret där du genomför de flesta beräkningar.

X, Y och Z används ofta vid symboliska beräkningar.

Med MODE kan du ändra lägesinställningar som bestämmer hur värden och grafer används, beräknas och visas (sidan 40).

ESC stänger en meny eller dialogruta.

◀, ▶, ▲, och ▼ flyttar markören.

APPS låter dig välja en applikationer (sidan 38).

CLEAR raderar inmatningsraden. Används även för att ta bort par med inmatningar och svar i historikområdet.

CATALOG låter dig välja funktioner och instruktioner från en lista (sidan 44).

Justera kontrasten genom att trycka ◆ − (ljusare) eller ◆ + (mörkare).

ENTER evalueran ett uttryck, kör en instruktion, väljer ett menyalternativ, osv.

Du kan använda ◆ ENTER om du vill visa resultatet av en beräkning som ett decimaltal.

Flytta markören

När du vill flytta markören i en särskild riktning trycker du på motsvarande markörtangent (◀, ▶, ▲ eller ▼).

I vissa TI-89 applikationer kan du även trycka på:

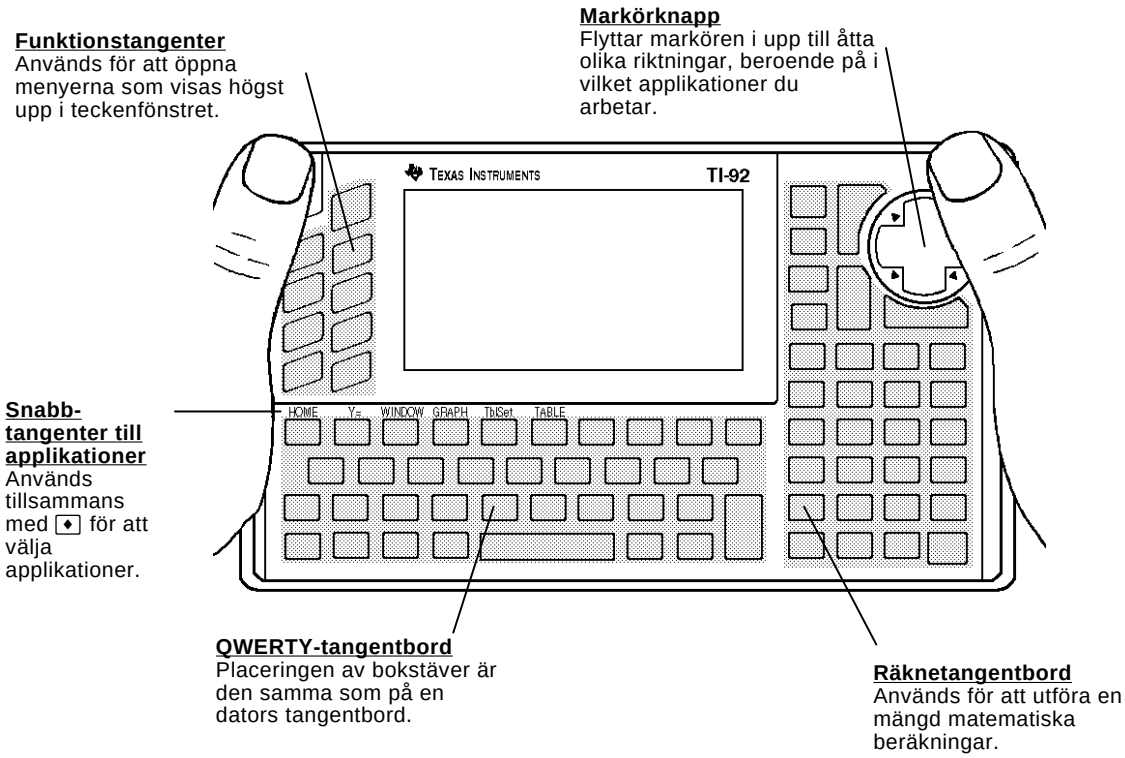
- 2nd ◀ eller 2nd ▶ när du vill flytta markören till början eller slutet av en rad.
- 2nd ▲ eller 2nd ▼ när du vill flytta markören en sida uppåt eller nedåt.
- ◆ ▲ eller ◆ ▼ när du vill flytta markören till början eller slutet av en sida.
- ▲ och ◀, ▲ och ▶, ▼ och ◀, eller ▼ och ▶ för att flytta diagonalt. (Tryck in tangenterna samtidigt.)

Tangentbordet TI-92 Plus

Tack vare TI-92 Plus:s greppvänliga form och tangentbordslayouten kan du snabbt få tillgång till vilket område av tangentbordet som helst, även när du håller enheten med båda händer.

Områden på tangentbordet

Tangentbordet är uppdelat i flera delar som var och en innefattar relaterade tangenter.

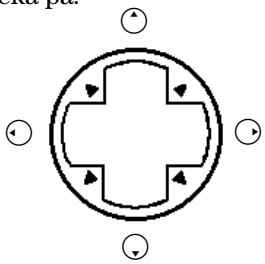


Markörknapp

När du flytta markören trycker du på en av markörknappens kanter. I den här handboken används symboler som ◀ och ▶ för att ange vilken sida av markörknappen du ska trycka på.

Du trycker exempelvis på ▶ för att flytta markören åt höger.

Obs! De diagonala riktningarna (↖ m.m.) används endast för geometri- och grafapplikationer.



Funktionstangenter

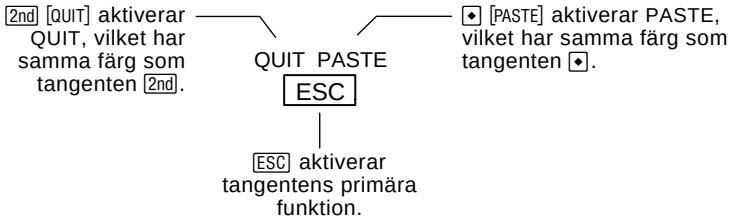
Tangent	Beskrivning
2nd (andra)	Aktiverar den andra funktionen hos nästa tangent du trycker ned. På tangentbordet är dessa funktioner markerade med samma färg som tangenten 2nd.
◆ (diamond)	Aktiverar tangenter som väljer vissa applikationer (se sidan 39), menyval och andra funktioner. På tangentbordet är dessa funktioner markerade med samma färg som tangenten ◆.
↑ (shift)	Skriver nästa tecken i versaler. ↑ används också tillsammans med ⊙ och ⊙ för att markera tecken i inmatningsraden vid redigering.
alpha (endast TI-89)	Används för att skriva in alfabetiska tecken, inklusive blanksteg. På tangentbordet är dessa funktioner markerade med samma färg som tangenten alpha.
(hand) (endast TI-92 Plus)	Används tillsammans med pekplattan för att manipulera geometriska objekt. används också för att rita på grafer.

***Obs!** Mer information om användningen av alpha och ↑, finns i avsnittet "Mata in alfabetiska tecken" på sidan 21.*

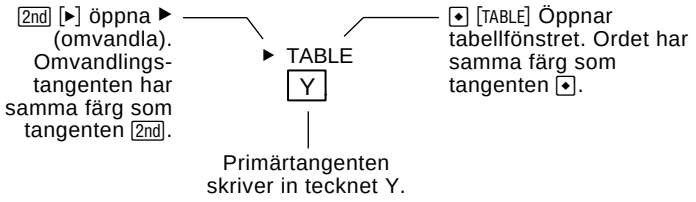
Exempel på användningen av 2nd och ◆

Tangenten ESC kan till exempel utföra tre olika operationer beroende på om du först trycker på 2nd eller ◆.

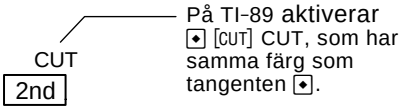
Följande exempel för TI-89 visar hur du använder funktionstangenterna 2nd eller ◆ tillsammans med tangenten ESC.



Följande exempel för TI-92 Plus visar hur du använder 2nd eller ◆ tillsammans med den alfabetiska tangenten Y.



Vissa tangenter har bara en extra funktion som aktiveras med antingen [2nd] eller [◆], beroende på i vilken färg funktionen är märkt och var på tangenten den är placerad.



När du trycker ned någon av tangenterna [2nd] eller [◆], visas indikatorn 2ND eller ◆ i statusraden i fönstrets underkant. Om du trycker ned en funktionstangent av misstag kan du trycka på den igen (eller trycka på [ESC]) för att deaktivera funktionsläget.

Andra viktiga tangenter som du behöver känna till

__Obs!__ En del tangentkombinationer skiljer sig mellan TI-89 och TI-92 Plus. I början av denna handbok finns en lista med alla skillnader.

Tangent	Beskrivning
[◆] [Y=]	Öppnar Y= Editor (kapitel 6).
[◆] [WINDOW]	Öppnar Window Editor (kapitel 6).
[◆] [GRAPH]	Öppnar fönstret Graph (kapitel 6).
[◆] [TblSet]	Ställer in parametrar för tabellfönstret (kapitel 13).
[◆] [TABLE]	Öppnar fönstret Table (kapitel 13).
TI-89: [◆] [CUT] [◆] [COPY] [◆] [PASTE] **TI-92 Plus:** [◆] X (cut) [◆] C (copy) [◆] V (paste)	Låter dig redigera inmatad text genom att klippa ut, kopiera och klistra in data.
[2nd] [⊞]	Växlar aktiv sida i ett delat fönster (kapitel 14).
[2nd] [CUSTOM]	Aktiverar/deaktiverar den anpassade menyn (sidan 37).
[2nd] [▶]	Omvandlar måttenheter (kapitel 4).
TI-89: [◆] [_] **TI-92 Plus:** [2nd] [_]	Utformar måttenheter (kapitel 4).
[←]	Raderar tecknet till vänster om markören (backsteg).
[2nd] [INS]	Växlar mellan att skriva över och infoga text vid inmatning av data (sidan 33).
[◆] [DEL]	Raderar tecknet till höger om markören.

Viktiga tangenter (fortsättning)

Tangent	Beskrivning
TI-89: [I] **TI-92 Plus:** [2nd] [I]	Matar in operatorn “with” som används vid symboliska beräkningar (kapitel 3).
[2nd] [∫] [2nd] [*d*]	Beräknar integraler och derivator (kapitel 3).
[2nd] [∠]	Anger en vinkel i polära, cylindriska eller sfäriska koordinater.
[2nd] [MATH]	Öppnar menyn MATH.
[2nd] [MEM]	Öppnar fönstret MEMORY (kapitel 21).
[2nd] [VAR-LINK]	Öppnar fönstret VAR-LINK för hantering av variabler (kapitel 21).
[2nd] [RCL]	Återkallar innehållet i en variabel (sidan 48).
TI-89: [2nd] [UNITS] **TI-92 Plus:** [♦] [UNITS]	Öppnar dialogrutan UNITS (kapitel 4).
[2nd] [CHAR]	Öppnar menyn CHAR där du kan välja grekiska tecken och andra internationella tecken, osv. (kapitel 18).
[2nd] [ANS], [2nd] [ENTRY]	Återkallar den senaste inmatningen respektive det senaste svaret (sidan 49).

Mata in alfabetiska tecken

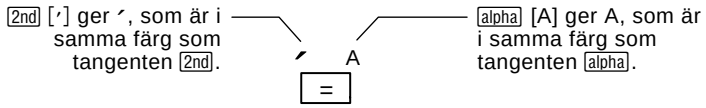
Alfabetiska tecken används i uttryck som x^2+y^2 för att ange variabelnamn (sidan 47), och i Text Editor (kapitel 18).

Skriva in alfabetiska tecken på en TI-89

Bokstäverna x, y, z och t används ofta i matematiska uttryck. Dessa bokstäver är primära tangenter på tangentbordet så att du snabbt och enkelt ska kunna skriva in dem.

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

Andra bokstäver är tillgängliga via funktionen [alpha] hos en annan tangent, liknande funktionsväljarna [2nd] och [◆] som beskrivs i föregående avsnitt. Till exempel:



Skriva in alfabetiska tecken på en TI-89 / TI-92 Plus

Om du vill:	På TI-89, tryck:	På TI-92 Plus, tryck:
Mata in ett enskilt alfabetiskt tecken som gemen.	[alpha] och därefter bokstavstangenten (statusraden visar ◖)	bokstavstangenten
Mata in ett enskilt alfabetiskt tecken som versal.	[↑] och därefter bokstavstangenten (statusraden visar ⬆)	[↑] och bokstavstangenten (statusraden visar ⬆)
Mata in ett blanksteg.	[␣] (alfafunktionen hos tangenten [(-)])	blanksteg
Aktivera skiftlås för gemener.	[2nd] [a-lock] (statusraden visar ▮)	(inga åtgärder krävs)
Aktivera skiftlås för versaler.	[↑] [a-lock] (statusraden visar ▮)	[2nd] [CAPS]
Deaktivera skiftlås.	[alpha] (deaktiverar alfalås för versaler)	[2nd] [CAPS] (deaktiverar alfalås för versaler)

***Obs!** På TI-89, du behöver inte använda [alpha] eller skiftlås för att mata in x, y, z eller t. Du måste dock använda [↑] eller skiftlås för versaler av X, Y, Z eller T.*

***Obs!** På TI-89, skiftlås deaktiveras alltid när du byter applikationer, t ex om du går från Text Editor till grundfönstret.*

Skriva in alfabetiska tecken... (fortsättning)

På TI-89, när någon form av skiftlås är aktiverat gäller följande:

- Om du vill skriva in en punkt, ett komma, eller något annat tecken som är primärfunktion hos en tangent måste du deaktivera skiftlåset.
- Om du vill skriva in ett tecken som är en andra funktion, t ex [2nd] [{], behöver du inte deaktivera skiftlåset. Efter du har skrivit in tecknet är skiftlås fortfarande aktiverat.

Automatiskt alfalås i dialogrutor på TI-89

Det finns tillfällen då du inte behöver trycka på [alpha] eller [2nd] [a-lock] för att skriva in alfabetiska tecken på en TI-89. Alfalåsning aktiveras automatiskt då en dialogruta visas. Funktionen för automatisk alfalåsning fungerar i följande dialogrutor:

Dialogruta	Alfalås
Catalog dialog box	Alla kommandon listas i alfabetisk ordning. Tryck ned en bokstav för att gå till det första kommandot som börjar med det tecknet. Mer information finns på sidan 44.
Units dialog box	Skriv in det första tecknet för en enhet eller konstant i respektive kategori. Mer information finns i kapitel 4.
Dialog boxes with entry fields	Detta inkluderar, men är inte begränsat till: Create New Folder, Rename och Save copy as. Mer information om dialogrutor finns på sidan 35.

***Obs!** Om du vill skriva in ett tal trycker du på [alpha] för att stänga av alfalåsningen. Tryck på [alpha] eller [2nd] [a-lock] för att aktivera låset igen.*

Alfalåsningen är **inte** aktiverad för dialogrutor som kräver numerisk inmatning. Dialogrutor som kräver numerisk inmatning är: Resize Matrix, Zoom Factors och Table Setup.

Speciella tecken

Använd menyn [2nd] [CHAR] för att välja bland ett antal olika specialtecken. Mer information finns under “Mata in speciella tecken” i kapitel 18.

Grundfönstret

> När du först slår på TI-89 / TI-92 Plus visas grundfönstret. I grundfönstret kan du utföra instruktioner, beräkna uttryck och visa resultat.

Visa grundfönstret

När du slår på räknaren TI-89 / TI-92 Plus efter att ha stängt av den med [2nd] [OFF], visas alltid grundfönstret. (Om räknaren TI-89 / TI-92 Plus stängde av sig själv med funktionen APD™, visas det fönster som var aktivt när räknaren stängdes av.)

Så här visar du grundfönstret:

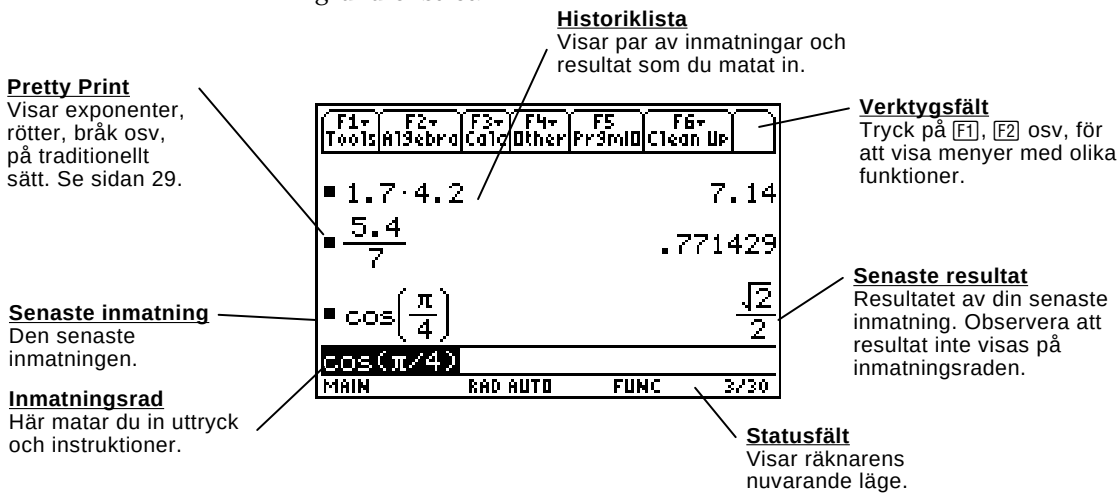
- Tryck på:
  TI-89: [HOME]
  TI-92 Plus: [◆] [HOME]

  – eller –
- Tryck på [2nd] [QUIT].

  – eller –
- Tryck på:
  TI-89: [APPS] [alpha] A
  TI-92 Plus: [APPS] A

Grundfönstrets delar

I följande exempel visas en kortfattad beskrivning av huvuddelarna i grundfönstret.



Historiklista

I historiklistan visas upp till åtta tidigare par av inmatningar och resultat (beroende på komplexiteten och höjden på de visade uttrycken). När fönstret är fullt försvinner informationen ut från överdelen av fönstret. Du kan använda historiklistan för att:

- Granska tidigare inmatningar och resultat. Du kan använda markören för att visa inmatningar och resultat som har rullat ut från fönstret.
- Hämta eller klistra in en tidigare inmatning eller ett resultat automatiskt på inmatningsraderna så att du kan återanvända eller redigera inmatningen eller resultatet. Se sidorna 50 och 52.

Rulla igenom historiklistan

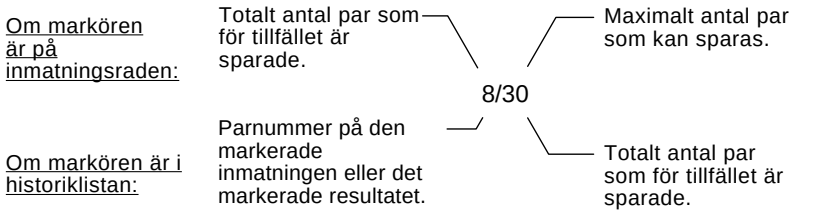
Markören befinner sig normalt på inmatningsraden. Du kan dock flytta markören till historiklistan.

Om du vill:	Gör du följande:
Visa inmatningar och resultat som har rullat ut från fönstret	1. Tryck på ⊕ från inmatningsraden för att markera det senaste resultatet. 2. Fortsätt att använda ⊕ för att flytta markören från resultatet till inmatningen, upp igenom historiklistan.
Gå till det äldsta och det senaste historikparet	Om markören är i historikområdet kan du trycka på [◆] ⊕ eller [◆] ⊖.
Visa en inmatning eller ett resultat som är för lång(t) för att få plats på en rad (▶ befinner sig i slutet av raden)	Flytta markören till inmatningen eller resultatet. Använd ⊕ och ⊖ för att rulla åt vänster och höger (eller [2nd] ⊕ och [2nd] ⊖ för att gå till slutet eller början).
Flytta tillbaka markören till inmatningsraden	Tryck på [ESC] eller ⊖ tills markören är tillbaka på inmatningsraden.

Obs! *Du kan se ett exempel på hur du visar ett långt svar, på sidan 28.*

Historikinformation i statusfältet

Använd historikindikatorn i statusfältet för att visa information om par av inmatningar och resultat. Exempel:



Som standard visas de sista 30 paren av inmatningar och resultat. Om historiklistan är full när du gör en ny inmatning (anges av 30/30), sparas det nya paret och det äldsta paret raderas. Historikindikatorn ändras inte.

Ändra historiklistan

Om du vill:	Gör du följande:
Ändra antalet par som kan sparas	Tryck på [F1] och välj 9:Format, eller tryck på **TI-89:** [◆][I] **TI-92 Plus:** [◆] F. Tryck därefter på ⊕, använd ⊕ eller ⊖ för att markera det nya antalet och tryck sedan två gånger på [ENTER].
Rensa historiklistan och ta bort alla sparade par	Tryck på [F1] och välj 8:Clear Home, eller skriv **ClrHome** på inmatningsraden.
Ta bort ett visst par	Flytta markören till inmatningen eller resultatet. Tryck på [←] eller [CLEAR].

Skriva tal

Med tangentbordet till TI-89 / TI-92 Plus kan du skriva positiva och negativa tal för dina beräkningar. Du kan också mata in tal i grundpotensform.

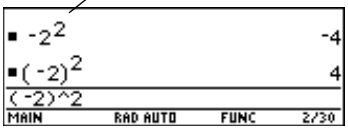
Skriva negativa tal

1. Tryck på tangenten för negation: [(-)]. (Använd inte tangenten för subtraktion: [−].)
2. Skriv talet.

Information om hur TI-89 / TI-92 Plus räknar med en negation i förhållande till andra funktioner finns i avsnittet "EOS™ (Equation Operating System)" i appendix B. Du bör till exempel veta att funktioner som x^2, beräknas före en negation.

Använd [(] och [)] för att inkludera parenteser om du är osäker på hur en negation evalueras.

Beräknas som $^-(2^2)$



***Viktigt!** Använd [−] för subtraktion och [(-)] för negation.*

Om du använder [−] i stället för [(-)] (eller tvärtom) visas kanske ett felmeddelande eller så kan du få oväntade resultat. Exempel:

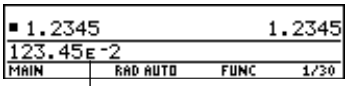
- 9 [×] [(-)] 7 = ⁻63
  – men –
  9 [×] [−] 7 visar ett felmeddelande.
- 6 [−] 2 = 4
  – men –
  6 [(-)] 2 = ⁻12 eftersom det tolkas som 6(⁻2), implicit multiplikation.
- [(-)] 2 [+] 4 = 2
  – men –
  [−] 2 [+] 4 subtraherar 2 från föregående svar och lägger sedan till 4.

Skriva tal i grundpotensform

1. Skriv den del av talet som föregår exponenten. Detta värde kan vara ett uttryck.
2. Tryck på:
   TI-89: [EE]
   TI-92 Plus: [2nd] [EE]
   E visas i teckenfönstret.
3. Skriv exponenten som ett heltal med upp till tre siffror. Du kan använda en negativ exponent.

När du skriver tal i grundpotensform visas inte svaren i grundpotensform.

Notationsformatet anges av inställningarna för lägen (sidan 29 - 31) och talets storlek.



Representerar 123,45 x 10^{-2}

Mata in uttryck och instruktioner

Du utför en beräkning genom att evaluera ett uttryck. Du påbörjar en funktion genom att utföra lämplig instruktion. Uttryck beräknas och resultaten visas enligt de inställningar för läge som beskrivs på sidan 29.

Definitioner

Uttryck

Består av tal, variabler, operatorer, funktioner och deras argument som beräknas till ett svar. Exempel: $\pi r^2 + 3$.

- Mata in ett uttryck i samma ordning som du normalt skriver det.
- På de flesta ställen där du blir uppmanad att mata in ett värde, kan du mata in ett uttryck.

Operator

Utför operationer som exempelvis +, −, ∗, ^.

- Operatorer kräver ett argument före och efter operatorn. Exempel: 4+5 och 5^2.

Obs!: *I appendix A beskrivs alla inbyggda funktioner och instruktioner.*

Funktion

Returnerar ett värde.

- Funktioner kräver ett eller flera argument (inom parenteser) efter funktionen. Exempel: √(5) och **min**(5,8).

Obs! *I den här handboken används termen* ***kommando*** *som en allmän referens till både funktioner och instruktioner.*

Instruktion

Påbörjar en åtgärd.

- Instruktioner kan inte användas i uttryck.
- Vissa instruktioner kräver inga argument. Exempel: **ClrHome**.
- Vissa kräver ett eller flera argument. Exempel: **Circle** 0,0,5.

För instruktioner ska argumenten inte vara inom parentes.

Implicit multiplikation

TI-89 / TI-92 Plus känner igen implicit multiplikation, förutsatt att den inte är i konflikt med en reserverad notation.

	Om du matar in:	TI-89 / TI-92 Plus tolkas det som:
Giltigt	2π	2∗ π
	4 sin(46)	4∗ sin(46)
	5(1+2) eller (1+2)5	5∗ (1+2) eller (1+2)∗ 5
	[1,2]a	[a 2a]
	2(a)	2∗ a
Ogiltigt	xy	Enstaka variabel med namnet xy
	a(2)	Funktionsanrop
	a[1,2]	Index till matriselement a[1,2]

Parenteser

Uttryck beräknas enligt hierarkin i EOS™ (Equation Operating System), vilket beskrivs i appendix B. Om du vill ändra ordningsföljden för en beräkning, eller om du vill försäkra dig om att ett uttryck beräknas i den ordning du vill, använder du parenteser.

Beräkningar inom parenteser görs först. I exempelvis 4(1+2), beräknas först (1+2) och därefter multipliceras svaret med 4.

Mata in ett uttryck

Skriv uttrycket och tryck därefter på [ENTER] för att beräkna det. Om du vill mata in ett funktions- eller instruktionsnamn på inmatningsraden kan du göra följande:

- Tryck på dess tangent, om den finns. Tryck till exempel på **TI-89:** [2nd] [SIN] eller **TI-92 Plus:** [SIN].

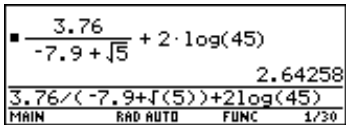
  – eller –
- Välj namnet på en meny, om det finns. Välj till exempel 2:abs på undermenyn Number, som finns på menyn MATH.

  – eller –
- Skriv in namnet bokstav för bokstav via tangentbordet. (På TI-89 använder du [alpha] och [2nd] [a-lock] för att skriva bokstäver.) Du kan använda gemener eller versaler som du själv vill. Skriv t ex in **sin(** eller **Sin(**.

Exempel

Räkna ut 3.76 ÷ (⁻7.9 + √5) + 2 log 45.

Skriv in funktionsnamnet i det här exemplet.

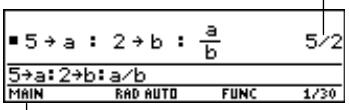
På TI-89:	**På TI-92 Plus:**	**Teckenfönster**
3 . 7 6 [÷] [(] [(-)] 7 . 9 [+] [2nd] [√]	3 . 7 6 [÷] [(] [(-)] 7 . 9 [+] [2nd] [√]	3.76/(⁻7.9+√([2nd] [√] infogar "√(" eftersom dess argument måste vara inom parentes.
5 [)] [)]	5 [)] [)]	3.76/(⁻7.9+√(5)) Använd [)] en gång för att avsluta √(5)-funktionen och en gång till för att avsluta (⁻7.9 + √5).
[+] 2 [2nd] [a-lock] L O G [alpha] [(] 4 5 [)] [ENTER]	[+] 2 L O G [(] 4 5 [)] [ENTER]	3.76/(⁻7.9+√(5))+2log(45) **log** kräver att du använder ( ) runt dess argument. 

Obs! *Du kan även välja* **log** *genom att trycka* ***TI-89:*** [CATALOG] ***TI-92 Plus:*** [2nd] [CATALOG] *(sidan 44).*

Mata in flera uttryck på en rad

Om du vill mata in mer än ett uttryck eller instruktion på samma gång, separerar du dem med ett kolon genom att trycka på [2nd] [:].

Visar endast det senaste resultatet.

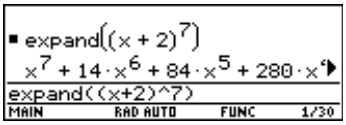


→ visas när du trycker på [STO▸] för att spara ett värde i en variabel.

Om längden på en inmatning eller på ett resultat är för lång för en rad

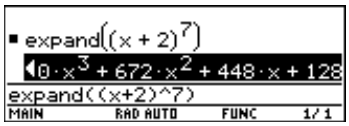
Om både inmatningen och dess resultat inte kan visas i historiklistan på en rad, visas resultatet på nästa rad.

Om längden på en inmatning eller på ett resultat är för lång för att få plats på en rad visas symbolen ▶ i slutet av raden.



Så här visar du hela inmatningen eller resultatet:

1. Tryck på ⊕ för att flytta markören från inmatningsraden upp till historiklistan. Det sista resultatet markeras.
2. Om det behövs använder du ⊕ och ⊖ för att markera inmatningen eller resultatet som du vill visa. Du flyttar exempelvis från resultatet till inmatningen med knappen ⊕, upp genom historiklistan.
3. Använd ⊕ och ⊖ eller [2nd] ⊕ och [2nd] ⊖ för att rulla åt höger och vänster.



Obs! När du rullar åt höger visas ◀ i början av raden.

4. Du återgår till inmatningsraden genom att trycka på [ESC].

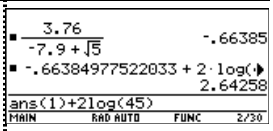
Fortsätta en beräkning

När du trycker på [ENTER] för att utföra en beräkning lämmas TI-89 / TI-92 Plus uttrycket kvar på inmatningsraden och markeras. Du kan fortsätta att använda det senaste svaret eller mata in ett nytt uttryck.

Om du trycker på:	TI-89 / TI-92 Plus händer följande:
[+], [−], [×], [÷], [^] eller [STO▸]	Inmatningsraden byts ut mot variabeln ans(1), med vilken du kan använda det senaste svaret som början på ett annat uttryck.
Vilken annan tangent som helst	Raderar innehållet på inmatningsraden och påbörjar en ny inmatning.

Exempel

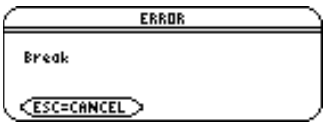
Räkna ut 3.76 ÷ (⁻7.9 + √5). Lägg sedan till 2 log 45 till svaret.

På TI-89:	På TI-92 Plus:	Teckenfönster
3 . 7 6 [÷] [(] [(-)] 7 . 9 [+] [2nd] [√] 5 [)] [)] [ENTER] [+] 2 [2nd] [a-lock] L O G [alpha] [(] 4 5 [)] [ENTER]	3 . 7 6 [÷] [(] [(-)] 7 . 9 [+] [2nd] [√] 5 [)] [)] [ENTER] [+] 2 L O G [(] 4 5 [)] [ENTER]	 When you trycker på [+], byts inmatningsraden ut mot variabeln ans(1), som innehåller det senaste svaret.

Stoppa en beräkning

När en beräkning pågår visas indikatorn BUSY till höger i statusfältet. Du stoppar beräkningen genom att trycka på [ON].

Det kan ta en liten stund innan "Break"-meddelandet visas.



Tryck på [ESC] för att återgå till det aktuella applikationermet.

Format på visade resultat

Ett resultat kan beräknas och sedan visas i flera format. I det här avsnittet beskrivs TI-89 / TI-92 Plus lägen och inställningar som påverkar visningsformaten. Information om hur du kontrollerar eller ändrar de aktuella inställningarna för lägen finns på sidan 40.

Pretty print-läget

Som standard visas Pretty Print = ON. Exponenter, rötter, bråk m.m. visas i samma format som du normalt skriver dem. Använd [MODE] för att aktivera och avaktivera pretty print-läget.

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

På inmatningsraden visas inte uttryck i pretty print-läge. Om pretty print-läget är aktiverat visas i historiklistan både inmatningen och resultatet i pretty print om du trycker på [ENTER].

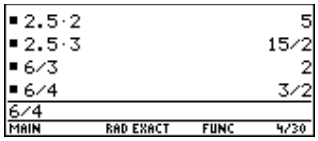
Läget Exact/Approx

Som standard är Exact/Approx inställt till AUTO. Du kan använda [MODE] för att välja bland tre inställningar.

Eftersom AUTO är en kombination av de två andra inställningarna bör du känna till alla tre inställningarna.

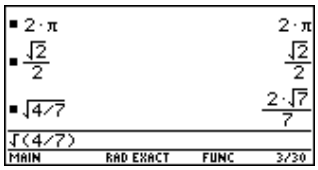
***Obs!** Genom att behålla bråkformat och symboliskt format, minskar EXACT antal fel med avrundning som kan uppstå vid mellanresultat i länkade beräkningar.*

EXACT – Alla resultat som inte är heltal visas i bråkform eller symboliskt format (1/2, π, $\sqrt{2}$ m.m.).



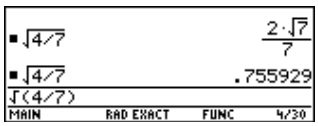
Visar hela tal som resultat.

Visar förenklade bråk som resultat.



Visar π symboliskt.

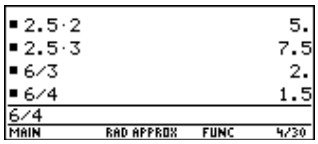
Visar en symbolisk form för rötter som inte kan beräknas till ett heltal.



Tryck på [•] [ENTER] för att tillfälligt åsidosätta EXACT-inställningen och visa resultatet som ett decimaltal.

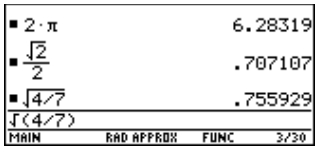
Läget Exact/Approx (fortsättning)

APPROXIMATE – Så långt det är möjligt visas alla numeriska resultat som decimaltal.



Bråk-talsresultat beräknas numeriskt.

__Obs!__ Resultat avrundas till precisionen som används i TI-89 / TI-92 Plus och visas enligt de aktuella inställningarna.

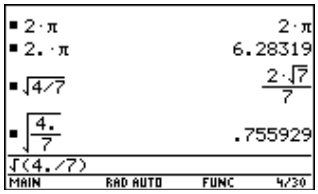


Symboliska former beräknas, där det är möjligt, numeriskt.

Eftersom odefinierade variabler inte kan beräknas behandlas de algebraiskt. Om exempelvis variabeln är odefinierad är $\pi r^2 = 3.14159 \cdot r^2$.

AUTO – Så långt det är möjligt används EXACT-formen, men när APPROXIMATE innehåller ett decimalkomma används en ungefärlig form. Vissa funktioner kan också visas som ett ungefärligt resultat, även om APPROXIMATE inte innehåller ett decimalkomma.

__Tips!__ Om du vill behålla EXACT-formen använder du bråk i stället för decimaltal. Använd exempelvis 3/2 i stället för 1,5.



En decimal i inmatningen tvingar resultatet att visas i decimalform.

I följande tabell jämförs de tre inställningarna.

__Tips!__ Om du vill beräkna en inmatning APPROXIMATE-formen i ungefärlig form, oavsett den aktuella inställningen, trycker du på ◆ ENTER.

Inmatning	Resultat med Exact	Resultat med Approximate	Resultat med Auto
8/4	2	2.	2
8/6	4/3	1.33333	4/3
8.5∗3	51/2	25.5	25.5
√(2)/2	$\frac{\sqrt{2}}{2}$	.707107	$\frac{\sqrt{2}}{2}$
π∗2	2·π	6.28319	2·π
π∗2.	2·π	6.28319	6.28319

En decimal i inmatningen tvingar resultatet att visas i decimalform i AUTO.

Läget Display Digits

Som standard är Display Digits = FLOAT 6, vilket innebär att resultat avrundas till max sex siffror. Du kan använda MODE för att välja andra inställningar. Inställningarna gäller för alla exponentiella format.

Internt beräknar och behåller TI-89 / TI-92 Plus alla decimalresultat med upp till 14 signifikanta siffror (även om högst 12 visas).

***Obs!** Oavsett inställningen för Display Digits, används alla decimalerna för interna decimaltalsberäkningar för att garantera så stor precision som möjligt*

***Obs!** Ett resultat visas automatiskt i grundpotensform om dess storlek inte kan visas i det angivna antalet siffror.*

Inställning	Exempel		Beskrivning
FIX (0 – 12)	123.	(FIX 0)	Resultat avrundas till det angivna antalet decimaler.
	123.5	(FIX 1)	
	123.46	(FIX 2)	
	123.457	(FIX 3)	
FLOAT	123.456789012		Antal decimaler varierar, beroende på resultatet.
FLOAT (1 – 12)	1.E 2	(FLOAT 1)	Resultat avrundas till det totala antalet angivna decimaler.
	1.2E 2	(FLOAT 2)	
	123.	(FLOAT 3)	
	123.5	(FLOAT 4)	
	123.46	(FLOAT 5)	
	123.457	(FLOAT 6)	

Läget Exponential Format

Som standard är Exponential Format = NORMAL. Du kan använda MODE för att välja bland tre inställningar.

***Obs!** I historiklistan visas ett tal i grundpotensform om dess absoluta värde är mindre än 0,001.*

Inställning	Exempel	Beskrivning
NORMAL	12345.6	Om ett resultat inte kan visas med det antal siffror som är angivet i läget Display Digits växlar TI-89 från NORMAL till SCIENTIFIC endast för detta resultat.
SCIENTIFIC	1.23456E 4	1.23456×10^4
		Exponent (10 upphöjt till).
		Alltid en siffra till vänster om decimalkommat.
ENGINEERING	12.3456E 3	12.3456×10^3
		Exponenten är en multipel av 3.
		Kan ha 1, 2 eller 3 siffror till vänster om decimalkommat.

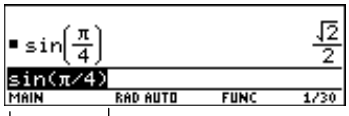
Redigera ett uttryck på inmatningsraden

Genom att veta hur du redigerar en inmatning kan du spara mycket tid. Om du gör fel när du matar in ett uttryck är det oftare lättare att rätta till felet än att skriva om hela uttrycket.

Ta bort markeringen från den föregående inmatningen

När du trycker på [ENTER] för att beräkna ett uttryck lämnar TI-89 / TI-92 Plus uttrycket kvar på inmatningsraden och markeras. För att kunna redigera uttrycket måste du först ta bort markeringen, annars tar du bort hela uttrycket genom att du skriver över det.

Du tar bort markeringen genom att flytta markören mot den sida av uttrycket som du vill redigera.



▶ flyttar markören till slutet av uttrycket.

◀ flyttar markören till början.

Flytta markören

När du har tagit bort markeringen flyttar du markören till lämplig plats i uttrycket.

Om du vill flytta markören:	Trycker du på:	
Åt vänster eller höger inuti ett uttryck.	◀ eller ▶	Håll ned knappen för att upprepa förflyttningen.
Till början av uttrycket.	[2nd] ◀	
Till slutet av uttrycket.	[2nd] ▶	

Obs! *Om du av misstag råkar trycka på ▲ i stället för ◀ eller ▶, flyttas markören upp i historiklistan. Tryck på [ESC] eller tryck på ▼ tills markören återgår till inmatningsraden.*

Ta bort ett tecken

Om du vill ta bort:	Trycker du på:	
Tecknet till vänster om markören.	[←]	Håll ned [←] för att ta bort flera tecken.
Tecknet till höger om markören.	[◆] [←]	
Alla tecken till höger om markören.	[CLEAR] (endast en gång)	Om det inte finns några tecken till höger om markören raderas hela inmatningsraden.

Rensa inmatningsraden

Så här rensar du inmatningsraden:

- Tryck på [CLEAR] om markören är i början eller slutet av inmatningsraden.

  – eller –
- Tryck på [CLEAR] [CLEAR] om markören inte är i början eller slutet av inmatningsraden. När du trycker på tangenten första gången tas alla tecken som är till höger om markören bort, och andra gången du trycker på tangenten rensas hela inmatningsraden.

Infoga eller skriva över tecken

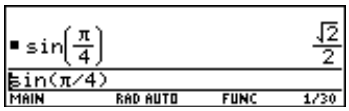
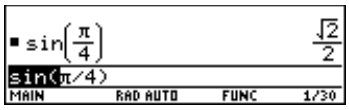
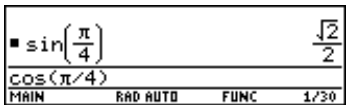
I TI-89 / TI-92 Plus finns både ett infogningsläge och ett överskrivningsläge. Som standard TI-89 / TI-92 Plus är infogningsläget aktiverat. Du växlar mellan de två lägena genom att trycka på [2nd] [INS].

__Tips!__ Du ser på markören om infogningsläget eller överskrivningsläget är aktiverat.

Om TI-89 / TI-92 Plus är i:	Kommer nästa tecken du skriver att:
Insert mode Smal markör mellan tecknen	Infogas vid markören.
Overtype mode Markören markerar tecknet	Ersätta det markerade tecknet.

Ersätta och ta bort flera tecken

Markera först tecknet eller tecknen. Ersätt eller ta sedan bort alla markerade tecken.

Om du vill:	Gör du följande:
Markera flera tecken	1. Flytta markören till ena sidan av tecknen som du vill markera.  Om du vill ersätta **sin(** med **cos(**, placerar du markören bredvid **sin**. 2. Håll ned [↑] och tryck på ◀ eller ▶ för att markera tecknen till vänster eller höger om markören.  Håll ned [↑] och tryck på ▶ ▶ ▶ ▶.
Ersätta de markerade tecknen – eller –	Skriv de nya tecknen. 
Ta bort de markerade tecknen	Tryck på [←].

__Tips!__ När du markerar tecken som du ska ersätta måste du komma ihåg att vissa funktionstangenter automatiskt lägger till en öppen parentes.

> För att det inte ska bli för trångt på tangentbordet använder TI-89 / TI-92 Plus menyer (i stället för endast tangenter) för att få tillgång till många funktioner. I det här avsnittet får du en översikt över hur du väljer ett alternativ på en meny. De olika menyerna beskrivs i olika kapitel i denna handbok.

Visa en meny

Tryck på:	För att visa:
[F1], [F2], osv.	En meny — Öppnas från verktygsfältet överst i de flesta applikationerfönster. På menyn kan du välja funktioner som passar.
[APPS]	Menyn APPLICATIONS — Öppnar en lista med applikationer som du kan välja från. Se sidan 38.
[2nd] [CHAR]	Menyn CHAR — Där du kan välja specialtecken (grekiska, matematiska osv) från olika kategorier.
[2nd] [MATH]	Menyn MATH — Där du kan välja från olika kategorier med matematiska funktioner.
TI-89: [CATALOG] **TI-92 Plus:** [2nd] [CATALOG]	Menyn CATALOG — Där du kan välja från en fullständig alfabetisk lista över de inbyggda funktionerna och instruktionerna. Låter dig även välja användardefinierade applikationer eller Flash-applikationer (om du laddat ner eller definierat några).
[2nd] [CUSTOM]	Menyn CUSTOM — Öppnar en meny där du själv kan lägga till tillgängliga funktioner, instruktioner eller tecken. På TI-89 / TI-92 Plus innefattar detta en standardmeny som du själv kan ändra och anpassa. Se sidan 37 och kapitel 17.

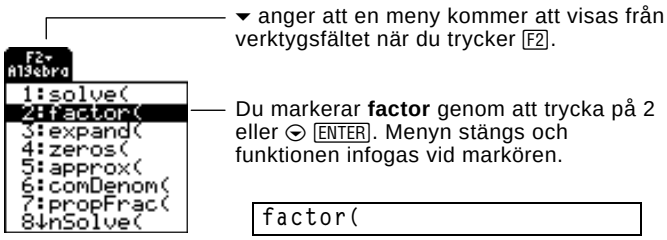
Välja ett alternativ på en meny

Du väljer ett menyalternativ på den markerade menyn på något av följande sätt:

- Tryck på siffran eller bokstaven som visas till vänster om menyalternativet. För bokstäver trycker du [alpha] på TI-89 och sedan bokstavstangenten.

  – eller –

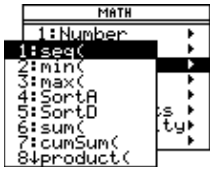
- Använd markörknappen ⊖ och ⊕ för att markera menyalternativet och tryck sedan på [ENTER]. (*Du kan* flytta markeringen till det sista menyalternativet genom att trycka på ⊕ från det första alternativet, eller tvärtom.)

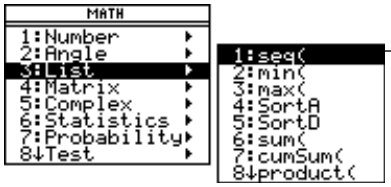


Menyalternativ som slutar med ▶ (undermenyer)

Om du väljer ett menyalternativ som slutar med ▶, visas en undermeny. Du kan därifrån välja ett alternativ på undermenyn.

***Obs!** På grund av den begränsade skärmstorleken överlappar din TI-89 dessa menyer som:*





List visar exempelvis en undermeny där du kan välja en speciell listfunktion.

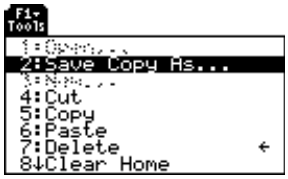
↓ anger att du kan använda markörknappen för att rulla ned till fler menyalternativ.

För menyalternativ som är undermenyer kan du använda markörknappen enligt instruktionen nedan.

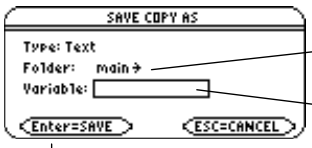
- Du visar undermenyn för det markerade alternativet genom att trycka på ⊙. (Detta är det samma som att välja alternativet.)
- Du stänger undermenyn utan att välja något genom att trycka på ⊙. (Detta är det samma som att trycka på [ESC].)
- Du kan hoppa direkt till det sista alternativet i en meny från det första genom att trycka på ⊙. Om du vill hoppa direkt till det första alternativet från det sista trycker du på ⊙.

Menyalternativ som innehåller ". . ." (dialogrutor)

Om du väljer ett menyalternativ som har tre prickar efteråt (. . .) visas en dialogruta när du väljer alternativet. Där kan du mata in ytterligare information.



Save Copy As ... visar exempelvis en dialogruta där du får mata in ett mapp- och ett variabelnamn.



→ anger att du kan trycka på ⊙ för att visa och välja från en meny.

En inmatningsruta visar att du måste ange ett värde. Använd [alpha] när det är nödvändigt. (Alfalås aktiveras automatiskt på TI-89. Se sidan 22.)

Efter att du har skrivit i en inmatningsruta, exempelvis Variable, måste du trycka på [ENTER] två gånger för att spara informationen och stänga dialogrutan.

Stänga en meny

Du stänger den markerade menyn utan att välja något alternativ genom att trycka på [ESC]. Beroende på om undermenyer visas eller inte kanske du måste trycka på [ESC] flera gånger för att stänga alla menyer.

Flytta från en meny till en annan

Om du vill flytta från en meny till en annan utan att välja något alternativ gör du något av följande:

- Tryck på tangenten (F1, F2, osv.) för den andra menyn.

  – eller –

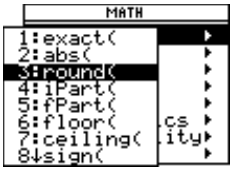
- Använd markörknappen för att flytta till nästa (tryck på ⊕) eller föregående (tryck på ⊖) meny. När du trycker på ⊕ från den sista menyn kommer du till den första menyn och tvärtom.

När du använder ⊕ måste du se till att inte markera ett menyalternativ som är en undermeny. Om du gör ⊕ det visas undermenyn i stället för nästa meny.

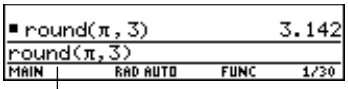
Exempel: Välja ett menyalternativ

Runda av värdet för π till tre decimaler. Starta på en tom inmatningsrad i grundfönstret:

1. Tryck på 2nd MATH för att visa menyn MATH.



2. Tryck på 1 för att visa undermenyn Number. (Eller tryck på ENTER eftersom det första alternativet automatiskt markeras.)
3. Tryck på 3 för att välja **round**. (Eller tryck på ⊖ ⊖ och ENTER.)
4. Tryck på 2nd [π] , 3) och sedan på ENTER för att evaluera uttrycket.



När du valde funktionen i steg 3 infogades automatiskt **round(** på inmatningsraden.

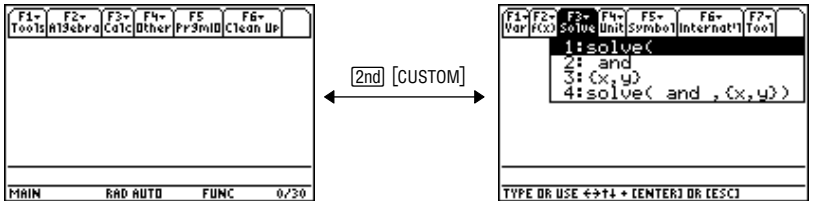
Använda menyn Custom

Din TI-89 / TI-92 Plus har en speciell meny som du kan redigera och aktivera/deaktivera när du vill. Du kan använda den förvalda anpassade menyn, eller skapa egna menyer enligt beskrivningarna i kapitel 17: Applikationermering.

Aktivera och deaktivera menyn

__Obs!__ Du kan också aktivera och deaktivera menyn genom att skriva **CustmOn** *eller* **CustmOff** *på inmatningsraden och trycka på* [ENTER].

När du aktiverar den anpassade menyn ersätts den normala menyn. När du deaktiverar den så återgår menyn till standardinställningen. Nedan följer ett exempel på användningen i grundfönstret:



Normalt verktygsfält i grundfönstret — Anpassad meny

Om menyn inte har redigerats visas den förvalda menyn.

__Tips!__ Med en anpassad meny får du snabbt åtkomst till de funktioner du använder oftast. I kapitel 17 beskrivs hur du skapar anpassade menyer och anpassar dem till dina behov.

Meny	Funktion
[F1] Var	Vanliga variabelnamn.
[F2] f(x)	Funktionsnamn som f(x), g(x) och f(x,y).
[F3] Solve	Funktioner för lösning av ekvationer.
[F4] Unit	Vanliga enheter som _m, _ft och _l.
[F5] Symbol	Symboler som #, ? och ~.
Internat'l **TI-89:** [2nd] [F6] **TI-92 Plus:** [F6]	Vanliga accentuerade tecken som è, é och ê.
Tool **TI-89:** [2nd] [F7] **TI-92 Plus:** [F7]	**ClrHome**, **NewProb**, och **CustmOff**.

Återställa den förvalda menyn

Om en anpassad meny visas och du vill återställa den till den förvalda följer du instruktionerna nedan:

1. Tryck på [2nd] [CUSTOM] i grundfönstret för att deaktivera den anpassade menyn och visa grundfönstrets normala menyrad.

__Obs!__ Den förra menyn raderas. Om menyn skapades med ett applikationer (kapitel 17), kan den återskapas genom att köra det applikationermet igen.

2. Öppna menyn [2nd] [F6] Clean Up och välj 3:Restore custom default.

   TI-89: [2nd] [F6]
   TI-92 Plus: [F6]
   Kommandot som används för att skapa den förvalda menyn klistras in på inmatningsraden.

3. Tryck [ENTER] för att genomföra kommandot och återskapa menyn.

Välja ett applikationer

I TI-89 / TI-92 Plus finns olika applikationer med vilka du kan lösa och utforska olika typer av problem. Du kan välja en applikation från en meny eller få tillgång till vanliga applikationer direkt från tangentbordet.

Från menyn APPLICATIONS

__Obs!__ Du stänger menyn utan att välja något alternativ genom att trycka på [ESC].

1. Tryck på [APPS] för att visa en meny som innehåller en lista över applikationerna.
2. Välj applikation genom att göra något av följande:
   - Använd markörknappen ⊖ eller ⊕ för att markera applikationen och tryck på [ENTER].

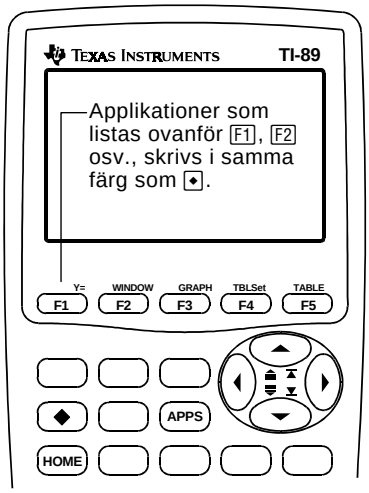
     – eller –
   - Tryck på siffran som motsvarar applikationen.

Applikationer:	Vad du kan göra:
FlashApps	Öppnar en lista med Flash-applikationen (om installerade).
Y= Editor	Definiera, redigera och välja funktioner eller ekvationer för grafer (kapitel 6 – 11).
Window Editor	Ställa in fönsterstorlek för att visa en graf (kapitel 6).
Graph	Visa grafer (kapitel 6).
Table	Visa en tabell med variabelvärden som motsvarar en angiven funktion (kapitel 13).
Data/Matrix Editor	Mata in och redigera listor, data och matriser. Du kan göra statistiska beräkningar och rita statistiska diagram (kapitel 15 och 16).
Applikationer Editor	Mata in och redigera applikationer och funktioner (kapitel 17).
Text Editor	Mata in och redigera en textsession (kapitel 18).
Numeric Solver	Skriva in ett uttryck eller ekvation. Definiera värden för alla variabler utom en och lös sedan för den okända variabeln (Kapitel 19).
Home	Mata in uttryck och instruktioner och göra beräkningar.

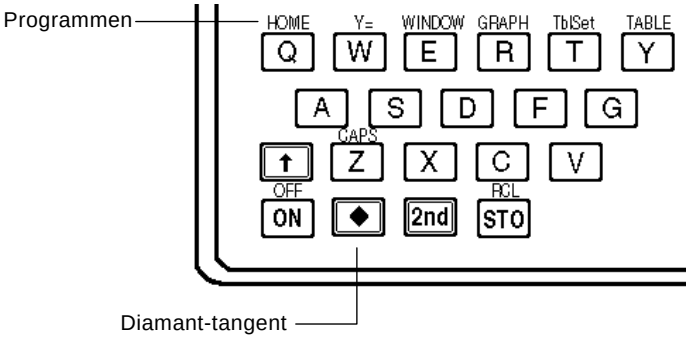
Från tangentbordet

Du kan öppna de vanligare applikationerna direkt från tangentbordet. På TI-89, betyder t ex [◆] [Y=] samma sak som att trycka [◆] och sedan [F1]. Denna handbok använder beteckningen [◆] [Y=], vilket liknar beteckningen som används för tangenternas “andra”-funktionen.

Applikationer:	Tryck:
Home	**TI-89:** [HOME] **TI-92 Plus:** [◆] [HOME]
Y= Editor	[◆] [Y=]
Window Editor	[◆] [WINDOW]
Graph	[◆] [GRAPH]
Table Setup	[◆] [TblSet]
Table Screen	[◆] [TABLE]



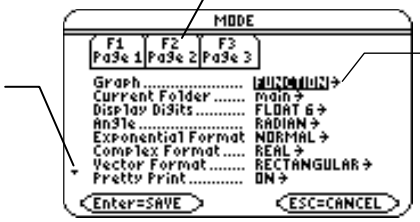
På TI-92 Plus, applikationer listas ovanför QWERTY-tangenterna.



Med hjälp av olika lägen styr du hur tal och grafer ska visas och tolkas. Lägesinställningar sparas med funktionen Constant Memory™ när räknaren TI-89 / TI-92 Plus är avstängd. Alla tal, även matriselement och listor visas enligt de aktuella lägesinställningarna.

Kontrollera inställningar för lägen

Tryck på [MODE] för att visa dialogrutan MODE, som innehåller en lista över alla lägen och deras aktuella inställningar.



Obs! Lägen som för tillfället är inaktiva är nedtonade. På andra sidan är exempelvis Split 2 App inte giltig när Split Screen = FULL. När du rullar igenom listan hoppar markören över de nedtonade inställningarna.

Ändra inställningar för lägen

Du kan göra följande från dialogrutan MODE:

1. Markera inställningen som du vill ändra. Använd ⊙ eller ⊙ (med [F1], [F2], eller [F3]) för att rulla igenom listan.
2. Tryck på ⊙ eller ⊙ för att visa en meny som innehåller en lista över de giltiga inställningarna. Den aktuella inställningen är markerad.
3. Välj en inställning. Gör något av följande:
   - Använd ⊙ eller ⊙ för att markera inställningen och tryck på [ENTER].

     – eller –
   - Tryck på siffran eller bokstaven som motsvarar inställningen.
4. Ändra andra inställningar för lägen om det behövs.
5. När du har gjort alla ändringar trycker du på [ENTER] för att spara ändringarna och stänga dialogrutan.

***Tips!** Om du vill stänga en meny och återgå till dialogrutan MODE utan att välja några alternativ trycker du på [ESC].*

Viktigt! Om du trycker på [ESC] i stället för [ENTER] för att stänga dialogrutan MODE, avbryts alla ändringar du har gjort.

Översikt över lägen

Obs! *Detaljerad information om ett visst läge finns i motsvarande avsnitt i denna handbok.*

Läge	Beskrivning
Graph	Grafer som kan plottas: FUNCTION, PARAMETRIC, POLAR, SEQUENCE, 3D eller DE.
Current Folder	Mapp som används för att lagra och hämta variabler i. Såvida du inte har skapat ytterligare mappar är endast mappen MAIN tillgänglig. Se avsnittet "Använda mappar för att spara oberoende uppsättningar variabler" i kapitel 5.
Display Digits	Maximalt antal siffror (FLOAT) eller ett fast antal decimaler (FIX) visas i decimaltalsresultat. Oavsett inställningen kan det totala antalet siffror som visas i decimaltalsresultat inte överstiga 12. Se sidan 31.
Angle	Enheter där vinklar tolkas och visas: RADIAN eller DEGREE.
Exponential Format	Notationsformat som används för att visa resultat: NORMAL, SCIENTIFIC eller ENGINEERING. Se sidan 31.
Complex Format	Format som används för att visa komplexa resultat, om resultatet över huvud taget kan visas: REAL (komplexa resultat visas inte om du inte använder en komplex inmatning), RECTANGULAR eller POLAR.
Vector Format	Format som används för att visa 2- och 3-elements-vektorer: RECTANGULAR, CYLINDRICAL eller SPHERICAL.
Pretty Print	Aktiverar och avaktiverar funktionen pretty print. Se sidan 29.
Split Screen	Delar fönstret i två delar och anger hur de båda delarna ska ordnas: FULL (ingen delning av fönstret), TOP-BOTTOM eller LEFT-RIGHT. Se kapitel 14.
Split 1 App	Applikationen överst eller till vänster i ett delat fönster. Om du inte använder ett delat fönster är detta det aktuella applikationermet.
Split 2 App	Applikationen längst ned eller till höger i ett delat fönster. Detta är endast aktivt i ett delat fönster.
Number of Graphs	I ett delat fönster kan du ordna båda sidorna av fönstret så att oberoende uppsättningar grafer kan visas.
Graph 2	Om Number of Graphs = 2, väljs en typ av graf i Split 2-delen av fönstret (nederst eller till höger). Se kapitel 12.
Split Screen Ratio	Relativa storlekar på de två delarna av ett delat fönster: 1:1, 1:2 eller 2:1 (endast TI-92 Plus).
Exact/ Approx	Beräknar uttryck och visar resultat i numerisk eller symbolisk form: AUTO, EXACT eller APPROXIMATE. Se sidan 29.

Lägen
(fortsättning)

Läge	Beskrivning
Base	Låter dig utföra beräkningar med tal inmodate i decimal (DEC), hexadecimal (HEX), eller binär (BIN) form.
Unit System	Låter dig mata in enheter för olika värden I ett uttryck, t ex 6_m * 4_m eller 23_m/_s * 10_s. Du kan även omvandla värden från en enhet till en annan I samma kategori, eller skapa egna användardefinierade måttenheter.
Custom Units	Låter dig välja egna standardvärden. Inställningen är gråmarkerad tills du väljer Unit System, 3:CUSTOM.
Language	Med denna inställning kan du anpassa din TI-89 / TI-92 Plus till ett valfritt språk utifrån de språk som finns installerade.

Använda menyn Clean Up för att påbörja ett nytt problem

I grundfönstret kan du använda menyn Clean Up när du vill starta en ny beräkning från nollställt läge utan att återställa minnet i din TI-89 / TI-92 Plus.

Clean Up verktygsmeny

Öppna menyn Clean Up i grundfönstret genom att trycka på:
TI-89: [2nd] [F6]
TI-92 Plus: [F6]

Menyval	Beskrivning
Clear a–z	Raderar (tar bort) alla variabler i aktuell katalog som har ett namn med bara ett tecken. Låsta eller sparade variabler tas inte bort. Du måste trycka [ENTER] för att bekräfta att du verkligen vill ta bort variablerna. Variabelnamn med bara ett tecken används ofta i symboliska beräkningar som: solve(a•x²+b•x+c=0,x) Om någon av variablerna redan tilldelats ett värde kan beräkningen ge felaktiga resultat. Du förhindrar detta genom att välja 1:Clear a–z innan du påbörjar beräkningen.
NewProb	Placerar **NewProb** på inmatningsraden. Du måste trycka [ENTER] för att exekvera kommandot. **NewProb** genomför flera operationer som låter dig påbörja ett nytt problem med nollställda variabler utan att återställa minnet: • Raderar alla variabler med namn som består av bara ett tecken i aktuell katalog (samma som 1:Clear a–z). Låsta eller sparade variabler raderas inte. • Stänger av alla funktioner och plottningar (**FnOff** och **PlotsOff**) i aktuellt grafläge. • Exekverar **ClrDraw**, **ClrErr**, **ClrGraph**, **ClrHome**, **ClrIO**, och **ClrTable**.
Restore custom default	Om du har en annan anpassad meny än förvald kan du återställa den förvalda. Se sidan 37.

Tips! *När du definierar en variabel som du vill behålla kan du ange ett namn som är längre än ett tecken. Genom detta kan den inte tas bort av misstag med kommandot* 1:Clear a–z.

Obs! *Information om hur du kontrollerar och återställer minnet eller systemförval hittar du i kapitel 21.*

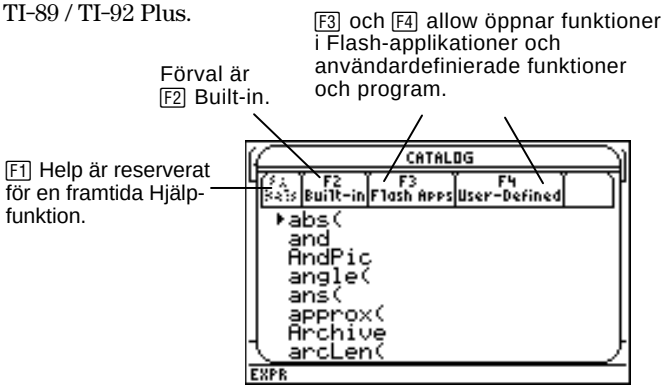
Dialogrutan CATALOG är ett enkelt sätt att komma åt alla inbyggda kommandon (funktioner och instruktioner) i en TI-89 / TI-92 Plus. I dialogrutan CATALOG kan du även använda funktioner som definerats i Flash-applikationer eller funktioner som du själv definierat (om du har laddat in applikationer eller definierat funktioner).

Öppna CATALOG

När du vill öppna dialogrutan CATALOG trycker du på:

TI-89: [CATALOG]
TI-92 Plus: [2nd] [CATALOG]

Förval för CATALOG är [F2] Built-in, vilket öppnar en lista med alla fördefinierade kommandon (funtioner och instruktioner) i en TI-89 / TI-92 Plus.



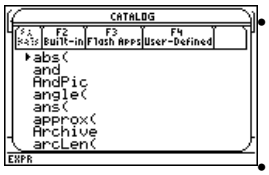
Obs! Alternativ som för tillfället inte är användbara är gråmarkerade. Till exempel, [F1] Help är reserverat för en framtida Hjälp-funktion. [F3] Flash Apps är gråmarkerat om du inte installerat några Flash-applikationer. [F4] User-Defined är gråmarkerat om du inte skapat några funktioner eller applikationer.

Välja från katalogen (CATALOG)

När du väljer ett kommando infogas dess namn på inmatningsraden vid markörpositionen. Du bör därför placera markören på rätt plats innan du väljer kommandot.

1. Tryck på:
   TI-89: [CATALOG]
   TI-92 Plus: [2nd] [CATALOG]
2. Tryck på Built-in.

Obs! Första gången du visar listan är det översta alternativet valt. Nästa gång du öppnar den är det senast valda alternativet markerat.



- Kommandon visas i alfabetisk ordning. Kommandon som inte börjar med en bokstav (+, %, √, Σ m.m.) visas i slutet av listan.
- Du stänger CATALOG utan att välja ett kommando genom att trycka på [ESC].

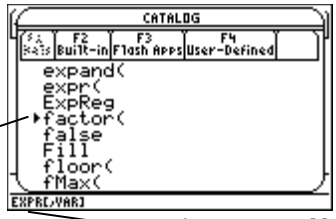
3. Flytta indikatorn ▶ till kommandot och tryck på [ENTER].

Om du vill flytta indikatorn ▶:	Trycker du på eller skriver:
Ett kommando i taget	⊖ eller ⊕
Ett fönster i taget	[2nd] ⊖ eller [2nd] ⊕
Till det första kommandot som börjar med en angiven bokstav	Bokstavstangenten. (På TI-89 ska du **inte** trycka på [alpha] först. Om du gör det måste du trycka [alpha] eller [2nd] [a-lock] igen innan du kan skriva in en bokstav.)

Tips! *Tryck på ⊕ för att flytta till botten av listan (om du är högst upp). Tryck på ⊖ för att flytta till toppen av listan (om du är längst ned).*

Hjälpinformation om parametrar

I statusfältet visas de parametrar som krävs och de som är valfria och vilken typ av parameter du ska använda, för det kommando som anges av symbolen ▶.



Markerat kommando och dess parametrar

Hakparenteser [ ] anger valfria parametrar.

Obs! *Detaljerad information om parametrar finns i appendix A.*

Från exemplet ovan är syntaxen för **factor**:

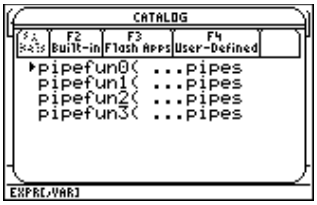
factor(*uttryck*)	krävs
– eller –	
factor(*uttryck*,*variabel*)	valfritt

Välja en funktion i en Flash-applikation

Flash-applikationer kan innehålla en eller flera funktioner. När du väljer en funktion sätts funktionens namn in på inmatningsraden vid markören. Du måste därför placera markören på rätt plats på inmatningsraden innan du väljer en funktion.

1. Tryck på:
   TI-89: [CATALOG]
   TI-92 Plus: [2nd] [CATALOG]

2. Tryck på [F3] Flash Apps. (Detta alternativ är gråmarkerat om du inte installerat några Flash-applikationer på din TI-89 / TI-92 Plus.)



- Listan visar funktionerna i alfabetisk ordning efter funktionsnamn. Den vänstra kolumnen visar funktioner. I den högra kolumnen visas vilket applikationer som innehåller den aktuella funktionen.
- Information om markerad funktion visas på statusraden.
- Om du vill avbryta utan att välja någon funktion trycker du på [ESC].

3. Flytta indikatorn ▶ till önskad funktion och tryck på [ENTER].

När du vill flytta indikatorn ▶:	Tryck på eller skriv:
En funktion i taget	⊖ eller ⊕
En sida i taget	[2nd] ⊖ eller [2nd] ⊕
Till den första funktionen som börjar med ett visst tecken	Bokstavstangenten. (På TI-89 ska du **inte** trycka på [alpha] först. Om du gör det måste du trycka på [alpha] eller [2nd] [a-lock] igen innan du kan skriva in en bokstav.)

Välja en användardefinierad funktion eller applikationer

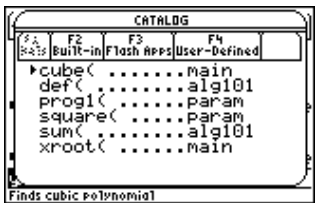
Du kan skapa egna funktioner eller applikationer och sedan använda [F4] User-Defined för att öppna dem. Instruktioner om hur du skapar funktioner finns i avsnitten "Skapa och evaluera användardefinierade funktioner" i kapitel 5 och "Mata in funktioner - översikt" i kapitel 17. I kapitel 17 finns även instruktioner om hur du skapar och kör applikationer.

När du väljer en funktion sätts funktionens namn in på inmatningsraden vid markören. Du måste därför placera markören på rätt plats på inmatningsraden innan du väljer en funktion.

1. Tryck på:
   TI-89: [CATALOG]
   TI-92 Plus: [2nd] [CATALOG]

2. Tryck på [F4] User-Defined. (Detta alternativ är gråmarkerat om du inte definierat några funktioner eller applikationer.)

__Obs!__ Använd fönstret VAR-LINK för att hantera variabler, mappar och Flash-applikationer. Se kapitel 21.



- Listan visar funktionerna i alfabetisk ordning efter funktionsnamn. Den vänstra kolumnen visar funktioner och applikationer. I den högra kolumnen visas vilken mapp som innehåller den aktuella funktionen.
- Om den första raden i en funktion eller ett applikationer består av en kommentar visas kommentaren på statusraden.
- Om du vill avbryta utan att välja någon funktion trycker du på [ESC].

3. Flytta indikatorn ▶ till önskad funktion eller applikationer och tryck på [ENTER].

När du vill flytta indikatorn ▶:	Tryck på eller skriv:
En funktion i taget	⊖ eller ⊕
En sida i taget	[2nd] ⊖ eller [2nd] ⊕
Till den första funktionen som börjar med ett visst tecken	Bokstavstangenten. (På TI-89 ska du **inte** trycka på [alpha] först. Om du gör det måste du trycka på [alpha] eller [2nd] [a-lock] igen innan du kan skriva in en bokstav.)

När du sparar ett värde sparas det som en namngiven variabel. Du kan sedan använda namnet i stället för värdet i uttryck. När TI-89 / TI-92 Plus namnet påträffas i ett uttryck substitueras variabelns sparade värde.

Regler för variabelnamn

Ett variabelnamn:

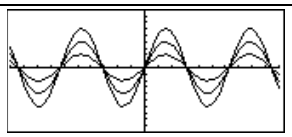
- Kan innehålla 1 till 8 tecken bestående av bokstäver och siffror. Detta inkluderar grekiska tecken (förutom π), bokstäver med accent och internationella bokstäver. Använd inte mellanslag.
  - Det första tecknet kan inte vara en siffra.
- Kan innehålla både stora (versaler) och små (gemener) bokstäver. Följande namn: AB22, Ab22, aB22 och ab22 refererar alla till samma variabel.
- Kan inte vara detsamma som ett namn som på förhand har tilldelats av TI-89 / TI-92 Plus. Sådana namn inkluderar:
  - Inbyggda funktioner (exempelvis **abs**) och instruktioner (exempelvis **LineVert**). Se appendix A.
  - Systemvariabler (exempelvis xmin och xmax, som används för att spara grafrelaterade värden). I appendix B finns en lista.

Exempel

Variabel	Beskrivning
myvar	OK
a	OK
Log	Inte OK, namnet är på förhand tilldelat funktionen **log.**
Log1	OK
3rdTotal	Inte OK, börjar med en siffra.
circumfer	Inte OK, mer än åtta tecken.

Datatyper

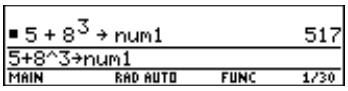
Du kan spara alla TI-89 / TI-92 Plus datatyper som variabler. I avsnittet **getType()** i appendix A finns en lista över datatyper. Några exempel:

Datatyp	Exempel
Uttryck	2,54, 1,25E 6, 2π, xmin/10, 2+3*i*, $(x-2)^2$, $\sqrt{2}/2$
Listor	{2 4 6 8}, {1 1 2}
Matriser	[1 0 0], $\begin{bmatrix} 1 & 0 & 0 \\ 3 & 4 & 6 \end{bmatrix}$
Teckensträngar	"Hello", "The answer is:", "xmin/10"
Bilder	
Funktioner	myfunc(arg), ellipse(x,y,r1,r2)

Spara ett värde i en variabel

__Obs!__ Om du har en TI-89 använder du [alpha] efter behov för att skriva in bokstäver.

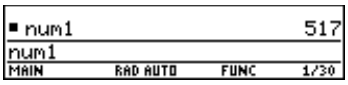
1. Mata in värdet du vill spara, som kan vara ett uttryck.
2. Tryck på [STO▸]. Symbolen för att spara (→) visas.
3. Skriv variabelnamnet.
4. Tryck på [ENTER].



Du kan lagra en variabel temporärt genom att använda operatorn "with". Se "Substituera värden och ange restriktioner" i kapitel 3.

Visa en variabel

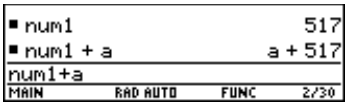
1. Skriv variabelnamnet.
2. Tryck på [ENTER].



Om variabeln är odefinierad visas variabelnamnet i resultatet.

__Obs!__ I kapitel 3 finns information om symbolisk hantering.

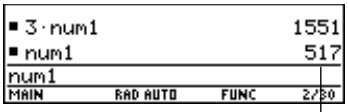
I detta exempel är variabeln odefinierad. Den används därför som en symbolisk variabel.



Använda ett värde i ett uttryck

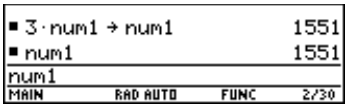
__Tips!__ Om du vill visa en lista över befintliga variabelnamn använder du [2nd] [VAR-LINK], enligt beskrivningen i kapitel 21.

1. Skriv variabelnamnet i uttrycket.
2. Tryck på [ENTER] för att beräkna uttrycket.



Variabelns värde ändras inte.

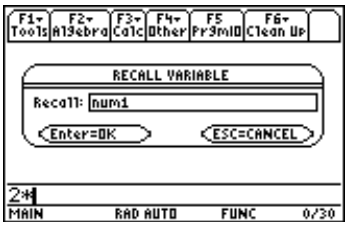
Om du vill att resultatet ska substituera variabelns föregående värde måste du spara resultatet.



Hämta en variabels värde

I vissa fall kanske du vill använda en variabels verkliga värde i ett uttryck i stället för variabelnamnet.

1. Tryck på [2nd] [RCL] för att visa en dialogruta.
2. Skriv variabelnamnet.
3. Tryck på [ENTER] två gånger.



I detta exempel kommer värdet som är sparat i num1 att infogas vid markörpositionen på inmatningsraden.

Återanvända en tidigare inmatning eller det senaste resultatet

Du kan återanvända en tidigare inmatning genom att beräkna inmatningen igen i befintligt skick eller genom att redigera inmatningen och sedan utföra beräkningen igen. Du kan också återanvända det senaste beräknade resultatet genom att infoga det i ett uttryck.

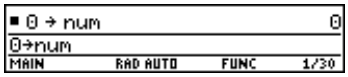
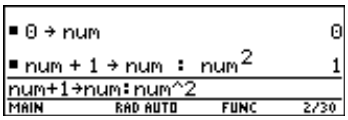
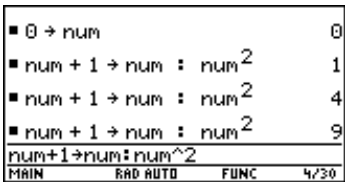
Återanvända uttrycket på inmatningsraden

När du trycker på [ENTER] för att beräkna ett uttryck stannar uttrycket kvar på inmatningsraden och markeras. Du kan skriva över inmatningen eller återanvända den.

Exempel: Ta med hjälp av en variabel reda på kvadraten på 1, 2, 3 osv.

Som visas i exemplet nedan ställer du först in startvärdet för variabeln och skriver sedan in uttrycket. Mata därefter in värdet igen för att öka variabeln och beräkna kvadraten.

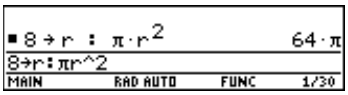
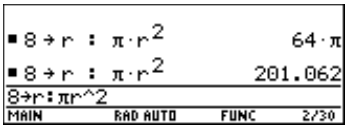
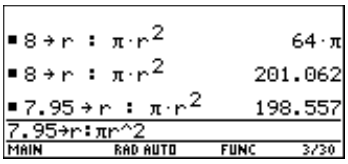
Tips! *Att utföra en inmatning flera gånger i befintligt skick är användbart för upprepade beräkningar som involverar variabler.*

På TI-89:	**På TI-92 Plus:**	**Teckenfönster**
0 [STO▸] [2nd] [a-lock] N U M [ENTER]	0 [STO▸] N U M [ENTER]	
N U M [alpha] [+] 1 [STO▸] [2nd] [a-lock] N U M [2nd] [:] N U M [^] 2 [ENTER]	N U M [+] 1 [STO▸] N U M [2nd] [:] N U M [^] 2 [ENTER]	
[ENTER] [ENTER]	[ENTER] [ENTER]	

Tips! *Genom att redigera en inmatning kan du göra mindre ändringar utan att behöva skriva om hela inmatningen.*

Utgå från ekvationen $A=\pi r^2$ och prova dig fram för att ta reda på radien av en cirkel som täcker 200 kvadratcentimeter.

I exemplet nedan används 8 som första gissning och det ungefärliga svaret visas i flyttalsform. Du kan redigera och köra beräkningen igen med t ex 7.95 och fortsätta tills svaret är så exakt som du vill ha det.

På TI-89:	**På TI-92 Plus:**	**Teckenfönster**
8 [STO▸] [alpha] R [2nd] [:] [2nd] [π] [alpha] R [^] 2 [ENTER]	8 [STO▸] R [2nd] [:] [2nd] [π] R [^] 2 [ENTER]	
[◆] [ENTER]	[◆] [ENTER]	
[◀] [◆] [DEL] 7 . 9 5 [ENTER]	[◀] [◆] [DEL] 7 . 9 5 [ENTER]	

Obs! *När inmatningen innehåller ett decimalkomma visas resultatet automatiskt i decimalform.*

Hämta en tidigare inmatning

Du kan återanvända en föregående inmatning som är sparad i historiklistan, även om inmatningen har rullat ut från fönstret. Den återkallade inmatningen *ersätter* den som för tillfället visas på inmatningsraden. Du kan därefter beräkna eller redigera den återkallade inmatningen.

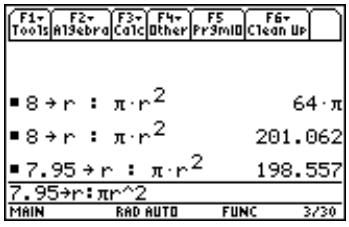
Obs! *Du kan också använda funktionen* ***entry*** *för att hämta en föregående inmatning. Se avsnittet* ***entry()*** *i appendix A.*

Om du vill återkalla:	**Trycker du på:**	**Resultat:**
Den senaste inmatningen (om du har ändrat inmatningsraden)	[2nd] [ENTRY] en gång	Om den senaste inmatningen fortfarande visas på inmatningsraden, hämtas den inmatning som föregår den.
Föregående inmatningar	[2nd] [ENTRY] flera gånger	För varje gång du trycker på tangenterna hämtas den inmatning som föregår den som visas på inmatningsraden.

Exempel:

Om inmatningsraden innehåller den sista inmatningen, hämtar [2nd] [ENTRY] denna inmatning.

Om inmatningsraden redigeras eller rensas, hämtar [2nd] [ENTRY] denna inmatning.



Hämta det senaste resultatet

Varje gång du beräknar ett inmatat uttryck sparar TI-89 / TI-92 Plus svaret till variabeln ans(1). Du infogar denna variabel på inmatningsraden genom att trycka på [2nd] [ANS].

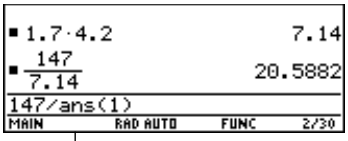
Exempel: Räkna ut arean av en bit trädgård som är 1,7 meter gånger 4,2 meter. Räkna sedan ut hur mycket den ger per kvadratmeter om trädgårdsbiten producerar totalt 147 tomater.

1. Räkna ut arean.

   1.7 [×] 4.2 [ENTER]

2. Räkna ut avkastningen.

   147 [÷] [2nd] [ANS] [ENTER]



Variabeln ans(1) infogas och dess värde används vid beräkningen.

Obs! *Se avsnittet* ***ans()*** *i appendix A.*

Liksom ans(1) alltid innehåller det senaste svaret, innehåller ans(2), ans(3) osv alltid tidigare svar. ans(2) innehåller exempelvis det näst sista svaret.

Automatiskt klistra in en inmatning eller ett resultat från historiklistan

Du kan välja vilken inmatning eller vilket svar som helst från historiklistan och automatiskt klistra in en kopia på inmatningsraden. På detta sätt infogar du en föregående inmatning eller ett tidigare resultat i ett nytt uttryck utan att behöva skriva om den föregående informationen.

Automatisk inklistring

Resultatet av att använda automatisk inklistring är likvärdigt med att använda [2nd] [ENTRY] och [2nd] [ANS] som beskrivs i föregående avsnitt, men med vissa skillnader.

Obs! Du kan också klistra in information genom att använda menyn [F1]. Se "Klippa ut, kopiera och klistra in information" i kapitel 5.

	Genom att klistra in kan du:	Med [2nd] [ENTRY] kan du:
För inmatningar:	*Infoga* tidigare inmatningar på inmatningsraden.	*Ersätta* innehållet på inmatningsraden med föregående inmatning.

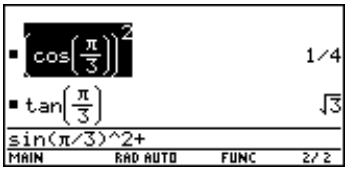
	Genom att klistra in kan du:	Med [2nd] [ANS] kan du:
För resultat:	Infoga *tidigare resultat* på inmatningsraden.	Infoga variabeln ans(1), som endast *innehåller det senaste resultatet*. Varje gång du gör en beräkning uppdateras ans(1) till det senaste resultatet.

Klistra in en inmatning eller ett resultat automatiskt

Tips! Du avbryter den automatiska inklistringen och återgår till inmatningsraden genom att trycka på [ESC].

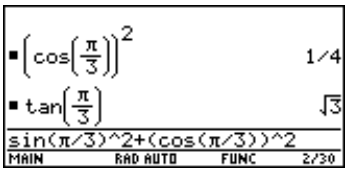
Tips! Du visar en inmatning eller ett resultat som är för långt för en rad (anges med ▸ i slutet av raden) genom att använda ⊕ och ⊖ eller [2nd] ⊕ och [2nd] ⊖.

1. Placera markören på inmatningsraden där du vill infoga inmatningen eller resultatet.
2. Tryck på ⊕ för att flytta markören upp till historiklistan. Det senaste resultatet markeras.
3. Använd ⊕ och ⊖ för att markera den inmatning eller det resultat som automatiskt ska klistras in.
   - Med ⊕ flyttar du från resultat till inmatning upp genom historiklistan.
   - Du kan använda ⊕ för att markera poster som har rullat ut från fönstret.



4. Tryck på [ENTER].

   Den markerade posten infogas på inmatningsraden.

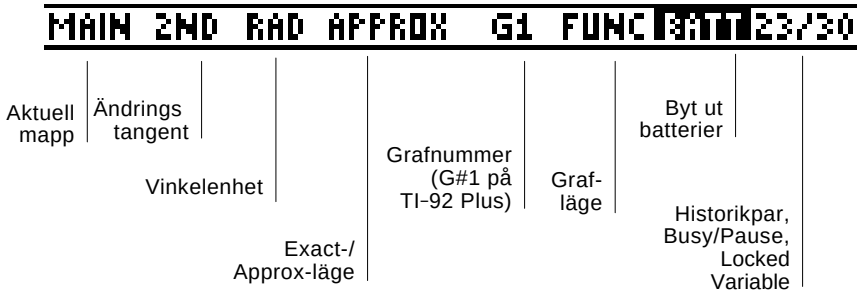


När du gör detta klistras hela inmatningen eller resultatet in. Om du bara behöver en del av inmatningen eller resultatet redigerar du inmatningsraden genom att ta bort det du inte behöver.

Indikatorer för statusfältet i fönstret

Statusfältet visas längst ned i alla applikationerfönster. Där visas information om det aktuella TI-89 / TI-92 Plus- läget, bland annat flera viktiga inställningar för lägen.

Indikatorer för statusfältet



Indikator	Betydelse
Aktuell mapp	Visar namnet på den aktuella mappen. Se "Använda mappar för att spara oberoende uppsättningar variabler" i kapitel 5. MAIN är den standardmapp som installeras automatiskt i TI-89 / TI-92 Plus.
Funktions-tangent	Visar vilken funktion som är aktiverad enligt beskrivningen nedan.
2nd	[2nd] — aktiverar den andra funktionen hos nästa tangent du trycker ned.
◆	[◆] — nästa tangent som trycks ned kommer att använda den diamant-markerade funktionen.
(TI-89)	[alpha] — skriver in bokstaven på nästa tangent i gemener.
(TI-89)	[2nd] [a-lock] — skiftlås för gemener är aktiverat. Tills detta stängs av kommer tecknet att skrivas in som gemen. Deaktivera läget genom att trycka på [alpha].
(TI-89)	[↑] [alpha] — skiftlås för versaler är aktiverat. Tills detta stängs av kommer tecknet att skrivas in som versal. Deaktivera läget genom att trycka på [alpha].
⬆	[↑] — skriver in tecknet på nästa tangent som versal. På TI-89, du kan använda [↑] för att skriva in ett tecken utan att använda [alpha].
Vinkelenhet	Visar i vilka enheter vinkelvärden tolkas och visas. Du ändrar detta läge genom att använda tangenten [MODE].
RAD	Radianer
DEG	Grader

***Obs!** Om du vill avbryta [2nd], [◆], [alpha] eller [↑], trycker du på samma tangent igen eller en annan funktionstangent.*

***Obs!** Om nästa tangent du trycker ned inte har någon andra funktion, eller bokstav, återgår tangenten till grundfunktionen.*

Statusfältet (fortsättning)

Indikator	Betydelse
Exact/ Approx-läge	Visar hur resultat beräknas och visas. Se sidan 29. Du ändrar detta läge genom att använda tangenten MODE.
AUTO	Automatisk
EXACT	Exakt
APPROX	Ungefärligt
Grafnummer	Om fönstret är delat och visar två oberoende grafer anger denna indikator vilken graf som är aktiv GR#1 eller GR#2. (Visar G#1 eller G#2 på TI-92 Plus.)
Grafläge	Anger vilken typ av graf som kan ritas. (Du ändrar detta läge med tangenten MODE.)
FUNC	y(x)-funktioner
PAR	x(t)- och y(t)-ekvationer i parameterform
POL	r(θ)-ekvationer i polär form
SEQ	u(n)-talföljder
3D	z(x,y) 3D-ekvationer
DE	y'(t) differentialekvationer
Batteri-indikator	Visas endast när batterierna börjar ta slut. Om BATT visas på en svart bakgrund bör du byta batterier så snart som möjligt.
Historikpar, Busy/Paus, Archived	Vilken information som visas i denna del av statusraden är beroende på vilkey applikation du använder.
23/30	Visas i grundfönstret och anger det antal inmatningar/svar som finns i historikområdet. Se sidan 24.
BUSY	En beräkning eller plottning pågår.
PAUSE	Du har tillfälligt avbrutit en plottning eller ett applikationer.
🔒	Variabeln som är öppnad i den aktuella redigeraren (Data/Matrix Editor, Applikationer Editor, eller Text Editor) är låst eller lagrad och kan inte ändras.

Visa applikationerversion och ID-nummer

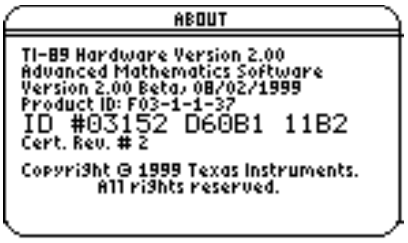
I vissa fall kan du behöva veta vilken version av mjukvara du använder eller vilket ID nummer din TI-89 / TI-92 Plus har.

Visa “About”-skärmen

När du befinner dig i grundfönstret, tryck [F1] och välj A:About.

Din skärm kommer att se annorlunda ut än den som visas till höger.

Tryck [ENTER] eller [ESC] för att stänga skärmen.



När behöver du känna till dessa uppgifter?

Informationen i fönstret About är till för situationer som:

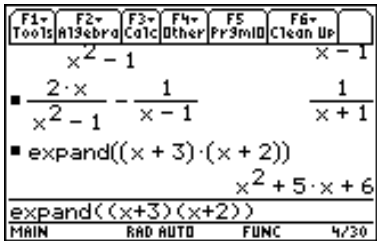
- Om du uppdaterar eller skaffar ny mjukvara till din TI-89 / TI-92 Plus kan du behöva lämna information om din nuvarande version av mjukvaran och/eller ID nummret på din räknare.
- Om du har problem med din TI-89 / TI-92 Plus och måste kontakta den tekniska supportavdelningen kan serienummret göra det lättare att hitta felet.

Algebra

3

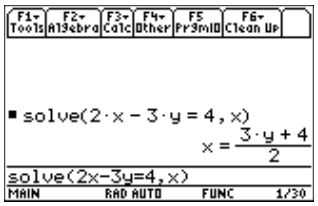
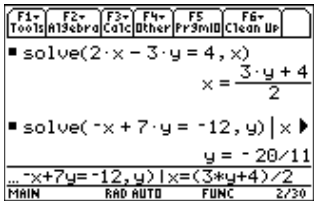
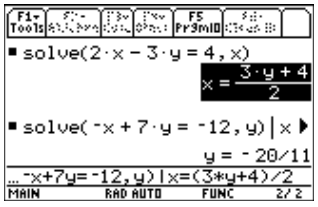
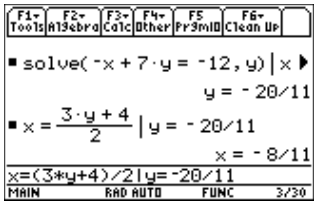
Översikt över algebra 58
Använda definierade eller odefinierade variabler 59
Använda Exact-, Approximate- och Auto-lägena..... 61
Automatisk förenkling..... 64
Fördröjd förenkling för vissa inbyggda funktioner 66
Substituera värden och ange restriktioner 67
Översikt över menyn Algebra..... 70
Vanliga algebraiska operationer..... 72
Översikt över menyn Calc..... 75
Vanliga beräkningsoperationer 76
Användardefinierade funktioner och symbolisk hantering..... 77
Felmeddelande om att minnet håller på att ta slut..... 79
Specialkonstanter som används i algebra..... 80

I detta kapitel finner du en översikt över användningen av de viktigaste symboliska verktygen inom algebra och "calculus".



Du kan enkelt genomföra symboliska beräkningar i grundfönstret.

Lös ekvationssystemet 2x − 3y = 4 och ⁻x + 7y = ⁻12. Lös den första ekvationen så att x uttrycks i termer med y. Substituera uttrycket för x i den andra ekvationen och räkna ut värdet av y. Byt sedan tillbaka värdet av y till den första ekvationen för att lösa ut x-värdet.

Steg	TI-89 Tangenter	TI-92 Plus Tangenter	Visa
1. Visa grundfönstret och rensa inmatningsraden. Lös ekvationen 2x − 3y = 4 för x. F2 1 väljer **solve(** *på menyn* Algebra. *Du kan också skriva in* **solve(** *direkt via tangentbordet eller välja det från* Katalog.	HOME CLEAR CLEAR F2 1 2 X − 3 Y = 4 , X) ENTER	♦ [HOME] CLEAR CLEAR F2 1 2 X − 3 Y = 4 , X) ENTER	
2. Börja med att lösa ekvationen ⁻x + 7y = ⁻12 för y, men tryck inte på ENTER än.	F2 1 (-) X + 7 Y = (-) 1 2 , Y)	F2 1 (-) X + 7 Y = (-) 1 2 , Y)	
3. Använd operatorn "with" för att substituera uttrycket för x som beräknades från den första ekvationen. Detta ger y-värdet. *Operatorn "with" visas som symbolen* **|** *i fönstret.* *Använd den automatiska inklistringsfunktionen för att markera det sista svaret i historiklistan och klistra in det på inmatningsraden.*	| ▲ ENTER ENTER	2nd [|] ▲ ENTER ENTER	
4. Markera ekvationen för x i historiklistan.	▲ ▲ ▲	▲ ▲ ▲	
5. Klistra automatiskt in det markerade uttrycket på inmatningsraden. Substituera sedan y-värdet som räknades ut i den andra ekvationen. Lösningen är: x = ⁻8/11 och y = ⁻20/11	ENTER | ▲ ENTER ENTER	ENTER 2nd [|] ▲ ENTER ENTER	

Detta exempel visar på en symbolisk beräkning. Det finns även en en-stegs-funktion för lösningar av ekvationssystem. (Se sidan 73.)

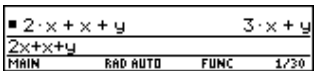
Använda definierade eller odefinierade variabler

> När du utför algebraiska operationer är det viktigt att du förstår innebörden av att använda odefinierade och definierade variabler. Annars finns det risk för att resultatet visas som ett tal i stället för ett algebraiskt uttryck, som du förväntade dig.

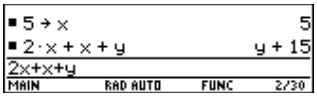
Hur odefinierade och definierade variabler behandlas

När du matar in ett uttryck som innehåller en variabel behandlas variabeln på ett av två sätt.

- Om variabeln är odefinierad behandlas den som en algebraisk symbol.



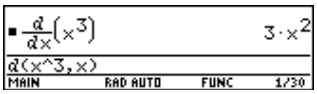
- Om variabeln är definierad (även om den är definierad som 0), substituerar värdet variabeln.



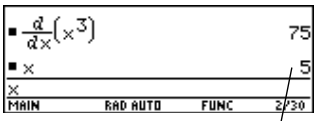
__Tips!__ När du definierar en variabel bör du använda mer än ett tecken i namnet. Låt namn med ett tecken vara odefinierade för algebraiska beräkningar.

För att förstå varför detta är viktigt visas ett exempel där vi antar att du vill hitta första derivatan av x^3 med avseende på x.

- Om x är odefinierad visas resultatet antagligen på det sätt du förväntade dig.

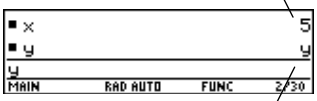
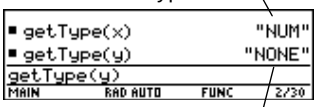


- Om x är definierad kan resultatet visas på ett sätt som du inte förväntade dig.



Resultatet 75 kan vara missvisande om du inte visste att 5 sparats i variabeln x förut.

Avgöra om en variabel är odefinierad

Metod:	Exempel:
Skriv variabelnamnet.	
Använd funktionen **getType**.	

Om den är definierad visas variabelns värde.

Om den är odefinierad visas variabelns namn.

Om den är definierad visas variabelns typ.

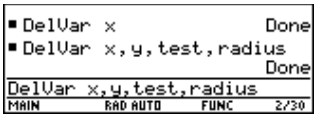
Om den är odefinierad visas "NONE".

__Obs!__ Använd [2nd] [VAR-LINK] för att visa en lista med definierade variabler, enligt beskrivningen i kapitel 21.

Ta bort en definierad variabel

Du kan "odefiniera" en definierad variabel genom att ta bort den.

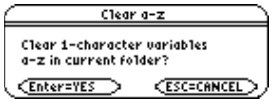
Om du vill ta bort:	Gör du följande:
En eller flera angivna variabler	Använd funktionen **DelVar**.



Du kan också ta bort variabler genom att använda fönstret VAR-LINK (2nd [VAR-LINK]), enligt beskrivning i kapitel 21.

Obs! *Information om mappar finns i kapitel 5.*

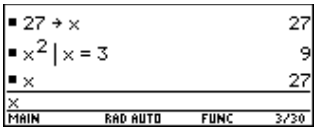
Alla variabler med en bokstav som namn (a – z) i den aktuella mappen	På menyn Clean Up i grundfönstret, välj 1:Clear a-z. Du blir uppmanad att trycka på ENTER för att bekräfta raderingen.



Tillfälligt åsidosätta en variabel

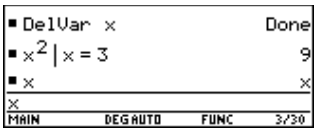
Genom att använda [|] för att skriva operatorn "with" (|) kan du göra följande:

- Tillfälligt åsidosätta en variabels definierade värde.



Obs! *Mer information om operatorn* **|** *finns på sidan 67.*

- Tillfälligt definiera ett värde för en odefinierad variabel.



När du vill mata in operatorn “with” (|), tryck på:

TI-89: [|]

TI-92 Plus: 2nd [|]

Lägesinställningarna Exact/Approx, som beskrivs kortfattat i kapitel 2, påverkar direkt den precision och noggrannhet med vilken räknaren beräknar ett resultat. I det här avsnittet beskrivs dessa inställningar, eftersom de är väsentliga vid algebraiska operationer.

EXACT

När Exact/Approx = EXACT, används exakt rationell aritmetik med upp till 614 siffror i täljaren och 614 siffror i nämnaren. Med inställningen EXACT gör du följande:

- Omvandlar irrationella tal till standardformat så långt det är möjligt utan att avrunda dem. Exempelvis omvandlas $\sqrt{12}$ till $2\sqrt{3}$ och ln(1000) omvandlas till 3 ln(10).
- Konverterar decimaltal till rationella tal. Exempelvis omvandlas 0.25 till 1/4.

Funktionerna **solve**, **cSolve**, **zeros**, **cZeros**, **factor**, ∫, **fMin** och **fMax** använder bara exakta symboliska algoritmer. Dessa funktioner beräknar inte uppskattade resultat i inställningen EXACT.

- Vissa ekvationer, exempelvis $2^{-x} = x$, har lösningar som inte kan beskrivas av funktioner och operatorer på TI-89 / TI-92 Plus.
- Med denna typ av ekvation räknar inte EXACT ut avrundade lösningar. Exempelvis har $2^{-x} = x$ den avrundade lösningen $x \approx 0.641186$, men den visas inte med inställningen EXACT.

Fördelar	Nackdelar
Resultaten är exakta.	När du använder mer komplicerade rationella tal och irrationella konstanter kan beräkningarna: • Använda mer minne, vilket kan göra att minnet tar slut innan lösningen är klar. • Ta längre tid. • Skapa skrymmande resultat som är svårare att förstå än ett decimaltal.

APPROXIMATE

När Exact/Approx = APPROXIMATE omvandlas rationella tal och irrationella konstanter till decimaltal. Följande undantag finns dock:

- Vissa inbyggda funktioner som förväntar sig att ett av dess argument ska vara heltal kommer att omvandla detta tal till ett heltal, om det är möjligt. Exempelvis omvandlas ***d***(y(x), x, 2.0) till ***d***(y(x), x, 2).
- Hela decimaltalsexponenter omvandlas till heltal. Exempelvis omvandlas $x^{2.0}$ till x^2, även i inställningen APPROXIMATE.

Funktioner som exempelvis **solve** och ∫ (integrate) kan använda både exakt symboliska och ungefärliga numeriska tekniker. Dessa funktioner hoppar över alla eller en del av sina exakta symboltekniker med inställningen APPROXIMATE.

Fördelar	Nackdelar
Om du inte behöver exakta resultat kan detta spara tid och/eller använda mindre minne än inställningen EXACT.	Resultat med odefinierade variabler eller funktioner visas ofta som ofullständiga lösningar. En koefficient som exempelvis borde bli 0 kanske i stället visas som ett litet tal, t ex 1.23457E-11.
Ungefärliga resultat är ibland mer kompakta och förståeliga än exakta resultat.	Vid exakta beräkningar, som exempelvis beräkning av gränsvärden och vid integration, har man mindre chans att få tillfredsställande resultat med inställningen APPROXIMATE.
Om du inte planerar att använda symboliska beräkningar liknar ungefärliga resultat de resultat man får med vanliga miniräknare.	Ungefärliga resultat är ibland mindre kompakta och förståeliga än exakta resultat. Du kanske exempelvis föredrar att visa 1/7 i stället för .142857.

AUTO

När Exact/Approx = AUTO används exakt rationell aritmetik överallt där operanderna är rationella tal. Annars används decimaltals-aritmetik efter att rationella operander har omvandlats till decimaltal. Decimaltal är med andra ord "bestämmande". Exempel:

1/2 – 1/3 görs om till 1/6
men
0.5 – 1/3 görs om till .16666666666667

Detta decimaltalsövertagande överlappar inte barriärer som exempelvis odefinierade variabler eller mellan element i listor eller matriser. Exempel:

(1/2 - 1/3) x + (0.5 – 1/3) y görs om till x/6 + .16666666666667 y
och
{1/2 - 1/3, 0.5 – 1/3} görs om till {1/6, .16666666666667}

I inställningen AUTO räknar funktioner, som exempelvis **solve,** ut så många lösningar som möjligt exakt och använder sedan, om det behövs, de ungefärliga numeriska metoderna för att räkna ut ytterligare lösningar. På liknande sätt använder ∫ (integral) ungefärliga numeriska metoder när exakta symboliska metoder inte fungerar.

Fördelar	Nackdelar
Exakta resultat visas när det är praktiskt och ungefärliga numeriska resultat visas när exakta resultat är opraktiska. Du kan ofta styra formatet för ett resultat genom att ange några koefficienter som antingen rationella tal eller som decimaltal.	Om du bara är intresserad av exakta resultat kan det vara bortkastad tid att söka efter ungefärliga resultat. Om du bara är intresserad av ungefärliga resultat kan det vara bortkastad tid att söka efter exakta resultat. Dessutom kanske minnet tar slut när du söker efter dessa exakta resultat.

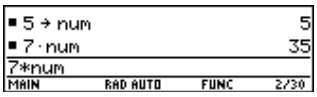
När du matar in ett uttryck på inmatningsraden och trycker på [ENTER], förenklas automatiskt uttrycket enligt reglerna för uttryckets standardförenkling.

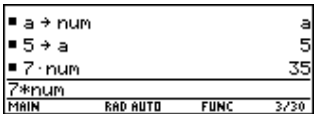
Regler för standard-förenkling

Alla av de följande reglerna tillämpas automatiskt. Du ser inte mellanresultaten.

- Om en variabel har ett definierat värde substituerar det värdet variabeln.

  Om variabeln definieras av en annan variabel, substitueras variabeln med denna variabels värde.



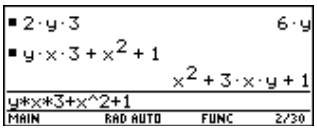


Obs! *Information om mappar finns i kapitel 5.*

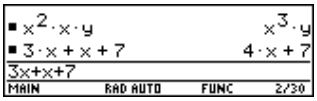
Standardförenkling ändrar inte variabler som använder sökvägsnamn för att ange en mapp. Exempelvis förenklas inte x+class\x till 2x.

Obs! *Se "Fördröjd förenkling för vissa inbyggda funktioner" på sidan 66.*

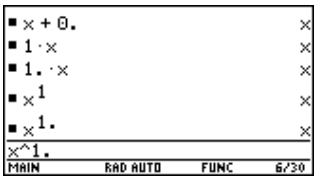
- För funktioner:
  - Argumenten förenklas. (Vissa inbyggda funktioner fördröjer förenkling av vissa av dess argument.)
  - Om funktionen är inbyggd eller användardefinierad tillämpas funktionsdefinitionen på de förenklade argumenten. Därefter substitueras funktionsformen med detta resultat.
- Numeriska uttryck kombineras.
- Produkter och summor sorteras i ordning.



  Produkter och summor som involverar odefinierade variabler sorteras efter den första bokstaven i variabelnamnet.
  - Odefinierade variabler från r till och med z antas vara verkliga variabler och placeras i alfabetisk ordning i början av summan.
  - Odefinierade variabler från a till och med q antas motsvara konstanter och placeras i alfabetisk ordning i slutet av summan (men före tal).
- Liknande faktorer och liknande termer samlas ihop.

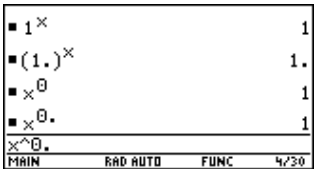


- Identiteter med nollor och ettor förkortas.

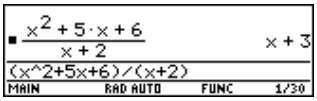


Detta decimaltal gör att resultatet visas som ett decimaltal.

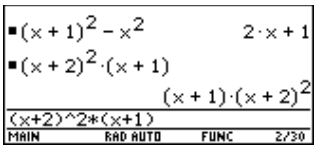
Om ett helt decimaltal anges som exponent, används det som ett heltal (och resultatet blir sålunda inte ett decimaltal).



- Polynomets största gemensamma faktor bryts ut och uttrycket förkortas.

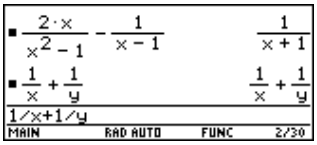


- Polynomet utvecklas så att det inte kan förkortas mer.



Ingen förkortning

- Gemensamma nämnare används om de inte kan förkortas.



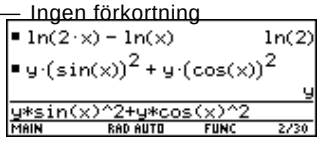
Ingen förkortning

- Funktionsidentiteter förkortas. Exempel:

  ln(2x) = ln(2) + ln(x)

  och

  $\sin(x)^2 + \cos(x)^2 = 1$



Hur lång tid tar förenklingsprocessen?

Beroende på komplexiteten hos en inmatning, ett resultat eller ett mellanled, kan det ta lång tid att utveckla ett uttryck och ta bort den gemensamma nämnaren som behövs för förenkling.

Du kan avbryta en förenklingsprocess som tar för lång tid genom att trycka på [ON]. Du kan därefter pröva med att förenkla endast en del av uttrycket. (Klistra in hela uttrycket automatiskt på inmatningsraden och ta sedan bort de delar som inte ska vara med.)

Fördröjd förenkling för vissa inbyggda funktioner

Vanligtvis förenklas variabler automatiskt till sin lägsta möjliga nivå innan de används i en funktion. För vissa funktioner fördröjs dock fullständig förenkling tills funktionen har utförts.

Funktioner som använder fördröjd förenkling

Funktioner som använder fördröjd förenkling har ett *var*-argument som är obligatoriskt och som utför funktionen med avseende på en variabel. Dessa funktioner har minst två argument med den allmänna formen:

funktion(*uttryck*, *var* [, ...])

Obs! *Alla funktioner som använder ett var-argument använder inte fördröjd förenkling.*

Exempel:

solve(x^2− x− 2=0,x)
d(x^2 − x− 2,x)
∫(x^2− x− 2,x)
limit(x^2 − x− 2,x,5)

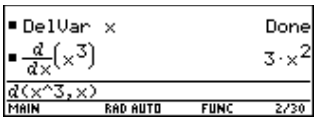
För en funktion som använder fördröjd förenkling:

1. Variabeln *var* förenklas till den lägsta nivå där variabeln fortfarande är en variabel (även om den fortfarande kan förenklas till ett icke-variabelvärde, t ex 5).
2. Funktionen utförs med hjälp av variabeln.
3. Om *var* kan förenklas ytterligare ersätts i så fall det värdet i resultatet.

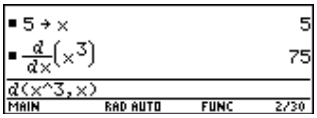
Obs! *Du kan definiera ett numeriskt värde för* var *om du vill, beroende på situationen.*

Exempel:

x kan inte förenklas.

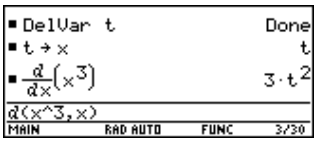


x är inte förenklad. Funktionen använder x^3 och substituerar x med 5.

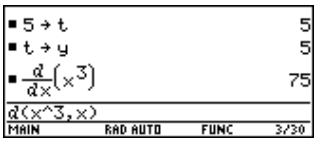


Obs! *I exemplet till höger beräknas derivatan av x^3 då x=5. Om x^3 ursprungligen förenklades till 75 skulle du få derivatan av 75, vilket inte är det du vill ha.*

x förenklas till t. Funktionen använder t^3.



x förenklas till t. Funktionen använder t^3 och substituerar t med 5.



Substituera värden och ange restriktioner

Med operatorn "with" (**|**) kan du tillfälligt substituera värden i ett uttryck eller ange restriktioner.

Skriva med operatorn "with"

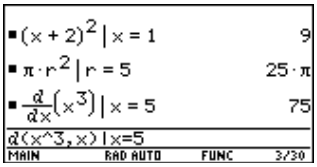
När du vill mata in operatorn “with” (**|**), tryck på:

TI-89: [|]

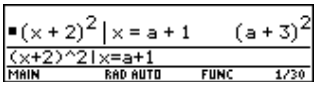
TI-92 Plus: [2nd] [|]

Substituera en variabel

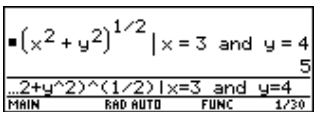
För varje förekomst av en angiven variabel kan du substituera den med ett numeriskt värde eller uttryck.



Förstaderivatan av x^3 med x = 5

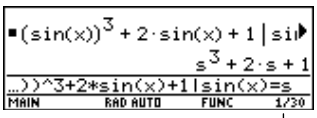


Om du vill substituera flera variabler på samma gång använder du den booleska operatorn **and**.



Substituera ett enkelt uttryck

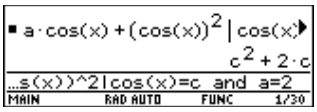
För varje förekomst av ett enkelt uttryck kan du substituera det med en variabel, ett numeriskt värde eller ett annat uttryck.



Substitution av sin(x) med s visar att uttrycket är ett polynom med avseende på sin(x).

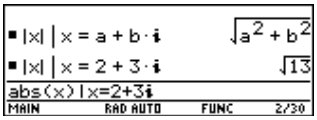
Obs! acos(x) *är inte samma som* a*cos(x).

Genom att substituera en ofta använd (eller lång) term, kan du visa resultat i en mer kompakt form.



Substituera komplexa värden

Du kan substituera komplexa värden på samma sätt som med reella värden.



Obs! *Översikt över komplexa tal finns i appendix B.*

Alla odefinierade variabler behandlas som verkliga tal i symboliska beräkningar. För att kunna utföra komplexa symboliska analyser måste du definiera en komplex variabel. Exempel:

Tips! *För att få den imaginära enheten* ***i****, trycker du på* [2nd] [i]. *Skriv inte bara* ***i*** *på tangentbordet.*

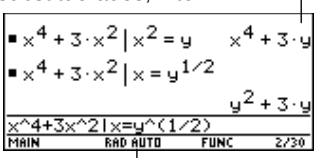
x+y***i***→z

Du kan därefter använda z som en komplex variabel. Du kan även använda z_. Mer information finns i avsnittet om _ (understrykning) i appendix A.

Gränser för substitueringar

- Substitutioner uppstår bara om det finns en *exakt* matchning för ersättningen.

Endast x^2 substituerades, inte x^4.



Ange substitutionen i enklare termer för att få en fullständig substitution.

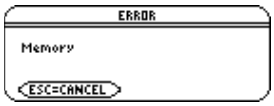
- Oändlig rekursion kan endast inträffa när du definierar en substitueringsvariabel uttryckt i sig själv.

Substituerar sin(x+1), sin(x+1+1), sin(x+1+1+1) osv.

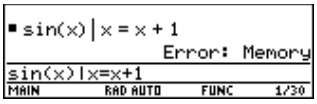
```
sin(x)|x=x+1
```

När du anger en substitution som orsakar en oändlig rekursion:

– Visas ett felmeddelande.



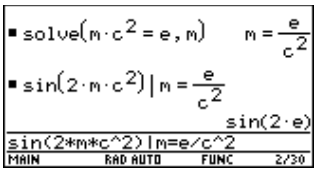
– När du trycker på [ESC], visas ett fel i historiklistan.



- Internt sorteras ett uttryck enligt reglerna för den automatiska förenklingen. Av den anledningen kanske inte produkter och summor matchar den ordning som du skrev dem i.

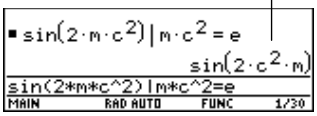
***Tips!** Använd funktionen **solve** för att få hjälp med att avgöra hur substitutionen ska se ut.*

– Som en allmän regel bör du substituera mot en enstaka variabel.



– Att substituera mot mer allmänna uttryck (antingen $m \cdot c^2 = e$ eller $c^2 \cdot m = e$) kanske inte fungerar så som du hade tänkt dig.

Ingen matchning för substitution



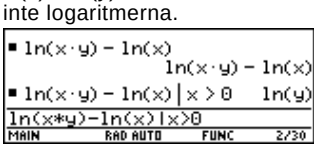
Ange restriktioner

Många identiteter och transformationer är endast giltiga inom ett begränsat definitionsområde. Exempel:

$\ln(x*y) = \ln(x) + \ln(y)$	bara om x och/eller y inte är negativa
$\sin^{-1}(\sin(\theta)) = \theta$	bara om $\theta \geq -\pi/2$ och $\theta \leq \pi/2$ radianer

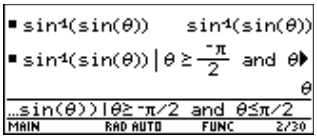
Använd operatorn "with" för att ange restriktioner.

Eftersom $\ln(x*y) = \ln(x) + \ln(y)$ inte alltid är giltiga, kombineras inte logaritmerna.



Med en restriktion är identiteten giltig och uttrycket förenklas.

Eftersom $\sin^{-1}(\sin(\theta)) = \theta$ inte alltid är giltig, förenklas inte uttrycket.



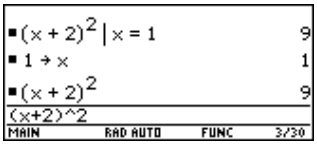
Med en restriktion kan uttrycket förenklas.

Tips! *Mata in* ln(x∗ y) *i stället för* ln(xy)*; annars tolkas xy som en enstaka variabel med namnet xy.*

Tips! *För ≥ eller ≤, press* [◆] [>] *eller* [◆] [<] *på tangentbordet. Du kan också använda* [2nd] [MATH] 8 *eller* [2nd] [CHAR] 2 *för att välja dem från en meny.*

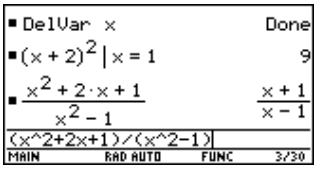
Använda substitutioner eller definiera en variabel

Ofta kan du nå samma resultat som vid substitution genom att definiera variabeln.

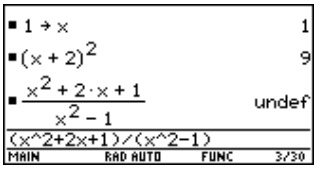


Substitution är dock att föredra i de flesta fall eftersom variabeln endast definieras för den aktuella beräkningen och inte påverkar efterföljande beräkningar.

Substitution av x=1 påverkar inte nästa beräkning.



Sparar du 1→x påverkar det efterföljande beräkningar.



Varning! *När x är definierad kan det påverka alla beräkningar som involverar x (tills du tar bort x).*

Översikt över menyn Algebra

Du kan använda menyn [F2] **Algebra** för att välja de senast använda algebraiska funktionerna.

Menyn Algebra

__Obs!__ I appendix A finns en fullständig beskrivning av varje funktion och dess syntax.

Från grundfönstret trycker du på [F2] för att visa:

Denna meny är även tillgänglig från menyn MATH. Tryck på [2nd] [MATH] och välj sedan 9:Algebra.

Menyalternativ	Beskrivning
solve	Löser ett uttryck med avseende på en angiven variabel. Detta returnerar endast verkliga lösningar, oavsett lägesinställningen Complex Format. Visar svar där lösningarna binds samman med “and” och “or”. (För komplexa lösningar väljer du A:Complex på menyn Algebra.)
factor	Faktoruppdelar ett uttryck med avseende på alla dess variabler eller med avseende på endast en variabel.
expand	Utvecklar ett uttryck med avseende på alla dess variabler eller med avseende på en variabel.
zeros	Anger värdet för en variabel som gör ett uttryck lika med noll. Visas i en lista.
approx	Beräknar ett uttryck med hjälp av decimaltals-aritmetik, så långt det är möjligt. Detta är detsamma som att använda [MODE] för att ange Exact/Approx = APPROXIMATE (eller använda [◆] [ENTER] för att beräkna ett uttryck).
comDenom	Beräknar en gemensam nämnare för alla termer i ett uttryck, förenklar uttrycket och anger det i bråkform.
propFrac	Förenklar ett uttryck som ett egentligt bråk.
nSolve	Beräknar en enstaka lösning för en ekvation som ett decimaltal (till skillnad från **solve**, som kan visa flera lösningar i rationellt eller symbolisk format).

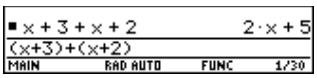
Menyalternativ	Beskrivning
Trig	Visar undermenyn: 1:tExpand(2:tCollect(tExpand — Expanderar trigonometriska uttryck med vinkelsummor och multipla vinklar. tCollect — Samlar produkten av heltalsexponenter av trigonometriska funktioner i vinkelsummor och multipla vinklar. **tCollect** är inversen till **tExpand**.
Complex	Visar undermenyn: 1:cSolve(2:cFactor(3:cZeros(Dessa funktioner är samma som **solve**, **factor** och **zeros**, men de beräknar också komplexa resultat.
Extract	Visar undermenyn: 1:getNum(2:getDenom(3:left(4:right(getNum — Tillämpar **comDenom** och returnerar sedan den resulterande täljaren. getDenom — Tillämpar **comDenom** och returnerar sedan den resulterande nämnaren. left — Returnerar den vänstra sidan av en ekvation eller olikhet. right — Returnerar den högra sidan av en ekvation eller olikhet.

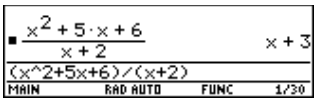
Obs! *Funktionerna* **left** *och* **right** *används också för att returnera en angivet antal element eller tecken från den vänstra eller högra sidan i en lista eller teckensträng.*

I det här avsnittet finns exempel som visar några av de funktioner som finns på verktygsmenyn F2 **Algebra**. Utförlig information om olika funktioner finns i appendix A. Vissa algebraiska operationer kräver ingen specialfunktion.

Addera och dividera polynom

Du kan addera eller dividera polynom direkt, utan att använda en specialfunktion.





Faktoruppdela och utveckla polynom

Använd funktionerna **factor** (F2 2) och **expand** (F2 3).

factor(*uttryck* [,*var*])

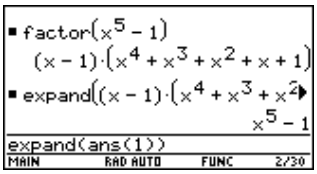
└ faktoruppdelning med avseende på en variabel

expand(*uttryck* [,*var*])

└ För partiell utveckling med avseende på en variabel

Faktoruppdela $x^5 - 1$. Utveckla sedan resultatet.

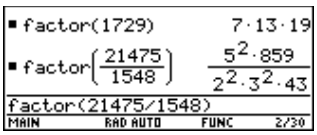
Lägg märke till att **factor** och **expand** är inversa operationer.



Söka efter primtalsfaktorer av ett tal

Med funktionen **factor** (F2 2) kan du göra mer än bara faktoruppdela ett algebraiskt polynom.

Du kan t ex söka efter primtalsfaktorer i ett rationellt tal (antingen ett heltal eller bråk med heltal).

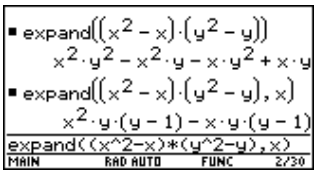


Söka efter en partiell utveckling

Med den valfria *var*-variabeln som hör till funktionen **expand** (F2 3), kan du göra en partiell utveckling som samlar variabler med samma exponent.

Utveckla $(x^2 - x)(y^2 - y)$ fullständigt med avseende på alla variabler.

Gör sedan en partiell utveckling med avseende på x.

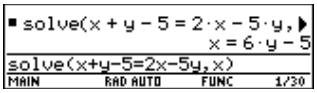


Lösa en ekvation

Använd funktionen **solve** ([F2] 1) för att lösa en ekvation med avseende på en angiven variabel.

solve(*ekvation*, *var*)

Lös x + y – 5 = 2x – 5y för x.



Lägg märke till att **solve** endast visar det slutliga resultatet.

Om du vill se mellanleden kan du manuellt lösa ekvationen steg för steg.

Obs! *En åtgärd som* [−] 2 [×] *subtraherar 2x från båda sidor.*

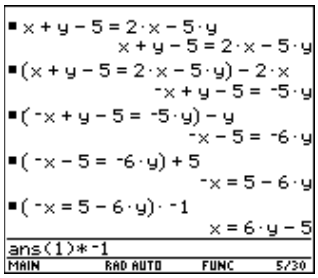
x [+] y [−] 5 [=] 2x [−] 5y

[−] 2 x

[−] y

[+] 5

[×] [(-)] 1



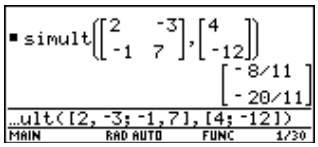
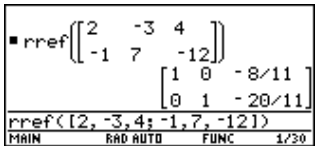
Lösa ett linjärt ekvationssystem

Tänk dig två uppsättningar ekvationer med två obekanta

2x – 3y = 4
⁻x + 7y = ⁻12

Lös detta ekvationssystem med någon av följande metoder.

Obs! *Matrisfunktionerna* **simult** *och* **rref** *finns inte på menyn* [F2] Algebra. *Använd* [2nd] [MATH] 4 *eller* Catalog.

Metod	Exempel
Använd funktionen **solve** för en en-stegs-lösning.	**solve(**2x−3y=4 and ⁻x+7y=⁻12,{x,y}**)**
Använd funktionen **solve** med ersättningen (**|**) för en stegvis beräkning.	Se förhandsvisningen i början av detta kapitel där lösningen för x = ⁻8/11 och y = ⁻20/11 visades.
Använd funktionen **simult** med en matris.	Mata in koefficienterna i en matris och resultatet som en konstant kolumnmatris. 
Använd funktionen **rref** med en matris.	Mata in koefficienterna i en utvidgad matris. 

Söka efter nollställen i ett uttryck

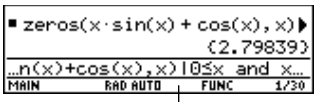
Använd funktionen **zeros** ([F2] 4).

zeros(*uttryck*, *var*)

__Tips!__ För ≥ eller ≤ tryck [◆] [>] eller [◆] [<]. Du kan också använda [2nd] [MATH] 8 eller [2nd] [CHAR] 2 för att välja dem från en meny.

Använd uttrycket x ∗ sin(x) + cos(x).

Sök efter nollställen med avseende på x i intervallet 0 ≤ x ≤ 3.



Använd operatorn "with" ([I]) för att ange intervallet.

Söka efter egentliga bråk och gemensamma nämnare

Använd funktionerna **propFrac** ([F2] 7) och **comDenom** ([F2] 6).

propFrac(*rationellt uttryck* [,*var*])

för egentliga bråk med avseende på en variabel

comDenom(*uttryck* [,*var*])

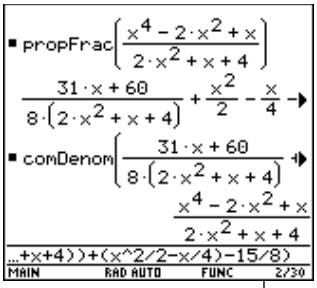
för gemensamma nämnare som samlar lika exponenter av en variabel

*__Obs!__ Du kan använda **comDenom** för ett uttryck, en lista eller en matris.*

Sök ett egentligt bråk för uttrycket $(x^4 - 2x^2 + x) / (2x^2 + x + 4)$.

Omvandla sedan resultatet till ett bråk med fullständigt utvecklad täljare och en fullständigt utvecklad nämnare.

Lägg märke till att **propFrac** och **comDenom** är inversa operationer.



Om du utför denna operation på räknaren, rullar funktionen **propFrac** ut från fönstret.

I detta exempel:

- $\frac{31x + 60}{8}$ är resten av $x^4 - 2x^2 + x$ dividerat med $2x^2 + x + 4$.
- $\frac{x^2}{2} - \frac{x}{4} - 15/8$ är kvoten.

Översikt över menyn Calc

Du kan använda menyn F3 **Calc** för att välja beräknings-funktioner som du ofta använder.

Menyn Calc

__Obs!__ En fullständig beskrivning av varje funktion och dess syntax finns i appendix A.

__Obs!__ Symbolen d för derivatan är en special-symbol. Det är inte samma symbol du får genom att skriva D *på tangentbordet. Använd* F3 1 *eller* 2nd [d].

Från grundfönstret trycker du på F3 för att visa:

Denna meny är även tillgänglig på menyn MATH. Tryck på 2nd [MATH] och välj sedan A:Calculus.

Menyalternativ	Beskrivning
d differentiate	Deriverar ett uttryck med avseende på en variabel.
∫ integrate	Integrerar ett uttryck med avseende på en variabel.
limit	Beräknar gränsvärdet av ett uttryck med avseende på en variabel.
Σ sum	Beräknar ett uttryck med diskreta variabelvärden inom ett intervall och beräknar sedan summan.
Π product	Beräknar ett uttryck med diskreta variabelvärden inom ett intervall och beräknar sedan produkten.
fMin	Söker efter värden på en angiven variabel där ett uttryck har minimum.
fMax	Söker efter värden på en angiven variabel där ett uttryck har maximum.
arcLen	Beräknar båglängden av ett uttryck med avseende på en variabel.
taylor	Beräknar Taylors polynomapproximation till ett uttryck med avseende på en variabel.
nDeriv	Beräknar den numeriska derivatan av ett uttryck med avseende på en variabel.
nInt	Beräknar en integral som ett decimaltal med Gauss kvadraturformel (en approximering med vägt medelvärde av integrandernas värden).
deSolve	Symboliska beräkningar som löser många första och andra ordningen differentialekvationer, med eller utan begynnelsevillkor.

I det här avsnittet finns exempel som beskriver en del av de funktioner som finns på menyn F3 **Calc**. I appendix A finns utförlig information om alla funktioner.

Integration och derivering

Använd funktionerna ∫ integrate (F3 2) och ***d*** differentiate (F3 1).

∫ (*uttryck*, *var* [,*undre*] [,*övre*])

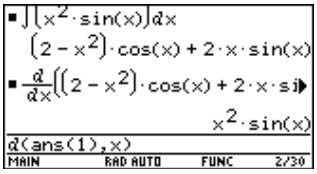
låter dig ange gränsvärden eller en integrationskonstant

d (*uttryck*, *var* [,*ordning*])

Obs! *Du kan endast integrera ett uttryck, du kan derivera ett uttryck, lista eller matris.*

Integrera $x^2 * \sin(x)$ med avseende på x.

Derivera resultatet med avseende på x.



För att få *d*, använd F3 1 eller 2nd [*d*]. Skriv inte D med tangentbordet.

Söka ett gränsvärde

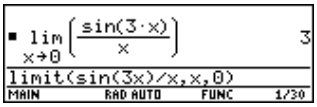
Använd funktionen **limit** (F3 3).

limit(*uttryck*, *var*, *punkt* [,*riktning*])*

negativt = från vänster
positivt = från höger
uteslutet eller 0 = båda

Obs! *Du kan söka efter ett gränsvärde för ett uttryck, en lista eller matris.*

Söka efter ett gränsvärde för sin(3x) / x då x går mot 0.



Söka efter ett Taylor-polynom

Viktigt: *Skalning i vinkelinställningen Degree med π/180 kan medföra att beräkningar visas i en annan form.*

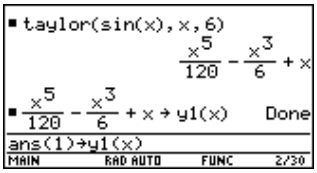
Använd funktionen **taylor** (F3 9).

taylor(*uttryck*, *var*, *ordning* [,*punkt*])

Om denna utesluts, är expansionspunkten 0

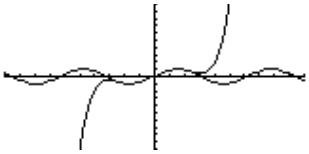
Sök efter ett 6:e ordningens Taylor-polynom för sin(x) med avseende på x.

Spara resultatet som en användardefinierad funktion med namnet y1(x).



Plotta sedan sin(x) och Taylor-polynomet.

```
Graph sin(x):Graph y1(x)
```



Du kan använda en användardefinierad funktion som ett argument för de inbyggda algebraiska funktionerna och beräkningsfunktionerna.

Information om hur du skapar en användar-definierad funktion

Se följande avsnitt:

- "Skapa och beräkna användardefinierade funktioner" i kapitel 5.
- "Plotta en funktion som är definierad i grundfönstret " och "Plotta en funktion som består av flera delar " i kapitel 12.
- "Översikt över att mata in en funktion" i kapitel 17.

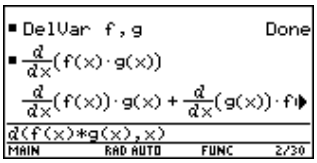
Odefinierade funktioner

Du kan använda funktioner som exempelvis f(x), g(t), r(θ) osv, som inte har tilldelats en definition. Dessa "odefinierade" funktioner ger symboliska resultat. Exempel:

Tips! *Välj* ***d*** *från menyn* Calc *genom att trycka på* F3 1 *(eller tryck* 2nd [d] *på tangentbordet).*

Använd **DelVar** för att försäkra dig om att f(x) och g(x) inte är definierade.

Sök sedan derivatan av f(x)∗ g(x) med avseende på x.



Funktioner bestående av ett uttryck

Du kan använda användardefinierade funktioner som består av ett enda uttryck. Exempel:

- Använd STO▸ för att skapa en användardefinierad secansfunktion där:

  $$\sec(x) = \frac{1}{\cos(x)}$$

  Sök sedan gränsvärdet för sec(x) då x går mot π/4.

Tips! *Du väljer* ***limit*** *på menyn* Calc *genom att trycka på* F3 3.

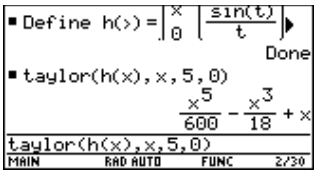
- Använd **Define** för att skapa en användardefinierad funktion h(x) där:

  $$h(x)=\int_0^x \sin(t) / t$$

  Sök sedan den 5:e ordningens Taylor-polynom för h(x) med avseende på x.

Tips! *Välj ∫ från menyn* Calc *genom att trycka på* F3 2 *(eller tryck* 2nd [∫] *på tangentbordet). Om du vill välja* ***taylor****, tryck* F3 9.

Definiera h(x)= ∫(sin(t)/t,t,0,x).

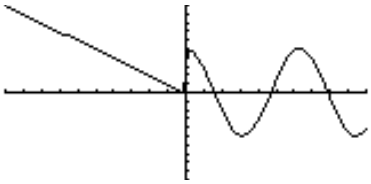


Funktioner bestående av flera respektive ett uttryck

Användardefinierade funktioner som består av flera uttryck bör endast användas som ett argument för numeriska funktioner (exempelvis **nDeriv** och **nInt**).

I vissa fall kan du dock skapa en ekvivalent funktion som består av ett uttryck. Tänk dig en delad funktion med två delar.

När:	**Använder du uttrycket:**
$x < 0$	^-x
$x \geq 0$	5 cos(x)

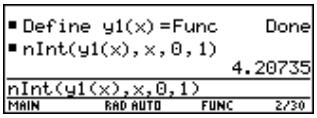


__Tips!__ Du kan använda din dator när du skriver in långa texter och sedan använda TI-GRAPH LINK för att skicka det till din TI-89 / TI 92-Plus. Se kapitel 18 för mer information.

- Skapa en användardefinierad funktion som består av flera uttryck som ser ut så här:

```
Func
  If x<0 Then
  Return ⁻x
Else
  Return 5cos(x)
EndIf
EndFunc
```

Definiera y1(x)=Func:If x<0 Then: ... :EndFunc



__Tips!__ Du väljer **nInt** *på menyn* Calc *genom att trycka på* F3 B:nInt.

Integrera sedan numeriskt y1(x) med avseende på x.

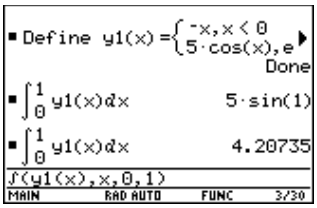
- Skapa en ekvivalent användardefinierad funktion som består av ett uttryck.

Använd den inbyggda funktionen **when**.

Integrera sedan y1(x) med avseende på x.

__Tips!__ Du väljer ∫ på menyn Calc *genom att trycka på* F3 2 *(eller tryck* 2nd [∫] *på tangentbordet).*

Definiera y1(x)=when(x<0, ⁻x, 5cos(x))



Tryck på ◆ ENTER för ett decimaltalsresultat.

Felmeddelande om att minnet håller på att ta slut

Mellanled sparas i minnet och tas sedan bort när uträkningen är klar. Beroende på hur komplex beräkningen är kan minnet ta slut innan resultatet kan beräknas.

Frigöra minne

- Ta bort variabler som inte behövs, särskilt de som är stora.
  - Använd [2nd] [VAR-LINK] enligt beskrivningen i kapitel 21 för att visa/ta bort variabler och/eller Flash-program.
- I grundfönstret:
  - Rensa historiklistan ([F1] 8) eller ta bort historikpar som inte behövs.
  - Du kan också använda [F1] 9 för att minska antalet historikpar som sparas.
- Använd [MODE] för att ange Exact/Approx = APPROXIMATE. (När det gäller resultat med fler siffror används på detta sätt mindre minne än med AUTO eller EXACT. När det gäller resultat som bara har några få siffror används på detta sätt mer minne.)

Förenkla problem

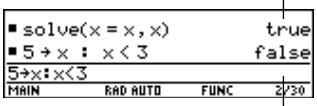
- Dela upp problemet i mindre delar.
  - Dela upp **solve**(a∗ b=0,*var*) i **solve**(a=0,*var*) och **solve**(b=0,*var*). Lös varje del och kombinera resultaten.
- Om flera odefinierade variabler uppstår endast i en viss kombination kan du substituera den kombinationen med en enskild variabel.
  - Om m och c endast uppstår som $m*c^2$, substituerar du $m*c^2$ med e.
  - I uttrycket $\frac{(a+b)^2+\sqrt{(a+b)^2}}{1-(a+b)^2}$ substituerar du (a+b) med c och använder $\frac{c^2+\sqrt{c^2}}{1-c^2}$. I lösningen substituerar du c med (a+b).
- För uttryck som kombinerats över en gemensam nämnare, substituerar du summor i nämnaren med unika, nya, odefinierade variabler.
  - I uttrycket $\frac{x}{\sqrt{a^2+b^2}+c}+\frac{y}{\sqrt{a^2+b^2}+c}$ substituerar du d med $\sqrt{a^2+b^2}+c$ och använder $\frac{x}{d}+\frac{y}{d}$. I lösningen substituerar du d med $\sqrt{a^2+b^2}+c$.
- Substituera kända numeriska värden med odefinierade variabler i ett tidigare skede, särskilt om de är enkla heltal eller tal i bråkform.
- Formulera om ett problem för att undvika exponenter i bråkform.
- Utelämna relativt små termer för att finna en approximation.

Resultatet av en beräkning kan inkludera en av de specialkonstanter som beskrivs i detta avsnitt. I vissa fall måste du också ange en konstant som en del av inmatningen.

true, false

Dessa anger resultatet av en identitet eller ett booleskt uttryck.

x=x är sant för alla värden på x.



5<3 är falskt.

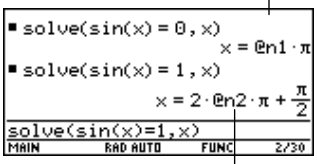
@n1 ... @n255

För @, tryck på:
TI-89: [◆] [STO▸]
TI-92 Plus: [2nd] R

Denna notation anger ett godtyckligt heltal.

När ett godtyckligt heltal inträffar flera gånger i samma session numreras varje förekomst i ordningsföljd. Välj [2nd] [F6] 2:NewProb för att återställa till @n1.

En lösning finns för varje heltalsmultipel av π.



Både @n1 och @n2 representerar ett godtyckligt heltal men notationen identifierar oberoende godtyckliga heltal.

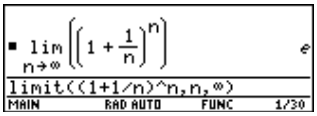
∞, *e*

För ∞, tryck på:
TI-89: [◆] [∞]
TI-92 Plus: [2nd] [∞]

För e tryck på:
TI-89: [◆] [e^x]
TI-92 Plus: [2nd] [e^x]

∞ motsvarar oändligheten och *e* motsvarar konstanten 2.71828... (basen för den naturliga logaritmen).

Dessa konstanter används ofta i både inmatningar och resultat.



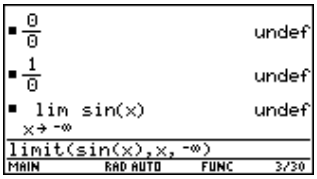
undef

Detta anger att resultatet är odefinierat.

Matematiskt odefinierat

±∞ (obestämt tecken)

Inte ett unikt gränsvärde



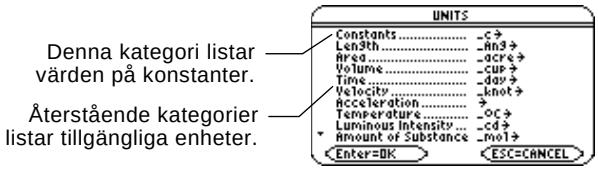
Konstanter och Måttenheter

4

Konstanter och måttenheter 82
Ange konstanter och måttenheter 83
Konvertera mellan måttenheter 85
Ange förvalda måttenheter för visningsläge 87
Skapa egna måttenheter 88
Fördefinierade konstanter och måttenheter 89

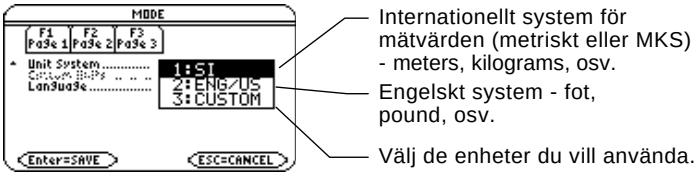
Med dialogrutan UNITS kan du välja konstanter och enheter från olika kategorier av måttenheter.

***Obs!** Namn på konstanter och enheter börjar alltid med ett understrykningstecken _.*



På s 3 (F3) i dialogrutan MODE, kan du vid visning av värden välja mellan tre olika system för måttenheter.

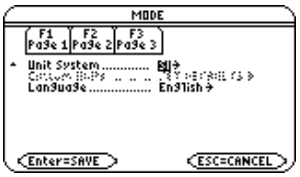
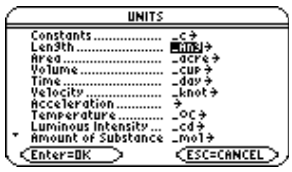
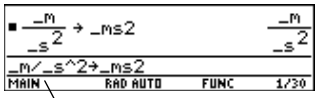
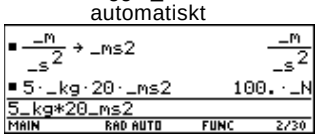
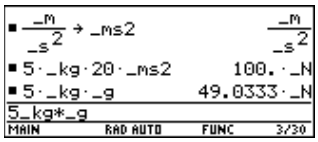
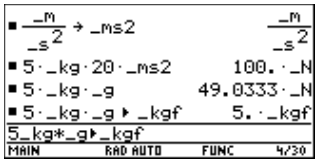
***Obs!** Du kan också använda* **getUnits()** *för en lista över alla förvalda enheter eller* **setUnits()** *om du vill ställa in förvalda enheter. Se appendix A.*



Med funktionerna för enheter kan du:

- Ange en enhet för värden i ett uttryck som 6_m ∗ 4_m eller 23_m/_s ∗ 10_s. Resultatet visas med förvalda enheter.
- Konvertera värden från en måttenhet till en annan inom samma kategori.
- Skapa dina egna enheter. Detta kan vara en kombination av existerande enheter eller unika, egen-definierade, enheter.

Med ekvationen f = m∗ a beräknar du kraften när m = 5 kilogram och a = 20 meter/sekund^2. Hur stor är kraften när a = 9.8 meter/sekund^2. (Detta är lika med tyngdkraftens acceleration benämnd _g). Konvertera resultatet från newton till kilopond.

Steg	TI-89 Tangenter	TI-92 Plus Tangenter	Visa
1. Öppna dialogrutan MODE, sidan 3. För Unit System väljer du SI för metriska måttenheter. *Resultaten visas i dessa förvalda enheter.*	MODE F3 ⊙ 1 ENTER	MODE F3 ⊙ 1 ENTER	
2. Skapa en enhet för acceleration i meter/sekund^2 och namnge den _ms2. *I dialogrutan UNITS kan du välja enheter från en alfabetisk lista med kategorier. Du kan använda* 2nd ⊖ *och* 2nd ⊕ *för att bläddra genom kategorierna en sida i taget.* *I stället för att skriva in* _m/_s^2 *varje gång du ska ange acceleration kan du nu använda* _ms2. *Du kan också använda UNITS för att välja* _ms2 *från kategorin* Acceleration.	2nd [UNITS] ⊖ ⊙ M ENTER ÷ 2nd [UNITS] ⊖ ⊖ ⊖ ⊙ S ENTER ^ 2 STO▸ ◆ [_] 2nd [a-lock] M S alpha 2 ENTER	◆ [UNITS] ⊖ ⊙ M ENTER ÷ ◆ [UNITS] ⊖ ⊖ ⊖ ⊙ S ENTER ^ 2 STO▸ 2nd [_] M S 2 ENTER	  Om du använder dialogrutan UNITS för att välja enhet läggs _ till automatiskt
3. Beräkna kraften när m = 5 kilogram (_kg) och a = 20 meter/sekund^2 (_ms2). *Om du vet förkortningen för en enhet kan du skriva in den från tangentbordet.*	5 ◆ [_] 2nd [a-lock] K G alpha × 2 0 ◆ [_] 2nd [a-lock] M S alpha 2 ENTER	5 2nd [_] K G × 2 0 2nd [_] M S 2 ENTER	
4. Med samma värde för m kan du nu beräkna kraften vid en acceleration motsvarande jordens gravitation (konstanten _g). *För* _g *kan du använda den fördefinierade konstanten i dialogrutan UNITS, eller skriva in* _g.	5 ◆ [_] 2nd [a-lock] K G alpha × 2nd [UNITS] ⊙ alpha G ENTER ENTER	5 2nd [_] K G × ◆ [UNITS] ⊙ G ENTER ENTER	
5. Konvertera till kilopond (_kgf). 2nd [▸] *visar konverteringsoperatorn* ▸.	⊙ 2nd [▸] ◆ [_] 2nd [a-lock] K G F alpha ENTER	⊙ 2nd [▸] 2nd [_] K G F ENTER	

Du kan använda menyer för att välja måttenheter från en lista med möjliga måttenheter eller så kan du skriva in dem från tangentbordet.

Från en meny

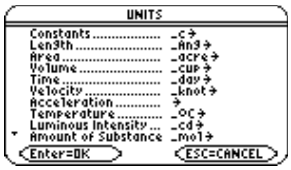
Följande exempel visar hur du väljer en måttenhet men du kan använda samma procedur för att välja en konstant.

I grundfönstret:

1. Skriv in värdet eller uttrycket.

```
6.3
```

2. Öppna dialogrutan UNITS.
   Tryck på:
   TI-89: [2nd] [UNITS]
   TI-92 Plus: [◆] [UNITS]

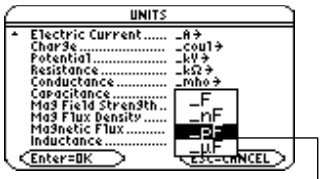


3. Använd ⊖ och ⊕ för att flytta markören till önskad kategori.

4. Välj markerad (förvald) måttenhet genom att trycka på [ENTER].

   – eller –

   Om du vill välja en annan enhet från kategorin trycker du på ⊙. Markera sedan enheten och tryck [ENTER].



Du kan även flytta markören genom att skriva in första bokstaven i ett enhetsnamn.

Den valda måttenheten visas på inmatningsraden. Namn på konstanter och måttenheter börjar alltid med ett understrykningstecken (_).

```
6.3_pF
```

__Tips!__ Använd [2nd] ⊖ och [2nd] ⊕ för att bläddra en sida i taget genom kategorierna.

__Obs!__ Om du skapat en egendefinerad måttenhet i den aktuella kategorin (sidan 88) är den med i listan.

Från tangentbordet

Om du vet vilken förkortning TI-89 / TI-92 Plus använder för en speciell konstant eller måttenhet kan du skriva in den direkt från tangentbordet. Exempel:

256_m

Första tecknet måste alltid vara ett understrykningstecken (_).
För _, tryck på:
TI-89: [◆] [_]
TI-92 Plus: [2nd] [_]

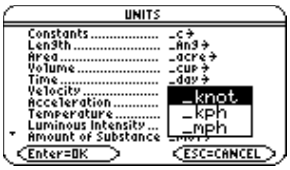
- Ett blanksteg eller multiplikationstecken (*) framför understrykningstecknet kan användas om du vill. Till exempel är 256_m, 256 _m och 256* _m detsamma.
  - Om du däremot lägger till måttenheter till en variabel måste du skriva in ett blanksteg eller * före understrykningstecknet. Till exempel behandlas x_m som en variabel, inte som x med en måttenhet.

__Obs!__ Du kan skriva in enheter i antingen versaler eller gemener.

Kombinera flera måttenheter

Ibland kan du behöva kombinera två eller flera måttenheter från olika kategorier.

Anta att du vill ange hastighet med meter per sekund. I dialogrutan UNITS innehåller kategorin Velocity inte denna måttenhet.



__Tips!__ Skapa egendefinierade måttenheter (sidan 88) för kombinationer som du ofta använder.

Du kan ange meter per sekund genom att kombinera _m och _s från kategorierna Length respektive Time.

```
3*9.8_m/_s
```

Kombinera enheterna _m och _s. Det finns ingen fördefinierad enhet för _m/_s.

Använda parenteser med enheter i en beräkning

För att beräkningarna ska bli korrekta kan du vara tvungen att använda parenteser ( ) för att gruppera värden med måttenheter. Detta gäller framförallt vid division. Exempel:

__Tips!__ Om du är minsta tveksam hur ett värde och dess enhet kommer att beräknas bör du använda parenteser ( ).

Beräkna:	Mata in:
$\frac{100_m}{2_s}$	`100_m/(2_s)` 50.·$\frac{_m}{_s}$

Du måste använda parenteser runt (2_s). Detta är viktigt vid division.

Om du utesluter parenteserna kan du få oväntade resultat som:

`100_m/2_s`	50.·_m·_s

Nedan finner du orsaken till att beräkningen blir missvisande. I en beräkning behandlas enheter på liknande sätt som variabler. Till exempel:

100_m beräknas som 100* _m
och
2_s beräknas som 2* _s

Utan parenteser beräknas inmatningen enligt:

$$100*_m / 2*_s = \frac{100*_m}{2} *_s = 50.\cdot_m\cdot_s$$

Du kan konvertera ett värde från en måttenhet till en annan i samma kategori. Detta gäller även användardefinierade enheter (sidan 88).

Alla enheter utom temperatur

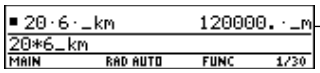
Om du använder en måttenhet i en beräkning konverteras och visas den automatiskt i den måttenhet som är inställd som förval för den kategorin, om du nu inte använder konverteringsoperatorn ▸ som beskrivs senare. Följande exempel förutsätter att du har ställt in förvalda måttenheter till systemet SI för metriska måttenheter (sidan 87).

Obs! *En lista över fördefinierade måttenheter finns på sidan 89.*

Tips! *I dialogrutan UNITS kan du välja måttenhet från en meny.*

Multiplicera 20 med 6 kilometer.

20∗ 6_km



Visas i förinställd enhet för Length, (_m i SI system).

Om du vill konvertera till en måttenhet som inte är förvald använder du konverteringsoperatorn ▸.

uttryck_enhet1 ▸ *_enhet2*

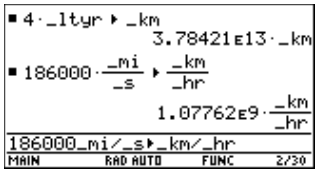
För ▸, tryck [2nd] [▸].

Konvertera 4 ljusår till kilometer:

4_ltyr ▸ _km

Konvertera 186000 miles/sekund till kilometer/timma:

186000_mi/_s ▸ _km/_hr



Om ett uttryck använder en kombination av måttenheter kan du specificera konvertering av vissa av måttenheterna. Måttenheter för vilka du inte anger någon konvertering visas med förvalda måttenheter.

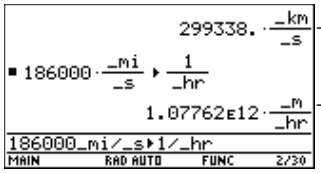
Konvertera 186000 miles/sekund från miles till kilometer:

186000_mi/_s ▸ _km

Konvertera 186000 miles/sekund från sekunder till timmar:

186000_mi/_s ▸ 1/_hr

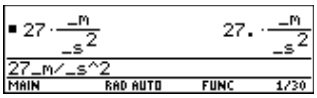
Eftersom ingen tidskonvertering har angetts visas svaret i förvald enhet (_s i detta exempel).



Eftersom ingen längdenhetskonvertering har angetts visas svaret i förvald enhet (_m i detta exempel).

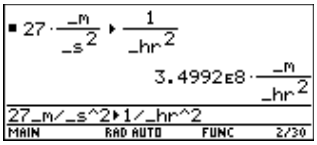
Om du vill ange meter per sekundkvadrat:

27_m/_s^2



Om du vill konvertera meter per sekundkvadrat från sekunder till timmar:

27_m/_s^2 ▸1/_hr^2



Temperaturvärden

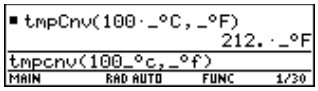
När du vill konvertera ett temperaturvärde måste du använda **tmpCnv()** istället för ▸ operatorn.

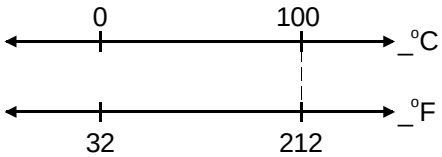
tmpCnv(*uttryck*_°*tempEnhet1*, _°*tempEnhet2*)

└ För °, tryck [2nd] [°].

Om du t ex vill konvertera 100_°C till _°F:

tmpCnv(100_°c, _°f)





Temperaturområden

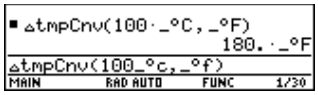
För Δ, tryck på:
TI-89: [◆] [⇧] [D]
TI-92 Plus: [2nd] G [⇧] D

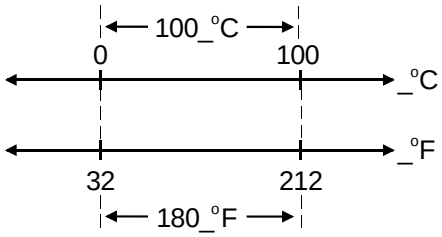
När du vill konvertera ett temperaturområde (differensen mellan två temperaturvärden) använder du **ΔtmpCnv()**.

ΔtmpCnv(*uttryck*_°*tempEnhet1*, _°*tempEnhet2*)

Konvertera t ex ett temperaturområde på 100_°C till dess motsvarande värde i _°F:

ΔtmpCnv(100_°c, _°f)





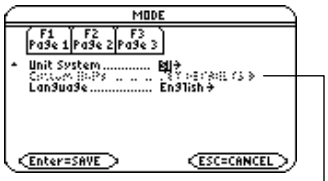
Ange förvalda måttenheter för visningsläge

Alla resultat som innehåller måttenheter visas i den förvalda måttenheten för aktuell kategori. Om t ex den förvalda måttenheten för Length är _m visas alla resultat innehållande längd i meter (även om du angett _km eller _ft i beräkningen).

Om du använder SI eller ENG/US system

Systemen SI och ENG/US för måttenheter (ställs in från sidan 3 i fönstret MODE) använder inbyggda, förvalda måttenheter som du inte kan ändra.

Mer information om de förvalda måttenheterna för dessa system hittar du på sidan 89.

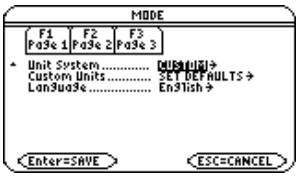


Om Unit System=SI eller ENG/US så är enheten Custom Units gråmarkerad. Du kan inte ange förval för enskilda enheter.

Anpassa förvalda inställningar

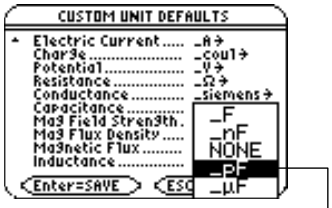
Anpassa förvalda inställningar:

1. Tryck [MODE] [F3] ⊕ 3 för att ställa in Unit System = CUSTOM.
2. Tryck ⊖ för att markera SET DEFAULTS.
3. Tryck ⊕ för att öppna dialogrutan CUSTOM UNIT DEFAULTS.
4. För varje kategori kan du markera dess förvalda måttenhet. Tryck på ⊕ och välj sedan en måttenhet från listan.
5. Tryck två gånger på [ENTER] för att spara dina ändringar och lämna dialogrutan MODE.



Obs! *Du kan även använda* **setUnits()** *eller* **getUnits()** *för att ställa in eller kontrollera förvalda måttenheter. Se Appendix A.*

Tips!: *När dialogrutan CUSTOM UNIT DEFAULTS först öppnas visar den de aktuella förvalda måttenheterna.*



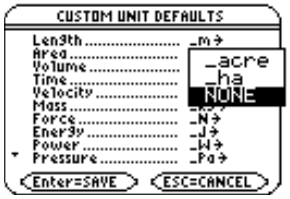
Du kan också flytta markören genom att trycka och sedan skriva in första bokstaven i en enhetsbeteckning.

Vad innebär förval NONE?

I många kategorier kan du ange NONE som förvald måttenhet.

Detta innebär att resultat i denna kategori visas i de förvalda värdena för dess komponenter.

Till exempel om Area = Length^2 så är Length en komponent av Area.



Obs! *NONE finns inte tillgängligt för baskategorier som Length och Mass som saknar komponenter.*

- Om förvalda enheter är Area = _acre och Length = _m (meter) visas resultat för area i måttenheten _acre.
- Om du anger Area = NONE visas resultat för area i måttenheten_m^2.

Skapa egna måttenheter

I många kategorier kan du skapa egna måttenheter utifrån en eller flera fördefinierade enheter. Du kan också skapa helt fristående måttenheter.

Varför ska du skapa egna måttenheter?

__Obs!__ Om du definierar en egen enhet för en existerande kategori kan du välja den från menyn i dialogrutan UNITS. Du kan däremot inte använda [MODE] för att välja denna enhet som förval vid visning.

Några orsaker till att skapa egna måttenheter kan vara:

- Du kanske vill ange längdmått med dekameter. Ange i detta fall 10_m som en ny måttenhet med beteckningen _dm.
- I stället för att skriva in _m/_s^2 som en enhet för acceleration kan du kombinera flera måttenheter till en ny enhet med beteckningen _ms2.
- Anta att du vill räkna ut hur många gånger en person blinkar. Du kan i detta fall använda _blinks som en giltig måttenhet utan att definiera den. Denna fristående enhet hanteras på samma sätt som en odefinierad variabel. Till exempel är 3_blinks samma som 3a.

Regler för namn på egendefinierade enheter

Reglerna för namngivning av enheter är liknande de för variabler.

- Namnet kan bestå av upp till 8 tecken.
- Första tecknet måste vara ett understrykningstecken. För _, tryck på:
  TI-89: [◆] [_]
  TI-92 Plus: [2nd] [_]
- Andra tecknet kan vara valfritt tecken förutom _ eller en siffra. Till exempel är _9f inte giltigt.
- Återstående tecken (upp till 6) kan vara valfria tecken förutom understrykningstecken.

Definiera en måttenhet

Du definierar en måttenhet på samma sätt som du definierar en variabel.

definition → *_nyEnhet*

└ För →, tryck på [STO▸].

__Obs!__ Användardefinierade enheter visas med gemener oavsett vilket teckenläge (versaler/gemener) du använde vid definitionen.

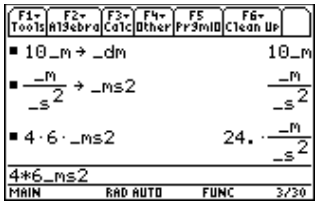
__Obs!__ Användardefinierade måttenheter som _dm sparas som variabler. Du kan ta bort dem på samma sätt som du tar bort variabler.

Definiera måttenheten dekameter:

10_m → _dm

Definiera en enhet för acceleration:

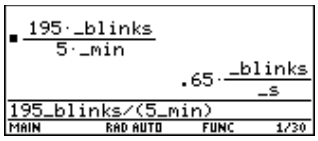
_m/_s^2 → _ms2



Förutsatt att förvald enhet för Length och Time är satt till _m och _s.

För att beräkna 195 blinkningar på 5 minuter som _blinks/_min:

195_blinks/(5_min)



Förutsatt att förvald enhet för Time är satt till _s.

Fördefinierade konstanter och måttenheter

Detta avsnitt listar fördefinierade konstanter och måttenheter kategorivis. Du kan välja dessa enheter från dialogrutan UNITS. Om du använder MODE för att ställa in förvalda enheter bör du observera att listan inte tar upp kategorier med bara en måttenhet.

Förval för SI och ENG/US

Systemen SI och ENG/US använder förvalda, inbyggda, måttenheter. I detta avsnitt markeras förvalen med (SI) och (ENG/US). I vissa kategorier använder båda systemen samma förvalda måttenheter.

Mer information om förvalet NONE finns på sidan 87. Observera att vissa kategorier saknar förvalda måttenheter.

Konstanter

Obs! *TI-89 / TI-92 Plus förenklar uttryck med måttenheter och visar resultatet i förvalda enheter. På grund av detta kan vissa konstanter visa andra värden än de exempel som visas i nedanstående tabell.*

Obs! *Se snabbtangenter för grekiska tecken (på insidan av omslaget).*

_c	ljusets hastighet	2.99792458E8_m/_s
_Cc	Coulombs konstant	8.9875517873682E9_N•_m²/_coul²
_g	tyngdkraftens acceleration	9.80665_m/_s²
_Gc	gravitationskonstanten	6.67259E-11_m³/_kg/_s²
_h	Plancks konstant	6.6260755E-34_J•_s
_k	Boltzmanns konstant	1.380658E-23_J/_°K
_Me	elektronens vilomassa	9.1093897E-31_kg
_Mn	neutronens vilomassa	1.6749286E-27_kg
_Mp	protonens vilomassa	1.6726231E-27_kg
_Na	Avogadros tal	6.0221367E23 /_mol
_q	elektronens laddning	1.60217733E-19_coul
_Rb	Bohrradie	5.29177249E-11_m
_Rc	allmänna gaskonstanten	8.31451_J/_mol/_°K
_Rdb	Rydbergs konstant	10973731.53413 /_m
_Vm	molvolym	2.241409E-2_m³/_mol
_ε0	kapacitivitet i vakuum	8.8541878176204E-12_F/_m
_σ	Stefan-Boltzmans konstant	5.6705119E-8_W/_m²/_°K⁴
_ϕ0	magnetiskt flödeskvantum	2.0678346161E-15_Wb
_μ0	permeabilitet i vakuum	1.2566370614359E-6_N/_A²
_μb	Bohrmagneton	9.2740154E-24_J•_m²/_Wb

Längd

_Ang	ångström	_mi	mile
_au	astronomisk enhet	_mil	1/1000 inch
_cm	centimeter	_mm	millimeter
_fath	fathom	_Nmi	nautisk mil
_fm	fermi	_pc	parsec
_ft	foot (ENG/US)	_rod	rod
_in	inch	_yd	yard
_km	kilometer	_μ	mikron
_ltyr	ljusår	_Å	ångström
_m	meter (SI)		

Area

_acre	acre (tunnland)	NONE (SI) (ENG/US)
_ha	hektar	

Volym

_cup ..... cup
_floz ..... fluid ounce
_flozUK.. British fluid ounce
_gal..... gallon
_galUK ... British gallon
_l..... liter
_ml..... milliliter
_pt..... pint
_qt..... quart
_tbsp..... matsked
_tsp..... tesked
NONE (SI) (ENG/US)

Tid

_day..... dag
_hr ..... timma
_min ..... minut
_ms..... millisekund
_ns..... nanosekund
_s ..... sekund (SI) (ENG/US)
_week..... vecka
_yr..... år
_µs ..... mikrosekund

Hastighet

_knot..... knop
_kph ..... kilometer per timma
_mph ..... miles per timma
NONE (SI) (ENG/US)

Acceleration

inga fördefinerade enheter

Temperatur

_°C..... °Celsius
För °, tryck [2nd] [°].
_°F..... °Fahrenheit
_°K..... °Kelvin
_°R..... °Rankine
(inget förval)

Ljusstyrka

_cd..... candela
(inget förval)

Materiemängd

_mol ..... mol
(inget förval)

Massa

_amu ..... atomic mass unit
_gm..... gram
_kg..... kilogram (SI)
_lb..... pound (ENG/US)
_mg..... milligram
_mton..... metriskt ton
_oz ..... ounce
_slug ..... slug
_ton ..... ton
_tonne .... metriskt ton
_tonUK... long ton

Kraft

_dyne ..... dyne
_kgf ..... kilopond
_lbf ..... pound force (ENG/US)
_N ..... newton (SI)
_tonf ..... ton force

Energi

_Btu..... British thermal unit (ENG/US)
_cal..... kalori
_erg ..... erg
_eV ..... elektronvolt
_ftlb ..... foot-pound
_J ..... joule (SI)
_kcal..... kilokalori
_kWh ..... kilowatt-timma
_latm ..... liter atmosfär

Effekt

_hp ..... hästkrafter (ENG/US)
_kW ..... kilowatt
_W..... watt (SI)

Tryck	_atm ..... atmosfär _bar ..... bar _inH2O... inches av vatten _inHg..... inches av kvicksilver _mmH2O. millimeter vatten	_mmHg... millimeter kvicksilver _Pa..... pascal (SI) _psi ..... pounds per square inch (ENG/US) _torr..... millimeter kvicksilver
Kinematisk viskositet	_St ..... stokes	
Ddynamisk viskositet	_P ..... poise	
Frekvens	_GHz ..... gigahertz _Hz ..... hertz (SI) (ENG/US)	_kHz ..... kilohertz _MHz ..... megahertz
Elektrisk ström	_A ..... ampere (SI) (ENG/US) _kA ..... kiloampere _mA..... milliampere	_µA ..... mikroampere
Laddning	_coul ..... coulomb (SI) (ENG/US)	
Spänning	_kV ..... kilovolt _mV ..... millivolt	_V..... volt (SI) (ENG/US) _volt..... volt
Resistans	_kΩ..... kilo ohm	_MΩ..... megaohm _ohm ..... ohm _Ω ..... ohm (SI) (ENG/US)
Konduktans	_mho.....mho (ENG/US) _mmho ...millimho	_siemens. siemens (SI) _µmho .... mikromho
Kapacitans	_F ..... farad (SI) (ENG/US) _nF ..... nanofarad _pF ..... pikofarad	_µF ..... mikrofarad
Magnetisk fältstyrka	_Oe ..... oersted	NONE (SI) (ENG/US)
Magnetisk flödestäthet	_Gs ..... gauss	_T..... tesla (SI) (ENG/US)
Magnetiskt flöde	_Wb ..... weber (SI) (ENG/US)	
Induktans	_henry.... henry (SI) (ENG/US) _mH..... millihenry _nH..... nanohenry	_µH ..... mikrohenry

Ytterligare funktioner i grundfönstret

5

Spara inmatningar i grundfönstret som ett Text Editor-skript 94
Klippa ut, kopiera och klistra in information 95
Skapa och beräkna användardefinierade funktioner 97
Använda mappar för att spara oberoende uppsättningar variabler 100
Om storleken på en inmatning eller ett resultat är för stor 103

I kapitel 2 beskrevs de grundläggande funktionerna för grundfönstret, för att du snabbt skulle kunna börja använda TI-89 / TI-92 Plus.

I det här kapitlet beskrivs ytterligare funktioner som du kan använda för att göra arbetet i grundfönstret ännu mer effektivt.

Eftersom detta kapitel består av olika fristående ämnen visas inget exempel i början.

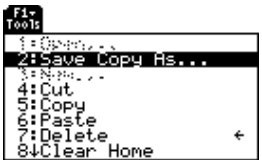
Spara inmatningar i grundfönstret som ett Text Editor-skript

Om du vill spara alla inmatningar i historiklistan kan du spara grundfönstret som en textvariabel. När du kör dessa inmatningar igen använder du Text Editor för att öppna variabeln som ett kommandoskript.

Spara inmatningarna i historiklistan

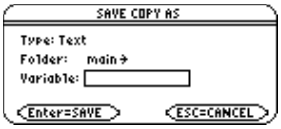
Gör följande från grundfönstret:

1. Tryck på F1 och välj 2:Save Copy As.



__Obs!__ Det är bara inmatningarna som sparas, inte resultaten.

2. Ange en mapp och en textvariabel som du vill använda för att lagra inmatningarna i.



Post	Beskrivning
Type	Ställs automatiskt in som Text och kan inte ändras.
Folder	Visar mappen som textvariabeln kommer att sparas i. Om du vill använda en annan mapp trycker du på ▶ för att visa en meny med befintliga mappar. Välj en mapp.
Variable	Skriv ett giltigt, oanvänt variabelnamn.

__Obs!__ Information om mappar finns på sidan 100.

3. Tryck på ENTER (när du har skrivit i en inmatningsruta, som exempelvis Variable, trycker du på ENTER två gånger).

Återställa sparade inmatningar

Eftersom inmatningarna är sparade i skriptformat kan du inte återskapa dem från grundfönstret. (På menyn F1 i grundfönstret är 1:Open inte tillgängligt.) Gör i stället följande:

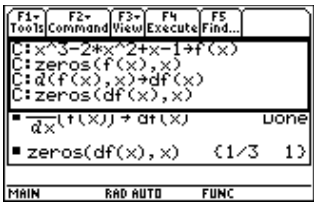
__Obs!__ Fullständig information om hur du använder Text Editor och utför ett kommandoskript finns i kapitel 18.

1. Använd Text Editor för att öppna variabeln som innehåller de sparade inmatningarna från grundfönstret.

   De sparade inmatningarna visas som en serie kommandorader som du kan utföra individuellt, i vilken ordning som helst.

2. Starta med markören på den första raden i skriptet och tryck upprepade gånger på F4 för att utföra kommandona rad för rad.

3. Visa det återställda grundfönstret.



Detta delade fönster visar Text Editor (med kommandoskriptet) och det återställda grundfönstret.

Klippa ut, kopiera och klistra in information

Genom att klippa ut, kopiera och klistra in kan du flytta eller kopiera information inom samma applikation eller mellan olika applikationer. Dessa funktioner använder Urklipp, som är ett område i minnet som fungerar som en tillfällig lagringsplats.

Klistra in automatiskt och klippa ut/kopiera/klistra in

Automatisk inklistring (beskrivs i kapitel 2) är ett snabbt sätt att kopiera en inmatning eller ett resultat i historiklistan och klistra in det/den på inmatningsraden.

1. Använd ⊕ och ⊖ för att markera posten i historiklistan.
2. Tryck på [ENTER] för att automatiskt klistra in posten på inmatningsraden.

För att kunna kopiera eller flytta information på inmatningsraden måste du använda funktionen klippa ut, kopiera eller klistra in. (Du kan kopiera från historiklistan men du kan inte klippa ut eller klistra in)

Klippa ut eller kopiera information till Urklipp

När du klipper ut eller kopierar information placeras den informationen i Urklipp. När du klipper ut tas information bort helt och hållet från den aktuella platsen (används för att flytta information) och när du kopierar finns informationen kvar.

1. Markera tecknen som du vill klippa ut eller kopiera.

   På inmatningsraden flyttar du markören till ena sidan av tecknen. Håll ned [↑] och tryck på ⊖ eller ⊕ för att markera tecken till vänster eller höger om markören.

2. Tryck på [F1] och välj 4:Cut eller 5:Copy.

Tips! *Du kan kopiera, klippa ut och klistra in utan att använda menyn [F1] i verktygsfältet. Tryck på:*
TI-89:
[◆] [CUT], [◆] [COPY], *eller* [◆] [PASTE]
TI-92 Plus:
[◆] X, [◆] C, *eller* [◆] V

Urklipp = (tom eller med föregående innehåll)

Efter du har klippt ut

Urklipp = x^4– 3x^3– 6x^2+8x

Efter du har kopierat

Urklipp = x^4– 3x^3– 6x^2+8x

Obs! *När du klipper ut eller kopierar information ersätter den informationen det föregående innehållet i Urklipp (om det finns något över huvud taget).*

Att klippa ut är inte det samma som att ta bort. När du tar bort information placeras informationen inte i Urklipp och kan därför inte användas igen.

Klistra in information från Urklipp

När du klistrar in infogas innehållet som finns i Urklipp vid markören på inmatningsraden. Innehållet i Urklipp ändras inte.

1. Placera markören där du vill klistra in informationen.
2. Tryck på [F1] och välj 6:Paste (eller använd genvägen):
   TI-89: [◆] [PASTE]
   TI-92 Plus: [◆] V

Exempel: Kopiera och klistra in

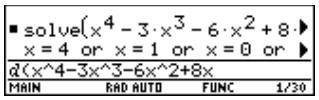
Anta att du vill återanvända ett uttryck utan att behöva skriva om det varje gång. Gör följande:

1. Kopiera informationen.
   a. Använd [↑] [▷] eller [↑] [◁] för att markera uttrycket.

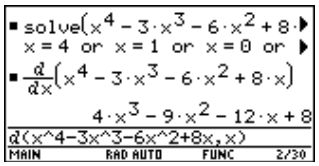
   b. Tryck på:
   TI-89: [◆] [COPY]
   TI-92 Plus: [◆] C
   c. I detta exempel trycker du på [ENTER] för att utföra beräkningen.

Tips! Du kan även återanvända uttryck genom att spara dem som användardefinierade funktioner. Se sidan 97.

2. Klistra in den kopierade informationen i en ny inmatning.
   a. Tryck på [F3] 1 för att välja funktionen ***d*** differentiate.
   b. Tryck på:
   TI-89: [◆] [PASTE]
   TI-92 Plus: [◆] V
   för att klistra in det kopierade uttrycket.

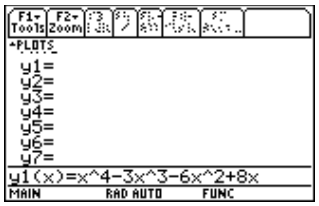


   c. Slutför den nya inmatningen och tryck på [ENTER].



Tips! Genom att kopiera och klistra in kan du enkelt överföra information från en applikation till ett annan.

3. Klistra in den kopierade informationen i ett annat program.
   a. Tryck på [◆] [Y=] för att visa Y= Editor.
   b. Tryck på [ENTER] för att definiera y1(x).
   c. Tryck på:
   TI-89: [◆] [PASTE]
   TI-92 Plus: [◆] V
   för att klistra in.



   d. Tryck på [ENTER] för att spara den nya inmatningen.

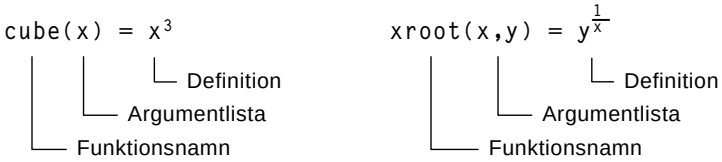
Skapa och beräkna användardefinierade funktioner

Med användardefinierade funktioner kan du spara mycket tid när du måste repetera samma uttryck (men med olika värden) flera gånger. Med användardefinierade funktioner kan du också utöka räknarens funktioner.

Format på funktioner

I följande exempel visas användardefinierade funktioner med ett argument och två argument. Du kan använda så många argument som behövs. I dessa exempel består definitionen av ett enda uttryck (eller en sats).

***Obs!** För funktionsnamn gäller samma regler som för variabelnamn. Se "Spara och hämta variabelvärden" i kapitel 2.*



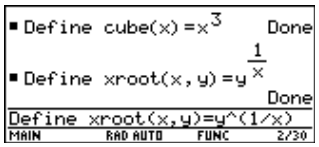
När du definierar funktioner och program måste du använda unika argumentnamn som inte kommer att användas i efterföljande funktions- eller programanrop.

I argumentlistan måste du använda samma argument som används i definitionen. cube(n) = x^3 ger exempelvis oväntade resultat när du beräknar funktionen.

Argument (x och y i dessa exempel) är platshållare som motsvarar de värden som överförts till funktionen. De motsvarar inte variablerna x och y såvida du inte specifikt överför x och y som argument när du beräknar funktionen.

Skapa en användardefinierad funktion

Använd någon av följande metoder:

Metod	Beskrivning
STO▸	Spara ett uttryck till ett funktionsnamn (inklusive argumentlistan).
Define-kommando	Definiera ett funktionsnamn (inklusive argumentlistan) som ett uttryck. 
Program Editor	Se kapitel 17 för information om hur du skapar en användardefinierad funktion.

Skapa en funktion med flera satser

***Obs!** Information om likheter och skillnader mellan funktioner och program finns i kapitel 17.*

Du kan också skapa en användardefinierad funktion vars definition består av flera programsatser. Definitionen kan inkludera många kontroll- och beslutsstrukturer (**If**, **ElseIf**, **Return** osv) som används i programmering.

Anta exempelvis att du vill skapa en funktion som summerar en serie baserade på heltalet (n):

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

När du skapar definitionen för en funktion med flera satser kan det vara praktiskt att först visualisera den i blockform.

Variabler som inte är med i argumentlistan måste deklareras som lokala.

Returnerar ett meddelande om n inte är ett heltal eller om nn≤0.

Summerar serien.

Returnerar summan.

```
Func
Local temp,i
If fPart(n)≠0 or n≤0
  Return "bad argument"
0→temp
For i,n,1,⁻1
  approx(temp+1/i)→temp
EndFor
Return temp
EndFunc
```

Func och **EndFunc** måste börja och avsluta funktionen.

Mer information om olika programsatser finns i appendix A.

När du matar in en funktion med flera satser i grundfönstret måste du mata in hela funktionen på en rad. Använd kommandot **Define** på samma sätt som med en funktion med en sats.

Använd argumentnamn som aldrig kommer att användas vid anrop av funktioner eller ett program.

Använd kolon för att avgränsa varje programsats.

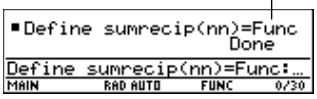
```
Define sumrecip(nn)=Func:Local temp,i: ... :EndFunc
```

***Tips!** Det är lättare att skapa en komplicerad funktion med flera satser i Program Editor än det är i grundfönstret. Se kapitel 17.*

I grundfönstret:

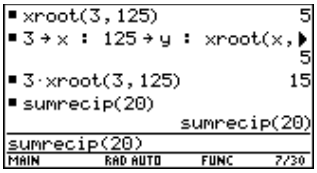
Funktioner med flera satser visas som "Func".

Mata in en funktion med flera satser på en rad. Kom ihåg att använda kolon.



Beräkna en funktion

Du kan använda en användardefinierad funktion på samma sätt som vilken funktion som helst. Beräkna den fristående eller använd den i ett annat uttryck.



Visa och redigera en funktionsdefinition

Om du vill:	Gör du följande:
Visa en lista med alla användardefinierade funktioner	Tryck på [2nd][VAR-LINK] för att visa VAR-LINK-fönstret. Du kanske måste använda menyn [F2] View för att ange variabeltypen Function. (Se kapitel 21.) – eller – Tryck på: **TI-89:** [CATALOG] [F4] **TI-92 Plus:** [2nd] [CATALOG] [F4]
Visa en lista på Flash-applikations-funktioner	Tryck på: **TI-89:** [CATALOG] [F3] **TI-92 Plus:** [2nd] [CATALOG] [F3]
Visa definitionen för en användardefinierad funktion	Markera funktionen i fönstret VAR-LINK och öppna menyn Innehåll. **TI-89:** [2nd] [F6] **TI-92 Plus:** [F6] – eller – Från grundfönstret trycker du på [2nd][RCL]. Skriv funktionsnamnet men inte argument-listan (exempelvis xroot) och tryck på [ENTER] två gånger. – eller – Från Program Editor öppnar du funktionen. (Se kapitel 17.)
Redigera definitionen	Från grundfönstret använder du [2nd][RCL] för att visa definitionen. Redigera definitionen. Använd därefter [STO▸] eller **Define** för att spara den nya definitionen. – eller – Från Program Editor öppnar du funktionen, redigerar den och sparar ändringarna. (Se kapitel 17.)

Obs! *Du kan visa en användardefinierad funktion I katalogens dialogruta, men du kan inte använda katalogen för att visa eller editera dess definition.*

I TI-89 / TI-92 Plus finns en inbyggd mapp med namnet MAIN, där alla variabler sparas. Genom att skapa ytterligare mappar kan du spara oberoende uppsättningar av användardefinierade variabler (även användardefinierade funktioner).

Mappar och variabler

Med mappar kan du på ett praktiskt sätt hantera variabler genom att ordna mapparna i relaterade grupper. Du kan exempelvis skapa separata mappar för olika program (Math, Text Editor osv.) eller klasser.

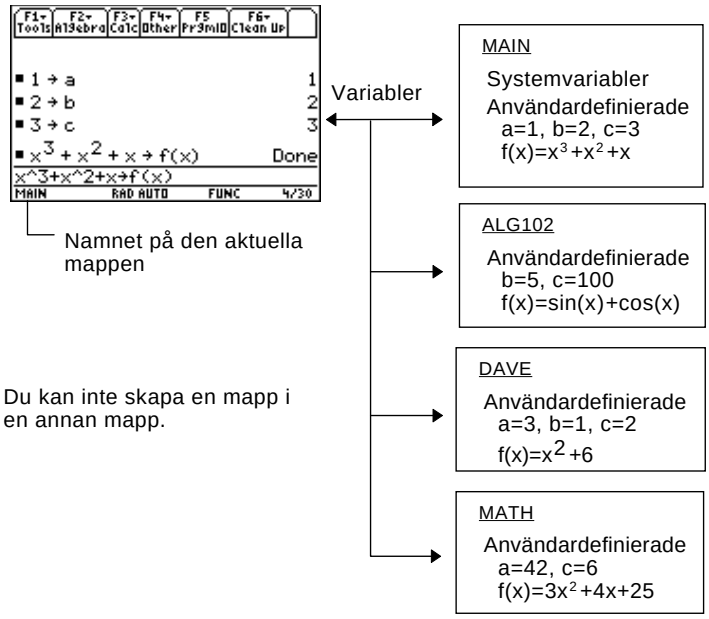
- Du kan spara en användardefinierad variabel i en befintlig mapp.
- En systemvariabel eller en variabel med ett reserverat namn kan dock endast sparas i mappen MAIN.

De användardefinierade variablerna i en mapp är oberoende av variablerna i andra mappar.

Exempel på variabler som endast kan sparas i mappen MAIN
Window-variabler (xmin, xmax osv.)
Table setup-variabler (TblStart, ΔTbl osv.)
Y= Editor-funktioner (y1(x) osv)

Du kan därför spara separata uppsättningar variabler i mappar med samma namn men med olika värden.

***Obs!** Användardefinierade variabler sparas i den "aktuella mappen" om du inte anger annat. Se "Använda variabler i olika mappar" på sidan 102.*



Du har alltid direkt tillgång till systemvariablerna i mappen MAIN, oavsett vilken som är den aktuella mappen.

Skapa en mapp från grundfönstret

Mata in kommandot **NewFold**.

NewFold *mappnamn*

└ Mappen som ska skapas. Den nya mappen blir automatiskt den aktuella mappen.

Skapa en mapp från VAR-LINK-fönstret

I VAR-LINK-fönstret, som beskrivs i kapitel 21, visas de befintliga variablerna och mapparna.

1. Tryck på [2nd] [VAR-LINK].
2. Tryck på [F1] Manage och välj 5:Create Folder.
3. Skriv in ett unikt katalognamn, högst åtta tecken, och tryck på [ENTER] två gånger.

När du har skapat en ny mapp från VAR-LINK-fönstret blir den mappen *inte* automatiskt den aktuella mappen.

Ställa in den aktuella mappen från grundfönstret

Mata in funktionen **setFold**.

setFold (*mappnamn)*

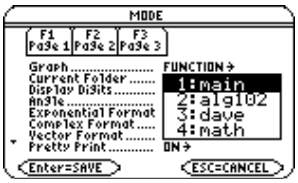
└ Funktionen **setFold** kräver att du skriver mappnamnet inom parentes.

När du utför **setFold** returneras namnet på den föregående aktuella mappen.

Ställa in den aktuella mappen från dialogrutan MODE

Så här använder du dialogrutan MODE:

1. Tryck på [MODE].
2. Markera inställningen Current Folder.
3. Tryck på ▶ för att visa en meny med befintliga mappar.
4. Välj en mapp. Gör något av följande:
   - Markera mappnamnet och tryck på [ENTER].
     – eller –
   - Tryck på numret eller bokstaven som motsvarar den mappen.
5. Tryck på [ENTER] för att spara ändringarna och stänga dialogrutan.



Tips! *Om du vill stänga menyn eller dialogrutan utan att spara ändringar trycker du på* [ESC].

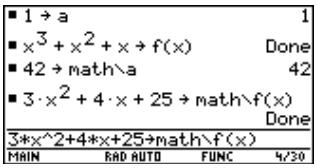
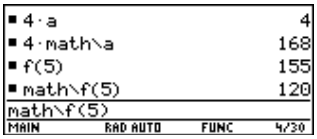
Använda variabler i olika mappar

Du kan använda en användardefinierad variabel eller funktion som inte är i den aktuella mappen. Mata in det fullständiga *mappnamnet* i stället för endast variabelnamnet.

Ett sökvägsnamn har följande format:

mappnamn\variabelnamn
– eller –
mappnamn\funktionsnamn

Exempel:

Om den aktuella mappen = MAIN	Mappar
	MAIN a=1 f(x)=x^3+x^2+x
	MATH a=42 f(x)=$3x^2+4x+25$

Obs! *I detta exempel antas att du redan har skapat en mapp med namnet MATH.*

Obs! *Information om VAR-LINK-fönstret finns i kapitel 21.*

Om du vill visa en lista över befintliga mappar och variabler trycker du på [2nd][VAR-LINK]. I VAR-LINK-fönstret kan du markera en variabel och trycka på [ENTER] för att klistra in variabelnamnet på inmatningsraden i grundfönstret. Om du klistrar in ett variabelnamn som inte är i den aktuella mappen klistras sökvägsnamnet (*mappnamn\variabelnamn*) in.

Ta bort en mapp från grundfönstret

Innan du tar bort en mapp måste du ta bort alla variabler som finns i den mappen.

- Ta bort en variabel genom att utföra kommandot **DelVar**.

  DelVar *var1* [, *var2*] [, *var3*] ...

Obs! *Du kan inte ta bort mappen MAIN.*

- Ta bort en tom mapp genom att utföra kommandot **DelFold**.

  DelFold *mapp1* [, *mapp2*] [, *mapp3*] ...

Ta bort en mapp från VAR-LINK-fönstret

Med VAR-LINK-fönstret kan du ta bort en mapp och dess variabler på samma gång. Se kapitel 21.

1. Tryck på [2nd][VAR-LINK].
2. Markera vilken eller vilka poster som ska tas bort och tryck på [F1]1 eller [←]. (Om du använder [F4] för att markera en mapp markeras dess variabler automatiskt.)
3. Tryck på [ENTER] för att bekräfta borttagningen.

Om storleken på en inmatning eller ett resultat är för stor

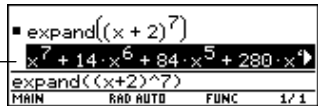
I vissa fall kan storleken på en inmatning eller ett resultat vara för lång och/eller för hög för att kunna visas i historiklistan. I andra fall kanske inte ett resultat kan visas på grund av för lite ledigt minne.

Om en inmatning är för lång eller ett resultat för långt

Flytta markören till historiklistan och markera inmatningen eller resultatet. Använd därefter markörknappen för att bläddra. Exempel:

- I följande exempel visas ett resultat som är för långt för att rymmas på en rad.

Tryck på ◀ eller 2nd ◀ för att rulla åt vänster.

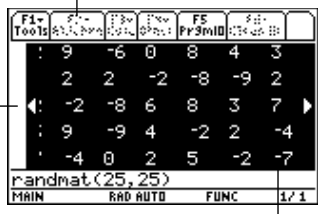


Tryck på ▶ eller 2nd ▶ för att rulla åt höger.

- I följande exempel visas ett resultat som både är för långt och för högt för att kunna visas i fönstret.

TI-89: Tryck på ⊕ eller ⇧ ⊕ för att rulla uppåt.
TI-92 Plus: Tryck på ⊕ eller ⇧ ⊕ för att rulla uppåt.

Tryck på ◀ eller 2nd ◀ för att rulla åt vänster.



Tryck på ▶ eller 2nd ▶ för att rulla åt höger.

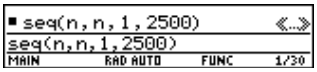
TI-89: Tryck på ⇧ ⊖ för att rulla nedåt.
TI-92 Plus: Tryck på ⇧ ⊖ för att rulla nedåt.

Obs! *I det här exemplet används funktionen* **randMat** *för att generera en 25 x 25-matris.*

För lite ledigt minne

Symbolen <<...>> visas när det finns för lite ledigt minne för att svaret ska kunna visas.

Exempel:



Obs! *I det här exemplet används funktionen* **seq** *för att generera en lista över heltal från 1 till 2 500 i ordningsföljd.*

När du ser symbolen <<...>> kan inte svaret visas, även om du markerar det och försöker rulla.

Pröva att göra följande:

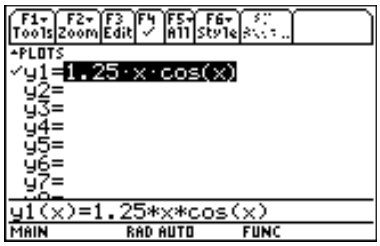
- Frigör minne genom att ta bort variabler som inte används och/eller Flash-applikationer enligt beskrivningen i kapitel 21.
- Försök dela upp problemet i mindre delar som kan beräknas och visas med mindre minne.

Grundläggande funktionsplottning

6

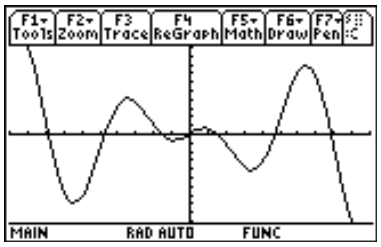
Översikt över grundläggande funktionsplottning 106
Översikt över stegen i funktionsplottning 107
Ställa in grafläget 108
Definiera funktioner som ska plottas 109
Välja vilka funktioner som ska plottas 111
Ange visningsformat för funktioner 112
Definiera visningsfönstret 113
Ändra grafformatet 114
Plotta de markerade funktionerna 115
Visa koordinater med den rörliga markören 116
Spåra en funktion 117
Använda zoomning för att utforska en graf 119
Använda matematikverktyg för att analysera funktioner 122

I det här kapitlet beskrivs hur du gör för att visa och utforska en graf. Innan du läser detta kapitel bör du känna till innehållet i kapitel 2.



Y= Editor visar en algebraisk representation.

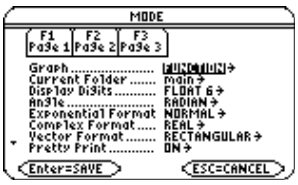
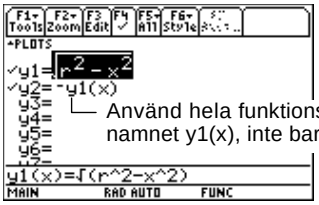
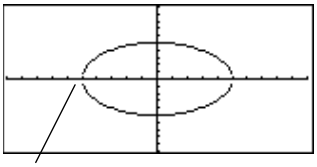
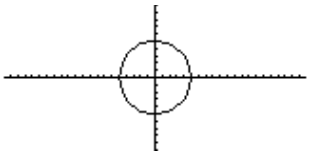
Graffönstret visar en grafisk representation.



Även om detta kapitel handlar om att plotta y(x)-funktioner, gäller de grundläggande stegen för alla graflägen. I senare kapitel finns specifik information om andra graflägen.

Översikt över grundläggande funktionsplottning

Plotta en cirkel med radien 5, med medelpunkt i origo. Visa cirkeln med standardvisnings-fönstret (**ZoomStd**). Använd sedan **ZoomSqr** för att justera visningsfönstret.

Steg	TI-89 Tangenter	TI-92 Plus Tangenter	Teckenfönster
1. Visa dialogrutan MODE. För Graph väljer du FUNCTION.	[MODE] [▶] 1 [ENTER]	[MODE] [▷] 1 [ENTER]	
2. Visa grundfönstret. Spara sedan radien, 5, i variabeln r.	[HOME] 5 [STO▸] [alpha] R [ENTER]	[◆] [HOME] 5 [STO▸] R [ENTER]	5→ r
3. Visa och rensa Y= Editor. Definiera sedan y1(x) = $\sqrt{r^2 - x^2}$, den övre halvan av en cirkel. *När du plottar en funktion måste du definiera separata funktioner för de övre och undre delarna av en cirkel.*	[◆] [Y=] [F1] 8 [ENTER] [ENTER] [2nd] [√] [alpha] R [^] 2 [−] X [^] 2 [)] [ENTER]	[◆] [Y=] [F1] 8 [ENTER] [ENTER] [2nd] [√] R [^] 2 [−] X [^] 2 [)] [ENTER]	
4. Definiera y2(x) = $^-\sqrt{r^2 - x^2}$, funktionen för den undre delen av cirkeln. *Den undre halvan är den den övre halvan med negativt tecken, så du kan definiera y2(x) = ⁻y1(x).*	[ENTER] [(-)] Y 1 [(] X [)] [ENTER]	[ENTER] [(-)] Y 1 [(] X [)] [ENTER]	 Använd hela funktions-namnet y1(x), inte bara y1.
5. Välj visningsfönstret ZoomStd, som automatiskt plottar funktionerna. *I standardvisningsfönstret är intervallet för både x- och y-axeln ⁻10 till 10. Detta intervall visas en längre sträcka längs x-axeln än y-axeln. Därför visas cirkeln som en ellips.*	[F2] 6	[F2] 6	 Lägg märke till det lilla mellanrummet mellan den övre och undre halvan.
6. Välj ZoomSqr. *ZoomSqr ökar intervallet längs x-axeln så att cirklar och kvadrater visas i korrekta proportioner.*	[F2] 5	[F2] 5	

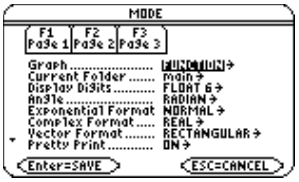
Obs! Det är ett mellanrum mellan den övre och den undre halvan av cirkeln eftersom varje halva är en separat funktion. De matematiska slutpunkterna för båda halvcirklarna är (–5, 0) och (5, 0). Beroende på visningsfönstret kan dock de *plottade* slutpunkterna för varje halvcirkel skilja sig något från deras *matematiska* slutpunkter.

Översikt över stegen i funktionsplottning

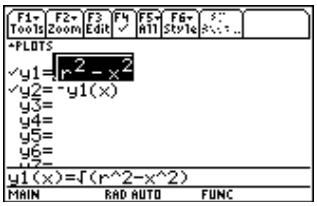
Du plottar en eller flera y(x)-funktioner genom att använda de allmänna stegen som beskrivs nedan. En mer detaljerad beskrivning av varje steg visas på sidorna som följer. Det är inte säkert du behöver följa alla steg varje gång du plottar en funktion.

Plotta funktioner

Ställ in grafläget ([MODE]) till FUNCTION. Ange vinkelenheten om det är nödvändigt.



Definiera funktioner i Y= Editor ([◆] [Y=]).



Välj ([F4]) som plottar funktioner.

Tips! Om du vill stänga av plottning av statistiska data (kapitel 16), trycker du på [F5] 5 *eller använder* [F4] *för att avmarkera funktionerna.*

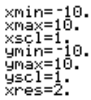
Ställ in visningsformatet för en funktion.

TI-89: [2nd] [F6]

TI-92 Plus: [F6]

Tips! Detta är ett valfritt alternativ. Vid användning av flera funktioner blir plottningen lättare att granska på detta sätt.

Definiera visningsfönstret ([◆] [WINDOW]).



Tips! Med [F2] Zoom *ändrar du också visningsfönstret.*

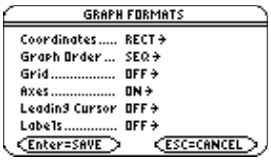
Ändra grafformatet, om det behövs.

[F1] 9

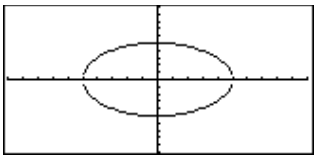
— eller —

TI-89: [◆] [1]

TI-92 Plus: [◆] F



Plotta de markerade funktionerna ([◆] [GRAPH]).



Utforska grafen

Du kan göra följande från graffönstret kan du:

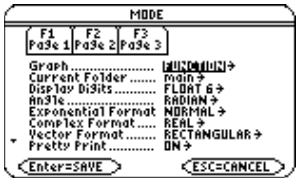
- Visa koordinaterna för vilken bildpunkt som helst genom att använda den rörliga markören, eller visa koordinaterna av en ritad punkt genom att spåra en funktion.
- Använd Zoom verktygsfältet [F2] för att zooma in eller ut på en del av grafen.
- Använd Math verktygsfältet [F5] för att söka efter nollställen, minimum, maximum osv.

Innan du plottar y(x)-funktioner måste du välja grafläget **FUNCTION**. Du kan också behöva ställa in enheten **Angle**, som påverkar hur räknaren plottar grafer av trigonometriska funktioner.

Grafläget

1. Tryck på MODE för att visa dialogrutan MODE, som visar de aktuella inställningarna för lägen.
2. Ställ in grafläget till FUNCTION. Se "Ställa in lägen" i kapitel 2.

Obs! *För grafer som inte använder komplexa tal ställer du in Complex Format = REAL. Annars kan grafer som använder potensform, exempelvis* $x^{1/3}$*, påverkas.*



Detta kapitel beskriver särskilt hur du plottar y(x)-funktioner. Du kan välja bland sex inställningar för graflägen.

Obs! *Andra inställningar för* Graph *läget beskrivs i senare kapitel.*

Inställningar för grafläget	Beskrivning
FUNCTION	y(x)-funktioner
PARAMETRIC	x(t)- och y(t)- ekvationer i parameterform
POLAR	r(θ)-ekvationer i polär form
SEQUENCE	u(n)-talföljder
3D	z(x,y) 3D-ekvationer
DIFFERENTIAL EQUATION	y'(t) differentialekvationer

Vinkelenheter

När du använder trigonometriska funktioner ställer du in Angle-läget till den vinkelenhet (RADIAN eller DEGREE) som du vill ange och visa vinkelvärden i.

Kontrollera statusfältet

I statusfältet längst ned i fönstret visas vilket det aktuella grafläget och den aktuella vinkelenheten är.

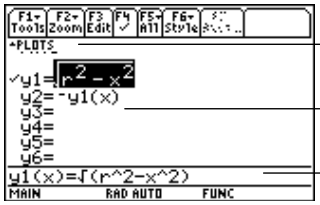
MAIN	RAD AUTO	FUNC
	Inställning för vinkelenhet	Grafläge

Definiera funktioner som ska plottas

I grafläget FUNCTION kan du plotta funktioner med namnen y1(x) till och med y99(x). Du definierar och redigerar dessa funktioner med hjälp av Y= Editor. (I Y= Editor visas funktionsnamnen för det aktuella grafläget. I exempelvis grafläget POLAR är funktionsnamnen r1(θ), r2(θ) osv.)

Definiera en ny funktion

1. Tryck på ◆[Y=] eller [APPS] 2 för att visa Y= Editor.



Plots — Du kan rulla ovanför y1= för att se en lista med statistiska plottningar. Se kapitel 16.

Funktionslistan — Du kan rulla genom listan med funktioner och definitioner.

Inmatningsraden — Här definierar och redigerar du den markerade funktionen i listan.

__Obs!__ I funktionslistan visas en förkortning av funktionsnamn, exempelvis y1, men på inmatningsraden visas hela namnet y1(x).

2. Tryck på ⊖ och ⊕ för att flytta markören till en odefinierad funktion. (Använd [2nd] ⊖ och [2nd] ⊕ för att bläddra en sida i taget.)

__Tips!__ För en odefinierad funktion behöver du inte trycka på [ENTER] eller [F3]. När du börjar skriva flyttar markören till inmatningsraden.

3. Tryck på [ENTER] eller [F3] för att flytta markören till inmatningsraden.
4. Skriv uttrycket för att definiera funktionen.
   - Den oberoende variabeln i plottning av funktioner är x.
   - Uttrycket kan referera till andra variabler, bland annat matriser, listor och andra funktioner. Endast flyttal eller listor med flyttal kan användas vid plottning.

__Tips!__ Om du av misstag råkar flytta markören till inmatningsraden trycker du på [ESC] för att flytta tillbaka den till funktionslistan.

5. När du är klar med uttrycket trycker du på [ENTER].

I funktionslistan visas nu den nya funktionen, som automatiskt markeras för plottning.

Redigera en funktion

Gör följande från Y= Editor:

1. Tryck på ⊖ och ⊕ för att markera funktionen.
2. Tryck på [ENTER] eller [F3] för att flytta markören till inmatningsraden.
3. Gör något av följande.
   - Använd ⊕ och ⊖ för att flytta markören i uttrycket så att du kan redigera det. Se "Redigera ett uttryck på inmatningsraden" i kapitel 2.

     — eller —
   - Tryck på [CLEAR] en eller två gånger för att ta bort det gamla uttrycket och skriv sedan det nya.

__Tips!__ Du avbryter eventuella ändringar genom att trycka på [ESC] i stället för [ENTER].

4. Tryck på [ENTER].

I funktionslistan visas nu den redigerade funktionen, som automatiskt markeras för plottning.

Ta bort en funktion

Gör följande från Y= Editor:

Om du vill ta bort:	Gör du följande:
En funktion från funktionslistan	Markera funktionen och tryck på [←] eller [CLEAR].
En funktion från inmatningsraden	Tryck på [CLEAR] en eller två gånger (beroende på markörens position) och tryck sedan på [ENTER].
Alla funktioner	Tryck på [F1] och välj sedan 8:Clear Functions. När du blir uppmanad att bekräfta trycker du på [ENTER].

Obs! *[F1] 8 tar inte bort eventuella statistiska plottningar (kapitel 16).*

Du behöver inte ta bort en funktion för att förhindra att den plottas. Enligt beskrivningen på sidan 111 kan du välja de funktioner som du vill plotta.

Genvägar för att flytta markören

I Y=Editor:

Tryck på:	När du vill:
[◆] [⊕] eller [◆] [⊖]	Gå till funktion 1 respektive den senast definierade funktionen. Om markören är på eller efter den senast definierade funktionen flyttar [◆] [⊖] markören till funktion 99.

Från grundfönstret eller ett program

Du kan också definiera och plotta en funktion från grundfönstret eller ett program.

- Använd kommandona **Define** och **Graph**. Läs följande:
  - "Plotta en funktion som är definierad i grundfönstret" och "Plotta en funktion som består av flera delar" i kapitel 12.
  - "Översikt över att mata in en funktion" i kapitel 17.
- Spara ett uttryck direkt till en funktionsvariabel. Läs följande:
  - "Spara och hämta variabelvärden" i kapitel 2.
  - "Skapa och beräkna användardefinierade funktioner" i kapitel 5.

Tips! *Användardefinierade funktioner kan ha nästan vilket namn som helst. Om du vill att de ska visas i Y= Editor måste du dock använda funktionsnamnen y1(x), y2(x) osv.*

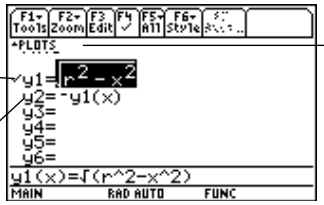
Välja vilka funktioner som ska plottas

Oavsett hur många funktioner som är definierade i Y= Editor kan du själv välja vilka du vill plotta.

Markera och avmarkera funktioner

Tryck på [◆][Y=] eller [APPS] 2 för att visa Y= Editor.

Symbolen "✓" anger vilka funktioner som ska plottas nästa gång du visar graffönstret.



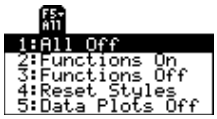
Markerad

Avmarkerad

Om PLOT-nummer visas, är de statistiska plottningarna markerade.

I detta exempel är Plots 1 och 2 markerade. Du visar dem genom att rulla ovanför y1=.

Tips! Du behöver inte markera en funktion när du anger eller redigerar den. Den markeras automatiskt.

Om du vill markera eller avmarkera:	Gör du följande:
En angiven funktion	1. Flytta markören för att markera funktionen. 2. Tryck på [F4]. Denna procedur markerar en avmarkerad funktion eller avmarkerar en markerad funktion.
Alla funktioner	1. Tryck på [F5] för att visa menyn **All**. 2. Välj ett menyalternativ. 

Tips! Du stänger av eventuella statistiska plottningar genom att trycka på [F5] 5 eller använda [F4] för att avmarkera dem.

Från grundfönstret eller ett program

Du kan också markera eller avmarkera funktioner från grundfönstret eller ett program.

- Använd kommandona **FnOn** och **FnOff** (tillgängliga på menyn [F4] Other i grundfönstret) för funktioner. Se appendix A.
- Använd kommandona **PlotsOn** och **PlotsOff** för statistiska plottningar. Se appendix A.

Ange visningsformat för funktioner

För varje definierad funktion kan du välja ett format som anger hur funktionen ska plottas. Detta är användbart när du plottar flera funktioner. Ange exempelvis en funktion som en heldragen kurva, en annan som en prickad kurva osv.

Visa eller ändra formatet för en funktion

Gör följande från Y= Editor:

1. Flytta markören för att markera en funktion.
2. Välj menyn Style:
   TI-89: Tryck på [2nd] [F6].
   TI-92 Plus: Tryck på [F6].

- Även om alternativet Line är markerat från början anges det aktuella formatet för funktionen av symbolen ✓.
- Du stänger menyn utan att göra några ändringar genom att trycka på [ESC].

3. Du gör en ändring genom att välja ett format.

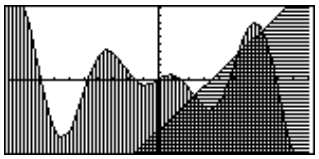
__Tips!__ Om du vill ange Line *som format för alla funktioner trycker du på* [F5] *och väljer* 4:Reset Styles.

Format	Beskrivning
Line	Kopplar ihop plottade punkter med en linje. Detta är standardinställningen.
Dot	Visar en punkt vid varje plottad punkt.
Square	Visar en fylld ruta för varje plottad punkt.
Thick	Kopplar ihop ritade punkter med en tjock linje.
Animate	En rund markör flyttas längs grafen men lämnar *inte* något spår.
Path	En rund markör flyttas längs grafen och *lämnar* ett spår.
Above	Skuggar området ovanför grafen.
Below	Skuggar området under grafen.

Om du använder skuggning ovanför eller under grafen

I TI-89 / TI-92 Plus finns fyra skuggmönster som används på ett roterande sätt. Om du anger att en funktion ska skuggas, används det första mönstret. Den andra skuggade funktionen använder det andra mönstret osv. Den femte skuggade funktionen använder det första mönstret igen.

När skuggade ytor korsar varandra överlappas deras mönster.



Från grundfönstret eller ett program

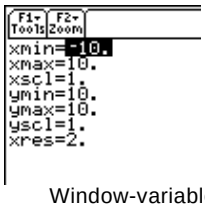
Du kan också ange format för en funktion från grundfönstret eller från ett program. Se kommandot **Style** i appendix A.

Definiera visningsfönstret

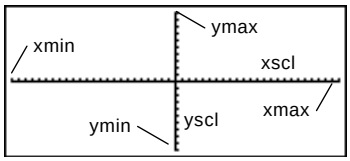
Visningsfönstret motsvarar den del av koordinatplanet som visas i graffönstret. Genom att mata in Window-variabler kan du definiera visningsfönstrets gränser och andra attribut. Funktionsgrafer, parametergrafer osv har sina egna oberoende uppsättningar Window-variabler.

Visa Window-variabler i Window Editor

Tryck på ◆ [WINDOW] eller [APPS] 3 för att visa Window Editor.



Window-variabler
(visas i Window Editor)



Motsvarande visningsfönster
(visas i graffönstret)

Variabel	Beskrivning
xmin, xmax, ymin, ymax	Gränser för visningsfönstret.
xscl, yscl	Avstånd mellan skalstreck på x- och y-axlarna.
xres	Anger bildpunktsupplösningen (1 till och med 10) för funktionsgrafer. Standardinställningen är 2. • Vid 1 beräknas och plottas funktionerna vid varje bildpunkt längs x-axeln. • Vid 10 beräknas och plottas funktionerna vid var tionde bildpunkt längs x-axeln.

Tips! *Du stänger av skalstreck genom att ange xscl=0 och/eller yscl=0.*

Tips! *Små värden av xres förbättrar grafens upplösning men kan öka den tid det tar att rita grafer.*

Ändra värden

Gör följande från Window Editor:

1. Flytta markören för att markera värdet du vill ändra.
2. Gör något av följande:
   - Skriv ett värde eller ett uttryck. Det gamla värdet tas bort när du börjar skriva.
     – eller –
   - Tryck på [CLEAR] för att ta bort det gamla värdet och skriv sedan det nya.
     – eller –
   - Tryck på ◀ eller ▶ för att ta bort markeringen och redigera sedan värdet.

Obs! *Om du skriver ett uttryck beräknas det när du flyttar markören till en annan Window-variabel eller avslutar Window Editor.*

Värdet sparas när du skriver det, du behöver inte trycka på [ENTER]. Med tangenten [ENTER] flyttar du bara markören till nästa Window-variabel.

Från grundfönstret eller ett program

Du kan också spara värden direkt till Window-variabler från grundfönstret eller ett program. Se “Spara och hämta variabelvärden” i kapitel 2.

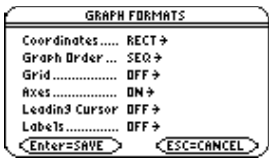
Ändra grafformatet

Du kan ställa in grafformatet så att referenselement, exempelvis axlar, stödraster och markörens koordinater, visas eller döljs. Funktionsgrafer, parametergrafer osv har sina egna oberoende uppsättningar för grafformat.

Visa inställningar för grafformat

Från Y= Editor, Window Editor eller graffönstret trycker du på [F1] och väljer 9:Format.

Tips! Du kan även öppna dialogrutan GRAPH FORMATS från Y= Editor, Window Editor och Graph fönstret. Tryck på:
TI-89: [◆] [I]
TI-92 Plus: [◆] F



- I dialogrutan GRAPH FORMATS visas de aktuella inställningarna.
- Avsluta utan att göra några ändringar genom att trycka på [ESC].

Tips! Du stänger av skalstreck genom att definiera visningsfönstret så att xscl och/eller yscl = 0.

Format	Beskrivning
Coordinates	Visar markörkoordinater i rektangulärt (RECT) eller polärt (POLAR) format, eller döljer (OFF) koordinaterna.
Graph Order	Plottar funktioner en i taget (SEQ) eller alla på samma gång (SIMUL).
Grid	Visar (ON) eller döljer (OFF) stödpunkter som motsvarar skalstrecken på axlarna.
Axes	Visar (ON) eller döljer (OFF) x- och y-axlarna.
Leading Cursor	Visar (ON) eller döljer (OFF) en referensmarkör som spårar funktionerna när de plottas.
Labels	Visar (ON) eller döljer (OFF) etiketter för x- och y-axlarna, förutsatt att axlarna visas.

Ändra inställningar

Gör följande från dialogrutan GRAPH FORMATS:

1. Flytta markören för att markera formatinställningen.
2. Tryck på ⊙ för att visa en meny med giltiga inställningar för det formatet.
3. Välj en inställning. Gör något av följande:
   - Flytta markören för att markera inställningen och tryck på [ENTER].

     – eller –
   - Tryck på siffran som motsvarar den inställningen.
4. När du har ändrat formatinställningarna trycker du på [ENTER] för att spara ändringarna och stänga dialogrutan GRAPH FORMATS.

Tips! Du stänger en meny eller dialogruta utan att göra några ändringar genom att trycka på [ESC] i stället för [ENTER].

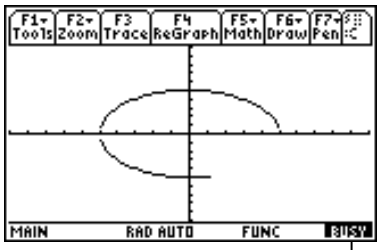
Plotta de markerade funktionerna

När du vill börja plotta de markerade funktionerna visar du graffönstret. Detta fönster använder det visningsformat och visningsfönster som du tidigare definierade.

Visa graffönstret

Tryck på [◆][GRAPH] eller [APPS] 4. De markerade funktionerna plottas automatiskt.

__Obs!__ Om du väljer en [F2] Zoom-åtgärd från Y= Editor eller Window Editor visas automatiskt Graph fönstret.



BUSY-indikatorn visas när plottning pågår.

Avbryta plottning

När plottning pågår:

- Du stoppar plottningen tillfälligt genom att trycka på [ENTER]. (Indikatorn PAUSE ersätter BUSY.) Du fortsätter genom att trycka på [ENTER] igen.
- Du avbryter plottning genom att trycka på [ON]. Du startar om plottningen från början genom att trycka på [F4] (ReGraph).

Ändra visningsfönstret

Beroende på olika inställningar kan en funktion plottas så att den exempelvis är för liten, för stor eller för mycket åt ena sidan i fönstret. Så här rättar du till detta:

- Omdefiniera visningsfönstret med andra gränser (sidan 113).
- Använd en Zoom-åtgärd (sidan 119).

Smart Graph

När du visar graffönstret visar funktionen Smart Graph omedelbart innehållet i föregående fönster, förutsatt att inget har ändrats som kräver omritning av grafen.

Funktionen Smart Graph uppdaterar fönstret och ritar bara om grafen om du har:

- Ändrat en inställning för läge som påverkar plottning, grafattribut för en funktion, en Window-variabel eller ett grafformat.
- Markerat eller avmarkerat en funktion eller statistisk plottning. (Om du bara markerar en ny funktion lägger funktionen Smart Graph till den funktionen i graffönstret.)
- Ändrat en definition för en markerad funktion eller värdet för en variabel i en markerad funktion.
- Tagit bort ett ritat objekt (kapitel 12).
- Ändrat definitionen för en statistisk plottning (kapitel 16).

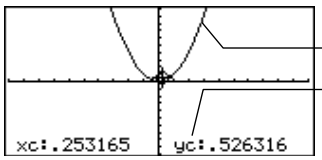
Visa koordinater med den rörliga markören

Om du vill visa koordinaterna för en punkt i graffönstret använder du den rörliga markören. Du kan flytta markören till vilken bildpunkt som helst i fönstret; markören är inte begränsad till en plottad funktion.

Rörlig markör

När du först visar graffönstret visas ingen markör. Tryck på någon av piltangenterna för att visa markören. Markören flyttas från fönstrets mittpunkt och dess koordinater visas.

Tips! *Om fönstret inte visar några koordinater ställer du in grafformatet (◆ I) så att Coordinates = RECT eller POLAR. Tryck på:*
TI-89: ◆ I
TI-92 Plus: ◆ F



y1(x)=x²

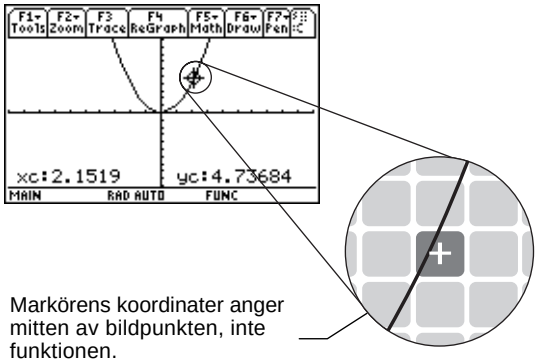
Bokstaven "c" anger att detta är markörens koordinater. Värdena sparas i system-variablerna xc och yc.

Rektangulära koordinater använder xc och yc. Polära koordinater använder rc och θc.

Tips! *Om du tillfälligt vill dölja markören och dess koordinater trycker du på* CLEAR, ESC *eller* ENTER. *Nästa gång du flyttar markören flyttas den från dess föregående position.*

Om du vill flytta den rörliga markören:	Trycker du på:
Till en angränsande bildpunkt	En piltangent i någon riktning.
I steg om 10 bildpunkter	2nd och sedan på markörknappen.

När du flyttar markören till en bildpunkt som verkar ligga "på" funktionskurvan, kan den ligga nära funktionen men inte på den.



Markörens koordinater anger mitten av bildpunkten, inte funktionen.

Så här ökar du noggrannheten:

- Använd verktyget **Trace** (beskrivs på nästa sida) för att visa koordinater som ligger på funktionskurvan.
- Använd en Zoom-åtgärd för att zooma in på en del av grafen.

Spåra en funktion

Om du vill visa de exakta koordinaterna för en ritad punkt på en plottad funktion använder du verktyget F3 **Trace**. Till skillnad från den rörliga markören rör sig spårningsmarkören endast längs en funktions plottade punkter.

Starta en spårning

Från graffönstret trycker du på F3.

Spårningsmarkören visas på funktionen vid det mittersta x-värdet i fönstret. Markörens koordinater visas längst ned i fönstret

Obs! *Om du plottar statistiska data (kapitel 16), visas spårningsmarkören på den lägst numrerade statistiska plottningen.*

Om flera funktioner plottas visas spårningsmarkören på den lägst numrerade funktionen som är markerad i Y= Editor. Funktionsnumret visas i den övre högra delen av fönstret.

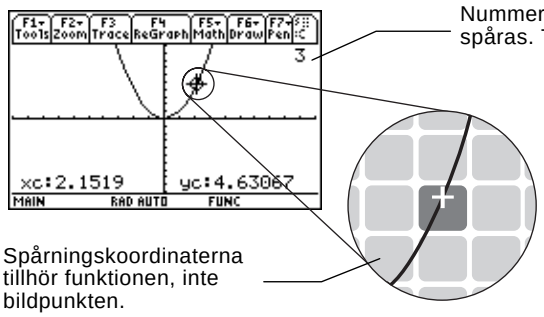
Flytta längs en funktion

Om du vill flytta spårningsmarkören:	Gör du följande:
Till föregående eller nästa plottade punkt	Tryck på ◀ eller ▶.
Ca 5 plottade punkter (kan vara mer eller mindre än 5, beroende på Windows-variabelns xres-värde)	Tryck på 2nd ◀ eller 2nd ▶.
Till en angivet x-värde på funktionen	Skriv x-värdet och tryck på ENTER.

Obs! *Om du anger ett x-värde måste det vara mellan xmin och xmax.*

Spårningsmarkören flyttar bara från plottad punkt till plottad punkt längs funktionen, inte från bildpunkt till bildpunkt.

Tips! *Om det inte visas några koordinater i fönstret ställer du in grafformatet så att Coordinates = RECT eller POLAR. Tryck på:*
TI-89: ◆ I
TI-92 Plus: ◆ F



Nummer på funktionen som spåras. Till exempel: y1(x).

Spårningskoordinaterna tillhör funktionen, inte bildpunkten.

Varje visat y-värde beräknas från x-värdet, det vill säga y=y*n*(x). Om funktionen är odefinierad för ett x-värde är y-värdet tomt.

Tips! *Använd QuickCenter (beskrivs på nästa sida) för att spåra en funktion som går ovanför eller nedanför fönstret.*

Du kan fortsätta att spåra en funktion som går ovanför eller nedanför visningsfönstret. Du kan inte se markören när den går utanför fönstret, men koordinatvärdet visar dess korrekta koordinater.

Flytta från funktion till funktion

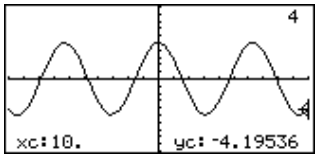
Tryck på ⊕ eller ⊖ för att flytta till föregående eller nästa markerade funktion vid samma x-värde. Det nya funktionsnumret visas på skärmen.

Föregående eller nästa funktion baseras på ordningsföljden av de markerade funktionerna i Y= Editor, inte så som funktionerna visas när de är avbildade som grafer i fönstret.

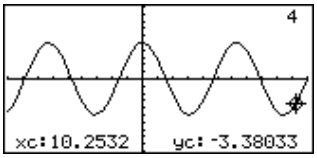
Automatisk panorering

Om du spårar en funktion utanför vänstra eller högra kanten av fönstret panoreras automatiskt fönstret till vänster eller höger. En kort fördröjning uppstår när den nya delen av grafen plottas.

Obs! *Automatisk panorering fungerar inte om statistisk plottning visas eller om en funktion använder ett skuggformat.*



Före automatisk panorering



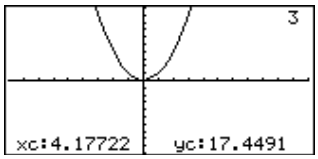
Efter automatisk panorering

Efter en automatisk panorering fortsätter markören spårningen.

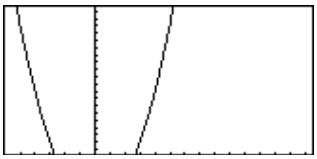
Använda QuickCenter

Om du spårar en funktion utanför över- eller underdelen av fönstret kan du trycka på [ENTER] för att centrera visningsfönstret vid markören.

Tips! *Du kan använda QuickCenter när som helst under en spårning, även när markören fortfarande visas i fönstret.*



Innan du använder QuickCenter



Efter att du har använt QuickCenter

Efter att du har använt QuickCenter avslutar markören spårningen. Om du vill fortsätta spåra trycker du på [F3].

Avbryta spårning

Du kan när som helst avbryta en spårning genom att trycka på [ESC].

En spårning avbryts även när du visar ett annat programfönster, exempelvis Y= Editor. Återgå till graffönstret och tryck på [F3] för att börja spåra:

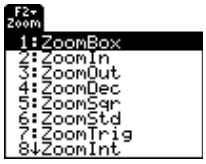
- Om funktionen Smart Graph ritade om fönstret visas markören vid det mellersta x-värdet.
- Om funktionen Smart Graph *inte* ritar om fönstret visas markören vid sin föregående position (innan du visade det andra programmet).

På menyn F2 **Zoom** finns flera verktyg som du kan använda för att justera visningsfönstret. Du kan också spara ett visningsfönster för senare användning.

Översikt över menyn Zoom

Obs! *Om du väljer ett Zoom verktyg från Y= Editor eller Window Editor visas automatiskt Graph fönstret.*

Tryck på F2 från Y= Editor, Window Editor eller graffönstret.



Instruktioner för hur du använder ZoomBox, ZoomIn, ZoomOut, ZoomStd, Memory och SetFactors finns senare i detta avsnitt.

Mer information om de övriga objekten finns i appendix A.

Obs! *Δx och Δy är avstånden från mittpunkten av en bildpunkt till mittpunkten av en intilliggande bildpunkt.*

Zoomverktyg	Beskrivning
ZoomBox	Ritar en ruta och zoomar in på den.
ZoomIn, ZoomOut	Markera en punkt och zoomar in och ut med värdet definierat i SetFactors.
ZoomDec	Anger Δx och Δy till 0,1 och centrerar origo.
ZoomSqr	Justerar Window-variabler så att en kvadrat eller cirkel visas i korrekta proportioner (i stället för en rektangel eller ellips).
ZoomStd	Ställer in Window-variabler till sina standardvärden. xmin = ¯10 ymin = ¯10 xres = 2 xmax = 10 ymax = 10 xscl = 1 yscl = 1
ZoomTrig	Ställer in Window-variabler till förinställda värden som ofta passar för att plotta trigonometriska funktioner. Centrerar origo och ställer in: Δx = π/24 (.130899... radianer eller 7.5 grader) ymin = ¯4 ymax = 4 xscl = π/2 (1.570796... radianer eller 90 grader) yscl = 0.5
ZoomInt	Markerar en ny mittpunkt och ställer sedan in Δx och Δy till 1 och ställer in xscl och yscl till 10.
ZoomData	Justerar Window-variabler så att alla markerade statistiska plottningar visas. Se kapitel 16.
ZoomFit	Justerar visningsfönstret så att hela intervallet av beroende variabelvärden för de markerade funktionerna visas. När funktioner plottas bevaras de aktuella xmin och xmax, och ymin och ymax justeras.
Memory	Sparar och hämtar inställningar för Window-variabler så att du kan återskapa ett anpassat visningsfönster.
SetFactors	Anger Zoom faktorer för ZoomIn och ZoomOut.

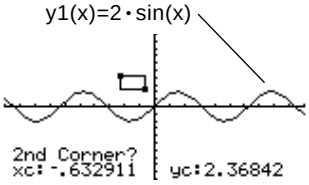
Zooma in med en zoomruta

1. Välj 1:ZoomBox på menyn [F2] Zoom.

   Ett meddelande frågar efter det 1st corner?

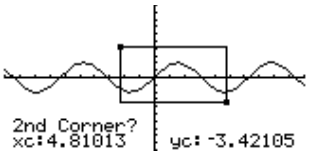
2. Flytta markören till ett hörn av rutan som du vill definiera och tryck på [ENTER].

   Markören ändras till en fyrkant och ett meddelande frågar efter det 2nd corner?



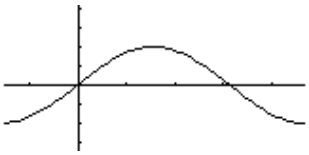
3. Flytta markören till det motsatta hörnet av zoomrutan.

   När du flyttar markören sträcks rutan ut.



4. När du har gjort en kontur av det område som du vill zooma in på trycker du på [ENTER].

   I graffönstret visas det inzoomade området.



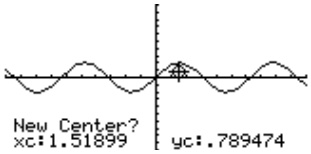
Tips! *Om du vill flytta markören i större steg använder du [2nd] ▶, [2nd] ▼, osv.*

Tips! *Du kan avbryta ZoomBox genom att trycka på [ESC] innan du trycker på [ENTER].*

Zooma in och ut på en punkt

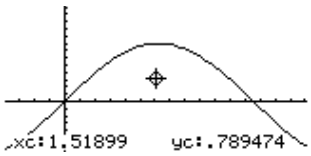
1. Välj 2:ZoomIn eller 3:ZoomOut på menyn [F2] Zoom.

   En markör visas och ett meddelande frågar efter New Center?



2. Flytta markören till punkten där du vill zooma in eller ut och tryck på [ENTER].

   Window-variablerna justeras efter de zoomfaktorer som är angivna i SetFactors.



- För ZoomIn delas x-variablerna med xFact och y-variablerna med yFact.

  $$\text{nytt xmin} = \frac{\text{xmin}}{\text{xFact}} \text{ osv.}$$

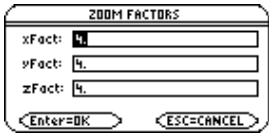
- För ZoomOut multipliceras variablerna med xFact och y-variablerna med yFact.

  nytt xmin = xmin ∗ xFact osv.

Ändra zoomfaktor

Zoomfaktorerna anger den förstoring och förminskning som används av ZoomIn och ZoomOut.

1. Välj C:SetFactors på menyn F2 Zoom för att visa dialogrutan ZOOM FACTORS.



Zoom faktorer måste vara ≥ 1, men de behöver inte vara heltal. Standardinställningen är 4.

Tips! Om du vill avsluta utan att spara ändringar trycker du på ESC.

2. Använd ⊖ och ⊕ för att markera värdet som du vill ändra. Gör sedan följande:
   - Skriv det nya värdet. Det gamla värdet tas automatiskt bort när du börjar skriva.

     – eller –
   - Tryck på ◐ eller ◑ för att ta bort markeringen och redigera sedan det gamla värdet.
3. Tryck på ENTER (när du har skrivit i en inmatningsruta måste du trycka på ENTER två gånger) för att spara ändringarna och stänga dialogrutan.

Spara eller hämta ett visningsfönster

Efter att ha använt olika Zoom verktyg kanske du vill återgå till ett föregående visningsfönster eller spara det aktuella.

1. Välj B:Memory på menyn F2 Zoom för att visa dess undermeny.

2. Välj objektet.

Välj:	Om du vill göra följande:
1:ZoomPrev	Återgå till det visningsfönster som visades innan föregående zoomning.
2:ZoomSto	Spara det aktuella visningsfönstret. (De aktuella Window-variabelvärdena sparas till systemvariablerna zxmin, zxmax osv.)
3:ZoomRcl	Hämtar det visningsfönster som senast sparades med ZoomSto.

Obs! Du kan bara spara en uppsättning Window-variabelvärden i taget. När du sparar en ny uppsättning skrivs den gamla uppsättningen över.

Återställa standard-visningsfönstret

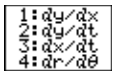
Du kan återställa Window-variabler till sina standardvärden när som helst.

Välj 6:ZoomStd på menyn F2 Zoom.

Menyn F5 **Math** finns i graffönstret och innehåller flera verktyg som du kan använda för att analysera plottade funktioner.

Översikt över menyn Math

Tryck på F5 från graffönstret.



På undermenyn Derivatives, är endast dy/dx tillgängligt för plottning av funktioner. De andra derivatorna är tillgängliga i andra graflägen (parameterform, polär form osv).

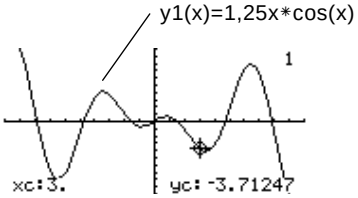
__Obs!__ När det gäller beräkningsresultat sparas markörkoordinater i systemvariablerna xc och yc (rc och θc om du använder polära koordinater). Derivator, integraler, avstånd osv sparas i systemvariabeln sysMath.

Matematiskt verktyg	Beskrivning
Value	Beräknar för en markerad y(x)-funktion y-värdet för ett givet x-värde.
Zero, Minimum, Maximum	Beräknar ett nollställe, minimum eller maximum.
Intersection	Beräknar koordinaterna för skärningspunkten mellan två funktioner.
Derivatives	Beräknar derivatans värde (lutningen) i en punkt.
∫f(x)dx	Beräknar den ungefärliga numeriska integralens värde över ett intervall.
Inflection	Beräknar koordinaterna för inflexionspunkten på en kurva. Andraderivatan ändrar där tecken (kurvan ändras från att vara konkav till konvex eller tvärtom).
Distance	Ritar linje och mäter avståndet mellan två punkter på samma funktion eller på två olika funktioner.
Tangent	Ritar en tangent till en punkt och visar tangentens ekvation.
Arc	Beräknar båglängden mellan två punkter längs en kurva.
Shade	Beroende på hur många funktioner som plottas. • Om endast en funktion plottas, skuggas funktionens område över eller under x-axeln. • Om två eller fler funktioner plottas, skuggas området mellan två funktioner inom ett intervall.

Beräkna y(x) vid en angiven punkt

1. Tryck på [F5] och välj 1:Value från graffönstret
2. Skriv x-värdet som måste vara ett verkligt värde mellan xmin och xmax. Värdet kan vara ett uttryck.
3. Tryck på [ENTER].

   Markören flyttar till det x-värde på den första funktionen som är markerad i Y= Editor. Funktionens koordinater visas.



4. Tryck på ⊖ eller ⊕ för att flytta markören mellan funktioner vid det angivna x-värdet. Motsvarande y-värde visas.

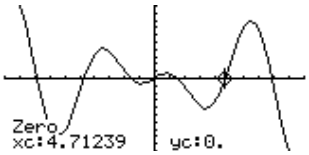
   Obs! Om du trycker på ⊖ eller ⊕ visas den rörliga markören. Du kanske inte kan flytta tillbaka den till det angivna x-värdet.

Tips! *Du kan också visa funktionskoordinater genom att spåra funktionen ([F3]), skriva ett x-värde och trycka på [ENTER].*

Beräkna ett nollställe, minimum eller maximum inom ett intervall

1. Tryck på [F5] och välj 2:Zero, 3:Minimum eller 4:Maximum från graffönstret.
2. Använd ⊖ och ⊕ för att välja lämplig funktion.
3. Ange den undre gränsen för x. Använd antingen ⊖ eller ⊕ för att flytta markören till den lägsta gränsen eller skriv dess x-värde.
4. Tryck på [ENTER]. Symbolen ▸ överst i fönstret anger den undre gränsen.
5. Ange den övre gränsen och tryck på [ENTER].

   Markören flyttas till lösningen och dess koordinater visas.

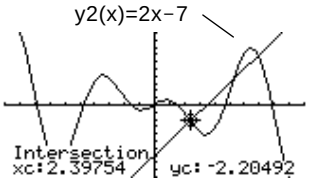


Tips!: *Att skriva x-värden är ett snabbt sätt att ange gränser.*

Beräknar koordinaterna för skärningspunkten mellan två funktioner inom ett intervall

1. Tryck på [F5] och välj 5:Intersection från graffönstret.
2. Välj den första funktionen med hjälp av ⊖ och ⊕ och tryck på [ENTER]. Markören flyttar till nästa plottade funktion.
3. Välj den andra funktionen och tryck på [ENTER].
4. Ange den undre gränsen för x. Använd ⊖ och ⊕ för att flytta markören till den undre gränsen eller skriv dess x-värde.
5. Tryck på [ENTER]. Symbolen ▸ överst i fönstret anger den undre gränsen.
6. Ange den övre gränsen och tryck på [ENTER].

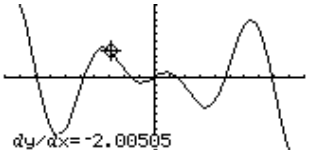
   Markören flyttas till skärningspunkten och dess koordinater visas.



Beräkna derivatans värde i en punkt (lutningen)

1. Tryck på [F5] och välj 6:Derivatives från graffönstret. Välj sedan 1:dy/dx på undermenyn.
2. Använd ⊖ och ⊕ för att välja en funktion.
3. Ange den punkt vars derivata ska beräknas. Flytta markören till punkten eller skriv dess x-värde.
4. Tryck på [ENTER].

   Derivatans värde i punkten visas.



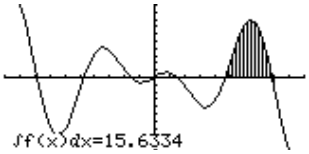
Beräkna den numeriska integralens värde över ett intervall

__Tips!__ Att skriva x-värden är ett snabbt sätt att ange gränser.

1. Tryck på [F5] och välj 7:∫f(x)dx från graffönstret.
2. Använd ⊖ och ⊕ för att välja en funktion.
3. Ange den undre gränsen för x. Använd ⊖ och ⊕ för att flytta markören till den undre gränsen eller skriv dess x-värde.
4. Tryck på [ENTER]. Symbolen ▸ överst i fönstret anger den undre gränsen.
5. Ange den övre gränsen och tryck på [ENTER].

   Intervallet skuggas och intergralens ungefärliga numeriska värde visas.

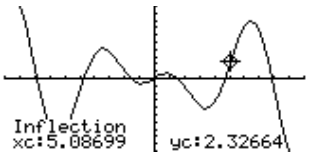
__Tips!__ Du tar bort det skuggade området genom att trycka på [F4] (ReGraph).



Beräknar koordinaterna för en inflexionspunkt inom ett intervall

1. Tryck på [F5] och välj 8:Inflection från graffönstret.
2. Använd ⊖ och ⊕ för att välja lämplig funktion.
3. Ange den undre gränsen för x. Använd ⊖ och ⊕ för att flytta markören till den undre gränsen eller skriv dess x-värde.
4. Tryck på [ENTER]. Symbolen ▸ överst i fönstret anger den undre gränsen.
5. Ange den övre gränsen och tryck på [ENTER].

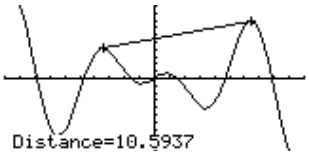
   Markören flyttar till inflexionspunkten (om det finns någon) inom intervallet, och koordinaterna visas.



Beräkna avståndet mellan två punkter

1. Tryck på [F5] och välj 9:Distance från graffönstret.
2. Använd ⊖ och ⊕ för att välja funktionen för den första punkten.
3. Ställ in den första punkten. Använd ◐ och ◑ för att flytta markören till punkten eller skriv dess x-värde.
4. Tryck på [ENTER]. Symbolen + markerar punkten.
5. Om den andra punkten är på en annan funktion använder du ⊖ och ⊕ för att välja den funktionen.
6. Ställ in den andra punkten. (Om du använder markören för att ange punkten ritas en linje när du flyttar markören.)
7. Tryck på [ENTER].

   Avståndet mellan de två punkterna visas tillsammans med linjen mellan dem.

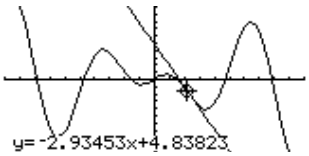


Rita en tangent

__Tips!__ Du kan ta bort en ritad tangent genom att trycka på [F4] (ReGraph).

1. Tryck på [F5] och välj A:Tangent från graffönstret.
2. Använd ⊖ och ⊕ för att välja lämplig funktion.
3. Ställ in tangeringspunkten. Flytta markören till punkten eller skriv dess x-värde.
4. Tryck på [ENTER].

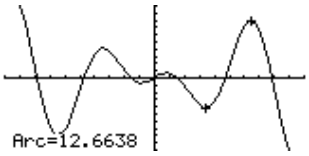
   Tangenten ritas och dess ekvation visas.



Beräkna en båglängd

1. Tryck på [F5] och välj B:Arc från graffönstret.
2. Använd ⊖ och ⊕ för att välja lämplig funktion.
3. Ställ in bågens första punkt. Använd ◐ och ◑ för att flytta markören eller skriv x-värdet.
4. Tryck på [ENTER]. Symbolen + markerar den första punkten.
5. Ställ in den andra punkten och tryck på [ENTER].

   Symbolen + markerar den andra punkten och båglängden visas.

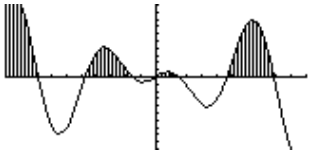


Skugga området mellan en funktion och x-axeln

Du kan bara ha en plottad funktion. Om du plottar två eller fler funktioner skuggas området mellan två funktioner.

1. Tryck på [F5] och välj C:Shade från graffönstret. Ett meddelande frågar efter Above X axis?
2. Välj ett av följande. Så här skuggar du funktionens område:
   - Ovanför x-axeln. Tryck på [ENTER].
   - Under x-axeln. Tryck på:
     TI-89: [alpha] [N]
     TI-92 Plus: N
3. Ange den undre gränsen för x. Använd ◀ och ▶ för att flytta markören till den understa gränsen eller skriv x-värdet.
4. Tryck på [ENTER]. Symbolen ▶ överst i fönstret anger den undre gränsen.
5. Ange den övre gränsen och tryck på [ENTER].

   Det inneslutna området skuggas.



Obs! *Om du inte trycker på ◀ eller ▶, eller om du anger ett x-värde då du ställer in gränsvärden, kommer xmin respektive xmax att användas som gränsvärden.*

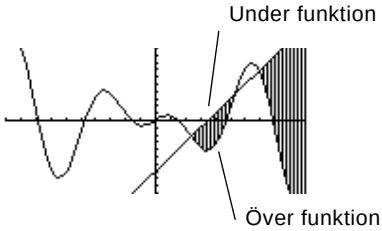
Tips! *Du tar bort det skuggade området genom att trycka på [F4] (ReGraph).*

Skugga området mellan två funktioner inom ett intervall

Minst två funktioner måste vara plottade. Om du plottar endast en funktion skuggas området mellan funktionen och x-axeln.

1. Tryck på [F5] och välj C:Shade från graffönstret. Ett meddelande frågar efter Above?
2. Använd ▼ eller ▲ för att välja en funktion. (Skuggningen visas *ovanför* denna funktion.)
3. Tryck på [ENTER]. Markören flyttar till nästa plottade funktion och ett meddelande frågar efter Below?
4. Använd ▼ eller ▲ för att välja en annan funktion. (Skuggningen visas *nedanför* denna funktion.)
5. Tryck på [ENTER].
6. Ange den undre gränsen för x. Använd ◀ och ▶ för att flytta markören till den undre gränsen eller skriv x-värdet.
7. Tryck på [ENTER]. Symbolen ▶ överst i fönstret markerar den undre gränsen.
8. Ange den övre gränsen och tryck på [ENTER].

   Det inneslutna området skuggas.



Obs! *Om du inte trycker på ◀ eller ▶, eller om du anger ett x-värde då du ställer in gränsvärden, kommer xmin respektive xmax att användas som gränsvärden.*

Tips! *Du tar bort det skuggade området genom att trycka på [F4] (ReGraph).*

Plotta parameterkurvor

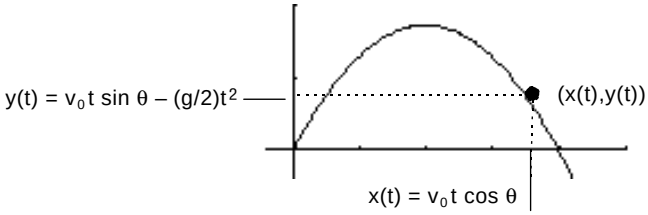
7

Förhandsvisning över plottning av kurvor i parameterform 128
Översikt över plottning av parameterekvationer 129
Skillnader mellan parameter- och funktionsplottning 130

I det här kapitlet beskrivs hur du plottar parameterekvationer på TI-89 / TI-92 Plus. Innan du använder det här kapitlet bör du känna till innehållet i kapitel 6: Grundläggande funktionsplottning.

Parameterekvationer består av en x- och en y-komponent som båda uttrycks som funktioner av samma oberoende variabel t.

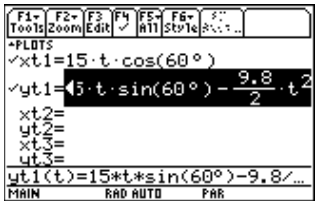
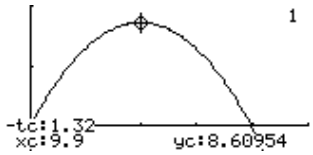
Du kan använda parameterekvationer för att plotta tidsberoende förlopp. Positionen hos en projektil har en horisontell (x) och en vertikal (y) komponent uttryckt som en funktion av tiden (t). Exempel:



Grafen visar banan för projektilen över tiden. Det förutsätts att endast tyngdaccelerationen (ingen dragande kraft m.m.) verkar på projektilen.

Förhandsvisning över plottning av kurvor i parameterform

Plotta den kurva i parameterform som beskriver positionen för en boll som sparkas i en vinkel (θ) på 60° med begynnelsehastigheten (v_0) 15 m/s. Tyngdaccelerationen är g = 9,8 m/s^2. Bortse från luftmotståndet. Vilken är den maximala höjden? När träffar bollen marken?

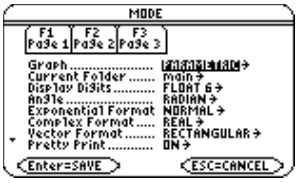
Steg	TI-89 Tangenter	TI-92 Plus Tangenter	Teckenfönster
1. Visa dialogrutan MODE. För grafläge väljer du PARAMETRIC.	[MODE] [▶] 2 [ENTER]	[MODE] [▶] 2 [ENTER]	
2. Visa och rensa Y= Editor. Definiera därefter den horisontella komponenten xt1(t) = v_0t cos θ. *Mata in värden för v_0 och θ.* **TI-89:** *Skriv* T [×] [2nd] [COS], *inte* T [2nd] [COS]. **TI-92 Plus:** *Skriv* T [×] [COS], *inte* T [COS]. *Mata in symbolen ° genom att skriva* [2nd][°] *eller* [2nd][MATH]2 1. *På så sätt tolkas ett tal som grader, oavsett vilken vinkelenhet som är vald.*	[◆] [Y=] [F1] 8 [ENTER] [ENTER] 1 5 T [×] [2nd] [COS] 6 0 [2nd] [°] [)] [ENTER]	[◆] [Y=] [F1] 8 [ENTER] [ENTER] 1 5 T [×] [COS] 6 0 [2nd] [°] [)] [ENTER]	xt1(t)=15t*cos(60°)
3. Definiera den vertikala komponenten yt1(t) = v_0t sin θ – (g/2)t^2. *Mata in värden för v_0,θ och g.*	[ENTER] 1 5 T [×] [2nd] [SIN] 6 0 [2nd] [°] [)] [−] [(] 9 . 8 [÷] 2 [)] T [^] 2 [ENTER]	[ENTER] 1 5 T [×] [SIN] 6 0 [2nd] [°] [)] [−] [(] 9 . 8 [÷] 2 [)] T [^] 2 [ENTER]	
4. Visa Window Editor. Mata in de lämpliga Window-variablerna för detta exempel. *Du kan trycka på* [▼] *eller* [ENTER] *för att mata in ett värde och flytta till nästa variabel.*	[◆] [WINDOW] 0 [▼] 3 [▼] . 0 2 [▼] [(-)] 2 [▼] 2 5 [▼] 5 [▼] [(-)] 2 [▼] 1 0 [▼] 5	[◆] [WINDOW] 0 [▼] 3 [▼] . 0 2 [▼] [(-)] 2 [▼] 2 5 [▼] 5 [▼] [(-)] 2 [▼] 1 0 [▼] 5	tmin=0. tmax=3. tstep=.02 xmin=-2. xmax=25. xscl=5. ymin=-2. ymax=10. yscl=5.
5. Plotta parameterekvationen för bollen.	[◆] [GRAPH]	[◆] [GRAPH]	
6. Välj Trace. Flytta sedan markören längs banan för att hitta: • y-värdet vid den maximala höjden. • t-värdet där bollen träffar marken.	[F3] [▶] eller [◀] om det behövs	[F3] [▶] eller [◀] om det behövs	

Översikt över plottning av parameterekvationer

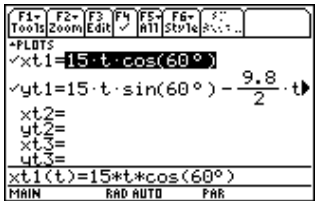
När du plottar parameterekvationer använder du samma allmänna steg som för y(x)-funktioner, enligt beskrivningen i kapitel 6: Grundläggande funktionsplottning. Eventuella skillnader som gäller för parameterekvationer beskrivs på följande sidor.

Plotta parameter-ekvationer

Ställ in grafläget (MODE) till PARAMETRIC. Ställ in vinkelenheten, om det behövs.



Definiera x- och y-komponenterna i Y= Editor (◆ [Y=]).



Tips! Stäng av alla statistik-plottningar (kapitel 16) genom att trycka på F5 *5 eller tryck på* F4 *för att avmarkera dem.*

Välj (F4) vilket markerar ekvationer som ska plottas. Markera x- eller y-komponenten, eller båda.

Tips! Detta är valfritt. Om du ska plotta flera funktioner är det lättare att skilja dem åt.

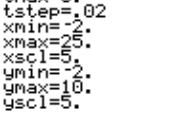
Ställ in visningsformatet för en ekvation. Du kan ställa in antingen x- eller y-komponenten.

TI-89: 2nd [F6]

TI-92 Plus: [F6]

Tips! F2 Zoom *ändrar också visningsfönstret.*

Definiera visnings-fönstret (◆ [WINDOW]).



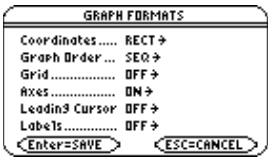
Ändra grafformatet om det behövs.

F1 9

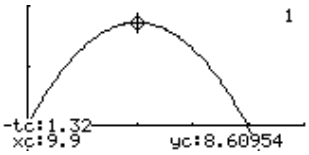
— eller —

TI-89: ◆ I

TI-92 Plus: ◆ F



Plotta de markerade ekvationerna (◆ [GRAPH]).



Utforska grafen

Du kan göra följande från graffönstret:

- Visa koordinaterna för en bildpunkt genom att använda den rörliga markören, eller för en plottad punkt genom att spåra en parameterekvation.
- Använd menyn F2 Zoom för att zooma in eller ut på en del av grafen.
- Använd menyn F5 Math för att hitta derivatan, tangenter osv. Vissa menyalternativ är inte tillgängliga för kurvor i parameterform.

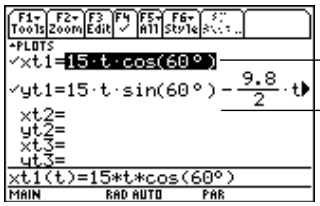
I det här kapitlet antas att du känner till hur du plottar y(x)-funktioner enligt beskrivningen i kapitel 6: Grundläggande funktionsplottning. I det här avsnittet beskrivs de skillnader som gäller för parameterekvationer.

Ställa in grafläget

Använd MODE för att ställa in Graph = PARAMETRIC innan du definierar ekvationer eller ställer in Window-variabler. Med Y= Editor och Window Editor kan du endast mata in information för det *aktuella* grafläget.

Definiera parameter-ekvationer i Y= Editor

För att kunna plotta en parameterekvation måste du definiera både dess x- och y-komponent. Om du endast definierar en komponent kan ekvationen inte plottas. (Du kan däremot använda enskilda komponenter för att generera en automatisk tabell enligt beskrivning i kapitel 13.)



Mata in x- och y-komponenter på separata rader.

Du kan definiera xt1(t) t.o.m xt99(t) och yt1(t) t.o.m. yt99(t).

Var försiktig när du använder implicit multiplikation med t. Exempel:

Obs! *När du använder t måste implicit multiplikation vara giltig i den rådande situationen.*

Ange:	I stället för:	Orsak:
t* cos(60)	tcos(60)	tcos tolkas som en användardefinierad funktion med namnet **tcos**, inte som en implicit multiplikation. I de flesta fall gäller detta obefintliga funktioner. Räknaren returnerar funktionsnamnet, inte ett tal.

Tips! *Du kan använda kommandot* ***Define*** *från grundfönstret (se appendix A) för att definiera funktioner och ekvationer för ett grafläge, oavsett vilket det aktuella läget är.*

Y= Editor underhåller en oberoende funktionslista för varje inställning av grafläget. Anta exempelvis följande:

- I grafläget FUNCTION definierar du en uppsättning y(x)-funktioner. Du kan byta till grafläget PARAMETRIC och definiera en uppsättning x- och y-komponenter.
- När du återgår till grafläget FUNCTION är y(x)-funktionerna fortfarande definierade i Y= Editor. När du återgår till grafläget PARAMETRIC är x- och y-komponenterna fortfarande definierade.

Välja parameter-ekvationer

Om du vill plotta en parameterekvation väljer du *antingen* dess x- eller y-komponent eller *båda*. När du anger eller redigerar en komponent väljs den automatiskt.

Det kan vara användbart att välja x- och y-komponenter separat, för exempelvis tabeller, som beskrivs i kapitel 13. Med flera parameterekvationer kan du välja och jämföra alla x-komponenter eller alla y-komponenter.

Välja visningsformat

Du kan ange format för antingen en x- eller y-komponent. Om du exempelvis ställer in x-komponenten till Dot ställs automatiskt y-komponenten in till Dot.

***Tips!** Använd formaten Animate och Path för intressanta rörliga projektileffekter.*

Formaten Above och Below är inte tillgängliga för parameter-ekvationer och är därför nedtonade på menyn Style i Y= Editor.

Window-variabler

Window Editor underhåller en oberoende uppsättning Window-variabler för varje inställning av grafläget (så som Y= Editor underhåller oberoende funktionslistor). För kurvor på parameter-form använder du följande Window-variabler.

***Obs!** Du kan använda negativt tstep. Om du gör det måste tmin vara större än tmax.*

Variabel	Beskrivning	
tmin, tmax	Minsta och största t-värde.	
tstep	Avståndet mellan succesiva t-värden. Parameterekvationer beräknas för:	
	x(tmin)	y(tmin)
	x(tmin+tstep)	y(tmin+tstep)
	x(tmin+2(tstep))	y(tmin+2(tstep))
	... får inte överskrida ...	... får inte överskrida...
	x(tmax)	y(tmax)
xmin, xmax, ymin, ymax	Gränser för visningsfönstret.	
xscl, yscl	Avstånd mellan skalstreck på x- och y-axeln.	

Standardvärdena (ställs in när du väljer 6:ZoomStd på menyn F2 Zoom) är:

tmin = 0,		xmin = ⁻10,	ymin = ⁻10,
tmax = 2π	(6,2831853... radianer eller 360 grader)	xmax = 10,	ymax = 10,
tstep =π/24	(,1308996... radianer eller 7,5 grader)	xscl = 1,	yscl = 1,

Du kanske måste ändra standardvärdena för t-variablerna (tmin, tmax, tstep) för att försäkra dig om att tillräckligt många punkter plottas.

Utforska en graf

Liksom med funktionsplottning kan du utforska en graf med följande verktyg.

Verktyg	För parameterkurvor:
Rörlig markör	Fungerar som med funktionsplottning.
F2 Zoom	Fungerar som med funktionsplottning, med följande undantag: • Endast x- (xmin, xmax, xscl) och y- (ymin, ymax, yscl) Window-variabler påverkas. • t-Window-variablerna (tmin, tmax, tstep) påverkas inte såvida du inte väljer 6:ZoomStd (vilket ställer in tmin = 0, tmax = 2π och tstep = π/24).
F3 Trace	Låter dig flytta markören längs grafen ett tsteg i taget. • När du påbörjar en spårning befinner sig markören på den först markerade parameterekvationen vid tmin. • QuickCenter gäller för alla riktningar. Om du flyttar markören ut från fönstret (längst upp/ned, till vänster/höger) trycker du på ENTER för att centrera visningsfönstret vid markörens position. • Automatisk panorering är inte tillgänglig. Om du flyttar markören ut från fönstrets vänstra eller högra sida panoreras inte visningsfönstret automatiskt. Du kan däremot använda QuickCenter.
F5 Math	Endast 1:Value, 6:Derivatives, 9:Distance, A:Tangent och B:Arc är tillgängliga för parameterekvationer. Dessa verktyg baseras på t-värden. Exempel: • 1:Value visar x- och y-värden för ett specifikt t-värde. • 6:Derivatives hittar dy/dx, dy/dt eller dx/dt vid en punkt definierad för ett specifikt t-värde.

Tips! *Under en spårning kan du även beräkna x(t) och y(t) genom att skriva t-värdet och trycka på* ENTER.

Tips! *Du kan använda QuickCenter när som helst under en spårning, även om markören fortfarande befinner sig i fönstret.*

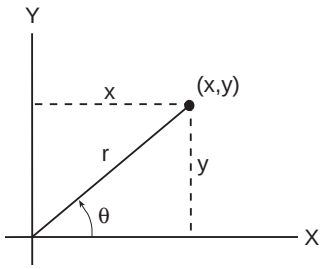
Plotta polära ekvationer

8

Förhandsvisning över polär plottning 134
Översikt över stegen i att plotta polära ekvationer 135
Skillnader mellan polär- och funktionsplottning 136

I det här kapitlet beskrivs hur du plottar polära ekvationer. Innan du läser detta kapitel bör du känna till innehållet i kapitel 6: Grundläggande funktionsplottning.

Tänk dig en punkt (x,y) som den i bilden nedan. I en polär ekvation är punktens avstånd (r) från origo en funktion av dess vinkel (θ) med den positiva x-axeln. Polära ekvationer uttrycks som r = f(θ).



Så här konverterar du mellan rektangulära koordinater (x,y) och polärkoordinater (r,θ):

$x = r \cos \theta \qquad r^2 = x^2 + y^2$

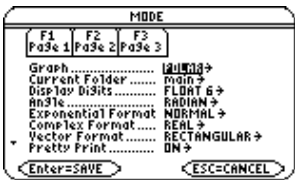
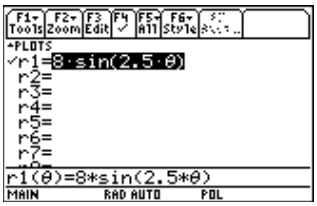
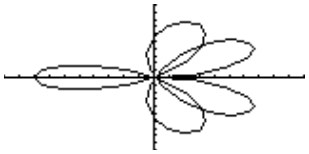
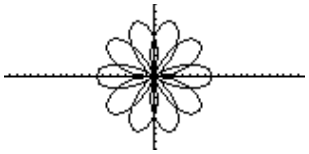
$y = r \sin \theta \qquad \theta = -\tan^{-1} \frac{x}{y} + \frac{\text{sign}(y) \cdot \pi}{2}$

Obs! Du hittar θ genom att använda funktionen angle (x+*i*y), som automatiskt utför beräkningen ovan.

Du kan visa koordinaterna för en punkt i antingen polär (r,θ) eller rektangulär (x,y) form.

Förhandsvisning över polär plottning

Grafen för den polära ekvationen A sin Bθ bildar formen av en ros. Plotta rosen för A=8 och B=2.5. Utforska sedan utseendet av rosen för andra värden av A och B.

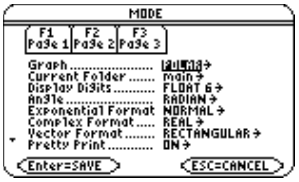
Steg	TI-89 Tangenter	TI-92 Plus Tangenter	Teckenfönster
1. Visa dialogrutan MODE. För grafläget väljer du POLAR. Som vinkelenhet (Angle) väljer du RADIAN.	[MODE] [▶] 3 [▼] [▼] [▼] [▶] 1 [ENTER]	[MODE] [▶] 3 [▼] [▼] [▼] [▶] 1 [ENTER]	
2. Visa och rensa Y= Editor. Definiera därefter polärekvationen r1(θ) = A sin Bθ. *Skriv 8 och 2.5 för A och B, respektive.*	[◆] [Y=] [F1] 8 [ENTER] [ENTER] 8 [2nd] [SIN] 2 . 5 [◆] [θ] [)] [ENTER]	[◆] [Y=] [F1] 8 [ENTER] [ENTER] 8 [SIN] 2 . 5 [θ] [)] [ENTER]	
3. Välj visningsfönstret ZoomStd, som plottar ekvationen. • *Grafen visar endast fem kronblad på rosen.* – *I standardvisningsfönstret är Window-variabeln θmax = 2π. De återstående kronbladen har θ värden som är större än 2π.* • *Rosen ser inte ut att vara symmetrisk.* – *Både x- och y-axeln sträcker sig från –10 till 10. Detta intervall sträcker sig dock över ett längre avstånd längs x-axeln än y-axeln.*	[F2] 6	[F2] 6	
4. Visa Window Editor och ändra θmax till 4π. *4π kommer att beräknas till ett tal när du stänger Window Editor.*	[◆] [WINDOW] [▼] 4 [2nd] [π]	[◆] [WINDOW] [▼] 4 [2nd] [π]	θmin=0. θmax=4π θstep=.13089969389957 xmin=-10. xmax=10. xscl=1. ymin=-10. ymax=10. yscl=1.
5. Välj ZoomSqr, som plottar om ekvationen. *ZoomSqr ökar intervallet längs x-axeln så att grafen visas i korrekta proportioner.*	[F2] 5	[F2] 5	
6. Du kan ändra värden för A och B om du behöver och sedan plotta om ekvationen.			

Översikt över stegen i att plotta polära ekvationer

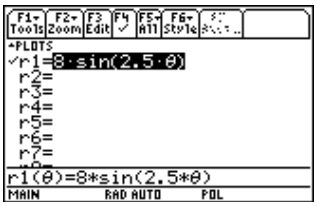
När du vill plotta polära ekvationer använder du samma allmänna steg som för y(x)-funktioner som beskrivs i kapitel 6: Grundläggande funktionsplottning. Skillnader som gäller för polära ekvationer beskrivs på följande sidor.

Plotta polära ekvationer

Ställ in grafläget ([MODE]) till POLAR. Ställ in vinkelenheten, om det behövs.



Definiera polär-ekvationen i Y= Editor ([◆] [Y=]).



Tips! *För att avmarkera statistiska plottningar (kapitel 16), trycker du på* [F5] 5 *eller använder* [F4] *för att avmarkera dem.*

Välj ([F4]), vilket markerar vilka ekvationer som ska plottas.

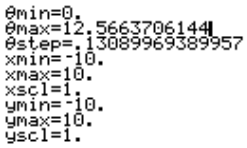
Ställ in visningsformatet för en ekvation.

TI-89: [2nd] [F6]

TI-92 Plus: [F6]

Tips! *(Valfritt) Om du ska plotta flera funktioner är det lättare att skilja dem åt.*

Definiera visnings-fönstret ([◆] [WINDOW]).



Tips! [F2] Zoom *ändrar också visningsfönstret.*

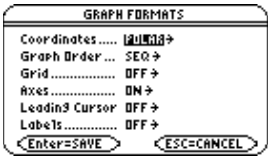
Ändra grafformatet om det behövs.

[F1] 9

— eller —

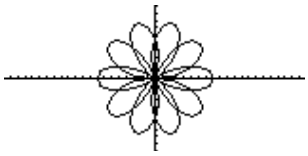
TI-89: [◆] [I]

TI-92 Plus: [◆] F



Tips! *För att visa r och θ, ställer du in Coordinates = POLAR.*

Plotta de markerade ekvationerna ([◆] [GRAPH]).



Utforska grafen

Du kan göra följande från graffönstret:

- Visa koordinaterna för en bildpunkt genom att använda den rörliga markören, eller för en plottad punkt genom att spåra en polär ekvation.
- Använd menyn [F2] Zoom för att zooma in eller ut på en del av grafen.
- Använd menyn [F5] Math för att hitta derivatan, tangenter osv. Vissa menyalternativ är inte tillgängliga för polära grafer.

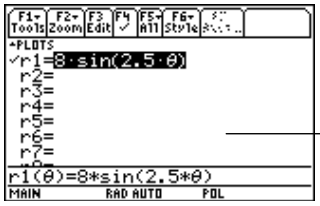
I detta kapitel antas att du känner till hur du plottar y(x)-funktioner (beskrivs i kapitel 6 Grundläggande funktionsplottning). I detta avsnitt beskrivs skillnader som gäller för polära ekvationer.

Ställa in grafläget

Använd MODE för att ställa in Graph = POLAR innan du definierar ekvationer eller ställer in Window-variabler. Med Y= Editor och Window Editor kan du endast mata in information för det *aktuella* grafläget.

Du bör också ställa in vinkelenheten (Angle) till de enheter (RADIAN eller DEGREE) du vill använda på θ.

Definiera polära ekvationer i Y= Editor



Du kan definiera polära ekvationer för r1(θ) t.o.m. r99(θ).

***Tips!** Du kan använda kommandot **Define** från grundfönstret (se appendix A) för att definiera funktioner och ekvationer för vilket grafläge som helst, oavsett vilket det aktuella läget är.*

Y= Editor underhåller en oberoende funktionslista för varje inställning av grafläget. Anta exempelvis följande:

- Du definierar i grafläget FUNCTION en uppsättning y(x)-funktioner. Du byter till grafläget POLAR och definierar en uppsättning r(θ)-ekvationer.
- När du återgår till grafläget FUNCTION är y(x)-funktionerna fortfarande definierade i Y= Editor. När du återgår till grafläget POLAR är r(θ)-ekvationerna fortfarande definierade.

Välja visningsformat

Formaten Above och Below är inte tillgängliga för polära ekvationer och är därför nedtonade på menyn Style i Y= Editor.

Window-variabler

Window Editor underhåller en oberoende uppsättning Window-variabler för varje inställning av grafläget (på samma sätt som Y= Editor) underhåller oberoende Window-variabler.

Variabel	Beskrivning
θmin, θmax	Minsta och största θ-värden.
θstep	Avståndet mellan succesiva θ-värden. Polära ekvationer beräknas för: r(θmin) r(θmin+θstep) r(θmin+2(θstep)) ... får inte överskrida... r(θmax)
xmin, xmax, ymin, ymax	Gränser för visningsfönstret.
xscl, yscl	Avstånd mellan skalstreck på x- och y-axeln.

Obs! *Du kan använda negativt θstep. Om du gör det måste θmin vara större än θmax.*

Standardvärdena (ställs in när du väljer 6:ZoomStd på menyn [F2] Zoom) är:

θmin = 0.		xmin = ⁻10,	ymin = ⁻10,
θmax = 2π	(6,2831853... radianer eller 360 grader)	xmax = 10,	ymax = 10,
θstep = π/24	(,1308996... radianer eller 7,5 grader)	xscl = 1,	yscl = 1,

Du kanske måste ändra standardvärdena för θ-variablerna (θmin, θmax, θstep) för att försäkra dig om att tillräckligt många punkter plottas.

Ställa in grafformatet

Visa koordinaterna som r- och θ-värden genom att använda

[F1] 9

— eller —

TI-89: [◆] [I]
TI-92 Plus: [◆] F

för att ange Coordinates = POLAR. Om Coordinates = RECT, kommer polära ekvationerna att plottas korrekt, men koordinaterna kommer att visas som x och y.

När du spårar en polärekvation visas θ-koordinaten även om Coordinates = RECT.

Utforska en graf

Liksom med funktionsplottning kan du utforska en graf med hjälp av följande verktyg. Koordinater visas i polär form eller rektangulär form enligt inställningen för grafformat.

Verktyg	För polärgrafer:
Rörlig markör	Fungerar som med funktionsplottning.
F2 Zoom	Fungerar som med funktionsplottning. • Endast x- (xmin, xmax, xscl) och y- (ymin, ymax, yscl) Window-variabler påverkas. • θ-Window-variablerna (θmin, θmax, θstep) påverkas inte, såvida du inte väljer 6:ZoomStd (som ställer in θmin = 0, θmax = 2π och θstep = π/24).
F3 Trace	Du kan flytta markören längs grafen ett θstep i taget. • När du påbörjar en spårning befinner sig markören på den först markerade ekvationen vid θmin. • QuickCenter gäller för alla riktningar. Om du flyttar markören utanför fönstret trycker du på ENTER för att centrera visningsfönstret vid markörens position. • Automatisk panorering är inte tillgänglig. Om du flyttar markören utanför fönstrets vänstra eller högra sida kommer visningsfönstret inte automatiskt att panoreras. Du kan däremot använda QuickCenter.
F5 Math	Endast 1:Value, 6:Derivatives, 9:Distance, A:Tangent och B:Arc är tillgängliga för polära ekvationer. Dessa verktyg baseras på θ-värden. Exempel: • 1:Value visar ett r-värde (eller x och y, beroende på grafformatet) för ett angivet θ-värde. • 6:Derivatives hittar dy/dx eller dr/dθ vid en punkt som definieras för ett angivet θ-värde.

Tips! *Under en spårning kan du även beräkna* r(θ) *genom att skriva θ-värdet och trycka på* ENTER.

Tips! *Du kan använda QuickCenter när som helst under en spårning, även om markören fortfarande befinner sig i fönstret.*

Plotta talföljder

9

Översikt över plottning av talföljder 140
Översikt över stegen i plottning av talföljder 141
Skillnader mellan plottning av talföljder och funktioner..... 142
Ställa in plottning med axlar för Time, Web eller Custom 146
Använda plottning med WEB-axlar 147
Använda plottningar med Custom-axlar 150
Använda talföljder för att generera tabeller 151

I det här kapitlet beskrivs hur du plottar talföljder. Innan du läser detta kapitel bör du känna till innehållet i kapitel 6: Grundläggande funktionsplottning.

Talföljder beräknas endast för konsekutiva heltalsvärden. De två allmänna typerna av talföljder är:

- **Icke-rekursiv** – Den n:te termen i talföljden är en funktion av den oberoende variabeln n.

  Varje term är oberoende av andra termer. I följande talföljd kan du beräkna u(5) direkt, utan att först beräkna u(1) eller någon annan föregående term.

  u(n) = 2 ∗ n för n = 1, 2, 3, ...

  n är alltid en serie konsekutiva heltal, med början vid ett positivt heltal eller noll.

  u(n) = 2 ∗ n ger talföljden 2, 4, 6, 8, 10, ...

***Obs!** En rekursiv talföljd kan referera till en annan talföljd. Exempel u2(n) = n^2 +u1(n− 1).*

- **Rekursiv** – Den n:te termen definieras i förhållande till en eller flera föregående termer som representeras av u(n− 1), u(n− 2) osv. Förutom föregående termer kan en rekursiv talföljd även vara definierad i förhållande till n (som exempelvis u(n) = u(n− 1) + n).

  I följande talföljd kan du inte beräkna u(5) utan att första räkna u(1), u(2), u(3) och u(4).

  u(n) = 2 ∗ u(n− 1) för n = 1, 2, 3, ...

  Den första termen är odefinierad eftersom den inte har någon föregående term. Du måste därför ange ett startvärde som ska användas för den första termen.

  Genom att använda startvärdet 1:

  u(n) = 2 ∗ u(n− 1) får du talföljden 1, 2, 4, 8, 16, ...

  Antal startvärden som du måste ange beror på hur djupt rekursionen går. Om exempelvis varje term definieras i förhållande till de föregående två termerna måste du ange startvärden för de två första termerna.

Översikt över plottning av talföljder

I en liten skog finns 4000 träd. Varje år fälls 20% av träden (och 80% finns kvar) och 1000 nya träd planteras. Använd en talföljd och räkna ut hur många träd som finns i skogen i slutet av varje år. Stabiliseras antalet träd efter ett visst antal år?

Från början	Efter 1 år	Efter 2 år	Efter 3 år	. . .
4000	.8 x 4000 + 1000	.8 x (.8 x 4000 + 1000) + 1000	.8 x (.8 x (.8 x 4000 + 1000) + 1000) + 1000	. . .

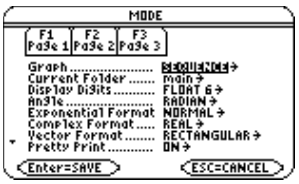
Steg	TI-89 Tangenter	TI-92 Plus Tangenter	Teckenfönster
1. Visa dialogrutan MODE. För Graph fläget väljer du SEQUENCE.	[MODE] ▶ 4 [ENTER]	[MODE] ▷ 4 [ENTER]	
2. Visa och rensa Y= Editor. Definiera därefter talföljden som u1(n) = iPart(.8∗ u1(n− 1)+1000). *Använd* **iPart** *för att ta heltalsdelen av resultatet. Inga delar av träd fälls.* *Öppna* **iPart(** *med* [2nd][MATH]. *Du kan också skriva in det eller välja det från* CATALOG.	[◆][Y=] [F1] 8 [ENTER] [ENTER] [2nd][MATH] 1 4 . 8 [alpha] U 1 [(] [alpha] N [−] 1 [)] [+] 1 0 0 0 [)] [ENTER]	[♦][Y=] [F1] 8 [ENTER] [ENTER] [2nd][MATH] 1 4 . 8 U 1 [(] N [−] 1 [)] [+] 1 0 0 0 [)] [ENTER]	F1▾ Tools F2▾ Zoom F3 Edit F4 ✓ F5▾ All F6▾ Style F7 Axes... *PLOTS ✓ u1=iPart(.8·u1(n − 1) + 10 ui1=4000 u2= ui2= u3= ui3= u4= u1(n)=iPart(.8*u1(n-1)+10... MAIN RAD AUTO SEQ
3. Definiera ui1 som det startvärde som kommer att användas som första term.	[ENTER] 4 0 0 0 [ENTER]	[ENTER] 4 0 0 0 [ENTER]	
4. Visa Window Editor. Ställ in n och plotta Window-variabler. *nmin=0 och nmax=50 beräknar storleken av skogen över 50 år.*	[◆][WINDOW] 0 ▼ 5 0 ▼ 1 ▼ 1 ▼	[♦][WINDOW] 0 ▽ 5 0 ▽ 1 ▽ 1 ▽	nmin=0. nmax=50. PlotStrt=1. PlotStep=1. xmin=0. xmax=50. xscl=10. ymin=0. ymax=6000. yscl=1000.
5. Ställ in Window-variablerna x och y till lämpliga värden för detta exempel.	0 ▼ 5 0 ▼ 1 0 ▼ 0 ▼ 6 0 0 0 ▼ 1 0 0 0	0 ▽ 5 0 ▽ 1 0 ▽ 0 ▽ 6 0 0 0 ▽ 1 0 0 0	
6. Visa graffönstret.	[◆][GRAPH]	[♦][GRAPH]	
7. Välj Trace. Flytta markören för att spåra år för år. Hur många år (nc) tar det för att antalet träd (yc) ska stabiliseras? *Spårningen börjar vid nc=0.* *nc är antalet år.* *xc = nc eftersom n plottas på x-axeln.* *yc = u1(n), antalet träd år n.*	[F3] ▶ och ◀ om det behövs	[F3] ▷ och ◁ om det behövs	1 Som standard används visningsformatet Square för talföljder. nc:27. xc:27. yc:4996.

Översikt över stegen i plottning av talföljder

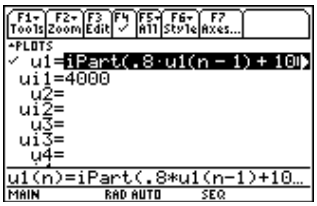
När du plottar talföljder använder du samma allmänna steg som för y(x)-funktioner, som beskrivs i kapitel 6: Grundläggande funktionsplottning. Skillnaderna beskrivs på följande sidor.

Plotta talföljder

Ställ in grafläget (MODE) till SEQUENCE. Ställ även in vinkelenheten (Angle) om det behövs.



Definiera sekvenser och, om det behövs, startvärden i Y= Editor (◆ [Y=]).



Tips! Du stänger av plottning av statistiska data (kapitel 16) genom att trycka på F5 *5 eller använda* F4 *för att avmarkera dem.*

Välj (F4) som definierar vilka talföljder som ska plottas. Välj inga startvärden.

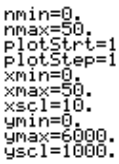
Ställ in visningsformatet för en talföljd.

TI-89: 2nd [F6]

TI-92 Plus: F6

Obs! För talföljder är standardformatet Square.

Definiera visningsfönstret (◆ [WINDOW]).



Tips! F2 Zoom *ändrar även visningsfönstret.*

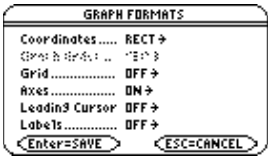
Ändra grafformatet om det behövs.

F1 9

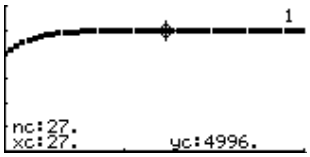
— eller —

TI-89: ◆ [1]

TI-92 Plus: ◆ F



Plotta de markerade talföljderna (◆ [GRAPH]).



Utforska grafen

Du kan göra följande från graffönstret:

- Visa koordinaterna för en bildpunkt genom att använda den rörliga markören, eller en plottad punkt genom att spåra en talföljd.
- Använd menyn F2 Zoom för att zooma in eller ut på en del av grafen.
- Använd menyn F5 Math för att beräkna en talföljd. Endast 1:Value är tillgängligt för talföljder.
- Plotta talföljder på axlarna Time (standard), Web eller Custom.

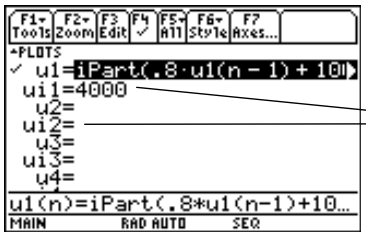
Tips! Du kan även beräkna en talföljd under spårning. Skriv värdet direkt med tangentbordet.

I det här kapitlet antas att du känner till hur du plottar y(x)-funktioner (beskrivs i kapitel 6: Grundläggande funktionsplottning). I det här avsnittet beskrivs de skillnader som gäller för talföljder.

Ställa in grafläget

Använd MODE för att ställa in Graph = SEQUENCE innan du definierar talföljder eller ställer in Window-variabler. I Y= Editor och Window Editor kan du mata in information för den *aktuella* inställningen av grafläget.

Definiera talföljder i Y= Editor

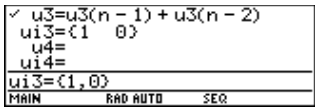


Du kan definiera talföljder u1(n) till och med u99(n).

Använd endast ui för rekursiva talföljder, som kräver ett eller flera startvärden.

Obs! *Du måste använda en lista för att mata in ett eller flera startvärden.*

Om en talföljd kräver mer än ett startvärde kan du mata in dem som en lista inom klammerparenteser { } och avgränsade av komman.



Mata in {1,0} även om {1 0} visas i talföljdslistan.

Om du inte matar in ett startvärde, trots att en talföljd kräver det visas ett felmeddelande när du plottar.

Obs! *Du kan välja andra axlar för grafen om du vill. Gäller bara för talföljder. TIME är standard-inställningen.*

I Y= Editor, Axes välja vilka axlar som ska användas för att plotta talföljderna. Mer detaljerad information finns på sidan 146.

Axlar	Beskrivning
TIME	Plottar n på x-axeln och u(n) på y-axeln.
WEB	Plottar u(n-1) på x-axeln och u(n) på y-axeln.
CUSTOM	Du kan välja x- och y-axlarna.

Tips! *Du kan använda kommandot **Define** från grundfönstret (se appendix A) för att definiera funktioner och ekvationer för vilket grafläge som helst, oavsett vilket som är det aktuella läget.*

Y= Editor underhåller en oberoende funktionslista för varje inställning av grafläget. Anta exempelvis följande:

- I grafläget FUNCTION definierar du en uppsättning y(x)-funktioner. Du ändrar grafläget till SEQUENCE och definierar en uppsättning u(n)-talföljder.
- När du återgår till grafläget FUNCTION är y(x)-funktionerna fortfarande definierade i Y= Editor. När du återgår till grafläget SEQUENCE är u(n)- talföljderna fortfarande definierade.

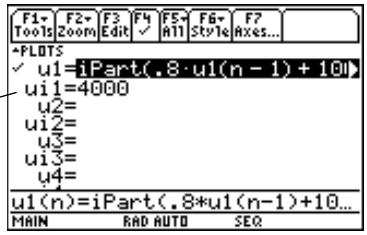
Markera talföljder

Med TIME- och WEB-axlarna plottas endast de markerade talföljderna. Om du angav en talföljd som kräver ett startvärde måste du ange det motsvarande ui-värdet.

***Obs!** Med axlarna TIME och CUSTOM beräknas alla definierade talföljder, även om de inte är plottade.*

Du kan markera en talföljd.

Du kan inte markera dess startvärde.



När du anger en talföljd i den anpassade inställningen och använder CUSTOM-axlarna plottas den, oavsett om den är markerad eller inte.

Välja visningsformat

Det är endast formaten Line, Dot, Square och Thick som är tillgängliga för talföljdsgrafer. Dot och Square markerar endast de diskreta heltalsvärdena (i ökningar av plotstep) för vilken en talföljd plottas.

Window-variabler

Window Editor underhåller en oberoende uppsättning Window-variabler för varje inställning av grafläget (på samma sätt som Y= Editor underhåller oberoende funktionslistor). Talföljdsgrafer använder följande Window-variabler.

***Obs!** Både nmin och nmax måste vara positiva heltal, även om nmin kan vara noll.*

***Obs!** nmin, nmax, plotstrt och plotstep måste vara heltal som är ≥ 1. Om du inte anger heltal kommer de att avrundas till heltal.*

Variabel	Beskrivning
nmin, nmax	Minsta och största n-värdena. Talföljder beräknas för: u(nmin) u(nmin+1) u(nmin+2) ... får inte överskrida ... u(nmax)
plotstrt	Termnummer som ska plottas först (beroende på plotstep). Om du exempelvis vill börja plotta med den andra termen i talföljden ställer du in plotstrt = 2. Den första termen beräknas vid nmin men plottas inte.
plotstep	Avståndet mellan succesiva n-värden *för plottning endast*. Detta påverkar inte hur talföljden beräknas, endast vilka punkter som beräknas. Anta exempelvis att plotstep = 2. Talföljden beräknas för varje efterföljande heltal men plottas bara för vartannat heltal.
xmin, xmax, ymin, ymax	Gränser för visningsfönstret.
xscl, yscl	Avståndet mellan skalstreck på x- och y-axlarna.

Window-variabler (fortsättning)

Standardvärdena (ställs in när du väljer 6:ZoomStd på menyn [F2] Zoom) är:

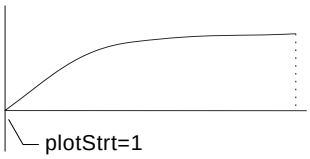
nmin = 1,	xmin = ¯10,	ymin = ¯10,
nmax = 10,	xmax = 10,	ymax = 10,
plotstrt = 1,	xscl = 1,	yscl = 1,
plotstep = 1,		

Du kanske måste ändra standardvärdena för n- och plot-variablerna för att försäkra dig om att tillräckligt många punkter plottas.

Du kan se hur plotstrt påverkar en graf i följande exempel på en rekursiv talföljd.

__Obs!__ Båda dessa grafer använder samma Window-variabler, förutom för plotstrt.

Denna graf plottas med början vid den första termen.

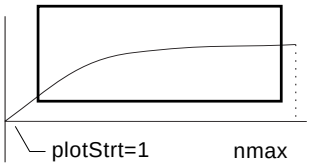


Denna graf plottas med början vid den nionde termen.

Med TIME-axlarna (från [2nd][F7] Axes i Y= Editor) kan du ställa in plotstrt = 1 och fortfarande plotta endast en markerad del av talföljden. Definiera bara ett visningsfönster som endast visar den del av koordinatplanet som du vill visa.

Du kan ange:

- xmin = första n-värdet att plotta
- xmax = nmax (även om du kan använda andra värden)
- ymin och ymax = förväntade värden för talföljderna.



Ändra grafformatet

Formatet Graph Order är inte tillgängligt.

- Med axeln TIME eller CUSTOM plottas alltid flera talföljder samtidigt.
- Med axlarna WEB plottas alltid flera talföljder i tur och ordning.

Utforska en graf

Liksom med funktionsplottning kan du utforska en graf med följande verktyg. Koordinater som visas, visas i rektangulär eller polär form, enligt inställningen av grafformatet.

Verktyg	För talföljdsgrafer:
Rörlig markör	Fungerar som vid funktionsplottning.
F2 Zoom	Fungerar som vid funktionsplottning. • Endast x- (xmin, xmax, xscl) och y- (ymin, ymax, yscl) Window-variablerna påverkas. • n- och plot-Window-variablerna (nmin, nmax, plotstrt, plotstep) påverkas inte, såvida du inte väljer 6:ZoomStd (som ställer in alla Window-variabler till standardvärdena).
F3 Trace	Beroende på om du använder axlarna TIME, CUSTOM eller WEB fungerar Trace på olika sätt. • Med axlarna TIME eller CUSTOM måste du flytta markören längs talföljden ett plotstep i taget. Om du vill flytta ca tio plottade punkter i taget trycker du på 2nd ▶ eller 2nd ◀. – När du påbörjar en spårning befinner sig markören på den först markerade talföljden vid det termnummer som angavs av plotstrt, även om den befinner sig utanför visningsfönstret. – QuickCenter gäller för alla riktningar. Om du flyttar markören utanför fönstret trycker du på ENTER för att centrera visningsfönstret vid markören. • Med WEB-axlar följer spårningsmarkören vävgrafen, inte talföljden. Se sidan 147.
F5 Math	Endast 1:Value är tillgängligt för talföljdsgrafer. • Med axlarna TIME och WEB visas u(n)-värdet (motsvaras av yc) för ett angivet n-värde. • Med axeln CUSTOM beror värdet som motsvarar x och y på vilken axel du väljer.

Tips! *Under en spårning kan du beräkna en talföljd genom att skriva ett värde för n och trycka på* ENTER.

Tips! *Du kan när som helst använda QuickCenter, även om markören fortfarande befinner sig i fönstret.*

När det gäller talföljder kan du välja olika typer av axlar för grafen. Exempel på de olika typerna visas senare i detta kapitel.

Visa dialogrutan AXES

I Y= Editor, Axlar:

- Beroende på den aktuella inställningen för Axes kan vissa alternativ vara nedtonade.
- Om du vill avsluta utan att göra några ändringar trycker du på [ESC].

Alternativ	Beskrivning
Axes	TIME – Plottar u(n) på y-axeln och n på x-axeln.
	WEB – Plottar u(n) på y-axeln och u(n–1) på x-axeln.
	CUSTOM – Du kan välja x- och y-axlarna.
Build Web	Endast aktiv när Axes = WEB. Detta anger om vävgrafen ska ritas manuellt (TRACE) eller automatiskt (AUTO).
	Se sidan 147 för mer information.
X Axis och Y Axis	Endast aktivt när Axes = CUSTOM. Med dessa kan du välja värdet eller talföljden som ska plottas på x- och y-axlarna.
	Se sidan 150 för mer information.

Du kan ändra dessa inställningar genom att använda samma procedur som du använder för att ändra andra typer av dialogrutor, exempelvis dialogrutan MODE.

Använda plottning med WEB-axlar

En plottning med WEB-axlar plottar grafer u(n) mot u(n− 1). Detta gör att du kan studera långsiktigt beteende för en rekursiv talföljd. I exemplen i detta avsnitt visas hur startvärdet kan påverka en talföljds beteende.

Giltiga funktioner för plottningar på WEB-axlar

En talföljd måste följa villkoren nedan, annars plottas den inte korrekt på WEB-axlarna. Talföljden:

- Måste vara rekursiv med endast en rekursionsnivå; u(n− 1) men inte u(n− 2).
- Kan inte referera till n direkt.
- Kan inte referera till andra definierade talföljder förutom till sig själv.

När du visar graffönstret

När du har markerat WEB-axlarna och visar graffönstret händer följande:

- referenslinjen y=x visas.
- Plottar de markerade talföljdsdefinitionerna som funktioner, med u(n− 1) som den oberoende variabeln. Detta konverterar en rekursiv talföljd till en icke-rekursiv form för plottning.

Ta till exempel följande sekvens $u1(n) = \sqrt{5-u1(n-1)}$ och ett initialt värde ui1=1. Först ritas referenslinjen y=x och därefter plottas $y = \sqrt{5-x}$.

Plotta vävgrafen

När talföljden har plottats kan vävgrafen visas manuellt eller automatiskt, beroende på hur ställer in Build Web i dialogrutan AXES.

Om Build Web =	Händer följande med vävgrafen:
TRACE	Den plottas inte förrän du trycker på F3. Vävgrafen plottas därefter steg för steg efter hand som du flyttar spårningsmarkören. **Obs!** Med WEB-axlar kan du inte spåra längs själva talföljden som i andra graflägen.
AUTO	Den ritas automatiskt. Du kan därefter trycka på F3 för att spåra vävgrafen och visa dess koordinater.

Vävgrafen:

1. Startar på x-axeln vid startvärdet ui (när plotstrt = 1).
2. Flyttar vertikalt (upp eller ned) till talföljden.
3. Flyttar horisontellt till referenslinjen y=x.
4. Repeterar denna vertikala och horisontella förflyttning tills n=nmax.

Obs! *Vävgrafen startar vid plotstrt. Värdet för n ökas med 1 varje gång vävgrafen flyttar till talföljden (plotstep ignoreras).*

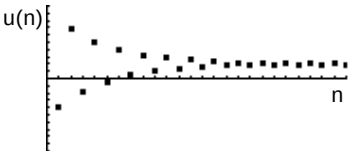
Exempel: Konvergens

1. Definiera i Y= Editor ([◆] [Y=]) u1(n) = ⁻0.8u1(n−1) + 3.6. Ställ in startvärdet till ui1 = ⁻4.
2. Ställ in Axes = TIME.
3. Ställ in Window-variablerna i Window Editor ([◆] [WINDOW]).

nmin=1.	xmin=0.	ymin=⁻10.
nmax=25.	xmax=25.	ymax=10.
plotstrt=1.	xscl=1.	yscl=1.
plotstep=1.		

4. Plotta talföljden ([◆] [GRAPH]).

   Som standard används visningsformatet Square för talföljder.

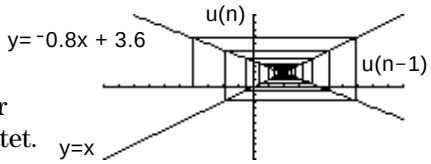


5. Ställ in Axes = WEB och Build Web = AUTO i Y= Editor.
6. Ändra Window-variablerna i Window Editor.

nmin=1.	xmin=⁻10.	ymin=⁻10.
nmax=25.	xmax=10.	ymax=10.
plotstrt=1.	xscl=1.	yscl=1.
plotstep=1.		

7. Plotta om talföljden.

   Vävgrafen visas alltid som linjer, oavsett vilket som är det aktuella visningsformatet.



Tips! Du kan flytta markören under en spårning till ett angivet värde genom att skriva värdet och trycka på [ENTER].

8. Tryck på [F3]. När du trycker på ▶ följer spårningsmarkören vävgrafen. I fönstret visas markörkoordinaterna nc, xc och yc (där xc och yc motsvarar u(n−1) och u(n)).

Tips! När värdet nc ändras befinner sig markören på talföljden. Nästa gång du trycker på ▶ förblir nc detsamma men markören befinner sig då på referenslinjen y=x.

När du spårar till större värden av nc kan du se hur xc och yc närmar sig konvergenspunkten.

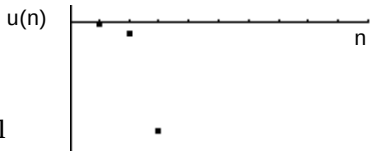
Exempel: Divergens

1. Definiera u1(n) = 3.2u1(n− 1) −.8(u1(n− 1))^2 i Y= Editor ([◆] [Y=]). Ställ in startvärdet ui1 = 4.45.
2. Ställ in Axes = TIME.
3. Ställ in Window-variablerna i Window Editor ([◆] [WINDOW]).

nmin=0.	xmin=0.	ymin=⁻75.
nmax=10.	xmax=10.	ymax=10.
plotstrt=1.	xscl=1.	yscl=1.
plotstep=1.		

4. Plotta talföljden ([◆] [GRAPH]).

   Eftersom talföljden snabbt divergerar till stora negativa värden plottas endast ett fåtal punkter.



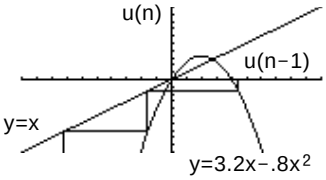
5. I Y= Editor, ställ in Axes = WEB och Build Web = AUTO.

6. Ändra Window-variablerna i Window Editor.

nmin=0.	xmin=¯10.	ymin=¯10.
nmax=10.	xmax=10.	ymax=10.
plotstrt=1.	xscl=1.	yscl=1.
plotstep=1.		

7. Plotta om talföljden.

   Vävgrafen visar hur snabbt talföljden divergerar till stora negativa värden.



Exempel: Oscillering

I detta exempel visas hur startvärdet kan påverka en talföljd.

1. Använd i Y= Editor ([◆] [Y=]) samma talföljd som definierades i divergensexemplet: u1(n) = 3.2u1(n−1) −.8(u1(n−1))². Ställ in startvärdet ui1 = 0.5.

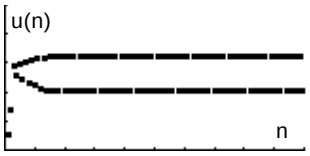
2. Ställ in Axes = TIME.

3. Ställ in Window-variablerna i Window Editor ([◆] [WINDOW]).

nmin=1.	xmin=0.	ymin=0.
nmax=100.	xmax=100.	ymax=5.
plotstrt=1.	xscl=10.	yscl=1.
plotstep=1.		

__Obs!__ Jämför denna graf med divergensexemplet. Detta är samma talföljd med ett annat startvärde.

4. Plotta talföljden ([◆] [GRAPH]).



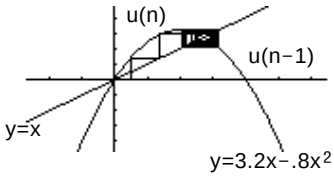
5. I Y= Editor, ställ in Axes = WEB och Build Web = AUTO.

6. Ändra Window-variablerna i Window Editor.

nmin=1.	xmin=¯2.68	ymin=¯4.7
nmax=100.	xmax=6.47	ymax=4.7
plotstrt=1.	xscl=1	yscl=1
plotstep=1.		

__Obs!__ Vävgrafen flyttas till en bana som oscillerar mellan de två stabila punkterna.

7. Plotta om talföljden.

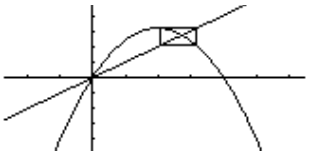


8. Tryck på [F3]. Använd därefter ⊙ för att spåra vävgrafen.

   Efter hand som du spårar till större värden av nc ser du att xc och yc oscillerar mellan 2.05218 och 3.19782.

__Obs!__ Genom att starta vävgrafen vid en senare term, visas den stabila oscillationen klarare.

9. Ställ in plotstrt=50 i Window Editor. Plotta därefter om talföljden.



Med axeln CUSTOM har du stor flexibilitet att plotta talföljder. Som du ser i följande exempel är axeln CUSTOM särskilt effektiv för att visa förhållanden mellan en talföljd och en annan.

Exempel: Modellen rovdjur-byte

Genom att använda modellen rovdjur-byte i biologi kan du beräkna antalet kaniner och vargar som håller populationerna i jämvikt i ett visst område.

R = Antal kaniner
M = Tillväxthastighet för kaniner om det inte finns några vargar (använd 0,05)
K = Hastighet med vilken vargar dödar kaniner (använd 0,001)
W = Antal vargar
G = Tillväxthastighet för vargar om det inte finns några kaniner (använd 0,0002)
D = Dödlighet för vargar om det inte finns några kaniner (använd 0,03)

$R_n = R_{n-1} (1 + M - K W_{n-1})$
$W_n = W_{n-1} (1 + G R_{n-1} - D)$

1. Definiera i Y= Editor ([◆][Y=]) talföljder och startvärden för R_n och W_n.

   u1(n) = u1(n− 1) ∗ (1 + 0.05 − 0.001 ∗ u2(n− 1))
   ui1 = 200
   u2(n) = u2(n− 1) ∗ (1 + 0.0002 ∗ u1(n− 1) − 0.03)
   ui2 = 50

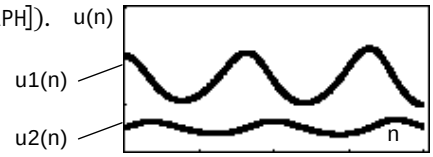
   Obs! *Anta att det från början är 200 kaniner och 50 vargar.*

2. Ställ in Axes = TIME.

3. Ställ in Window-variablerna i Window Editor ([◆][WINDOW]).

nmin=0.	xmin=0.	ymin=0.
nmax=400.	xmax=400.	ymax=300.
plotstrt=1.	xscl=100.	yscl=100.
plotstep=1.		

4. Plotta talföljden. ([◆][GRAPH]).



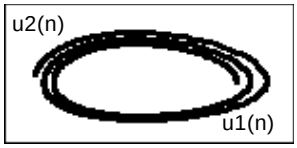
   Obs! *Använd [F3] för att individuellt spåra antalet kaniner u1(n) och vargar u2(n) med tiden (n).*

5. I Y= Editor, ställ in Axes = CUSTOM, X Axis = u1 och Y Axis = u2.

6. Ändra Window-variablerna i Window Editor.

nmin=0.	xmin=84.	ymin=25.
nmax=400.	xmax=237.	ymax=75.
plotstrt=1.	xscl=50.	yscl=10.
plotstep=1.		

7. Plotta om talföljden.



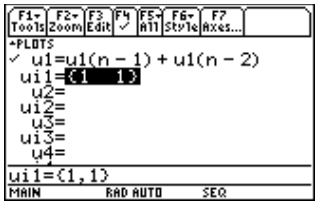
   Obs! *Använd [F3] för att spåra både antalet kaniner (xc) och vargar (yc) över en period som sträcker sig över 400 generationer.*

I föregående avsnitt beskrevs hur du plottar en talföljd. Du kan också använda en talföljd för att generera en tabell. I kapitel 13 finns detaljerad information om tabeller.

Exempel: Fibonacci-följden

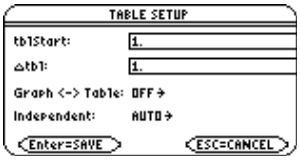
I en Fibonacci-följd är de första två termerna 1 och 1. Varje efterföljande term är summan av de två föregående termerna.

1. Definiera i Y= Editor ([◆] [Y=]) talföljden och ställ in startvärdena som i bilden.



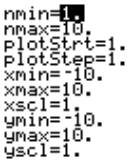
Du måste mata in {1,1}, trots att {1 1} visas i talföljdslistan.

2. Ställ in tabellparametrarna ([◆] [TblSet]) till:
   tblStart = 1
   Δtbl = 1
   Independent = AUTO

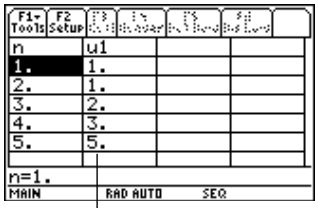


Detta alternativ är nedtonat om du inte använder TIME-axeln.

3. Ställ in Window-variablerna ([◆] [WINDOW]) så att nmin har samma värde som tblStart.



4. Visa tabellen [◆] [TABLE]).



Fibonacci-följden är i kolumn 2.

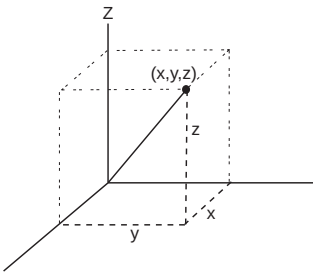
5. Rulla ned tabellen (⊖ eller [2nd] ⊖) för att visa mer av talföljden.

3D-plottning

10

Nya funktioner för plottning av 3D-grafer 154
Översikt över stegen i plottning av 3D-ekvationer 156
Skillnader mellan 3D- och funktionsplottning 157
Flytta markören i 3D..... 160
Rotera en 3D-graf med variabeln eyeψ 162
Animera en 3D-graf interaktivt..... 164
Ändra koordinataxlarna och visningsformatet 165
Konturplottning..... 167
Exempel: Konturer hos en komplex modulär yta..... 170
Implicit plottning 171
Exempel: Implicit plottning av en mer komplicerad ekvation..... 173

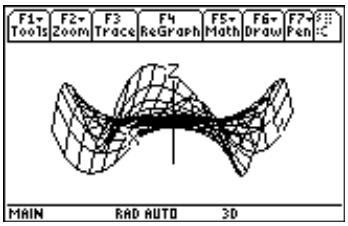
I det här kapitlet beskrivs hur du plottar 3D-ekvationer. Innan du läser kapitlet bör du känna till innehållet i kapitel 6: Grundläggande funktionsplottning.



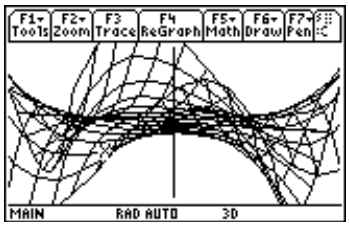
I en 3D-graf av en ekvation för z(x,y) definieras positionen för en punkt som i bilden till höger.

Utökat visningsläge är en ny funktion som är användbar för alla typer av 3D-grafer. Med det utökade visningsläget kan du studera 3D-grafer i detalj. Till exempel:

Normalt visningsläge



Utökat visningsläge



Tryck på × (multiplikationstangenten, inte bokstaven X) när du vill växla visningsläge.

När du plottar en 3D-graf används det utökade visningsläget automatiskt om:

- Du ställer in eller ändrar format till CONTOUR LEVELS eller IMPLICIT PLOT.
- Den föregående grafen använde utökat visningsläge.

Om du trycker på en piltangent för att animera grafen enligt senare beskrivning växlas visningsläget automatiskt till normalt. Du kan inte animera en graf i utökat visningsläge.

Tips! *Om du vill visa en graf längs x-, y- eller z-axeln skriver du in tecknen X, Y respektive Z.*

Tips! *Om du vill växla från en formatstil till nästa (och hoppa över* IMPLICIT PLOT*), tryck på:*

TI-89: alpha [F]
TI-92 Plus: F

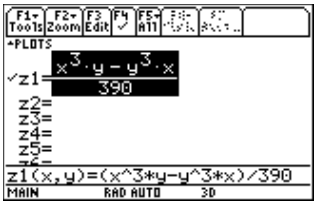
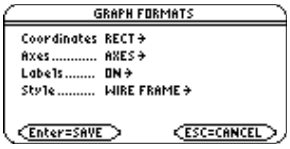
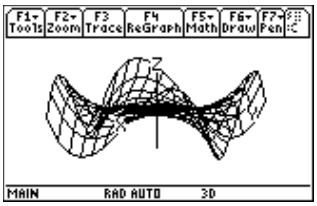
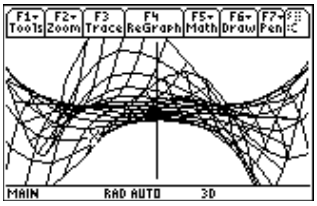
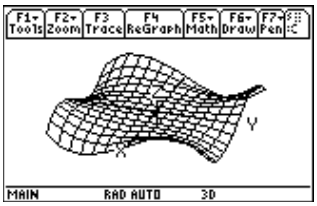
Detta bibehåller den aktuella visningen (expanderad eller normal).

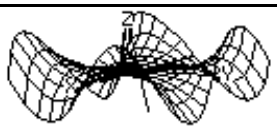
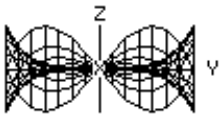
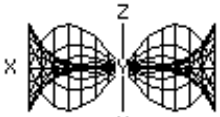
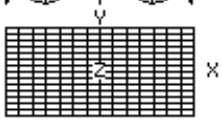
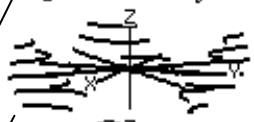
Obs! *Om du vill växla till* IMPLICIT PLOT *(via dialog-rutan GRAPH FORMAT), tryck på:*

TI-89: ◆ |
TI-92 Plus: ◆ F

Nya funktioner för plottning av 3D-grafer

Plotta 3D-ekvationen z(x,y) = ($x^3y - y^3x$) / 390. Animera grafen genom att med piltangenterna ändra eye-variabeln som bestämmer visningsvinkeln. Du kan sedan visa grafen i olika format.

Steg	TI-89 Tangenter	TI-92 Plus Tangenter	Visa
1. Öppna dialogrutan MODE. Välj 3D i grafläge.	[MODE] ⊙ 5 [ENTER]	[MODE] ⊙ 5 [ENTER]	
2. Visa och rensa Y= Editor. Definiera 3D-ekvationen z1(x,y) = ($x^3y - y^3x$) / 390. *Observera att implicerad multiplika-tion används i tangentkombinationen.*	[◆] [Y=] [F1] 8 [ENTER] [ENTER] [(] X [^] 3 Y [−] Y [^] 3 X [)] [÷] 3 9 0 [ENTER]	[◆] [Y=] [F1] 8 [ENTER] [ENTER] [(] X [^] 3 Y [−] Y [^] 3 X [)] [÷] 3 9 0 [ENTER]	
3. Ändra grafens format för att visa och märka axlarna. Ställ in Style = WIRE FRAME. *Du kan animera grafer i alla format men* WIRE FRAME *uppdateras snabbast.*	[◆] [I] ⊖ ⊙ 2 ⊖ ⊙ 2 ⊖ ⊙ 1 [ENTER]	[◆] F ⊖ ⊙ 2 ⊖ ⊙ 2 ⊖ ⊙ 1 [ENTER]	
4. Välj ZoomStd för att automatiskt plotta grafen. *Medan grafen beräknas (innan den plottas) visas hur många procent av beräkningen som är klar i övre, vänstra delen av skärmen.*	[F2] 6	[F2] 6	
Obs! *Om du redan plottat en 3D-graf kan det hända att grafen visas i utökat läge. När du animerar grafen återställs visningen automatiskt till normalläge. (Förutom animeringar kan du utföra samma operationer i normalt och i utökat visningsläge.)*	[×] (tryck [×] för att växla mellan normalt och utökat visningsläge)	[×] (tryck [×] för att växla mellan normalt och utökat visningsläge)	
5. Animera grafen genom att minska värdet för Windowvariabeln eyeϕ. ⊖ *eller* ⊕ *kan påverka* eyeθ *och* eyeψ *men i mindre utsträckning än* eyeϕ. *Om du vill animera grafen kontinuerligt trycker du ned markörtangenten i ungefär 1 sekund och släpper den sedan. Stoppa genom att trycka* [ENTER].	⊖ ⊖ ⊖ ⊖ ⊖ ⊖ ⊖ ⊖	⊖ ⊖ ⊖ ⊖ ⊖ ⊖ ⊖ ⊖	

Steg	TI-89 Tangenter	TI-92 Plus Tangenter	Visa
6. Återställ grafen till utgångsläget. Flytta sedan visningsvinkeln längs “visningssfären” runt grafen. *Mer information om visningssfär finns på sidan 164.*	0 (noll, inte bokstaven O) ◀ ◀ ◀	0 (noll, inte bokstaven O) ◀ ◀ ◀	
7. Visa grafen längs x -axeln, y -axeln och z -axeln.	X	X	
Denna graf har samma form längs y -axeln och x -axeln.	Y	Y	
	Z	Z	
8. Återställ grafen till utgångsläget.	0	0	
9. Visa grafen i olika grafformat.	I (tryck I för att växla från ett format till nästa)	F (tryck F för att växla från ett format till nästa)	 HIDDEN SURFACE  CONTOUR LEVELS (kan kräva extra beräkningstid) WIRE AND CONTOUR WIRE FRAME

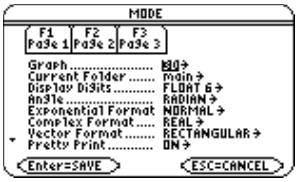
Obs! Du kan också visa grafen som en implicit plottning genom att använda dialogrutan GRAPH FORMATS (F1 9 eller **TI-89:** ◆ I **TI-92 Plus:** ◆ F). Om du trycker **TI-89:** I **TI-92 Plus:** F för att växla mellan format visas inte implicit plottning.

Översikt över stegen i plottning av 3D-ekvationer

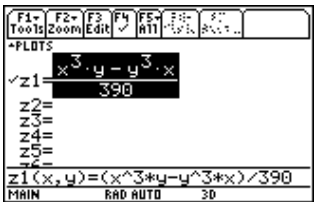
När du plottar 3D-ekvationer använder du samma allmänna steg som för y(x)-funktioner (beskrivs i kapitel 6: Grundläggande funktionsplottning). Skillnader som gäller för 3D-ekvationer beskrivs på sidorna som följer.

Plotta 3D-ekvationer

Ställ in grafläget ([MODE]) till 3D. Om det behövs ställer du även in vinkelenheten (Angle).



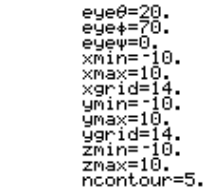
Definiera 3D-ekvationer i Y= Editor ([◆] [Y=]).



Välj ([F4]) vilken ekvation som ska plottas. Du kan endast välja en 3D-ekvation.

Tips! Om du vill stänga av alla plottningar av statistiska data (kapitel 16) trycker du på [F5] 5 eller använder [F4] för att avmarkera dem.

Definiera visningskuben ([◆] [WINDOW]).



Obs! För 3D-grafer kallas visningsfönstret visningskuben. Med [F2] Zoom ändrar du även visningskuben.

Ändra grafformatet om det behövs.

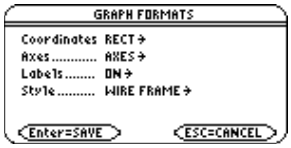
[F1] 9

— eller —

TI-89: [◆] [I]

TI-92 Plus: [◆] F

Tips! För att det ska vara lättare att se orienteringen av 3D-grafer aktiverar du Axes och Labels.



Plotta den markerade ekvationen ([◆] [GRAPH]).

Obs! Innan grafen visas, visas i fönstret hur många procent som beräknats.

Utforska grafen

Du kan göra följande från graffönstret:

- Spåra ekvationen.
- Använda menyn [F2] Zoom för att zooma in eller ut på en del av grafen. Vissa av menyalternativen är nedtonade eftersom de inte är tillgängliga för 3D-grafer.
- Använda menyn [F5] Math för att beräkna ekvationens värde vid en angiven punkt. Endast 1:Value är tillgängligt för 3D-grafer.

Tips! Du kan också beräkna z(x,y) samtidigt som du spårar. Skriv x-värdet och tryck på [ENTER] och skriv y-värdet och tryck på [ENTER].

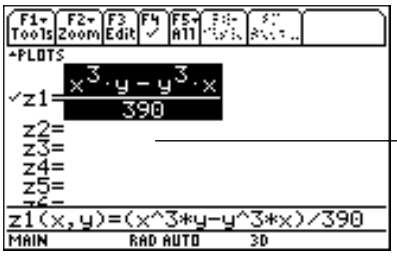
Skillnader mellan 3D- och funktionsplottning

I det här kapitlet antas det att du känner till hur du plottar y(x)-funktioner (beskrivs i kapitel 6: Grundläggande funktionsplottning). I det här avsnittet beskrivs skillnaderna som gäller för 3D-ekvationer.

Ställa in grafläget

Använd MODE för att ställa in Graph = 3D innan du definierar ekvationer eller ställer in Window-variabler. I Y= Editor och Window Editor kan du endast mata in information för det *aktuella* grafläget.

Definiera 3D-ekvationer i Y= Editor



Du kan definiera 3D-ekvationer för z1(x,y) till och med z99(x,y).

***Tips!** Du kan använda kommandot **Define** från grundfönstret (se appendix A) för att definiera funktioner och ekvationer för vilket grafläge som helst, oavsett vilket som är det aktuella läget.*

Y= Editor underhåller en oberoende funktionslista för varje inställning av grafläget. Anta exempelvis följande:

- I grafläget FUNCTION definierar du en uppsättning y(x)-funktioner. Du byter till 3D-grafläge och definierar en uppsättning z(x,y)-ekvationer.
- När du återgår till grafläget FUNCTION är y(x)-funktionerna fortfarande definierade i Y= Editor. När du återgår till 3D-grafläge är z(x,y)-ekvationerna fortfarande definierade.

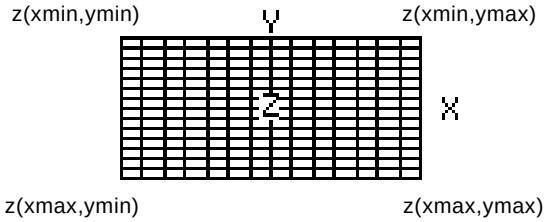
Välja visningsformat

Eftersom du endast kan plotta en 3D-ekvation i taget är visningsformaten inte tillgängliga. I Y= Editor är menyn Style nedtonad.

För 3D-ekvationer kan du däremot använda:
F1 9
— eller —
TI-89: ◆ I
TI-92 Plus: ◆ F
för att ställa in Style-formatet till WIRE FRAME eller HIDDEN SURFACE.
Se "Ändra koordinataxlarna och visningsformatet" på sidan 165.

Window-variabler

Window Editor underhåller en oberoende uppsättning Window-variabler för varje inställning av grafläget (på samma sätt som Y= Editor underhåller oberoende funktionslistor). 3D-grafer använder följande Window-variabler.

Variabel	Beskrivning
eyeθ, eyeϕ, eyeψ	Vinklar (alltid i grader) används för att visa grafen. Se "Rotera en 3D-graf med variabeln eyeψ" på sidan 162.
xmin, xmax, ymin, ymax, zmin, zmax	Gränser för visningskuben.
xgrid, ygrid	Avståndet mellan xmin och xmax och mellan ymin och ymax delas upp i angivet antal stödpunkter. Ekvationen z(x,y) beräknas vid varje stödrasterpunkt där stödlinjerna genomskär varandra. Avståndet mellan succesiva värden utefter x- och y-axlarna beräknas enligt följande: $x\ \text{ökning} = \frac{xmax - xmin}{xgrid}$ $y\ \text{ökning} = \frac{ymax - ymin}{ygrid}$ Antalet stödlinjer är xgrid + 1 och ygrid + 1. När exempelvis xgrid = 14 och ygrid = 14 består stödrastret XY av 225 (15 × 15) stödpunkter. 
ncontour	Antalet konturer som fördelas jämnt längs det visade intervallet av z-värden. Se sidan 168 för mer information.

Obs! *Om du inte anger ett heltal för* xgrid *eller* ygrid *rundas det av till närmaste heltal som är ≥ 1.*

Obs! *I 3D-läge finns inga Windowvariabler, för scl, så du kan inte ange skalstreck på axlarna.*

Standardvärdena (ställs in när du väljer 6:ZoomStd på menyn F2 Zoom) är:

eyeθ = 20.	xmin = ⁻10.	ymin = ⁻10.	zmin = ⁻10.
eyeϕ = 70.	xmax = 10.	ymax = 10.	zmax = 10.
eyeψ = 0.	xgrid = 14.	ygrid = 14.	zscl = 1.

Obs! *När du ökar värdet på stödlinjevariablerna minskas plottningshastigheten.*

Du kanske måste öka standardvärdena för stödlinjevariablerna (xgrid, ygrid) för att försäkra dig om att tillräckligt många punkter plottas.

Ställa in grafformatet

Formaten Axes och Style är specifika för 3D-graffläge. Se "Ändra koordinataxlarna och visningsformatet" på sidan 165.

Utforska en graf

Liksom med funktionsplottning kan du utforska en graf med följande verktyg. Koordinater som visas, visas i rektangulär eller cylindrisk form enligt inställningen för grafformatet. (I 3D-plottning visas cylindriska koordinater när du använder:
[F1] 9
— eller —
TI-89: [◆] [I]
TI-92 Plus: [◆] F
för att ställa in Coordinates = POLAR.)

Verktyg	För 3D-grafer:
Rörlig markör	Den rörliga markören är inte tillgänglig.
[F2] Zoom	Fungerar på i stort sätt samma sätt som för funktionsplottning, men kom ihåg att du nu använder tre dimensioner i stället för två. • Endast följande zoomningar är tillgängliga: 2:ZoomIn 5:ZoomSqr A:ZoomFit 3:ZoomOut 6:ZoomStd B:Memory C:SetFactors • Endast x- (xmin, xmax), y- (ymin, ymax) och z- (zmin, zmax, zscl) Window-variablerna påverkas. • grid- (xgrid, ygrid) och eye- (eyeθ, eyeϕ, eyeψ) Window-variablerna påverkas inte såvida du inte väljer 6:ZoomStd (vilket återställer dessa variabler till standardvärdena).
[F3] Trace	Du kan flytta markören längs en stödlinje från en stödpunkt till nästa på 3D-ytan. • När du påbörjar en spårning visas markören i mitten av xy-stödrastret. • QuickCenter är tillgängligt. När som helst under en spårning, oavsett var markören befinner sig, kan du trycka på [ENTER] för att centrera visningskuben vid markören. • Markörförflyttningar begränsas till x- och y-riktningar. Du kan inte flytta markören bortom visningskubens kanter, som ställs in med xmin, xmax, ymin och ymax.
[F5] Math	Endast 1:Value är tillgängligt för 3D-grafer. Med detta verktyg visas z-värdet för ett angivet x- och y-värde. När du har valt 1:Value skriver du x-värdet och trycker på [ENTER]. Skriv därefter y-värdet och tryck på [ENTER].

Tips! *Se "Flytta markören i 3D" på sidan 160.*

Tips! *Under en spårning kan du även beräkna z(x,y). Skriv x-värdet och tryck på [ENTER] och skriv därefter y-värdet och tryck på [ENTER].*

Flytta markören i 3D

> När du flyttar markören längs en 3D-yta kan det vara svårt att se varför markören rör sig som den gör. 3D-grafer har två oberoende variabler (x, y) i stället för en, och x- och y-axlarna har en annan orientering än i andra graflägen.

Hur du flyttar markören

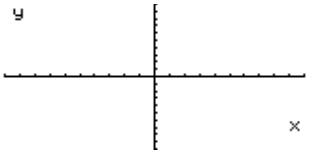
__Obs!__ Du kan flytta markören inom x- och y-gränserna som anges av Window-variablerna xmin, xmax, ymin och ymax.

På en 3D-yta följer markören alltid en stödlinje.

Markörknapp	Flyttar markören till nästa stödpunkt i:
⊙	Positiv x-riktning
⊙	Negativ x-riktning
⊙	Positiv y-riktning
⊙	Negativ y-riktning

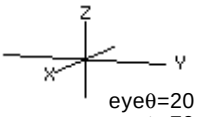
Även om reglerna är enkla, kan själva markörförflyttningen vara förvirrande om du inte känner till axlarnas orientering.

I 2D-plottning har x- och y-axlarna alltid samma orientering i förhållande till graffönstret.



__Tips!__ Om du vill visa axlar och etiketter från i Y= Editor, Window Editor eller i Graffönstret, tryck på:
__TI-89:__ ◆ |
__TI-92 Plus__: ◆ F

I 3D-plottning har x- och y-axlarna annan orientering i förhållande till graffönstret. Du kan också rotera och/eller höja visningsvinkeln.

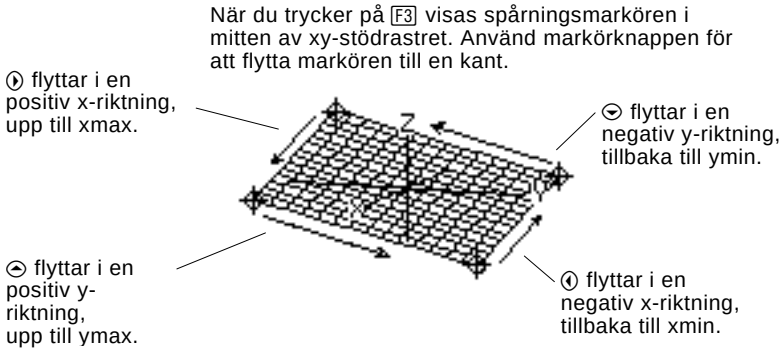


eyeθ=20
eyeϕ=70
eyeψ=0

Exempel på hur du flyttar markören

I följande graf visas ett lutande plan som har ekvationen z1(x,y) = ⁻(x + y) / 2. Anta att du vill spåra runt de gränser som visas.

__Tips!__ Genom att visa och etikettera axlarna är det lättare att se mönstret i markörförflyttningen.



__Tips!__ Om du vill flytta stödpunkter närmare varandra kan du öka Window-variablerna xgrid och ygrid.

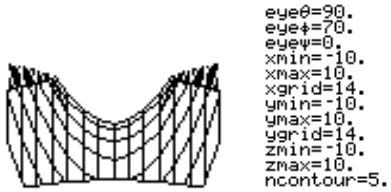
När spårningsmarkören befinner sig på en inre punkt i det visade planet, flyttas markören från en stödpunkt till nästa längs en av stödlinjerna. Du kan inte flytta diagonalt över stödrastret.

Lägg märke till att stödlinjerna kanske inte visas parallellt mot axlarna.

Exempel på markören på en dold yta

På mer komplexa former kan det se ut som om markören inte befinner sig på en stödpunkt. Detta är en optisk illusion som uppstår när markören befinner sig på en dold yta.

Anta exempelvis att du har sadelformen $z1(x,y) = (x^2 - y^2) / 3$.
I följande graf visas vyn när du tittar ned utefter y-axeln.



Titta nu på samma form vid 10° från x-axeln (eyeθ = 10).

***Tips!** För att skära bort framsidan av sadeln i detta exempel, ställer du in xmax=0 för att endast visa negativa x-värden.*

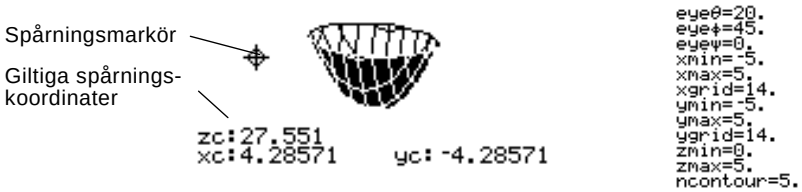
Du kan flytta markören så att den inte ser ut att vara på en stödpunkt.

Om du klipper bort den främre sidan kan du se att markören i själva verket befinner sig på en stödpunkt på den dolda baksidan.

Exempel på markören när den är "utanför" kurvan

Även om markören endast kan röra sig längs en stödlinje kommer du att se många situationer när markören inte ser ut att vara på 3D-ytan över huvud taget. Detta inträffar när Z-axeln är för kort eller lång för att visa z(x,y) för motsvarande x- och y-värden.

Anta exempelvis att du spårar paraboloid $z(x,y) = x^2 + y^2$ plottad med de angivna Window-variablerna. Du kan enkelt flytta markören till en position som visas hedan:



***Tips!** Med QuickCenter kan du centrera visningskuben vid markören. Tryck bara på* [ENTER].

Även om markören i själva verket spårar paraboloiden, ser den ut att vara utanför kurvan eftersom spårningskoordinaterna:

- xc och yc är inom visningskuben.

  – men –

- zc är utanför visningskuben.

När zc är utanför visningskubens z-gräns visas markören fysiskt vid zmin eller zmax (även om de korrekta spårningskoordinaterna visas i fönstret).

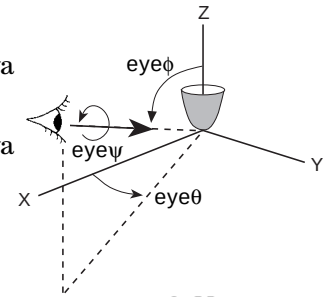
Rotera en 3D-graf med variabeln eyeψ

I 3D-läge kan du använda Windows-variablerna eyeθ och eyeϕ för att ställa in visningsvinklar som bestämmer synlinjen. En ny Windows-variabel, eyeψ, låter dig rotera grafen runt denna synlinje.

Hur visningsvinkeln mäts

Visningsvinkeln består av tre element:

- eyeθ — vinkel i grader från den postiva x -axeln.
- eyeϕ — vinkel i grader från den postiva z -axeln.
- eyeψ — vinkel i grader med vilken grafen roteras moturs runt synlinjen som ställts in med eyeθ och eyeϕ.



__Obs!__ När eyeψ=0 *så är z-axeln vertikal på skärmen. När* eyeψ=90 *så är z-axeln roterad 90° moturs och horisontell.*

I Window Editor ([◆] [WINDOW]) skriver du alltid in eyeθ, eyeϕ och eyeψ i grader oavsett aktuell vinkelenhet.

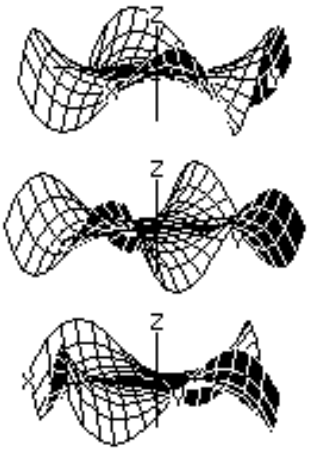
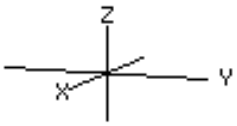
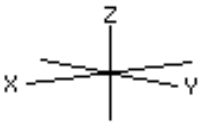
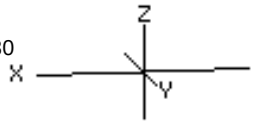
```
eyeθ=20.
eyeϕ=70.
eyeψ=0.
xmin=-10.
xmax=10.
xgrid=14.
ymin=-10.
ymax=10.
ygrid=14.
zmin=-10.
zmax=10.
ncontour=5.
```

Skriv inte in symbolen °. Skriv t ex 20, 70 och 0, inte 20°, 70° och 0°.

Resultat av att ändra eyeθ

Vyn i graffönstret är alltid orienterad längs visningsvinkeln. Ur denna synpunkt kan du ändra eyeθ till att rotera visningsvinkeln runt Z-axeln.

__Obs!__ I detta exempel ökas eyeθ *stegvis med* 30.

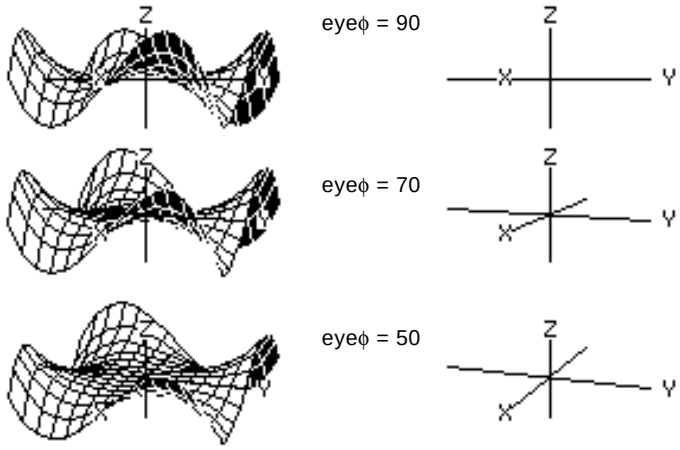
z1(x,y) = (x³y – y³x) / 390	I detta exempel är eyeϕ = 70
	eyeθ = 20 
	eyeθ = 50 
	eyeθ = 80 

Resultat av att ändra eyeϕ

Genom att ändra eyeϕ kan du luta visningsvinkeln i förhållande till xy-planet. Om 90 < eyeϕ < 270 är visningsvinkeln under xy-planet.

***Obs!** Detta exempel startar på xy-planet (*eyeϕ = 90*) och ökar* eyeϕ *stegvis med* 20 *för att öka vårdet på visningvinkeln.*

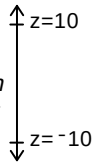
z1(x,y) = (x³y – y³x) / 390	I detta exempel är eyeθ° = 20



Resultat av att ändra eyeψ

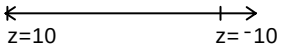
Visningen i graffönstret är alltid orienterad längs visningsaxlarna som ställts in med eyeθ and eyeϕ. Genom att ändra eyeψ kan du rotera grafen runt denna synlinje.

***Obs!:** Under rotation sträcks och krymps axlarna för att passa skärmens höjd och bredd. Detta kan orsaka en del förvrängningar som i exemplet.*



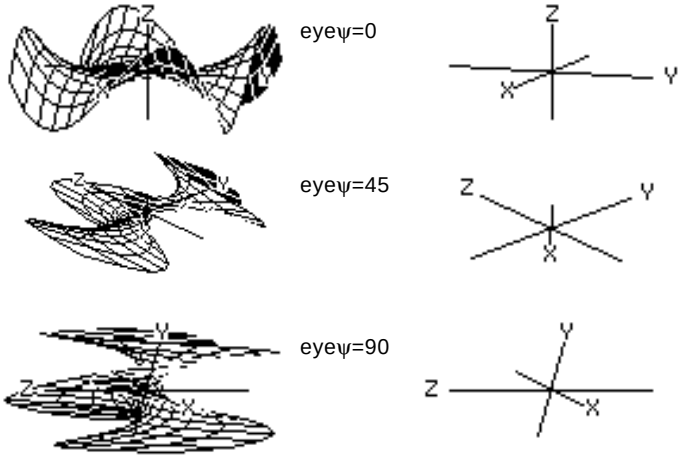
När eyeψ=0, *sträcker sig z -axeln över hela skärmens höjd.*

När eyeψ=90 *sträcker sig z -axeln över hela skärmens bredd.*



När z -axeln roterar 90° utökas längden (¯10 till 10 i detta exempel) till nästan den dubbla längden. Samtidigt sträcks eller krymps x- och y-axlarna.

z1(x,y)=(x³y–y³x) / 390	I detta exempel, eyeθ=20 och eyeϕ=70



I grundfönstret eller från ett program

Värdena för eye lagras i systemvariablerna eyeθ, eyeϕ och eyeψ. Du kan öppna och spara dessa variabler efter behov.

TI-89: Skriv in ϕ eller ψ genom att trycka [◆] [(] [alpha] [F] eller [◆] [(] [Y], respektive. Du kan också trycka [2nd] [CHAR] och använda menyn Greek.

TI-92 Plus: Skriv in ϕ eller ψ genom att trycka [2nd] G F eller [2nd] G Y respektive. Du kan också trycka [2nd] [CHAR] och använda menyn Greek.

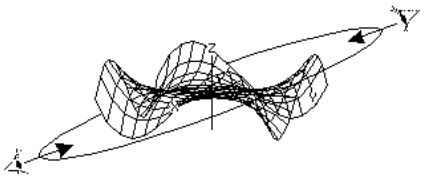
> När du har plottat en 3D-graf kan ändra visningsvinkel interaktivt med markören. Se exemplet på sidan 154.

Visningssfär

__Obs!__ Visningssfären påverkar Window-variablerna eye *i olika utsträckning.*

När du använder ◀ och ▶ för att animera en graf kan du föreställa dig att du flyttar visningsvinkeln längs dess "visningssfär" runt grafen.

Flyttning längs denna sfär kan medföra att z -axeln svajar lite under animeringen (se exemplet på sidan 154).



Animera grafen

__Obs!__ Om grafen visas i utökat visningsläge återgår visningen till normalläge automatiskt då du trycker ned en markörtangent.

__Tips!__ När grafen är animerad kan du stoppa och starta animeringen i samma riktning med:

__TI-89:__ [ENTER] *eller* [alpha] [␣]
__TI-92 Plus:__ [ENTER] *eller blanksteg*

__Tips!__ Under en animering kan du växla till nästa grafformat genom att trycka på:

__TI-89:__ [I]
__TI-92 Plus:__ F

__Tips!__ Om du vill visa en graf där vinklarna för eye *visas hittar du mer information på sidan 162.*

Om du vill:	Gör så här:
Animera grafen stegvis	Tryck ned och släpp genast markörknappen.
Flytta längs visningssfären: Flytta visningspunktens höjd: (ökar och minskar framförallt eyeϕ)	◀ eller ▶ ▲ eller ▲
Animera grafen kontinuerligt	Tryck ned markören i ungefär en sekund och släpp sedan upp den. **TI-89:** Om du vill stoppa, tryck på [ESC], [ENTER], [ON], eller [◆] [␣] (blanksteg). **TI-92 Plus:** Om du vill stoppa, tryck på [ESC], [ENTER], [ON], eller blanksteg.
Växla mellan 4 animeringshastigheter (öka eller minska de stegvisa ändringarna i Window-variabeln eye)	Tryck [+] eller [−].
Ändra visningsvinkel för en icke-animerad graf för att visa den längs x-, y- eller z-axeln	Tryck X, Y respektive Z.
Återställ till ursprungsvärden för eye	Tryck 0 (noll, inte bokstaven O).

Animera en serie grafbilder

Du kan även animera en graf genom att spara en serie bilder av grafen och sedan bläddra genom dessa. Mer information om animering av grafbilder hittar du i kapitel 12. Denna metod ger dig större kontroll över Window-variablerna, speciellt eyeψ (sidan 162) som roterar grafen.

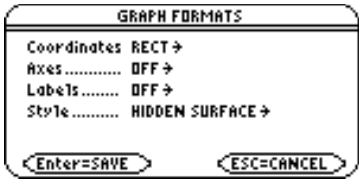
Ändra koordinataxlarna och visningsformatet

Med standardinställningarna visas dolda ytor på en 3D-graf men inte axlarna. Du kan dock ändra på grafformatet när som helst.

Visa dialogrutan GRAPH FORMATS

Gör följande från Y= Editor, Window Editor eller graffönstret:

F1 9
— eller —
TI-89: ◆ I
TI-92 Plus: ◆ F



- I dialogrutan visas de aktuella inställningarna för grafformatet.
- Du avslutar utan att göra några ändringar genom att trycka på ESC.

Du ändrar någon av dessa inställningar med samma procedur som du använder för att ändra andra typer av dialogrutor, exempelvis dialogrutan MODE.

Exempel på inställningar för Axes

Visa de giltiga inställningarna för Axes genom att markera den aktuella inställningen och trycka på ⊙.

$z1(x,y) = x^2+.5y^2$

- AXES – Visar standard-xyz-axlar.

- BOX – Visar tredimensionella lådaxlar.

  Kanterna på lådan bestäms av Window-variablerna xmin, xmax osv.

  I många fall är origo (0,0,0) inuti lådan och inte vid ett hörn.

  Om exempelvis xmin = ymin = zmin = ⁻10 och xmax = ymax = zmax = 10, är origo i mitten av lådan.

Tips! *Att ställa in Labels = ON är användbart när du visar olika typer av 3D-axlar.*

Exempel på inställningar för Style

Du visar de giltiga inställningarna för Style genom att markera den aktuella inställningen och trycka på ⊙.

__Tips!__ WIRE FRAME är snabbare att plotta och kan vara mer praktiskt när du experimenterar med olika former.

- WIRE FRAME – Visar 3D-formen som en genomskinlig trådram.

- HIDDEN SURFACES – Använder skuggning för att skilja på de två sidorna i 3D-formen.

Kommande avsnitt i detta kapitel beskriver CONTOUR LEVELS, WIRE AND CONTOUR (sidan 167) och IMPLICIT PLOT (sidan 171).

Optiska illusioner

Den vinkel i vilken du tittar på en graf (Window-variablerna eyeθ, eyeϕ och eyeψ) kan resultera i optiska illusioner som gör att du tappar perspektivet på en graf.

De flesta optiska illusioner brukar uppstå när eye-vinklar är i en negativ kvadrant av koordinatsystemet.

Optiska illusioner kan vara mer märkbara med lådaxlar. Exempelvis kanske det inte är helt uppenbart vad som är lådans framsida.

__Obs!__ I dessa exempel visas graferna så som de visas i fönstret.

__Obs!__ I dessa exempel används artificiell skuggning (som inte visas i fönstret) för att visa lådans framsida.

Tittar ned ovanifrån xy-planet	Tittar upp underifrån xy-planet
eyeθ = 20, eyeϕ = 55, eyeψ = 0	eyeθ = 20, eyeϕ = 120, eyeψ= 0

Du kan förminska effekterna av optiska illusioner genom att använda dialogrutan GRAPH FORMATS för att ställa in Style = HIDDEN SURFACE.

I en konturgraf ritas en linje som sammanbinder näraliggande punkter i 3D-grafen som har samma värde för z. Detta avsnitt beskriver grafformaten CONTOUR LEVELS samt WIRE AND CONTOUR.

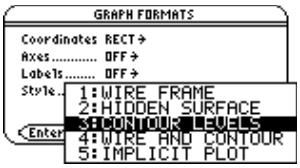
Välj format för grafen

Definiera en ekvation i 3D-grafläge och plotta den på samma sätt som vanligt med följande undantag: Visa dialogrutan GRAPH FORMATS genom att trycka [F1] 9 i Y= Editor, Window editor eller graffönstret. Ställ sedan in:

Style = CONTOUR LEVELS

– eller –

Style = WIRE AND CONTOUR

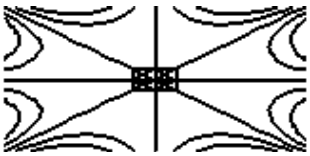
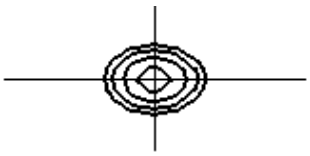
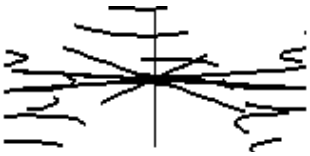
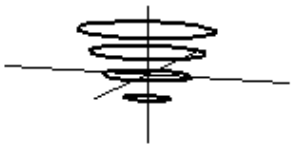
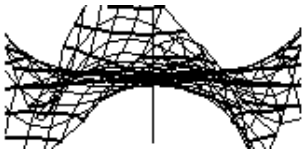
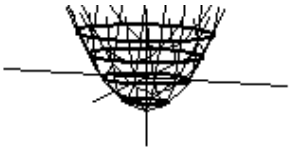


Tips! *Om du vill växla från en formatstil till nästa (och hoppa över* IMPLICIT PLOT*), tryck på:*
TI-89: [|]
TI-92 Plus: F

Obs! *Om du trycker*
TI-89: [|]
TI-92 Plus: F
för att välja CONTOUR LEVELS *ändras inte visningsvinkel, vy eller formatet för* Labels *som det gör när du använder*
TI-89: [◆] [|]
TI-92 Plus: [◆] F

- I visningsläget CONTOUR LEVELS visas endast konturerna.
  - Visningsvinkeln är förinställd att visa konturerna längs z-axeln. Du kan ändra visningsvinkel som du vill.
  - Grafen visas i utökat visningsläge. Växla mellan normalt och utökat läge genom att trycka [×].
  - Format för Labels sätts till OFF automatiskt.
- I visningsläget WIRE AND CONTOUR ritas konturerna i ett wireframe-fönster. Visningsvinkel, visningsläge (utökat eller normalt) och formatet för Labels behåller sina tidigare inställningar.

Obs! *Dessa exempel använder samma Window-variabler för* x-, y- *och* z *som en ZoomStd visningskub. Om du använder dig av ZoomStd, tryck Z för att ställa in siktlinjen till z -axeln.*

Format	**z1(x,y)=(x³y−y³x) / 390**	**z1(x,y)=x²+.5y²−5**
	Synlinjen längs z-axeln	
CONTOUR LEVELS		
	Inställningar eyeθ=20, eyeϕ=70, eyeψ=0	
CONTOUR LEVELS		
WIRE AND CONTOUR		

Obs! *Blanda inte ihop stödlinjerna med konturerna. Konturerna är mörkare.*

Hur bestäms värden för z?

Du kan ställa in Window-variabeln ncontour ([◆] [WINDOW]) för att bestämma antalet konturer som ska ritas upp jämnt fördelat längs z-axeln. Se nedan:

$$\text{stegstorlek} = \frac{\text{zmax} - \text{zmin}}{\text{ncontour} + 1}$$

Konturernas värde för z är:

zmin + stegstorlek
zmin + 2(stegstorlek)
zmin + 3(stegstorlek)
⋮
zmin + ncontour(stegstorlek)

```
eyeθ=20.
eyeϕ=70.
eyeψ=0.
xmin=-10.
xmax=10.
xgrid=14.
ymin=-10.
ymax=10.
ygrid=14.
zmin=-10.
zmax=10.
ncontour=5.
```

Förvalt värde är 5. Du kan ange ett värde från 0 till 20.

Om ncontour=5 och du använder standardfönstret (zmin=⁻10 och zmax=10) är varje steg 3,333. Fem konturer ritas för z=⁻6,666, ⁻3,333, 0, 3,333 och 6,666.

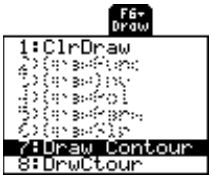
Observera att en kontur inte ritas för ett z-värde om 3D-grafen inte är definierad vid det z-värdet.

Rita en kontur för z-värdet av en markerad punkt interaktivt

Om en konturgraf visas kan du markera en punkt på grafen och rita en kontur för motsvarande z-värde.

1. Öppna menyn Draw genom att trycka på:
   TI-89: [2nd] [F6]
   TI-92 Plus: [F6]
2. Välj 7:Draw Contour.
3. Antingen:
   - Skriv in punktens x -värde och tryck [ENTER]. Skriv sedan in punktens y -värde och tryck [ENTER].

     – eller –
   - Flytta markören till önskad punkt (markören flyttas längs stödrastret). Tryck på [ENTER].



__Tips!__ Existerande konturer försvinner inte från skärmen. Om du vill ta bort tidigare inställda konturer öppnar du fönstret Window editor ([◆] [WINDOW]) och ställer in ncontour=0.

Vi antar t ex att den aktuella grafen är $z1(x,y)=x^2+.5y^2-5$.
Om du anger x=2 och y=3 ritas en kontur för z=3,5.

Rita konturer för specifika z-värden

Öppna menyn Rita i graffönstret och välj 8:DrwCtour. Grundfönstret visas automatiskt med DrwCtour på inmatningsraden. Du kan sedan ange en eller flera värden för z eller generera en sekvens av z-värden.

Några exempel:

__Tips!__ Om du vill ta bort förvalda konturer trycker du [◆] [WINDOW] *och anger* ncontour=0.

DrwCtour 5	Ritar en kontur för z=5.
DrwCtour {1,2,3}	Ritar konturer för z=1, 2 och 3.
DrwCtour seq(n,n, ¯10,10,2)	Ritar konturer för en sekvens av z-värden från ¯10 till 10 i steg om 2 (¯10, ¯8, ¯6, etc.).

Angivna konturer ritas i den aktuella 3D-grafen. (Konturen ritas inte ut om det angivna värdet för z ligger utanför visningskuben eller om 3D-grafen inte är definierad för aktuellt z-värde)

Noteringar om konturplottning

Konturplottning:

- Du kan använda markörtangenterna (sidan 164) för att animera konturplottningen.
- Du kan inte spåra ([F3]) de faktiska konturerna. Däremot kan du spåra linjerna som ritas vid läget Style=WIRE AND CONTOUR.
- Det kan ta en stund att utföra de initiala beräkningarna.
- På grund av de ibland långa beräkningstiderna kan du börja att experimentera med 3D-ekvationer i läget Style=WIRE FRAME. Beräkningstiden är där betydligt kortare. När du är säker på att du har rätt Window-variabler inställda öppnar du dialogrutan för grafformat och ställer in Style=CONTOUR LEVELS eller WIRE AND CONTOUR.
  TI-89: [◆] [|]
  TI-92 Plus: [◆] F

Exempel: Konturer hos en komplex modulär yta

Den komplexa modulära ytan av z(a,b)=abs(f(a+b*i*)) visar alla komplexa nollställen av alla polynom y=f(x). Detta teorem är välkänt inom den komplexa analysen men har inte tidigare kunnat visualiseras.

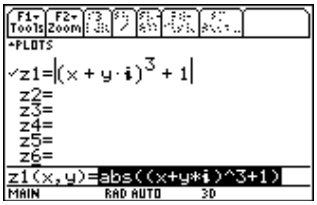
Exempel

I detta exempel antar vi att $f(x)=x^3+1$. Genom att ersätta den generella komplexa formen $x+yi$ med x kan du uttrycka den komplexa ytans ekvation med $z(x,y)=abs((x+y*i)^3+1)$.

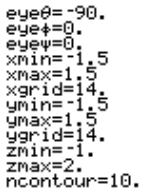
1. Använd [MODE] för att ange Graph=3D.

2. Tryck [◆] [Y=] och definiera ekvationen:

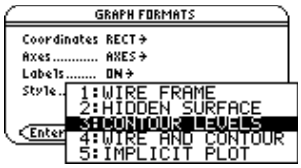
   z1(x,y)=abs((x+y*i)^3+1)



3. Tryck [◆] [WINDOW] och ställ in Window-variablerna enligt bilden.



4. Öppna dialogrutan Grafformat:
   TI-89: [◆] [I]
   TI-92 Plus: [◆] F
   Aktivera axlarna, ställ in Style = CONTOUR LEVELS, och återgå till Window-redigeraren.



5. Tryck [◆] [GRAPH] för att plotta ekvationen.

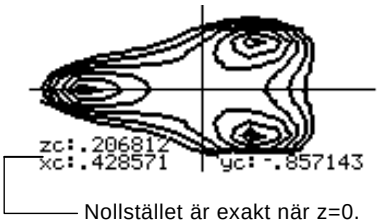
   Det tar en stund att beräkna grafen så ha tålamod. När grafen visas kommer den komplexa modulära ytan att snudda xy -planet exakt vid de komplexa nollställena av polynomet:

   $$-1, \frac{1}{2}+\frac{\sqrt{3}}{2}i, \text{ och } \frac{1}{2}-\frac{\sqrt{3}}{2}i$$

Obs! *För mer exakta uppskattningar kan du öka Window-variablerna xgrid och ygrid. Ökad upplösning kräver dock längre beräkningstid.*

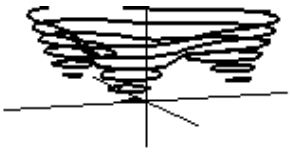
6. Tryck [F3] och flytta spårningsmarkören till nollstället i den fjärde kvadranten.

   Koordinaterna ger uppskattningsvis $.428-.857i$ som nollställe.



Tips! *När du animerar grafen växlar visningsläget till normalt. Använd [×] för att växla mellant normalt och utökat visningsläge.*

7. Tryck [ESC]. Använd sedan markörtangenterna för att animera grafen och visa den från olika eye-vinklar.



Detta exempel visar eyeθ=70, eyeϕ=70 och eyeψ=0.

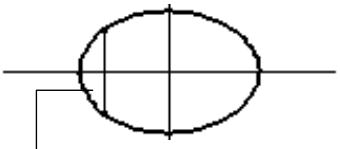
Implicit plottning

Implicit plottning används framförallt för att plotta 2D-grafer i implicit form som inte kan plottas i funktionsläget. Tekniskt sett är en implicit graf en 3D-konturgraf som har plottats med endast en kontur vid z=0.

Explicit och implicit form

I 2D funktionsplottningsläge har ekvationer den explicita formen y=f(x), där y är unikt för varje värde av x.

Många ekvationer är i implicit form som f(x,y)=g(x,y) där du inte explicit kan beräkna y i termer av x eller x i termer av y.



Eftersom y inte är unikt för varje x så kan du inte plotta detta i funktionsläget.

Tips! *Du kan plotta många implicita former om du:*

- *Uttrycker dem som parameterekvationer. Mer information finns i kapitel 7.*
- *Delar upp dem i separata, explicita, funktioner. Mer information finns i kapitel 6.*

Genom att använda implicit plottning i 3D-grafläget kan du plotta dessa implicita ekvationer utan att lösa ut för y eller x.

Ändra den implicita formen till en ekvation med lösningen noll.

f(x,y)−g(x,y)=0

I Y= Editor, skriv in den sida av ekvationen som inte är noll. Detta är giltigt eftersom en implicit beräkning automatiskt sätter svaret till noll.

z1(x,y)=f(x,y)−g(x,y)

I ellipsekvationen till höger skriver du t ex in den implicita formen i Y= Editor.

If $x^2+.5y^2=30$,

then $z1(x,y)=x^2+.5y^2-30$.

Välja grafformat

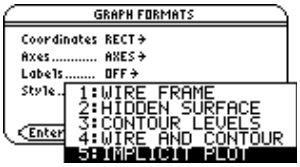
I 3D-grafläge definierar du ekvationer och plottar dessa som alla andra 3D-ekvationer med följande undantag. Visa dialogrutan GRAPH FORMATS genom att trycka ◆ [I] i Y= Editor, Window editor eller graffönstret.

TI-89: ◆ [I]

TI-92 Plus: ◆ F

Ställ sedan in:

Style = IMPLICIT PLOT



Obs! *I graffönstret kan du för att växla mellan de olika grafformaten trycka på:*

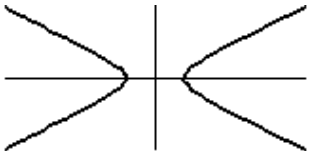
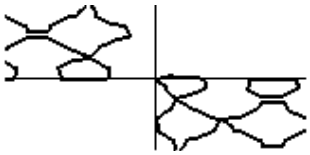
TI-89: [I]
TI-92 Plus: F

för att återgå till IMPLICIT PLOT *måste du dock använda:*

TI-89: ◆ [I]
TI-92 Plus: ◆ F

- Visningsvinkeln ställs initialt in så att du betraktar grafen längs z-axeln. Du kan ändra visningsvinkel efter egna önskemål.
- Grafen visas i utökat visningsläge. Om du vill växla mellan normalt och utökat visningsläge trycker du [×].
- Formatet Labels ställs automatiskt in till OFF.

Obs! *Dessa exempel använder samma Window-variabler för* x-, y- *och* z *som en ZoomStd visningskub. Om du använder dig av ZoomStd, tryck Z för att ställa in synlinjen längs z -axeln.*

Format	$x^2-y^2=4$ z1(x,y)=x^2-y^2-4	sin(x)+cos(y)=$e^{(x*y)}$ z1(x,y)=sin(x)+cos(y)$-e^{(x*y)}$
IMPLICIT PLOT		

Noteringar om implicit plottning

För implicit plottning gäller:

- Windowvariabeln ncontour (sidan 168) har ingen effekt. Endast konturen för z=0 plottas oavsett värdet i ncontour. Den plottade grafen visar var den implicita formen bryter xy-planet.
- Du kan använda markörtangenterna (sidan 164) för att animera grafen.
- Du kan inte spåra ([F3]) de faktiska konturerna. Däremot kan du spåra de osynliga linjerna (wire frame) i grafen som motsvarar 3D-ekvationen.
- Det kan ta en stund att utföra de initiala beräkningarna.
- På grund av de ibland långa beräkningstiderna kan du börja att experimentera med 3D-ekvationer i läget Style=WIRE FRAME. Beräkningstiden för detta är betydligt kortare. När du sedan har ställt in önskade Window-variabler andvänder du [◆] [I] för att ange Style=IMPLICIT PLOT.
  TI-89: [◆] [I]
  TI-92 Plus: [◆] F

Exempel: Implicit plottning av en mer komplicerad ekvation

Du kan använda grafformatet IMPLICIT PLOT för att plotta och animera en mer komplicerad ekvation som inte kan plottas på annat sätt. Även om det kan ta lång tid att beräkna dessa ekvationer är resultatet värt väntan.

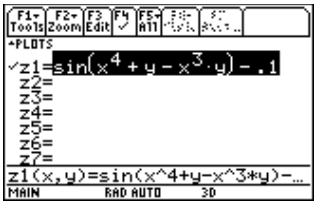
Exempel

Plotta ekvationen sin(x4+y−x3 y) = .1.

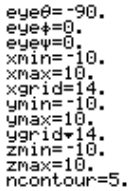
1. Använd [MODE] för att ange Graph=3D.

2. Tryck [◆] [Y=] och definiera ekvationen:

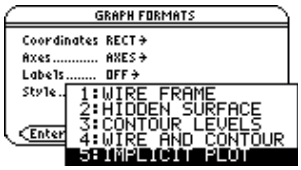
   z1(x,y)=sin(x^4+y−x^3y)−.1



3. Tryck [◆] [WINDOW] och ställ in Windowvariablerna enligt värdena till höger.



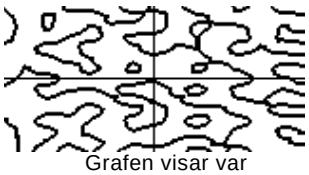
4. Tryck
   TI-89: [◆] [|]
   TI-92 Plus: [◆] F
   Aktivera axlarna, ställ in Style = IMPLICIT PLOT och återgå till Window editor.



Obs!: *För större precision kan du öka värdet för Windowvariablerna* xgrid *och* ygrid*s. Detta förlänger dock beräkningstiden.*

5. Tryck [◆] [GRAPH] för att plotta ekvationen.

   Det tar en stund att utföra beräkningarna så ha tålamod.



Grafen visar var $\sin(x^4+y-x^3y) = .1$

Tips! *När du animerar grafen växlar visningsläget till normal. Tryck* [×] *för att växla mellan normalt och utökat visningsläge.*

6. Använd markörtangenterna för att animera grafen och visa den från olika vinklar.

Exemplet visar eyeθ= ¯127.85, eyeϕ=52.86 och eyeψ= ¯18.26.

Plottning av differentialekvationer

11

Plottning av differentialekvationer - översikt 176
Plottning av lösningar till differentialekvationer - steg för steg 178
Skillnader mellan funktionsplottning och plottning av lösningar till differentialekvationer 179
Ange begynnelsevillkor 184
Definiera ett system för ekvationer av högre ordning 186
Exempel på en ekvation av andra ordningen 187
Exemple på en ekvation av tredje ordningen 189
Ställa in axlar till Time eller Custom 190
Exempel på axlar (Time och Custom) 191
Jämförelse av RK och Euler 193
Exempel på funktionen deSolve( ) 196
Felsökning av grafformat 197

I detta kapitel beskrivs hur du kan lösa differentialekvationer grafiskt med din TI-89 / TI-92 Plus. Innan du påbörjar detta kapitel bör du vara insatt i funktionerna i kapitel 6: Grundläggande plottning.

Din TI-89 / TI-92 Plus kan lösa första ordningens ordinära differentialekvationer som t ex:

y' = .001 y ∗ (100 − y)

eller kopplade första ordningens differentialekvationer som:

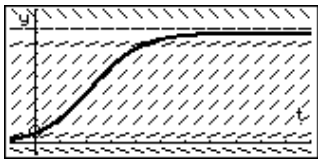
y1' = ⁻y1 + 0.1 ∗ y1 ∗ y2
y2' = 3 ∗ y2 − y1 ∗ y2

Du kan lösa andra- och tredje ordningens ekvationer (eller högre) genom att dela upp dem i första ordningens ekvationer som i exemplet:

y'' + y = sin(t) kan definieras som y1' = y2
y2' = ⁻y1 + sin(t)

Genom att ange begynnelsevillkor kan du plotta olika lösningskurvor till en differentialekvation.

Du kan även plotta ett lutningsfält eller ett riktningsfält som hjälper dig att visualisera hur en grupp av lösningskurvor ser ut.

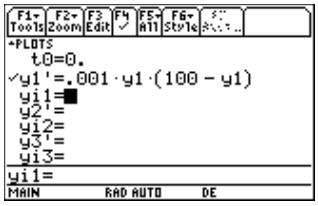
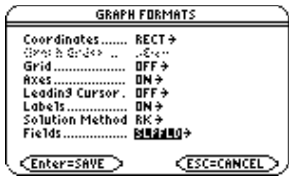
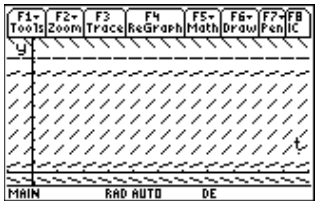


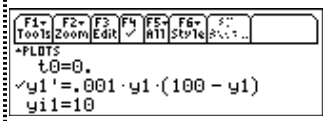
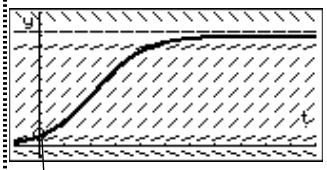
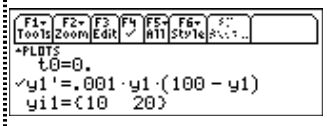
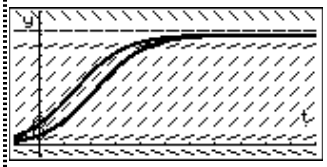
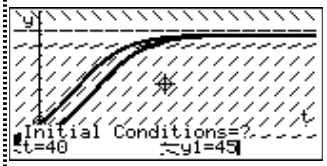
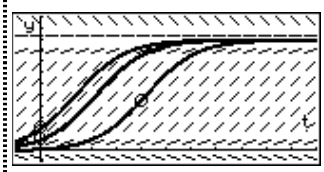
Vid plottning använder TI-89 / TI-92 Plus numeriska metoder som uppskattar de exakta lösningarna. Med den nya funktionen **deSolve()** kan du lösa en del differentialekvationer symboliskt. I detta kapitel finns en kort introduktion till **deSolve()**. Se appendix A för detaljerad information.

Obs! *En differentialekvation är:*

- *av första ordningen* när endast förstaderivata anges.
- *ordinär* när alla derivator som förekommer beror av en och samma oberoende variabel.

Plotta lösningen till den logistiska första ordningens ekvation y' = .001y∗ (100− y). Börja med att endast plotta riktningsfältet. Skriv sedan in startvärden i Y= Editor och interaktivt i plottningsfönstret.

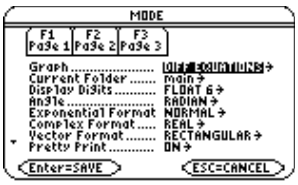
Steg	TI-89 Tangenter	TI-92 Plus Tangenter	Teckenfönster
1. Öppna dialogrutan MODE. För inställningen Graph, välj DIFF EQUATIONS.	[MODE] ▶ 6 [ENTER]	[MODE] ▶ 6 [ENTER]	
2. Öppna och töm Y= Editor. Definiera ekvationen: y1'(t)=.001y1∗ (100− y1) *Tryck [×] för att skriva in ∗som används ovan. Använd inte implicit multiplikation mellan variablerna och parenteserna. Om du gör detta hanteras det som ett funktionsanrop.* *Skriv inte in något startvärde för* yi1.	◆ [Y=] [F1] 8 [ENTER] [ENTER] . 0 0 1 Y 1 [×] [(] 1 0 0 [−] Y 1 [)] [ENTER]	◆ [Y=] [F1] 8 [ENTER] [ENTER] . 0 0 1 Y 1 [×] [(] 1 0 0 [−] Y 1 [)] [ENTER]	 **Viktigt!** Med y1' markerat kommer din TI-89 / TI-92 Plus att plotta lösningskurvan för y1, inte derivatan y1'.
3. Öppna dialogrutan GRAPH FORMATS. Ställ in Axes = ON, Labels = ON, Solution Method = RK och Fields = SLPFLD. ***Viktigt!** När du vill plotta lösningen till en differentialekvation måste* Fields *vara satt till* SLPFLD *eller* FLDOFF. *Om* Fields=DIRFLD, *uppstår ett fel då du försöker plotta grafen.*	◆ [I] ▼ ▼ ▶ 2 ▼ ▼ ▶ 2 ▼ ▶ 1 ▼ ▶ 1 [ENTER]	◆ F ▼ ▼ ▶ 2 ▼ ▼ ▶ 2 ▼ ▶ 1 ▼ ▶ 1 [ENTER]	
4. Öppna Window Editor och ställ in Windowvariablerna enligt exemplet till höger.	◆ [WINDOW] 0 ▼ 1 0 ▼ . 1 ▼ 0 ▼ [(-)] 1 0 ▼ 1 1 0 ▼ 1 0 ▼ [(-)] 1 0 ▼ 1 2 0 ▼ 1 0 ▼ 0 ▼ . 0 0 1 ▼ 2 0	◆ [WINDOW] 0 ▼ 1 0 ▼ . 1 ▼ 0 ▼ [(-)] 1 0 ▼ 1 1 0 ▼ 1 0 ▼ [(-)] 1 0 ▼ 1 2 0 ▼ 1 0 ▼ 0 ▼ . 0 0 1 ▼ 2 0	t0=0. tmax=10. tstep=.1 tplot=0. xmin=-10. xmax=110. xscl=10. ymin=-10. ymax=120. yscl=10. ncurves=0. diftol=.001 fldres=20.
5. Öppna graffönstret. *Eftersom du inte angav något startvärde ritas bara riktningsfältet (enligt specifikationen* Fields=SLPFLD *i dialogrutan* GRAPH FORMATS*).*	◆ [GRAPH]	◆ [GRAPH]	

Steg	TI-89 Tangenter	TI-92 Plus Tangenter	Teckenfönster
6. Återgå till Y= Editor och skriv in ett startvärde: yi1=10	◆ [Y=] ENTER 1 0 ENTER	◆ [Y=] ENTER 1 0 ENTER	
7. Återgå till graffönstret. *Begynnelsevärdet som skrivs in i Y= Editor sparas alltid vid positionen t0. Grafen påbörjas vid startvärdet och plottas först till höger och sedan till vänster.*	◆ [GRAPH]	◆ [GRAPH]	 Begynnelsevärdet är markerat med en cirkel.
8. Gå tillbaks till Y= Editor och ändra yi1 så att den innehåller två begynnelsevärden som en lista: yi1={10,20}	◆ [Y=] ⊖ ENTER 2nd [{] 1 0 , 2 0 2nd [}] ENTER	◆ [Y=] ⊖ ENTER 2nd [{] 1 0 , 2 0 2nd [}] ENTER	
9. Återgå till graffönstret.	◆ [GRAPH]	◆ [GRAPH]	
10. Om du vill välja ett startvärde interaktivt, tryck på: **TI-89:** 2nd [F8] **TI-92 Plus:** [F8] Mata in t=40 och y1=45 vid prompten. *När du anger ett värde interaktivt kan du ange andra värden för t än t0 som du skriver in i Y= Editor eller Window Editor.* *I stället för att skriva in t och y1 efter att du tryckt på **TI-89:** 2nd F8 **TI-92 Plus:** F8, kan du flytta markören på skärmen och sedan trycka på ENTER.* *Du kan använda F3 om du vill spåra kurvor för begynnelsevillkor som du skrivit in i Y= Editor. Du kan däremot inte spåra kurvor för startvärden som angetts interaktivt.*	2nd [F8] 4 0 ENTER 4 5 ENTER	F8 4 0 ENTER 4 5 ENTER	 

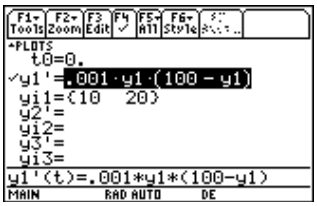
När du vill plotta lösningar till differentialekvationer använder du samma grundsteg som används för funktioner av typen y(x), vilket beskrivs i kapitel 6: Grunderna i funktionsplottning. Skillnaderna beskrivs på följande sidor.

Plotta differential-ekvationer

Ställ in grafläget ([MODE]) till DIFF EQUATIONS. Ställ också in Angle, om nödvändigt.



Definiera ekvationerna och eventuellt startvärden i Y= Editor ([◆] [Y=]).



Tips! Om du vill stänga av alla statistikplottningar, tryck på [F5] 5 *eller använd* [F4] *för att avmarkera dem. Se kapitel 16.*

Välj ([F4]) vilka ekvationer som ska plottas.

Ställ in visningsstil för en ekvation.

TI-89: [2nd] [F6]

TI-92 Plus: [F6]

Ange grafformat. Lösningsmetod och Fält är unika för differentialekvationer.

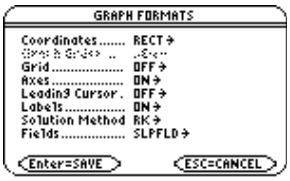
[F1] 9

— eller —

TI-89: [◆] [I]

TI-92 Plus: [◆] F

Obs! Inställningen för Fields *är kritisk beroende på ekvationens ordning (sidan 197).*



Obs! Giltiga inställningar för Axes *beror på inställningen för* Fields *(sidorna 190 och 197).*

Ställ in axlarna utifrån formatet för Fält.

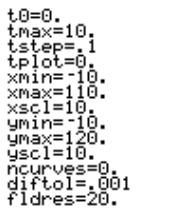
TI-89: [2nd] [F7]

TI-92 Plus: [F7]

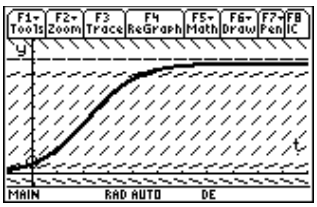
Obs! Beroende på inställningarna för Solution Method *och* Fields *visas olika Windowvariabler.*

Tips! [F2] Zoom *ändrar också visningsfönster.*

Definiera visningsfönster ([◆] [WINDOW]).



Plotta ekvationerna ([◆] [GRAPH]).



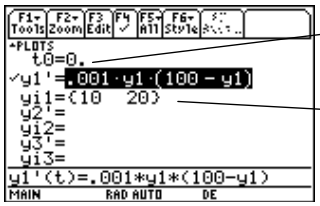
Skillnader mellan funktionsplottning och plottning av lösningar till differentialekvationer

I detta kapitel förutsätts att du redan känner till plottning av y(x)-funktioner som beskrivs i kapitel 6: Grunderna i funktionsplottning. Skillnader beskrivs på följande sidor.

Ställa in värden för Graph läget

Använd [MODE] för att ange Graph = DIFF EQUATIONS innan du definierar differentialekvationer eller ställer in Windowvariabler. I Y= Editor och Window Editor kan du bara skriva in värden utifrån den aktuella inställningen för Graph.

Definiera differential-ekvationer med Y= Editor



Använd t0 för att ange begynnelsevärde för t. Du kan även ställa in t0 i Window Editor.

Använd yi för att ange ett eller flera begynnelsevärden för motsvarande differentialekvation.

Du kan definiera differential-ekvationerna y1'(t) till y99'(t).

***Tips!** Du kan använda kommandot* **Define** *från grundfönstret när du vill definiera funktioner och ekvationer.*

Använd inte formatet y(t) när du skriver in ekvationer i Y= Editor. Till exempel:

Skriv så här: y1' = .001y1∗ (100− y1)
Inte så här: y1' = .001y1(t)∗ (100− y1(t))

Använd inte implicit multiplikation mellan en variabel och ett uttryck inom en parentes. Om du gör så hanteras det som ett funktionsanrop.

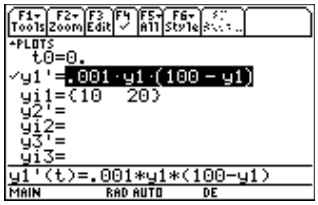
Du kan bara skriva in första ordningens ekvationer i Y= Editor. Om du vill plotta andra eller tredje ordningens ekvationer måste du ange dem som ett system av första ordningens ekvationer. Mer information hittar du på sidan 186.

Detaljerad information om hur du anger begynnelsevillkor finns på sidan 184.

Välja differential-ekvationer

Du kan använda [F4] för att markera en differentialekvation men inte för att ange dess begynnelsevillkor.

Viktigt: Val av y1' plottar lösningskurvan för y1, inte derivatan för y1', beroende på axelinställningarna.



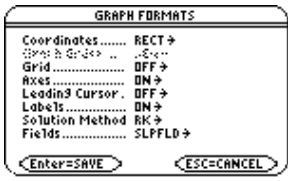
Välja visningsstil

På menyn Style finns bara alternativen Line, Dot, Square, Thick, Animate och Path tillgängliga. Dot och Square markerar bara de diskreta värden (i tstep steg) vid vilka lösningen till differentialekvationen plottas.

TI-89: [2nd] [F6]
TI-92 Plus: [F6]

Ställa in grafformat

I fönstret Y= Editor, Window Editor eller skärmen Graph, tryck:
[F1] 9
— eller —
TI-89: [◆] [I]
TI-92 Plus: [◆] F



De format som påverkas av differentialekvationer är:

Grafformat	Beskrivning
Graph Order	Ej tillgängligt.
Solution Method	Anger vilken metod som används för att lösa ekvationen. • RK — Runge-Kuttas metod. Mer information om den algoritm som används för denna metod finns i appendix B. • EULER — Eulers metod. De olika metoderna låter dig välja mellan större noggrannhet eller snabbare beräkningar. RK är noggrannare än EULER men tar längre tid för beräkningarna.
Fields	Anger om ett fält ska ritas för lösningen till differentialekvationen. • SLPFLD — Ritar ett riktningsfält för en första ordningens ekvation med t på x-axeln och lösningen på y-axeln. Mer information om hur riktningsfält används finns i exemplet på sidan 176. • DIRFLD — Ritar ett fasplan för en andra ordningens ekvation (eller för ett system med två första ordningens ekvationer). Axlarnas användning bestäms av deras inställning. Mer information om hur fasplan används finns i exemplet på sidan 187. • FLDOFF — Visar inget fält. Detta kan användas för alla ekvationer, men du måste använda det för tredje ordningens ekvationer och ekvationer av högre ordning. Du måste skriva in samma begynnelsevillkor för alla ekvationerna i Y= Editor (sidan 184). Se även exemplet på sidan 189.

***Viktigt!** Inställningen för grafformatet* Fields *är viktigt för korrekt plottning av lösningar till differential-ekvationer. Mer information hittar du under felsökning på sidan 197.*

***Tips!** Om du trycker på* [ENTER] *medan ett fält plottas stannar räknaren i pausläge när fältet är ritat och innan lösningen plottas. Tryck på* [ENTER] *igen för att fortsätta.*

***Tips!** Om du vill avbryta en plottning trycker du på* [ON].

Ställa in axlarna

I Y= Editor är Axes tillgängligt (eller inte tillgängligt) beroende på aktuellt grafformat.

Om Axes tillgängligt kan du välja axlar som används vid plottning av lösningen till differentialekvationen. Mer information finns på sidan 190.
TI-89: [2nd] [F7]
TI-92 Plus: [F7]

Axlar	Beskrivning
TIME	Plottar t på x-axeln och y (lösningarna till de valda differentialekvationerna) på y-axeln.
CUSTOM	Du kan själv välja x- och y-axel.

Windowvariabler

Grafer av lösningar till differentialekvationer använder nedanstående Windowvariabler. Beroende på Solution Method och grafformatet i Fields visas inte alla nedanstående variabler i Window Editor ([◆] [WINDOW]) samtidigt.

Variabel	Beskrivning
t0	Tidpunkt vid vilken begynnelsevillkoren som angetts i Y= Editor inträffar. Du kan ange t0 i Window Editor och i Y= Editor. (Om du ställer in t0 i Y= Editor sätts tplot automatiskt till samma värde.)
tmax, tstep	Används för att bestämma värdet för t när ekvationen plottas: y'(t0) y'(t0+tstep) y'(t0+2∗ tstep) ... får inte överskrida... y'(tmax) Om Fields = SLPFLD så ignoreras tmax. Lösningen plottas stegvis från t0 till båda ytterkanterna av skärmen enligt värdet i tstep.
tplot	Först plottas värdet för t. Om detta inte är ett stegvis värde för tstep börjar plottningen vid nästa värde för tstep. I vissa ekvationer är den första punkten som beräknas och plottas vid t0 kanske inte visuellt intressant. Genom att ange tplot till ett högre värde än t0 kan du påbörja plottningen i ett intressant område och därigenom minska beräkningstiden och undvika onödig information i graffönstret.

Obs! *Om* tmax < t0, *måste* tstep *ha ett negativt värde.*

Obs! *Om* Fields=SLPFLD *så ignoreras* tplot *och antas ha samma värde som* t0.

Windowvariabler (fortsättning)

xmin, xmax, ymin, ymax	Gränser för visningfönstret.
xscl, yscl	Avstånd mellan skalstreck på x- och y-axeln.
ncurves	Antal lösningskurvor (0 till 10) som ritas automatiskt om du inte anger ett startvärde. Förvald inställning är ncurves = 0.

__Obs!__ Information om hur inställningarna för Fields påverkar användningen av ncurves finns på sidan 184.

När ncurves används sätts t0 temporärt till skärmens mittpunkt och startvärdena distribueras jämnt längs y-axeln där:

$$\text{stegstorlek} = \frac{\text{ymax} - \text{ymin}}{\text{ncurves} + 1}$$

Värdena y för startvärden är:

ymin + stegstorlek
ymin + 2∗ (stegstorlek)
⋮
ymin + ncurves∗ (stegstorlek)

diftol	(Endast Solution Method = RK) Toleransen som används av metoden RK för att bestämma en steglängd för lösningen av ekvationen måste vara ≥1E⁻14.
fldres	(Endast Fields = SLPFLD eller DIRFLD) Antal kolumner (1 till 80) som används för att rita ett riktningsfält eller ett fasplan över hela skärmens bredd.
Estep	(Endast Solution Method = EULER) Iterationer mellan värden för tstep måste vara ett heltal >0. För högre noggrannhet kan du öka värdet för Estep utan att plotta extra punkter.
dtime	(Endast Fields = DIRFLD) Tidpunkt vid vilken fasplanet ritas.

Standardvärden (som anges om du väljer 6:ZoomStd från verktygsmenyn F2 Zoom) är:

t0 = 0.	xmin = ⁻1.	ymin = ⁻10.	ncurves = 0.
tmax = 10.	xmax = 10.	ymax = 10.	diftol = .001
tstep = .1	xscl = 1.	yscl = 1.	Estep = 1.
tplot = 0.			fldres = 14.
			dtime = 0.

Du kan vara tvungen att ändra standardvärdet för t-variablerna för att garantera att tillräckligt antal punkter plottas.

Systemvariabeln fldpic

När ett fält ritas så sparas automatiskt en bild av fältet i systemvariablen fldpic. Om du sedan utför en operation som plottar om lösningen till ekvationen, men inte påverkar fältet, så använder din TI-89 / TI-92 Plus bilden i fldpic istället för att plotta om fältet. Detta kan snabba upp plottningen avsevärt.

fldpic raderas automatiskt när du lämnar läget för differentialekvationer eller visar en graf i läget Fields = FLDOFF.

Utforska en graf

Liksom vid funktionsplottning kan du utforska en graf med nedanstående verktyg. Koordinater visas i rektangulär eller polär form enligt inställningarna för grafformatet.

Verktyg	För grafer av differentialekvationer:
Frihands-markör	Fungerar på samma sätt som för funktionsgrafer.
[F2] Zoom	Fungerar på samma sätt som för funktionsgrafer. • Endast Windowvariablerna x (xmin, xmax, xscl) och y (ymin, ymax, yscl) påverkas. • Windowvariablerna t (t0, tmax, tstep, tplot) påverkas inte såvida du inte väljer 6:ZoomStd (vilket återställer alla Windowvariabler till sina standardvärden).
[F3] Trace	Används om du vill flytta markören längs kurvan ett steg (tstep) i taget. Om du vill flytta markören ungefär tio punkter i taget trycker du på [2nd] ▶ eller [2nd] ◀. Om du anger startvärden i Y= Editor eller använder Windowvariabeln ncurves för att plotta kurvorna automatiskt kan du spåra kurvorna. Om du använder: **TI-89:** [2nd] [F8] **TI-92 Plus:** [F8] Dvs. IC i graffönstret för att ange startvärden kan du inte spåra kurvorna. QuickCenter gäller för alla riktningar. Om du flyttar markören utanför skärmen (upptill, nedtill, vänster eller höger), tryck på [ENTER] för att centrera markören och grafen i fönstret. Använd ▲ eller ▼ för att visa resultaten på alla plottade kurvor.
[F5] Math	Endast 1:Value finns tillgängligt. • Med axlarna TIME visas lösningen för y(t) (representerat av yc) för ett specificerat t-värde. • Med axlarna CUSTOM beror värdena som hör ihop med x och y på de axlar du väljer.

Tips! *Under en spårning kan du flytta markören till en särskild punkt genom att skriva in ett värde för t och sedan trycka på [ENTER].*

Tips! *Du kan använda QuickCenter när som helst under en spårning, även om markören fortfarande är kvar på skärmen.*

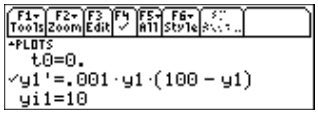
Ange begynnelsevillkor

Du kan ange begynnelsevillkor i Y= Editor, låta din TI-89 / TI-92 Plus beräkna begynnelsevillkor automatiskt eller välja dem interaktivt från graffönstret.

Ange startvärden i Y= Editor

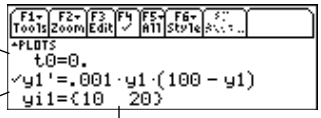
Du kan ange ett eller flera begynnelsevillkor i Y= Editor. Om du vill ange fler än ett villkor skriver du in dem som en kommaseparerad lista med parenteser { } runt.

När du vill ange begynnelsevillkor för ekvationen y1' använder du raden yi1 osv.



Använd t0 när du vill ange vid vilket t ett begynnelsevillkor gäller. Dett är också det första t som utvärderas för grafen.

Plotta en grupp av grafer genom att ange en lista med startvärden.

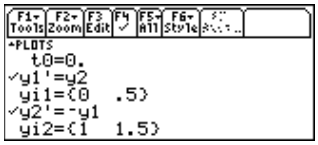


Skriv in {10,20} även om {10 20} visas.

__Obs!__ Information om hur du skapar ett system för andra och tredje ordningens ekvationer eller högre finns på sidan 186.

För andra ordningens ekvationer eller högre måste du definiera ett system med första ordningens ekvationer i Y= Editor.

Om du anger begynnelsevillkor så måste du ange samma antal villkor för varje ekvation i systemet. Om du inte gör det uppstår ett fel av typen Dimension error.



Om du inte anger ett begynnelse-villkor i Y= Editor

Om du inte anger begynnelsevillkor så bestämmer Windowvariabeln ncurves (◆ [WINDOW]) antalet lösningskurvor som ska plottas automatiskt. Förvalt värde är ncurves = 0. Du kan ange ett värde mellan 0 och 10. Grafformatet inställt i Fields och inställningen för Axes bestämmer dock om ncurves används.

__Tips!__ Om du inte anger begynnelsevillkor kan du använda SLPFLD *(med* ncurves=0) *eller* DIRFLD *om du bara vill visa ett riktningsfält eller ett fasplan.*

__Obs!__ SLPFLD *kan bara användas för en enkel första ordningens ekvation.* DIRFLD *används endast för andra ordningens ekvationer (eller för system med två ekvationer av första ordningen).*

Om Fields =	Gäller följande:
SLPFLD	ncurves används för att plotta grafer om det inte är satt till noll.
DIRFLD	ncurves ignoreras. Inga kurvor plottas.
FLDOFF	ncurves används om Axes = TIME (eller om Axes = Custom och x-axeln är t). I andra fall uppstår ett fel av typen Diff Eq setup.

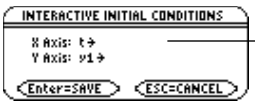
När ncurves används sätts t0 temporärt till skärmens mittpunkt. Värdet hos t0 som det ställts in i Y= Editor eller Window Editor ändras inte.

Välja begynnelsevillkor interaktivt från graffönstret

Obs! Med inställningen SLPFLD *eller* DIRFLD *kan du välja begynnelsevillkor interaktivt även om du angett begynnelsevillkor i Y= Editor.*

Obs! Om du valt FLDOFF *kan du ange begynnelsevillkor interaktivt. Om du skriver in tre eller fler ekvationer så måste du ange ett villkor (inte en lista) för varje ekvation i Y= Editor. Om du inte gör detta uppstår ett fel av typen* Dimension error.

När lösningen till en differentialekvation plottas (oavsett om en lösningskurva visas) kan du markera en punkt i graffönstret och använda den som begynnelsevillkor.

Om Fields =	Gör så här:
SLPFLD – eller – DIRFLD	1. Tryck på: **TI-89:** [2nd] [F8] **TI-92 Plus:** [F8] 2. Ange ett begynnelsevillkor genom att antingen: • Flytta markören till önskad punkt och tryck på [ENTER]. – eller – • Skriv in ett värde för var och en av koordinaterna och tryck på [ENTER]. – Om du valt SLPFLD (endast första ordningens ekvationer) skriver du in värden för t0 och y(t0). – Om du valt DIRFLD (endast andra ordningens ekvationer eller system med två första ordningens ekvationer) skriver du in begynnelsevillkor för båda y(t0) där t0 är värdet som angetts i Y= Editor eller Window Editor. En cirkel visar startvärdet och lösningskurvan plottas.
FLDOFF	1. Tryck på: **TI-89:** [2nd] [F8] **TI-92 Plus:** [F8] Du uppmanas att välja de axlar som du vill ange ett startvärde för.  t är en giltig markering. Med den kan du ange ett värde för t0. Dina val används för axlarna i grafen. 2. Du kan acceptera de förvalda värdena eller skriva in egna. Tryck sedan på [ENTER]. 3. Ange begynnelsevillkor enligt beskrivningarna för SLPFLD eller DIRFLD.

Att observera vid spårning av lösningskurva

När du anger startvärden i Y= Editor eller använder ncurves för att plotta kurvorna automatiskt kan du använda [F3] för att spåra kurvorna.

Du kan däremot inte spåra en kurva där du angett begynnelsevillkor interaktivt. Dessa kurvor plottas inte utan ritas upp.

I Y= Editor måste du skriva in alla ekvationer som första ordningens ekvationer. Om du har en *n:e*-ordningens ekvation måste du konvertera den till ett system med *n* första ordningens ekvationer.

Konvertera en ekvation till ett första ordningens system

Ett system med flera ekvationer kan definieras på flera olika sätt. Nedan följer en allmän metod.

1. Skriv om den ursprungliga differentialekvationen efter behov.
   a. Lös ekvationen för sistaderivatan.
   b. Uttryck lösningen i termer av y och t.
   c. I ekvationens högra led ersätter eller tar du bort alla referenser till derivatavärden.

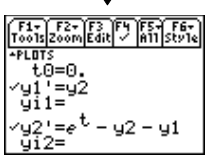
I stället för:	Använd:
y	y1
y'	y2
y''	y3
y'''	y4
$y^{(4)}$	y5
⋮	⋮

   d. I ekvationens vänstra led ersätter du alla derivatavärden enligt nedanstående.

I ställer för:	Använd:
y'	y1'
y''	y2'
y'''	y3'
$y^{(4)}$	y4'
⋮	⋮

2. På användbara rader i Y= Editor skriver du in systemet av ekvationer som:

y1' = y2
y2' = y3
y3' = y4
– upp till –
y*n*' = *din n:e-grads ekvation*



$y'' + y' + y = e^x$

$y'' = e^x - y' - y$

$y'' = e^t - y' - y$

$y'' = e^t - y2 - y1$

Ersätt inte på vänster sida nu.

$y2' = e^t - y2 - y1$

Obs! *Du skapar en första ordningens ekvation genom att se till så att högra sidan av ekvationen inte innehåller några derivatavariabler.*

Obs! *Utifrån de ersättningar som gjorts ovan motsvarar y'-raderna i Y= Editor:*
y1' = y'
y2' = y''
osv.
Detta exempel med ekvation av andra ordningen skrivs alltså in på raden y2'.

I system som detta är lösningen till ekvationen y1' lösningen till ekvationen av *n:e* ordningen. Du kan avmarkera alla andra ekvationer ur systemet.

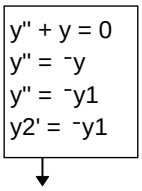
Andra ordningens differentialekvation y''+y = 0 representerar en enkel harmonisk svängning. Konvertera detta till ett system med ekvationer för Y= Editor. Plotta sedan lösningen för startvärdena y(0) = 0 och y'(0) = 1.

Exempel

1. Tryck på [MODE] och ställ in Graph=DIFF EQUATIONS.

2. Definiera ett system med ekvationer för andra ordningens ekvation enligt beskrivningen på sidan 186.

   Skriv om ekvationen och ersätt de variabler som behövs.

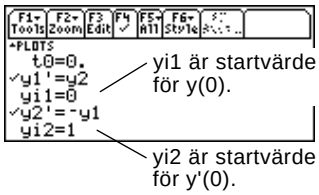


__Obs!__ t0 är den tidpunkt vid vilken begynnelsevillkoret inträffar. Det är också det första värdet för t som utvärderas för grafen. Förvalt värde är t0=0.

3. I Y= Editor ([◆][Y=]) skriver du in systemet med ekvationer.

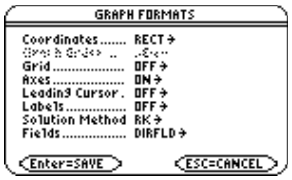
4. Ange begynnelsevillkor:

   yi1=0 och yi2=1



__Viktigt!__ För andra ordningens ekvationer måste du ange Fields=DIRFLD *eller* FLDOFF.

5. Tryck på:
   [F1] 9
   — eller —
   TI-89: [◆] [I]
   TI-92 Plus: [◆] F
   och ställ in Axes = ON, Labels = OFF, Solution Method = RK och Fields = DIRFLD.



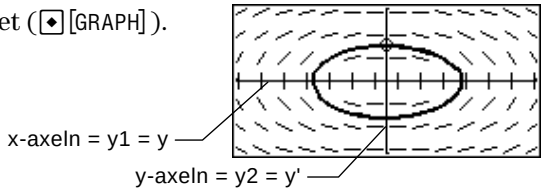
__Viktigt!__ Fields=DIRFLD *kan inte plotta längs en tidsaxel. Ett fel av typen* Invalid Axes *uppstår om* Axes=TIME *eller om t är satt till en axel av typen* CUSTOM.

6. I Y= Editor, tryck på:
   TI-89: [2nd] [F7]
   TI-92 Plus: [F7]
   och kontrollera att Axes = CUSTOM med y1 och y2 längs axlarna.

7. Ställ in Window-variablerna i Window Editor ([◆][WINDOW]).

t0=0.	xmin=⁻2.	ncurves=0.
tmax=10.	xmax=2.	diftol=.001
tstep=.1	xscl=1.	fldres=14.
tplot=0.	ymin=⁻2.	dtime=0.
	ymax=2.	
	yscl=1.	

8. Öppna graffönstret ([◆][GRAPH]).

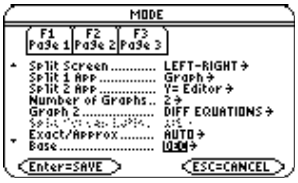


Om du väljer ZoomSqr ([F2] 5) ser du att banan i fasplanet faktiskt är en cirkel. Om du använder ZoomSqr kommer dina Windowvariabler emellertid att ändras.

Använd en delad skärm för att utforska den harmoniska oscillatorn i detalj. Du kan plotta hur y och y' ändras med avseende på tiden (t).

***Obs!** När du vill visa olika grafer i båda delarna av en delad skärm måste använda läget 2-graph.*

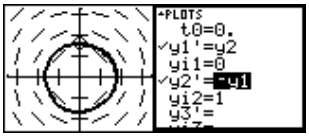
9. Tryck på [MODE] och ändra inställningarna enligt sidan 2, som visas i exemplet till höger. Stäng dialogrutan och grafen ritas om.



10. Tryck på [2nd] [⊞] för att växla till den högra delen av den delade skärmen.

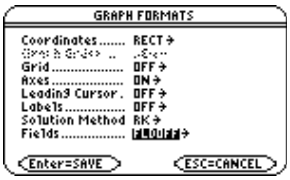
11. Använd [F4] och välj y1' och y2'.

    Den högra sidan använder samma ekvation som den vänstra. Inga ekvationer är dock initialt valda i den högra sidan.



***Viktigt!** Eftersom du inte kan plotta längs en tidsaxel med* Fields=DIRFLD *måste du ändra inställningen för* Fields. FLDOFF *stänger av alla fält.*

12. Tryck på:
    [F1] 9
    — eller —
    TI-89: [◆] [I]
    TI-92 Plus: [◆] F
    och ställ in Fields = FLDOFF.



13. I Y= Editor, tryck på:
    TI-89: [2nd] [F7]
    TI-92 Plus: [F7]
    och kontrollera att Axes = TIME.

***Obs!** När du anger läget för 2 grafer sätts alla Windowvariabler för den högra sidan till förvalda värden.*

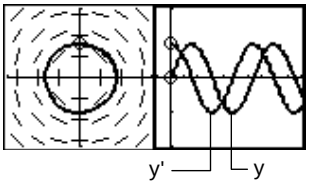
14. I Window Editor, ändra ymin och ymax enligt exemplet till höger.

ymin= ⁻ 2.

ymax=2.

15. Tryck på [◆] [GRAPH] för att öppna graffönstret för graf 2.

    Den vänstra sidan visar banan i fasplanet. På den högra sidan visas lösningskurvan och dess derivata.



16. Återgå till fullskärmsvisning av orginalgrafen genom att först trycka på [2nd] [⊞] för att växla till vänster sida. Tryck sedan på [MODE] och ändra inställningen för Split Screen.

Split Screen = FULL

Exemple på en ekvation av tredje ordningen

Skriv ett system av ekvationer för en ekvation av tredje ordningen y'''+2y''+2y'+y = sin(x) i Y= Editor. Plotta lösningen som en funktion av tiden. Använd startvärdena y(0) = 0, y'(0) = 1 och y''(0) = 1.

Exempel

1. Tryck på [MODE] och ställ in Graph=DIFF EQUATIONS.

2. Definiera ett system av ekvationer för tredje ordningens ekvation enligt beskrivningen på sidan 186.

   Skriv om ekvationen och ersätt eventuella variabler.

y''' + 2y'' + 2y' + y = sin(x)
y''' = sin(x) – 2y'' – 2y' – y
y''' = sin(t) – 2y'' – 2y' – y
y''' = sin(t) – 2y3 – 2y2 – y1
y3' = sin(t) – 2y3 – 2y2 – y1

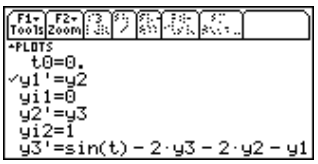
Obs! t0 är den tidpunkt vid vilken begynnelsevillkoret börjar gälla. Förvalt värde är t0=0.

3. I Y= Editor ([◆] [Y=]) skriver du in systemet med ekvationer.

4. Ange begynnelsevillkor:

   yi1=0, yi2=1, och yi3=1

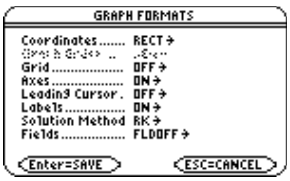
5. Kontrollera att endast y1' är markerat. Använd [F4] för att avmarkera alla andra ekvationer.



Viktigt! Lösningen till ekvationen y1' är lösningen på tredje ordningens ekvation.

Viktigt! För tredje ordningens ekvationer eller högre måste du ställa in Fields=FLDOFF. *I annat fall uppstår ett fel av typen* Undefined vid plottningen.

6. Tryck på:
   [F1] 9
   — eller —
   TI-89: [◆] [I]
   TI-92 Plus: [◆] F
   och ställ in Axes = ON, Labels = ON, Solution Method = RK och Fields = FLDOFF.



Obs! Med inställningen Axes=TIME *plottas lösningen till ekvationen över tid (t).*

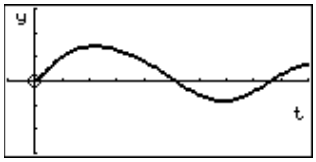
7. I Y= Editor, tryck på:
   TI-89: [2nd] [F7]
   TI-92 Plus: [F7]
   och ställ in Axes = TIME.

8. Ställ in Windowvariablerna i Window Editor ([◆] [WINDOW]).

t0=0.	xmin=⁻1.	ncurves=0.
tmax=10.	xmax=10.	diftol=.001
tstep=.1	xscl=1.	
tplot=0.	ymin=⁻3.	
	ymax=3.	
	yscl=1.	

Tips! Om du vill hitta en lösning vid en viss tidpunkt använder du [F3] för att spåra i grafen.

9. Öppna graffönstret ([◆] [GRAPH]).



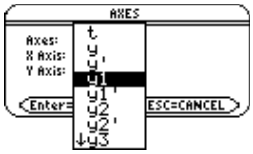
Ställa in axlar till Time eller Custom

Inställning av axlarna ger dig stor flexibilitet vid plottning av olika differentialekvationer. Anpassade axlar är särskilt effektiva för att visa olika samband.

Öppna dialogrutan AXES

I Y= Editor, tryck på:
TI-89: [2nd] [F7]
TI-92 Plus: [F7]

Om Fields = SLPFLD så är Axes inte tillgängligt.
TI-89: [2nd] [F7]
TI-92 Plus: [F7]

Enhet	Beskrivning
Axes	TIME — Plottar t längs x axeln och y (lösningarna på alla valda differentialekvationer) längs y-axeln. CUSTOM — Välj själv x- och y-axel.
X Axis, Y Axis	Endast aktivt när Axes = CUSTOM. Du väljer själv vad du vill plotta längs x- och y axeln.  t — tid y — lösningar (y1, y2, etc.) till alla valda differential-ekvationer y' — värdet av alla valda differentialekvationer (y1', y2', etc.) y1, y2, etc. — Lösningen till motsvarande differential-ekvation, oavsett om ekvationen är markerad eller inte y1', y2', etc. — Värdet av den högra sidan av motsvarande differentialekvation, oavsett om ekvationen är markerad eller inte

Obs! *t är inte giltigt för* någon axel *när* Fields=DIRFLD. *Om du väljer t kommer ett fel av typen* Invalid axes *att uppstå vid plottningen.*

Exempel på axlar (Time och Custom)

Använd den biologiska modellen för jägare/byte för att bestämma det antal kaniner och rävar som krävs för att uppnå jämvikt i en avgränsad biotop. Plotta lösningen med både tidsaxlar och anpassade axlar.

Modellen för jägare/byte

Använd de två kopplade första ordningens differentialekvationer:

y1' = ⁻y1 + 0.1y1 ∗ y2 och y2' = 3y2 − y1 ∗ y2

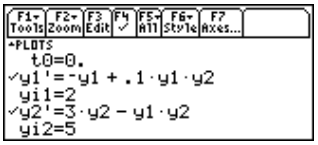
där:

y1 = Rävpopulation
yi1 = Startvärde för rävpopulation (2)
y2 = Kaninpopulation
yi2 = Startvärde för kaninpopulation (5)

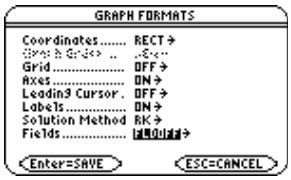
1. Använd [MODE] och ställ in Graph = DIFF EQUATIONS.

Tips! Du kan minska plottningstiderna genom att radera alla andra ekvationer i Y= Editor. Med FLDOFF *beräknas alla ekvationer även om de inte är markerade.*

2. Definiera differential-ekvationerna och startvärdena i Y= Editor ([◆] [Y=]).



3. Tryck på:
[F1] 9
— eller —
TI-89: [◆] [I]
TI-92 Plus: [◆] F
och ställ in Axes = ON, Labels = ON, Solution Method = RK samt Fields = FLDOFF.



4. I Y= Editor, tryck:
TI-89: [2nd] [F7]
TI-92 Plus: [F7]
och ställ in Axes = TIME.

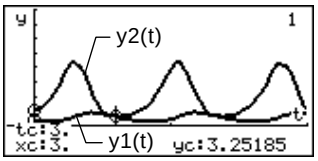
5. Ställ in Windowvariablerna i Window Editor ([◆] [WINDOW]).

t0=0.	xmin=⁻1.	ncurves=0.
tmax=10.	xmax=10.	diftol=.001
tstep=π/24	xscl=5.	
tplot=0.	ymin=⁻10.	
	ymax=40.	
	yscl=5.	

6. Plotta differentialekvationerna ([◆] [GRAPH]).

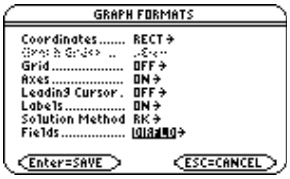
Tips! Du kan använda ⊕ och ⊖ för att flytta spårnings-markören mellan kurvorna för y1 och y2.

7. Tryck på [F3] för att spåra. Tryck sedan på 3 [ENTER] för att se antalet rävar (yc för y1) och kaniner (yc för y2) vid t=3.



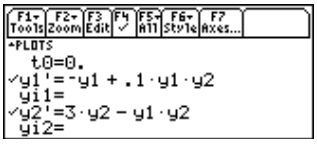
Obs! I detta exempel används DIRFLD *för två relaterade differential-ekvationer som inte representerar en andra ordningens ekvation.*

8. Återgå till Y= Editor. Tryck på:
   [F1] 9
   — eller —
   TI-89: [◆] [I]
   TI-92 Plus: [◆] F
   och ställ in Fields = DIRFLD.

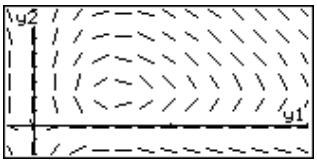


9. Tryck på:
   TI-89: [2nd] [F7]
   TI-92 Plus: [F7]
   och kontrollera att axlarna är inställda som i exemplet till höger.

10. I Y= Editor, tar du bort startvärdena för yi1 och yi2.



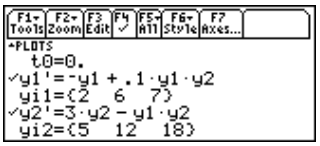
11. Återgå till graffönstret som nu endast visar fasplanet.



Tips! Du kan använda en lista för att ange flera startvärden.

12. Om du vill plotta en grupp lösningar återgår du till Y= Editor och skriver in startvärden enligt nedan:

    yi1={2,6,7} och yi2={5,12,18}

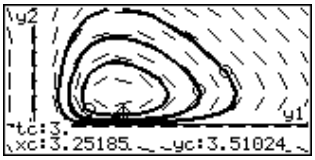


13. Återgå till graffönstret som nu visar en kurva för varje par av startvärden.

Tips! Du kan använda ⊕ och ⊖ för att flytta spårnings-markören mellan de olika kurvorna.

14. Tryck [F3] för att spåra. Tryck sedan 3 [ENTER] för att se antalet rävar (xc) och kaniner (yc) vid t=3.

    Eftersom t0=0 och tmax=10, kan du spåra inom området $0 \le t \le 10$.



Tänk dig en logistisk tillväxtmodell dP/dt = .001∗ P∗ (100− P), med begynnelsevillkoret P(0) = 10. Använd funktionen **BldData** för att jämföra plottningspunkterna som beräknats med metoderna RK och Euler. Plotta därefter dessa punkter tillsammans med en graf över ekvationens exakta lösning.

Exempel

1. Tryck på [MODE] och ställ in Graph=DIFF EQUATIONS.
2. Uttryck första ordningens ekvationen i termer av y1' och y1.

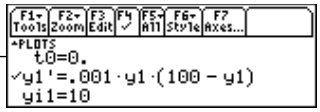
   y1'=.001y1∗ (100− y1)

   Använd inte implicit multiplikation mellan en variabel och ett uttryck inom parenteser.Om du gör så hanteras det som ett funktionsanrop.

__Tips!__ Du kan minska plottningstiderna genom att radera alla andra ekvationer i Y= Editor. Med FLDOFF *beräknas alla ekvationer även om de inte är markerade.*

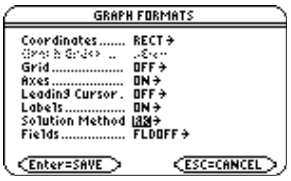
3. Skriv in ekvationen i Y= Editor ([◆] [Y=]).
4. Ange begynnelsevillkor:

   yi1=10



t0 är den tidpunkt då startvärdet börjar gälla. Förvalt värde är t0=0.

5. Tryck:
   [F1] 9
   — eller —
   TI-89: [◆] [I]
   TI-92 Plus: [◆] F
   och ställ in Solution Method = RK och Fields = FLDOFF.



6. Ställ in Windowvariablerna i Window Editor ([◆] [WINDOW]).

t0=0.	xmin=⁻1.	ncurves=0.
tmax=100.	xmax=100.	diftol=.001
tstep=1.	xscl=1.	
tplot=0.	ymin=⁻10.	
	ymax=10.	
	yscl=1.	

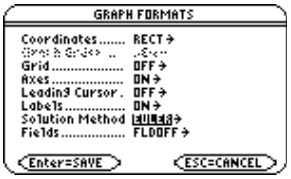
Viktigt! Ändra tstep från .1 (dess förval) till 1. I annat fall beräknar **BldData** för många rader för variabeln och ett fel av typen Dimension error uppstår.

__Obs!__ Du behöver inte plotta grafen innan du använder **BldData**. *Mer information om* **BldData** *finns i appendix A.*

7. Från grundfönstret
   TI-89: [HOME]
   TI-92 Plus: [◆] [HOME]
   använd **BldData** för att skapa en datavariabel med plottnings-punkterna från metoden RK.

```
BldData rklog
```

8. Återgå till Y= Editor, tryck:
   [F1] 9
   — eller —
   TI-89: [◆] [I]
   TI-92 Plus: [◆] F
   och ställ in
   Solution Method = EULER.

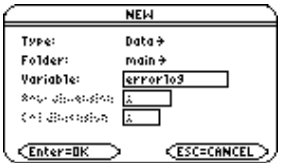


9. Återgå till grundfönstret och använd **BldData** för att skapa en datavariabel med plottningspunkterna från metoden Euler.

```
BldData eulerlog
```

Obs! errorlog *kan användas för att kombinera data från* rklog *och* eulerlog *så att du kan jämföra de två uppsättningarna data bredvid varandra.*

10. Använd Data/Matrix Editor (APPS 6 3) för att skapa en ny variabel med namnet errorlog.

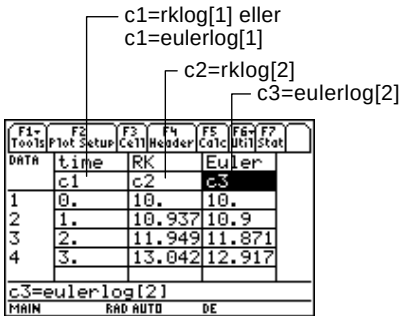


Obs: rklog[1] *och* rklog[2] *refererar till kolumn 1 och 2 i* rklog. *Detsamma gäller för* eulerlog[2].

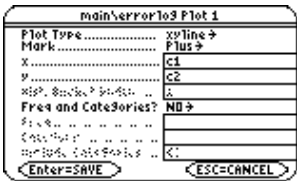
Tips! *Bläddra genom datavariablerna för att se hur värdena för RK och Euler skiljer sig för samma tidsvärde.*

11. I denna nya datavariabel definerar du referenserna för kolumnhuvudena c1, c2 och c3 till data i rklog och eulerlog. Skriv även in rubriker enligt exemplet till höger.

    Definiera en referens genom att flytta markören till önskad kolumn och tryck på F4. Skriv in uttrycket (som rklog[1] för c1), och tryck på ENTER.



12. I Data/Matrix Editor, tryck F2. Tryck sedan på F1 och definiera Plot 1 för data hos RK data som i exemplet till höger.



13. Definiera Plot 2 för data hos Euler. Använd värdena som visas till höger.

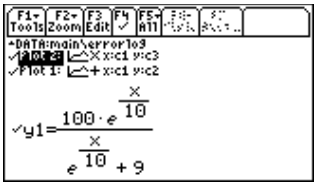
Plot Type=xyline
Mark=Cross
x=c1
y=c3

14. Återgå till Y= Editor, tryck MODE och ställ in Graph = FUNCTION.

Obs! *Information om hur du använder* **deSolve()** *för att hitta en exakt, allmän lösning finns på sidan 196.*

15. Den exakta lösningen till differentialekvationen visas nedan. Skriv in den som y1.

    y1 = (100∗ *e*^(x/10))/(*e*^(x/10)+9)



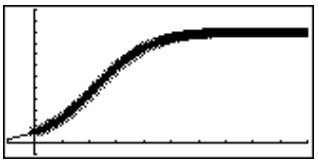
Du kan använda ⊕ för att bläddra uppåt och se Plot 1 och Plot 2.

16. Ställ in Windowvariablerna i Window Editor.

xmin=⁻10. ymin=⁻10. xres=2.
xmax=100. ymax=120.
xscl=10. yscl=10.

Obs! *Den oskarpa linjen i grafen visar skillnaderna mellan värdena från RK och Euler.*

17. Öppna graffönstret ([◆] [GRAPH]).

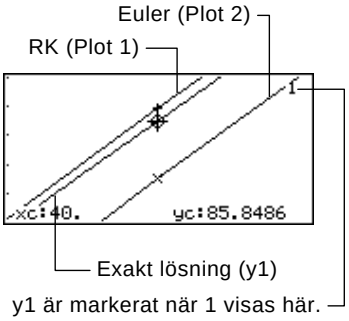


18. I Window Editor, ställ in Windowvariablerna för zoomning så att du kan utforska skillnaderna mer detaljerat.

xmin=39.7 ymin=85.5 xres=2.
xmax=40.3 ymax=86.
xscl=.1 yscl=.1

19. Återgå till graffönstret.

20. Tryck [F3] för att spåra och tryck sedan ⊕ eller ⊖ tills y1 är markerat. (1 visas i övre högra hörnet.) Skriv in 40.



Genom att flytta spårningsmarkören och spåra varje lösning till xc = 40 ser du att:

- Den exakta lösningen (y1) är 85.8486 avrundat till sex siffror.
- Lösningen enligt RK (Plot 1) är 85.8952.
- Lösningen enligt Euler (Plot 2) är 85.6527.

Du kan även använda Data/Matrix Editor för att öppna datavariabeln errorlog och bläddra till time = 40.

Exempel på funktionen deSolve()

Med den funktionen **deSolve()** kan du lösa många första och andra ordningens ekvationer exakt.

Exempel

För en allmän lösning kan du använda följande syntax. För en särskild lösning hittar du mer information i appendix A.

deSolve(*1aEller2aGradOdn***,** *oberoendeVar***,** *beroendeVar***)**

Använd den logistiska första ordningens ekvation från exemplet på sidan 176 och finn den allmänna lösningen för y med avseende på t.

Tips! För högsta noggrannhet, använd 1/1000 i stället för .001. Flyttalsvärden kan orsaka avrundningsfel.

Obs! Detta exempel innehåller ingen plottning så du kan använda valfritt läge för Graph.

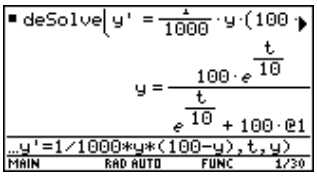
deSolve(y' = 1/1000 y∗ (100− y),t,y)

Använd inte implicit multiplikation mellan en variabel och ett uttryck inom parenteser.Om du gör så hanteras det som ett funktionsanrop.

För ', tryck [2nd] [′].

Innan du använder **deSolve()** måste du radera alla existerande variabler av typen t och y. I annat fall uppstår ett fel.

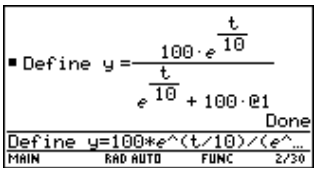
1. Från grundfönstret
   TI-89: [HOME]
   TI-92 Plus: [◆] [HOME]
   använd **deSolve()** för att hitta den allmänna lösningen.



@1 anger en konstant. Du kan använda andra konstanter (@2, osv.).

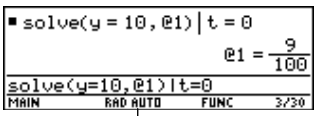
2. Använd lösningen till att definiera en funktion.
   a. Tryck [▲] för att markera lösningen i historiklistan. Tryck sedan på [ENTER] för att automatiskt klistra in den på inmatningsraden.
   b. Infoga instruktionen **Define** i radens början. Tryck sedan på [ENTER].

Tips! Använd [2nd] [◀] för att flytta markören till inmatningsradens början.



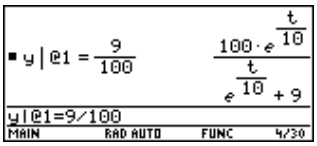
Obs! Om du har en annan konstant (@2, etc.) löser du för den konstanten i stället.

3. Med begynnelsevillkoret y=10 med t=0, använd **solve()** för att hitta @1-konstanten.



För @, tryck
TI-89: [◆] [STO▸]
TI-92 Plus: [2nd] R

4. Evaluera den allmänna lösningen (y) med konstanten @1=9/100 för att hitta den partikulär lösningen.



Du kan också använda **deSolve()** för att lösa problemet direkt. Skriv in:

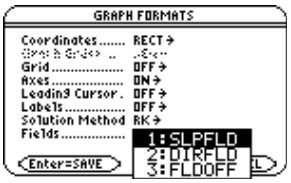
deSolve(y' = 1/1000 y∗ (100− y) and y(0)=10,t,y)

Felsökning av grafformat

I detta avsnitt hittar du information och hjälp med plottning av differentialekvationer. Ofta beror problemen på inställningarna för Fields.

Ställa in graf-formatet Fields

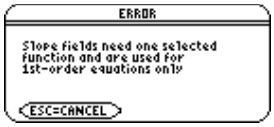
I Y= Editor, Window Editor eller graffönstret, tryck:
F1 9
— eller —
TI-89: ◆ I
TI-92 Plus: ◆ F



Vilken ordning av ekvation plottar du?

Om ekvationen är av:	så är giltiga inställningar:
första ordningens	SLPFLD eller FLDOFF
andra ordningens (system med två första ordningens ekvationer)	DIRFLD eller FLDOFF
tredje ordningens eller högre (system med tre eller fler första ordningens ekvationer)	FLDOFF

Eftersom inställningen Fields = SLPFLD är förvald så visas oftast felmeddelandet till höger.



När du får detta eller andra felmeddelanden:

- Kontrollera i tabellen ovan att du har rätt inställning för Fields för den ekvation du vill beräkna. Ändra till en giltig inställning.
- För en särskild inställning av Fields, kontrollera tabellen nedan för mer information om respektive inställning.

Fields=SLPFLD

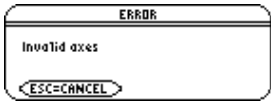
I Y= Editor	Använd F4 för att markera en, och endast en, första ordningens ekvation. Du kan ange flera ekvationer men endast en i taget kan vara markerad. Den markerade ekvationen får inte innehålla några referenser till andra ekvationer i Y= Editor. Till exempel: Om y1'=y2 uppstår ett fel av typen Undefined variable vid plottning.
I graffönstret	Om riktningsfältet ritas men ingen kurva plottas anger du ett startvärde som i beskrivningen på sidan 184.

Fields=DIRFLD

I Y= Editor

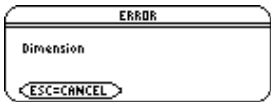
Skriv in ett giltigt system med två första ordningens ekvationer. Information om hur du skapar ett giltigt system för en andra ordningens ekvation finns på sidan 186.

Ange Axes = CUSTOM:
TI-89: [2nd] [F7]
TI-92 Plus: [F7]
Om Axes = TIME, visas felet Invalid axes när du försöker att plotta grafen.



Om du anger startvärden i Y= Editor så måste ekvationerna som refereras av de anpassade axlarna ha samma antal begynnelsevillkor.

I annat fall uppstår ett fel av typen Dimension error då du försöker plotta grafen.



Med anpassade axlar

Ställ in giltiga axlar för ditt system med ekvationer.

Ange inte t för någon av axlarna. Om du gör det uppstår ett fel av typen Invalid axes då du försöker plotta grafen.

De två axlarna måste referera till olika ekvationer i ditt system av ekvationer. Till exempel är y1 vs. y2 giltigt medan y1 vs. y1' ger ett fel av typen Invalid axes.

I graffönstret

Om fasplanet ritas men ingen kurva plottas anger du ett begynnelsevillkor i Y= Editor eller väljer ett begynnelsevillkor interaktivt i graffönstret enligt beskrivningen på sidan 184. Om du redan angett begynnelsevillkor, välj ZoomFit:
TI-89: [F2] [alpha] A
TI-92 Plus: [F2] A

Windowvariabeln ncurves ignoreras tillsammans med DIRFLD. Förvalda kurvor ritas inte automatiskt.

Observera

Med inställningen DIRFLD bestäms vilka ekvationer som ska plottas av vilka ekvationer som refereras av de anpassade axlarna, oavsett vilka ekvationer som är markerade i Y= Editor.

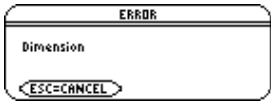
Om ditt system av ekvationer refererar till t så plottas fasplanet (inte kurvorna) i förhållande till den tidpunkt som anges i Windowvariabeln dtime.

Fields=FLDOFF

I Y= Editor

Om du skriver in en ekvation av andra ordningen eller högre måste du skriva in den som ett system av ekvationer enligt beskrivningen på sidan 186.

Alla ekvationer (markerade eller ej) måste ha samma antal begynnelsevillkor. I annat fall uppstår ett fel av typen Dimension error.

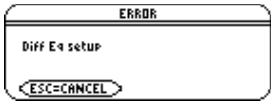


När du vill ange Axes = TIME eller CUSTOM, tryck på:
TI-89: [2nd] [F7]
TI-92 Plus: [F7]

Med anpassade axlar

Om X Axis is not t så måste du ange minst ett begynnelsevillkor för varje ekvation i Y= Editor (oavsett om ekvationen är markerad eller ej).

I annat fall uppstår ett fel av typen Diff Eq setup då du försöker plotta grafen.



I graffönstret

Om ingen kurva plottas anger du ett begynnelsevillkor enligt beskrivningen på sidan 184. Om du redan angett begynnelsevillkor i Y= Editor, välj ZoomFit:
TI-89: [F2] [alpha] A
TI-92 Plus: [F2] A

Första ordningens ekvationer kan se annorlunda ut med inställningen FLDOFF än med SLPFLD. Detta beror på att FLDOFF använder Windowvariablerna tplot och tmax (sidan 181) vilka ignoreras med SLPFLD.

Observera

Använd FLDOFF och Axes = Custom för att plotta första ordningens ekvationer som inte kan plottas med SLPFLD. Du kan till exempel plotta t mot y1' (där SLPFLD plottar t mot y1). Om du anger flera ekvationer av första ordningen kan du plotta en ekvation eller dess lösning mot en annan genom att ange dem som axlar.

Om du använder tabellfönstret för att utforska differential-ekvationer

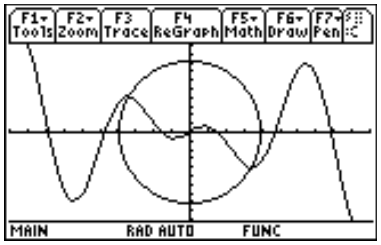
Du kan använda tabellfönstret för att se de olika plottningspunkterna för lösningen till en differentialekvation. I tabellen kan dock vissa ekvationer vara andra än de som plottas. I tabellen visas endast markerade ekvationer, oavsett om dessa kommer att plottas eller inte, beroende på dina inställningar för Fields och Axes.

Ytterligare graffunktioner

12

Översikt över ytterligare grafverktyg 202
Spara datapunkter från en graf 203
Plotta en funktion som är definierad i grundfönstret 204
Plotta en funktion som består av flera delar 206
Plotta en familj av kurvor 208
Använda tvågrafsläget 209
Rita en funktion eller en invers på en graf..... 212
Rita en linje, cirkel eller textetikett på en graf..... 213
Spara och öppna en bild av en graf..... 217
Animera en serie grafbilder 219
Spara och öppna en grafdatabas..... 220

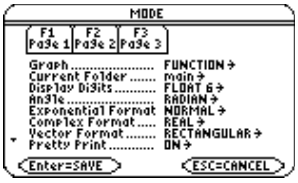
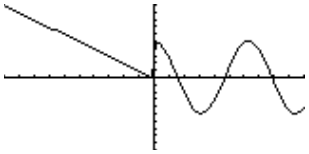
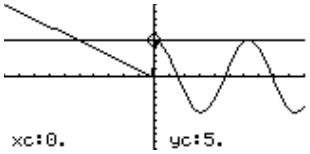
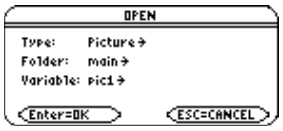
I det här kapitlet beskrivs ytterligare funktioner som du kan använda för att skapa grafer på TI-89 / TI-92 Plus. Informationen gäller allmänt för alla inställningar av grafläget.



I det här kapitlet antas att du redan känner till de grundläggande procedurerna för att definiera och välja funktioner, ställa in Window-variabler och visa grafer, som beskrivs i kapitel 6: Grundläggande funktionsplottning.

Översikt över ytterligare grafverktyg

Plotta från grundfönstret den delade funktionen: y = − x då x < 0 och y = 5 cos(x) då x ≥ 0. Rita en horisontell linje tvärs över överdelen av cosinuskurvan. Spara sedan en bild av grafen som visas.

Steg	TI-89 Tangenter	TI-92 Plus Tangenter	Teckenfönster
1. Visa dialogrutan MODE. För grafläge väljer du FUNCTION. För vinkelenhet (Angle) väljer du RADIAN.	[MODE] ▶ 1 ▼ ▼ ▼ ▶ 1 [ENTER]	[MODE] ▶ 1 ▼ ▼ ▼ ▶ 1 [ENTER]	
2. Visa grundfönstret. Använd kommandot **Graph** och funktionen **when** för att ange den delade funktionen. *När du trycker på [F4] 2 och väljer* **Graph** *på menyn* Other *läggs automatiskt ett mellanslag till.*	[HOME] [F4] 2 [2nd] [a-lock] W H E N [alpha] [(] X [2nd] [<] 0 [,] [(-)] X [,] 5 [×] [2nd] [COS] X [)] [)]	[◆] [HOME] [F4] 2 W H E N [(] X [2nd] [<] 0 [,] [(-)] X [,] 5 [×] [COS] X [)] [)]	Graph when(x<0,⁻x,5*cos(x))
3. Kör kommandot **Graph**, som automatiskt visar graffönstret. *Grafen använder de aktuella Window-variablerna, som behandlas som standardvärden ([F2] 6) för detta exempel.*	[ENTER]	[ENTER]	
4. Rita en horisontell linje över överdelen av cosinuskurvan. *Räknaren är kvar i "linje"-läge tills du väljer en annan operation eller trycker på [ESC].*	[2nd] [F7] 5 ▲ (tills linjen är på plats) [ENTER]	[F7] 5 ▲ (tills linjen är på plats) [ENTER]	
5. Spara en bild av grafen. Använd PIC1 som variabelnamn för bilden. *Se till att ställa in Type = Picture. Som standard är inställningen GDB.*	[F1] 2 ▶ 2 ▼ ▼ P I C [alpha] 1 [ENTER] [ENTER]	[F1] 2 ▶ 2 ▼ ▼ P I C 1 [ENTER] [ENTER]	
6. Ta bort den horisontella linjen. *Du kan också trycka på [F4] för att plotta om grafen.*	[2nd] [F6] 1	[F6] 1	
7. Öppna den sparade bildvariabeln för att åter visa grafen med linjen. *Se till att ställa in Type = Picture. Som standard är inställningen GDB.*	[F1] 1 ▶ 2 (om den inte visas ställer du in Variable = pic1) [ENTER]	[F1] 1 ▶ 2 (om den inte visas ställer du in Variable = pic1) [ENTER]	

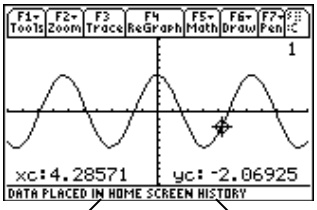
Spara datapunkter från en graf

Från grundfönstret kan du spara uppsättningar av koordinater och/eller matematiska resultat för senare analys. Du kan spara informationen som en enradig matris (vektor) i grundfönstret eller som datapunkter i en systemdatavariabel som kan öppnas i Data/Matrix Editor.

Spara punkterna

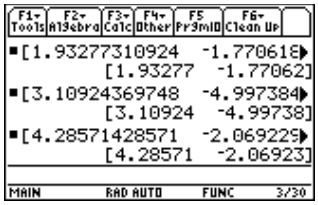
1. Visa grafen. (I detta exempel visas y1(x)=5∗ cos(x).)
2. Visa koordinaterna eller de matematiska resultaten som du vill spara.
3. **TI-89:** ◆ (-) (Grundfönstret) eller ◆ , (sysData-variabel)
   TI-92 Plus: ◆ H (Grundfönstret) eller ◆ D (sysData-variabel)
4. Upprepa processen om det behövs.

Tips! *Du visar koordinaterna eller de matematiska resultaten genom att spåra en funktion med* F3 *eller utföra en* F5 Math*-funktion (exempelvis Minimum eller Maximum). Du kan också använda den rörliga markören.*



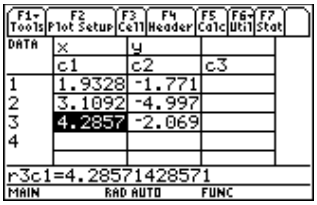
TI-89: ◆ (-)
TI-92 Plus: ◆ H

Visade koordinater läggs till i historielistan i grundfönstret (men inte på inmatningsraden) som en enradig matris eller vektor.



TI-89: ◆ ,
TI-92 Plus: ◆ D

Visade koordinater sparas i en datavariabel med namnet sysData, som du kan öppna i Data/Matrix Editor.



Tips! *Använd ett delat fönster för att visa en graf och grundfönstret eller en graf och Data/Matrix Editor på samma gång.*

Anmärkningar om variabeln SysData

- När du trycker på: **TI-89:** ◆ , **TI-92 Plus:** ◆ D
  - Om variabeln sysData inte finns skapas den i mappen MAIN.
  - Om variabeln sysData redan finns läggs nya data till i slutet av befintliga data. Befintliga rubriker eller kolumnhuvuden (för de påverkade kolumnerna) tas bort, rubriker ersätts med lämpliga rubriker för den nya informationen.
- Du kan ta bort och rensa variabeln sysData precis som du kan med andra variabler. Du kan däremot inte låsa den.
- Om grundfönstret innehåller en funktion eller statistisk plottning som refererar till det aktuella innehållet i sysData, fungerar inte kommandot.

Plotta en funktion som är definierad i grundfönstret

I många fall kan du skapa en funktion eller ett uttryck i grundfönstret och sedan plotta den/det. Du kan kopiera ett uttryck till Y= Editor eller plotta det direkt från grundfönstret utan att använda Y= Editor.

Vilken är den oberoende variabeln för ett visst grafläge?

I Y= Editor måste alla funktioner vara definierade uttryckt i det aktuella graflägets oberoende variabel.

Grafläge	Oberoende variabel
Function	x
Parametric	t
Polar	θ
Sequence	n
3D	x, y
Differential Equation	t

Kopiera från grundfönstret till Y= Editor

Tips! I stället för att använda [F1] 5 eller [F1] 6 för att kopiera och klistra in, använd:

TI-89: [◆] [COPY] eller [◆] [PASTE].
TI-92 Plus: [◆] C (copy) eller [◆] V(paste).

Tips! *Om du vill kopiera ett uttryck från historiklistan i grundfönstret till inmatningsraden kan du använda funktionen för automatisk inklistring eller kopiera och klistra in.*

Tips! Define *finns på menyn* [F4] *i grundfönstret.*

Tips! [2nd][RCL] *är användbart om ett uttryck är sparat till en variabel eller funktion som inte passar i Y= Editor, exempelvis* a1 *eller* f1(x).

Om du har ett uttryck i grundfönstret kan du använda någon av följande metoder för att kopiera det till Y= Editor.

Metod	Beskrivning
Kopiera och klistra in	1. Markera uttrycket i grundfönstret. Tryck på [F1] och välj 5:Copy. 2. Visa Y= Editor, markera funktionen och tryck på [ENTER]. 3. Tryck på [F1] och välj 6:Paste. Tryck sedan på [ENTER].
[STO▸]	Spara uttrycket till ett Y=-funktionsnamn. `2x^3+3x^2-4x+12→y1(x)` Använd det fullständiga funktionsnamnet: y1(x), inte bara y1.
Define, kommando	Definiera uttrycket som en användardefinierad Y=-funktion. `Define y1(x)=2x^3+3x^2-4x+12`
[2nd] [RCL]	Om uttrycket redan är sparat till en variabel: 1. Visa Y= Editor, markera funktionen och tryck på [ENTER]. 2. Tryck på [2nd] [RCL]. Skriv variabelnamnet som innehåller uttrycket och tryck på [ENTER] två gånger. **Viktigt!** Om du vill hämta en funktionsvariabel som exempelvis f1(x), skriv bara f1, inte hela funktionsnamnet. 3. Tryck på [ENTER] för att spara det hämtade uttrycket i funktionslistan i Y= Editor.

Plotta direkt från grundfönstret

Med kommandot **Graph** kan du plotta ett uttryck från grundfönstret utan att använda Y= Editor. Till skillnad från Y= Editor, kan du med **Graph** mata in ett uttryck uttryckt i vilken oberoende variabel som helst, oavsett vilken variabel som hör till det aktuella grafläget.

__Tips! Graph__ finns på menyn F4 i grundfönstret.

__Obs! Graph__ använder de aktuella inställningarna för Window-variabler.

Om uttrycket är uttryckt i termer av:	Använder du kommandot Graph enligt detta exempel:
Det aktuella grafläges oberoende variabel	`graph 1.25x*cos(x)` För funktionsplottning är x den oberoende variabeln.
En oberoende variabel som inte är samma som den för det aktuella grafläget	`graph 1.25a*cos(a),a` Ange den oberoende variabeln, annars kan ett fel uppstå.

__Tips!__ Om du vill skapa en tabell från grundfönstret använder du kommandot __Table__. Det liknar __Graph__. Båda delar samma uttryck.

Graph fungerar inte med sekventiella plottningar eller med differentialekvationer. För parameter-, polär- och 3D-grafer använder du följande syntax:

I grafläget PARAMETRIC:	**Graph** *xuttryck*, *yuttryck*, *t*
I grafläget POLAR:	**Graph** *uttryck*, *θ*
I grafläget 3D:	**Graph** *uttryck*, *x*, *y*

Kommandot **Graph** kopierar inte uttrycket till Y= Editor. I stället placeras funktioner som har valts tillfälligt i Y= Editor. Du kan spåra, zooma, visa och redigera uttryck för **Graph** i tabellfönstret på samma sätt som funktioner i Y= Editor.

Rensa graffönstret

Varje gång du utför kommandot **Graph** läggs det nya uttrycket till de befintliga uttrycken. Så här rensar du graferna:

- Kör kommandot **ClrGraph** (finns på menyn F4 Other i grundfönstret).

  – eller –
- Visa Y= Editor. Nästa gång du visar graffönstret kommer de funktioner som du valde i Y= Editor att plottas.

Extra fördelar med användardefinierade funktioner

Du kan definiera en användardefinierad funktion uttryckt i vilken oberoende variabel som helst. Exempel:

Definierad uttryckt i "aa".

```
define f1(aa)=1.25a*cos(aa)
graph f1(x)
```

Refererar till funktionen genom att använda den oberoende variabeln för det aktuella grafläget.

och:

```
define f1(aa)=1.25a*cos(aa)
f1(x)→y1(x)
```

Plotta en funktion som består av flera delar

För att kunna plotta en delad funktion måste du först definiera funktionen genom att ange gränserna och uttrycken för varje del. Funktionen **when** är mycket användbar för funktioner med två delar. För funktioner med tre eller fler delar kan det vara lättare att skapa en användardefinierad funktion med flera satser.

Använda funktionen When

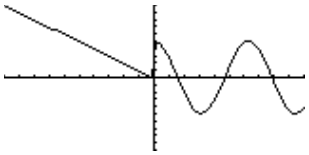
__Tips!__ Resultat från plottningar i graph kan variera.

För att definiera en funktion med två delar använder du syntaxen:

when(*villkor*, *santuttryck*, *falsktuttryck*)

Anta exempelvis att du vill plotta en funktion med två delar.

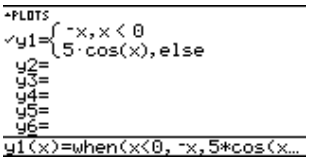
När:	**Använder du uttrycket:**
$x < 0$	^{-}x
$x \geq 0$	$5 \cos(x)$



I Y= Editor:

Funktionen visas i pretty print-läge.

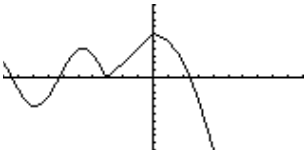
Mata in funktionen på den här formen.



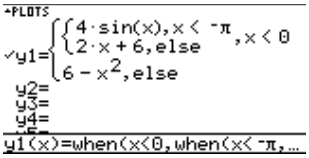
__Tips!__ Mata in **when** *genom att skriva in det, eller välj det från CATALOG.*

För tre eller fler delar kan du använda nästlade **when**-funktioner.

När:	**Använder du uttrycket:**
$x < -\pi$	$4 \sin(x)$
$x \geq -\pi$ och $x < 0$	$2x + 6$
$x \geq 0$	$6 - x^2$



I Y= Editor:



där:

```
y1(x)=when(x<0,when(x<⁻π,4*sin(x),2x+6),6-x^2)
```

Denna nästade funktion används då x<0.

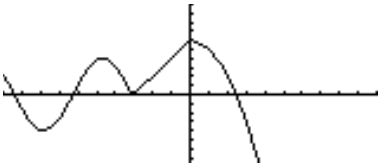
Nästlade funktioner blir snabbt komplexa och svåra att se.

Använda en användardefinierad funktion med flera satser

För tre eller fler delar kanske du vill skapa en användardefinierad funktion med flera satser.

Betrakta exempelvis den föregående tredelade funktionen.

När:	Använder du uttrycket:
$x < -\pi$	4 sin(x)
$x \geq -\pi$ och $x < 0$	2x + 6
$x \geq 0$	$6 - x^2$



__Obs!__ Information om likheter och skillnader mellan funktioner och program finns i kapitel 17.

__Tips!__ Resultat från plottningar i graph kan variera.

En användardefinierad funktion med flera satser kan ha många av de kontroll- och villkorsstrukturer (**If**, **ElseIf**, **Return** osv.) som används i programmering. När du skapar strukturen för en funktion kan det vara praktiskt att först visualisera den i blockform.

```
Func
  If x<-π Then
  Return 4*sin(x)
ElseIf x>=-π and x<0 Then
  Return 2x+6
Else
  Return 6-x^2
EndIf
EndFunc
```

Func och **EndFunc** måste starta och avsluta funktionen.

Information om enskilda programsatser finns i appendix A.

När du matar in en funktion med flera satser i Y= Editor eller grundfönstret måste du mata in hela funktionen på en rad.

Använd kolon (:) för att avgränsa varje programsats.

```
Func:If x<-π Then:Return 4*sin(x): ... :EndIf:EndFunc
```

I Y= Editor:

Endast “Func” visas för en funktion med flera satser.

Mata in en funktion med flera satser på en rad. Kom ihåg att ta med kolon.

Från grundfönstret eller ett program

Från grundfönstret kan du även använda kommandot **Define** för att skapa en användardefinierad funktion med flera satser. Se sidan 204 för mer information om hur du kopierar en funktion från grundfönstret till Y= Editor.

Från Program Editor (kapitel 17) kan du skapa en användardefinierad funktion. Du kan exempelvis använda Program Editor för att skapa en funktion med namnet f1(xx). I Y= Editor ställer du in y1(x) = f1(x).

Genom att mata in en lista i ett uttryck kan du plotta en separat funktion för varje värde i listan. (Du kan inte plotta en familj av kurvor i grafläget SEQUENCE eller 3D.)

Exempel på hur du kan använda Y= Editor

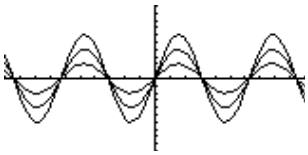
Tips! *Resultat från plottningar i graph kan variera.*

Tips! *Omslut listelement inom klammerparenteser ([2nd] [{] och [2nd] [}]) och avgränsa dem med komman.*

Obs! *Komman visas på inmatningsraden men inte i funktionslistan.*

Mata in uttrycket {2,4,6} sin(x) och plotta funktionerna.

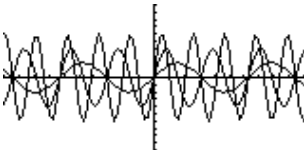
```
*PLOTS
✓y1={2  4  6}·sin(x)
 y2=
 y3=
 y4=
 y5=
 y6=
 y7=
y1(x)={2,4,6}*sin(x)
```



Plottar tre funktioner:
2 sin(x), 4 sin(x), 6 sin(x)

Mata in uttrycket {2,4,6} sin({1,2,3} x) och plotta funktionerna.

```
*PLOTS
✓y1={2  4  6}·sin({1  2 ▶
 y2=
 y3=
 y4=
 y5=
 y6=
 y7=
y1(x)={2,4,6}*sin({1,2,3}…
```



Plottar tre funktioner:
2 sin(x), 4 sin(2x), 6 sin(3x)

Exempel på hur du kan använda kommandot Graph

På liknande sätt kan du använda kommandot **Graph** från grundfönstret eller ett program, enligt beskrivning på sidan 205.

```
graph {2,4,6}sin(x)
graph {2,4,6}sin({1,2,3}x)
```

Visa grafer samtidigt med listor

Tips! *Om du vill ställa in grafformat i Y= Editor, Window Editor, eller Graffönstret, tryck på:*
TI-89: [◆] [|]
TI-92 Plus: [◆] F

När grafformatet är inställt till Graph Order = SIMUL, plottas funktionerna i grupper enligt elementnumret i listan.

```
*PLOTS
✓y1={2  4  6}·sin(x)
✓y2={1  2  3}·x+4
✓y3=cos(x)
```

För dessa exempelfunktioner plottas tre grupper.

- 2 sin(x), x+4, cos(x)
- 4 sin(x), 2x+4
- 6 sin(x), 3x+4

Funktionerna inom varje grupp plottas samtidigt men grupperna plottas i tur och ordning.

Spåra en familj av kurvor

När du trycker på ⊖ eller ⊕ flyttas markören till nästa eller föregående kurva i samma familj innan den flyttas till nästa eller föregående markerad funktion.

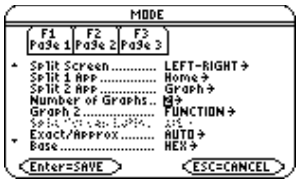
Använda tvågrafsläget

I tvågrafsläget dupliceras de TI-89 / TI-92 Plus grafrelaterade funktionerna, vilket ger dig två oberoende grafräknare. Tvågrafsläget än endast tillgängligt i delat fönsterläge. Mer information om delade fönster finns i kapitel 14.

Ställa in läget

Det är flera lägesinställningar som påverkar tvågrafsläget, men det är bara två inställningar som krävs. Båda finns på sidan 2 i dialogrutan MODE.

1. Tryck på [MODE]. Tryck därefter på [F2] för att visa sidan 2.
2. Ställ in följande obligatoriska lägen.
   - Split Screen = TOP-BOTTOM eller LEFT-RIGHT
   - Number of Graphs = 2



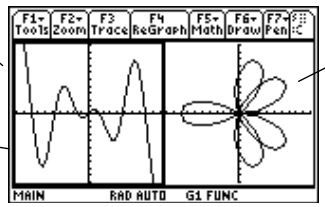
3. (Valfritt) Du kan även ställa in följande lägen.

   Sidan 1:
   - Graph = grafläget för den övre eller den vänstra sidan av delningen

   Sidan 2:
   - Split 1 App = program för den övre eller den vänstra sidan av delningen
   - Split 2 App = program för den undre eller den högra sidan av delningen
   - Graph 2 = grafläget för den undre eller den högra sidan av delningen
   - Split Screen Ratio = relativa storlekar av de två sidorna (endast TI-92 Plus)
4. Tryck på [ENTER] för att stänga dialogrutan.

Tvågrafsfönstret

Ett tvågrafsfönster liknar ett vanligt delat fönster.



Graf 1: vänstra eller översta sidan

Tjock kant anger aktiv grafsida

Graf 2: högra eller understa sidan

Grafläget för den aktiva grafen

Aktiv grafsida: GR#1 eller GR#2

Oberoende grafrelaterade funktioner

Både graf 1 och graf 2 har oberoende:

- Graflägen (FUNCTION, POLAR m.m.). Andra lägen, som exempelvis Angle, Display Digits m.m. delas och påverkar båda graferna.
- Window Editor-variabler.
- Parametrar för tabellinställningar och tabellfönster.
- Grafformat som exempelvis Coordinates, Axes m.m.
- Graffönster.
- Y= Editor. Båda graferna delar dock gemensamma definitioner för funktioner och statistiska plottningar.

Obs! *Y= Editor är bara helt oberoende när de två sidorna använder olika graflägen (beskrivs nedan).*

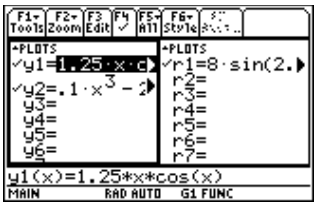
Oberoende grafrelaterade program (Y= Editor, graffönstret m.m.) kan visas på sidorna av fönstret på samma gång.

Program som inte är grafrelaterade (grundfönstret, Data/Matrix Editor m.m.) delas och kan endast visas på en sida i taget.

Y= Editor i tvågrafsläge

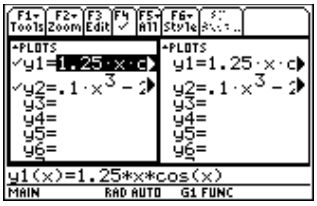
Även i tvågrafsläge finns det egentligen bara en Y= Editor, som underhåller en enda funktionslista för varje inställning av grafläget. Om däremot båda sidorna använder samma grafläge kan du från varje sida välja olika funktioner från den listan.

- När båda sidorna använder olika graflägen visas olika funktionslistor på varje sida.



- När båda sidorna använder samma grafläge visas samma funktionslista på båda sidorna.
  - Du kan använda [F4] för att välja olika funktioner och statistiska plottningar (anges med symbolen ✓) för varje sida.
  - Om du ställer in ett visningsformat ([2nd] [F6]) för en funktion, används det formatet för båda sidorna. (**TI-89**: [2nd] [F6] **TI-92 Plus:** [F6])

Obs! *Om du gör en ändring i den aktiva Y= Editor (omdefinierar en funktion, ändrar ett format m.m.) visas inte den ändringen på den inaktiva sidan förrän du växlar till den.*



Anta att graf 1 och graf 2 är inställda för funktionsplottning. Även om samma funktionslista visas på båda sidorna kan du välja (✓) olika funktioner för plottning.

Använda ett delat fönster

Fullständig information om delade fönster finns i kapitel 14.

- Du växlar från ena grafsidan till den andra genom att trycka på [2nd] [⊞] (den sekundära funktionen av [APPS]).

***Obs!** Du kan endast visa program som inte är grafrelaterade (exempelvis grundfönstret) på en sida i taget.*

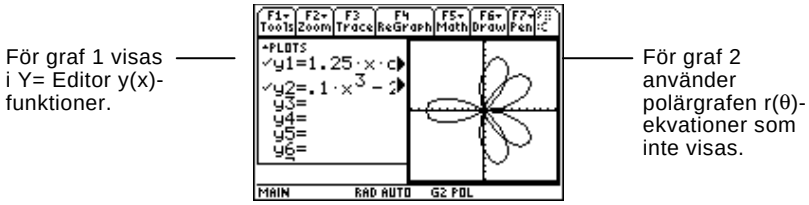
- Så här visar du olika program:
  - Växla till en grafsida och visa programmet på vanligt sätt.

    – eller –
  - Använd [MODE] för att ändra Split 1 App och/eller Split 2 App.
- Så här avslutar du tvågrafsläget:
  - Använd [MODE] för att ställa in Number of Graphs = 1, eller avsluta det delade fönstret genom att ställa in Split Screen = FULL.

    – eller –
  - Tryck på [2nd] [QUIT] två gånger. Detta avslutar alltid ett delat fönster och du återgår till en grundfönster i full storlek.

Kom ihåg att de två sidorna är fristående

I tvågrafsläge kan de två sidorna se ut att vara relaterade även om de inte är det. Exempel:



Från grundfönstret eller ett program

När du har ställt in tvågrafsläget gäller grafrelaterade åtgärder för den aktiva grafsidan. Exempel:

```
10→xmax
```

påverkar antingen graf 1 eller graf 2, beroende på vilken som är aktiv när du kör kommandot.

Du växlar de aktiva sidorna genom att trycka på [2nd] [⊞] eller använda funktionen **switch**, **switch**(1) eller **switch**(2).

Rita en funktion eller en invers på en graf

För jämförelseändamål kan du med kommandot **DrawFunc** rita en funktion över den aktuella grafen. Oftast är den ritade funktionen en variation av grafen. Du kan också rita inversen av en funktion. (Dessa funktioner är inte tillgängliga för 3D-grafer.)

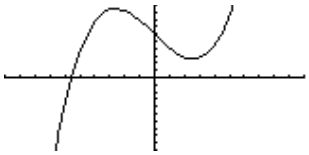
Rita en funktions, parameterkvationers eller polär ekvation

Kör **DrawFunc**, **DrawParm** eller **DrawPol** från grundfönstret eller en applikation. Du kan inte rita en funktion eller ekvation interaktivt från grundfönstret.

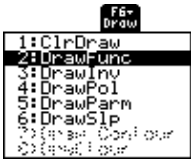
DrawFunc *uttryck*
DrawParm *uttryck1, uttryck2* [,*tmin*] [,*tmax*] [,*tstep*]
DrawPol *uttryck* [,*θmin*] [,*θmax*] [,*θstep*]

Exempel:

1. Definiera y1(x)=.1x^3 – 2x+6 i Y= Editor och plotta funktionen.



2. I graffönstret trycker du på:
   TI-89: [2nd] [F6]
   TI-92 Plus: [F6]
   och väljer 2:DrawFunc.

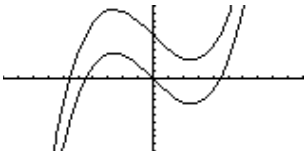


3. I grundfönstret matar du in vilken funktion som ska ritas.

```
DrawFunc y1(x)-5
```

4. Tryck på [ENTER] för att plotta funktionen i grundfönstret.

   Du kan inte spåra, zooma eller utföra en beräkning på en ritad funktion.



Obs! *När du vill öppna grundfönstret och klistra in* **DrawFunc** *på inmatningsraden, tryck på:*
TI-89: [2nd] [F6] 2
TI-92 Plus: [F6] 2

Tips! *Om du vill radera den ritade funktionen, tryck* [F4]

— eller —

TI-89: [2nd] [F6] *och välj* 1:ClrDraw.
TI-92 Plus: [F6] *och välj* 1:ClrDraw.

Rita inversen av en funktion

Kör **DrawInv** från grundfönstret eller ett program. Du kan inte rita en invers funktion interaktivt från graffönstret.

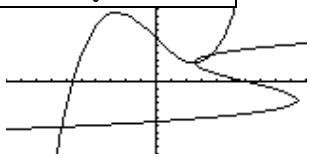
DrawInv *uttryck*

Använd exempelvis grafen av y1(x)=.1x^3 – 2x+6, enligt bilden nedan.

1. I graffönstret, tryck på:
   TI-89: [2nd] [F6]
   TI-92 Plus: [F6]
   och välj 3:DrawInv.
2. Ange funktionen.
3. Tryck på [ENTER].

   Inversen plottas som (y,x) i stället för (x,y).

```
DrawInv y1(x)
```



Obs! *När du vill öppna grundfönstret och klistra in* **DrawInv** *på inmatningsraden, tryck på:*
TI-89: [2nd] [F6] 3
TI-92 Plus: [F6] 3

Rita en linje, cirkel eller textetikett på en graf

Du kan rita ett eller fler objekt i graffönstret, oftast för jämförelseändamål. Rita exempelvis en horisontell linje för att visa att två delar av en graf har samma y-värde. (Vissa objekt är inte tillgängliga för 3D-grafer.)

Ta bort alla ritade objekt

Ett ritat objekt är inte en del av själva grafen. Objektet ritas "ovanpå" grafen och stannar kvar i fönstret tills du tar bort det.

Gör följande från graffönstret:

- **TI-89:** [2nd] [F6]
  TI-92 Plus: [F6]
  och välj 1:ClrDraw.

  – eller –
- Tryck på [F4] för att rita om.

Tips! Du kan också mata in **ClrDraw** *på inmatningsraden i grundfönstret.*

Du kan även göra andra saker som gör att funktionen Smart Graph ritar om grafen (som exempelvis att ändra Window-variablerna eller avmarkera en funktion i Y= Editor).

Rita en punkt eller en frihandslinje

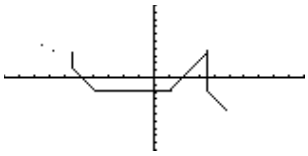
Gör följande från graffönstret:

1. **TI-89:** [2nd] [F7]
   TI-92 Plus: [F7]
   och välj 1:Pencil.
2. Flytta markören till önskad plats.

Om du vill rita en:	Gör du följande:
Punkt (bildpunktsstorlek)	Tryck på [ENTER].
Frihandslinje	**TI-89:** Tryck på och håll ned [↑] och flytta markören så att en linje ritas. **TI-92 Plus:** Tryck och håll ned [✎], flytta markören för att rita linjen. När du vill sluta rita linjen släpper du upp [↑] eller [✎].

Tips! När du ritar en frihandslinje kan du flytta markören diagonalt.

Efter att du har ritat punkten eller linjen är du fortfarande i "pennläge".



- Om du vill fortsätta att rita flyttar du markören till en annan plats.
- Avsluta genom att trycka på [ESC].

Obs! Om du börjar rita på en vit bildpunkt ritas en svart punkt eller linje. Om du börjar rita på en svart bildpunkt ritas en vit punkt eller linje (kan användas som radergummi).

Ta bort olika delar av ett ritobjekt

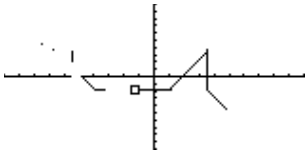
I graffönstret:

1. **TI-89:** [2nd] [F7]
   TI-92 Plus: [F7]
   och välj 2:Eraser. Markören visas som en liten fyrkant.
2. Flytta markören till önskad plats.

Om du vill ta bort:	Gör du följande:
Området under rutan	Tryck på [ENTER].
Längs en frihandslinje	**TI-89:** Tryck och håll ned [↑], flytta markören för att radera linjen. **TI-92 Plus:** Tryck och håll ned [✎], flytta markören för att radera linjen. Avsluta genom att släppa upp [↑] eller [✎].

Obs! *Med dessa tekniker kan du också ta bort delar av plottade funktioner.*

Efter att du har raderat är du fortfarande i "raderingsläge".



- Om du vill fortsätta att radera flyttar du rutan till en annan plats.
- Avsluta genom att trycka på [ESC].

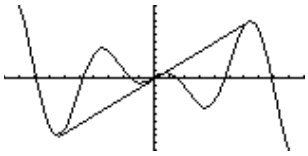
Rita en linje mellan två punkter

I graffönstret:

1. **TI-89:** [2nd] [F7]
   TI-92 Plus: [F7]
   och välj 3:Line.
2. Flytta markören till första punkten och tryck på [ENTER].
3. Flytta till den andra punkten och tryck på [ENTER]. (När du flyttar dras en linje från den första punkten till det ställe där markören befinner sig.)

Tips! *Använd [2nd] för att flytta markören i större steg; [2nd] ⊙ osv.*

Efter att du har ritat linjen är du fortfarande i "linjeläge".



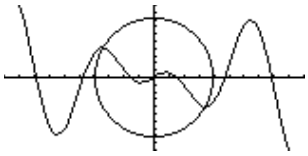
- Om du vill fortsätta att rita en till linje flyttar du markören till en ny första punkt.
- Avsluta genom att trycka på [ESC].

Rita en cirkel

I graffönstret:

1. **TI-89:** [2nd] [F7]
   TI-92 Plus: [F7]
   och välj 4:Circle.
2. Flytta markören till cirkelns medelpunkt och tryck på [ENTER].
3. Flytta markören för att ställa in radien och tryck på [ENTER].

Tips! *Använd [2nd] för att flytta markören i större steg; [2nd] ⊙ osv.*



Rita en horisontell eller vertikal linje

Gör följande från graffönstret:

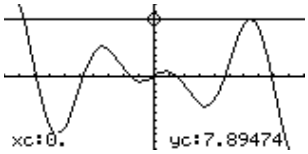
1. **TI-89:** [2nd] [F7]
   TI-92 Plus: [F7]
   och välj 5:Horizontal eller 6:Vertical. En horisontell eller vertikal linje och en blinkande markör visas i fönstret.

   Om linjen från början visas på en axel kan den vara svår att se. Du kan dock lätt upptäcka den blinkande markören.

__Tips!__ Använd [2nd] för att flytta markören i större steg; [2nd] ⊙ osv.

2. Använd markörknappen för att flytta linjen till önskad plats. Tryck därefter på [ENTER].

Efter att du har ritat linjen är du fortfarande i "linjeläge".



- Om du vill fortsätta att rita flyttar du markören till en annan plats.
- Avsluta genom att trycka på [ESC].

Rita en tangent

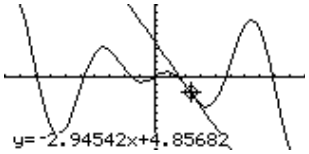
Om du vill rita en tangent använder du menyn [F5] Math. Gör följande från graffönstret:

1. Tryck på [F5] och välj A:Tangent.
2. Om det behövs använder du ⊖ och ⊕ för att välja funktionen.

__Tips!__ Om du vill ange en punkt för en tangent kan du skriva dess x-värde och trycka på [ENTER].

3. Flytta markören till punkten för tangenten och tryck på [ENTER].

   Tangenten ritas och dess ekvation visas.



Rita en linje baserat på en punkt och en lutning

Om du vill rita en linje genom en angiven punkt med en angiven lutning kör du kommandot **DrawSlp** från grundfönstret eller ett program. Använd följande syntax:

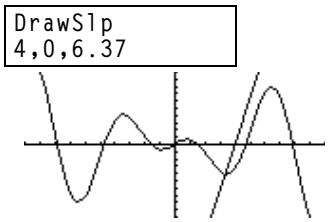
DrawSlp x, y, *lutning*

Du kan också få tillgång till **DrawSlp** från grundfönstret.

1. **TI-89:** [2nd] [F6]
   TI-92 Plus: [F6]
   och välj 6:DrawSlp. Detta växlar till grundfönstret och klistrar in **DrawSlp** på inmatningsraden.

2. Slutför kommandot och tryck på [ENTER].

   Räknaren växlar automatiskt till graffönstret och ritar linjen.



Skriv textetiketter

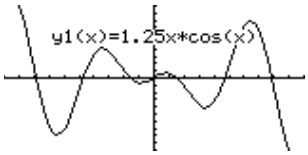
Gör följande från graffönstret:

1. **TI-89:** [2nd] [F7]
   TI-92 Plus: [F7]
   och välj 7:Text.

__Tips!__ Textmarkören anger det övre vänstra hörnet på nästa tecken som du skriver.

2. Flytta textmarkören till platsen där du vill börja skriva.
3. Skriv textetiketten.

Efter att du har skrivit texten är du fortfarande i "textläge".



- Om du vill fortsätta flyttar du markören till en annan plats.
- Avsluta genom att trycka på [ENTER] eller [ESC].

Från grundfönstret eller ett program

Det finns kommandon med vilka du kan rita de objekt som beskrivs i detta kapitel. Det finns också kommandon (exempelvis **PxlOn**, **PxlLine** osv) med vilka du kan rita objekt genom att ange exakt bildpunktsplacering i fönstret.

En lista över tillgängliga ritkommandon finns i "Rita i graffönstret" i kapitel 17.

Spara och öppna en bild av en graf

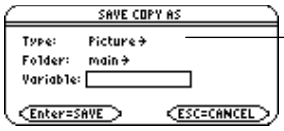
Du kan spara en bild av det aktuella graffönstret i en PICTURE- (eller PIC-) variabel. Sedan kan du öppna den variabeln och visa bilden. Detta sparar endast bilden, inte de grafinställningar som användes för att skapa bilden.

Spara en bild av hela graffönstret

I en bild kan det ingå plottade funktioner, axlar, skalstreck och ritade objekt. Bilden har inga indikatorer för undre och övre gränser, meddelanden eller markörkoordinater.

Visa graffönstret så som du vill spara det. Gör sedan följande:

1. Tryck på [F1] och välj 2:Save Copy As.
2. Ange typ (Picture), mapp och ett unikt variabelnamn.
3. Tryck på [ENTER]. När du har skrivit i en inmatningsruta som exempelvis Variable måste du trycka på [ENTER] två gånger.



Viktigt! Som standard är Type = GDB (för grafdatabas). Ställ in Type = Picture.

Spara en del av graffönstret

Du kan definiera en rektangulär ruta som endast omsluter den del av graffönstret som du vill spara.

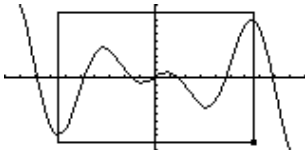
__Obs!__ Du kan inte spara en del av en 3D-graf.

__Tips!__ Använd ⊖ och ⊕ för att flytta till över- eller underdelen och använd ⊙ och ⊙ för att flytta sidorna.

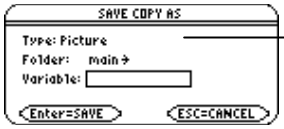
1. **TI-89:** [2nd] [F7]
   TI-92 Plus: [F7]
   och välj 8:Save Picture.

   En ruta visas runt fönstrets yttre kant.

2. Ange rutans första hörn genom att flytta dess övre och vänstra sida. Tryck därefter på [ENTER].
3. Ange det andra hörnet genom att flytta den undre och högra sidan. Tryck därefter på [ENTER].



4. Ange mappen och ett unikt variabelnamn.
5. Tryck på [ENTER]. När du har skrivit i en inmatningsruta som exempelvis Variable, måste du trycka på [ENTER] två gånger.



Obs! När du sparar en del av en graf anges Type = Picture automatiskt.

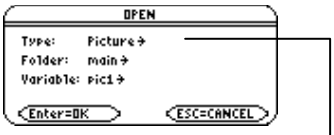
Öppna en grafbild

När du öppnar en grafbild läggs den ovanpå det aktuella graffönstret. Om du bara vill visa bilden använder du Y= Editor för att avmarkera eventuella andra funktioner innan du öppnar grafbilden.

Gör följande från graffönstret:

1. Tryck på [F1] och välj 1:Open.
2. Markera typ (Picture), mapp och variabeln som innehåller den grafbil du vill öppna.
3. Tryck på [ENTER].

Obs! *Om ett variabelnamn inte visas i dialogrutan finns det inga grafbilder i mappen.*



Viktigt! Som standard är Type = GDB (för grafdatabas). Ställ in Type = Picture.

En grafbild är ett ritobjekt. Du kan inte spåra kurvor i en bild.

För bilder som sparats från en del av graffönstret

När du trycker på [F1] och väljer 1:Open läggs bilden ovanpå graffönstret, med början i det övre vänstra hörnet. Om bilden sparades från en del av graffönstret kan bilden (sidan 217) se ut som om den har flyttats från den underliggande grafen.

Du anger vilken fönsterbildpunkt som ska användas som det övre vänstra hörnet med hjälp av kommandona som listas under "Från grundfönstret eller ett program" nedan.

Ta bort en grafbild

Picture-variabler som inte behövs tar upp minne. Om du vill ta bort en variabel använder du VAR-LINK-fönstret ([2nd] [VAR-LINK]), enligt beskrivningen i kapitel 21.

Från grundfönstret eller ett program

Om du vill spara och öppna (hämta) en grafbild använder du kommandona **StoPic**, **RclPic**, **AndPic**, **XorPic** och **RplcPic**, enligt beskrivning i appendix A.

Om du vill visa en serie grafbilder som en animering använder du kommandot **CyclePic**. Se sidan 219.

Så som har beskrivits tidigare i detta kapitel kan du spara en bild av en graf. Genom att använda kommandot **CyclePic** kan du bläddra igenom en serie grafbilder för att skapa en animering.

Kommandot CyclePic

Innan du använder **CyclePic** måste du ha en serie grafbilder med samma basnamn och som är numrerade i följd med början på 1 (exempelvis pic1, pic2, pic3, . . .).

När du bläddrar igenom bilderna använder du följande syntax:

CyclePic *picNamnSträng*, *n* [,*vänta*] [,*cykler*] [,*riktning*]

- *riktning*: 1 = framåt/circulär cykel; ⁻1= framåt/bakåt
- *cykler*: antal gånger som cykeln ska upprepas
- *vänta*: sekunder mellan r
- *n*: antal bilder i cykeln
- *picNamnSträng*: basnamnet på bilderna inom citattecken, t ex "pic"

Exempel

I detta exempelprogram (med namnet **cyc**) genereras 10 vyer av en 3D-graf, med varje vy roterad 10° ytterligare runt Z-axeln. Information om varje kommando finns i appendix A. Information om hur du använder Program Editor finns i kapitel 17.

Kommentarer inleds med ◉. Tryck på:
TI-89: [◆] [)]
TI-92 Plus: [2nd] X

För ϕ, tryck på:
TI-89: [◆] [(] [alpha] F
TI-92 Plus: [◆] G F

För #, tryck
TI-89: [CATALOG]
TI-92 Plus: [2nd][CATALOG]
och välj från listan.

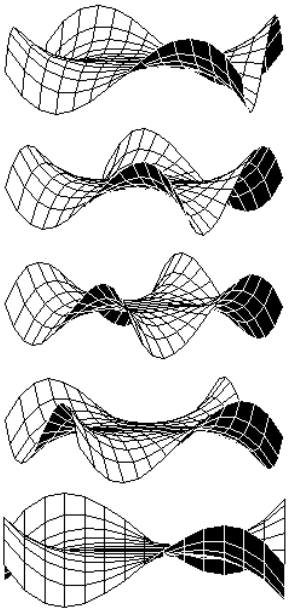
För &, tryck:
TI-89: [◆] [×]
TI-92 Plus: [2nd] H

Obs! *På grund av dess komplexitet tar det flera minuter för detta program att köras.*

Programlista

```
:cyc()
:Prgm
:local i
:◉Set mode and Window-variables
:setMode("graph","3d")
:70→eyeϕ
:⁻10→xmin
:10→xmax
:14→xgrid
:⁻10→ymin
:10→ymax
:14→ygrid
:⁻10→zmin
:10→zmax
:1→zscl
:◉Define the function
:(x^3*y-y^3*x)/390→z1(x,y)
:◉Generate pics and rotate
:For i,1,10,1
:  i*10→eyeθ
:  DispG
:  StoPic #("pic" & string(i))
:EndFor
:◉Display animation
:CyclePic "pic",10,.5,5,⁻1
:EndPrgm
```

Varannan graf från programmet



När du har matat in detta program i Program Editor går du till grundfönstret och matar in `cyc()`.

Spara och öppna en grafdatabas

En grafdatabas är en uppsättning av alla element som definierar en viss graf. Genom att spara en grafdatabas som en GDB-variabel kan du återskapa den grafen vid ett senare tillfälle genom att öppna dess sparade databasvariabel.

Element i en grafdatabas

En grafdatabas kan bestå av följande:

- Lägesinställningar ([MODE]) för Graph, Angle, Complex Format och Split Screen (bara om du använder tvågrafsläget).
- Alla funktioner i Y= Editor ([◆] [Y=]), även visningsformat och vilka funktioner som är markerade.
- Tabellparametrar ([◆] [TblSet]), Window-variabler ([◆][WINDOW]) och grafformat ([F1] 9 — eller — **TI-89**: [◆][I] **TI-92 Plus**: [◆]F).

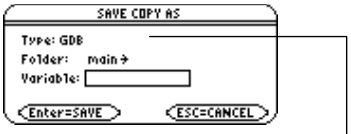
__Obs!__ I tvågrafsläget sparas elementen för båda graferna i en enda databas.

En grafdatabas innehåller inte ritobjekt eller statistiska plottningar.

Spara den aktuella grafdatabasen

Gör följande från Y= Editor, Window Editor, tabellfönstret eller graffönstret:

1. Tryck på [F1] och välj 2:Save Copy As.
2. Ange mappen och ett unikt variabelnamn.
3. Tryck på [ENTER]. När du har skrivit i en inmatningsruta, som exempelvis Variable, måste du trycka på [ENTER] två gånger.



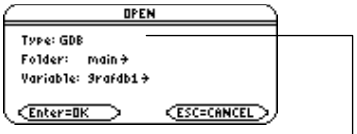
Obs! Om du är i graffönstret, se till att Type=GDB.

Öppna en grafdatabas

Varning! När du öppnar en grafdatabas ersätts all information i den aktuella databasen. Du kanske vill spara den aktuella grafdatabasen innan du öppnar en sparad databas.

Gör följande från Y= Editor, Window Editor, tabellfönstret eller graffönstret:

1. Tryck på [F1] och välj 1:Open.
2. Välj mappen och variabeln som innehåller grafdatabasen som du vill öppna.
3. Tryck på [ENTER].



Obs! Om du är i graffönstret, se till att Type=GDB.

Ta bort en grafdatabas

Oanvända GDB-variabler tar upp minne i räknaren. Du kan ta bort dem genom att använda VAR-LINK-fönstret ([2nd] [VAR-LINK]), som beskrivs i kapitel 21.

Från grundfönstret eller ett program

Du kan spara och öppna (hämta) en grafdatabas genom att använda kommandona **StoGDB** och **RclGDB**, vilket beskrivs i appendix A.

Tabeller

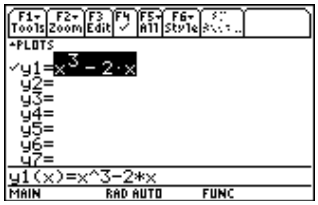
13

Översikt över tabeller 222
Översikt över stegen i tabellgenerering 223
Mata in tabellparametrar 224
Visa en automatisk tabell 226
Skapa en manuell tabell 229

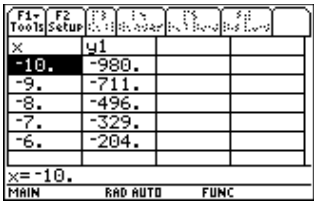
I kapitel 6: Grundläggande funktionsplottning lärde du dig att definiera och plotta en funktion.

Genom att använda en tabell kan du visa en definierad funktion i tabellform.

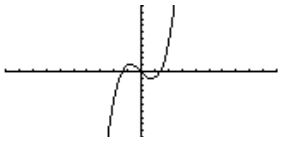
I Y= Editor visas en algebraisk representation.



Obs! *Tabeller är inte tillgängliga i 3D grafläget.*



I tabellfönstret visas en numerisk representation.



I graffönstret visas en grafisk representation.

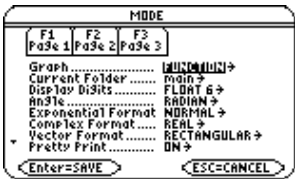
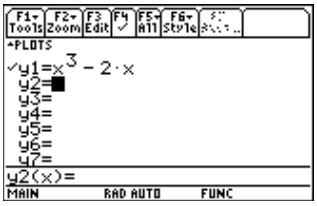
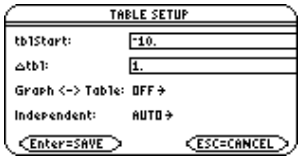
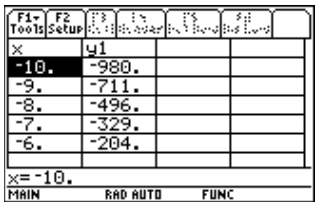
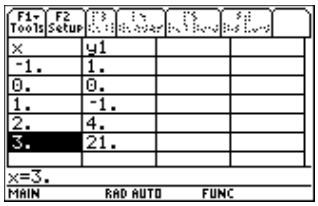
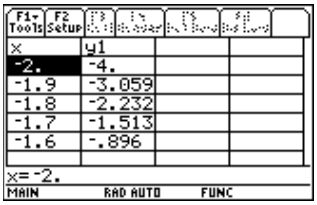
I tabellen visas en lista med värden för den oberoende variabeln och där visas också motsvarande värde för det beroende variabeln.

Oberoende variabel

$$y(x) = x^3 - 2x$$

Beroende variabel

Beräkna funktionen $y=x^3-2x$ för varje heltal mellan ⁻10 och 10. Hur många teckenbyten finns det och var uppstår de?

Steg	TI-89 Tangenter	TI-92 Plus Tangenter	Teckenfönster
1. Visa dialogrutan MODE. Välj grafläget FUNCTION.	[MODE] [▶] 1 [ENTER]	[MODE] [▶] 1 [ENTER]	
2. Visa och rensa Y= Editor. Definiera därefter $y1(x) = x^3 - 2x$.	[◆] [Y=] [F1] 8 [ENTER] [ENTER] X [^] 3 [−] 2 X [ENTER]	[◆] [Y=] [F1] 8 [ENTER] [ENTER] X [^] 3 [−] 2 X [ENTER]	
3. Ställ in tabellparametrarna till: tblStart = ⁻10 Δtbl = 1 Graph < - > Table = OFF Independent = AUTO	[◆] [TblSet] [(-)] 1 0 [▼] 1 [▼] [▶] 1 [▼] [▶] 1 [ENTER]	[◆] [TblSet] [(-)] 1 0 [▼] 1 [▼] [▶] 1 [▼] [▶] 1 [ENTER]	
4. Visa tabellfönstret.	[◆] [TABLE]	[◆] [TABLE]	
5. Rulla igenom tabellen. Lägg märke till att y1 byter tecken vid x = ⁻1, 1, och 2. *Du bläddrar en sida i taget genom att använda [2nd] [▼] och [2nd] [▲].*	[▼] och [▲] om det behövs	[▼] och [▲] om det behövs	
6. Zooma in på teckenbytet mellan x = ⁻2 och x = ⁻1 genom att ändra tabellparametrarna till: tblStart = ⁻2 Δtbl = .1	[F2] [(-)] 2 [▼] . 1 [ENTER] [ENTER]	[F2] [(-)] 2 [▼] . 1 [ENTER] [ENTER]	

Översikt över stegen i tabellgenerering

Du genererar en tabell med värden för en eller flera funktioner med hjälp av de allmänna stegen som visas här nedan. Mer specifik information om hur du matar in tabellparametrar och visar tabellen finns på sidorna som följer.

Generera en tabell

Obs! *Tabeller är inte tillgängliga i 3D grafläget.*

Tips! *Mer information om hur du definierar och markerar funktioner med Y= Editor finns i kapitel 6.*

Tips! *Du kan ange:*

- *En automatisk tabell*
  - *Baserad på startvärden.*
  - *Som matchar en graf.*
- *En manuell (fråga) tabell.*

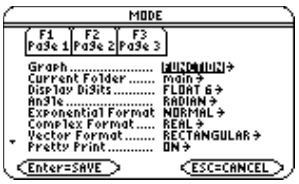
Ställ in Graph-läget och om det behövs vinkelenheten (Angle) ([MODE]).

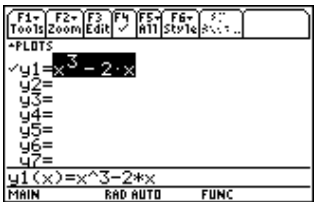
Definiera funktioner i Y= Editor ([◆] [Y=]).

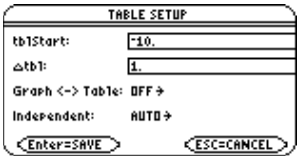
Markera ([F4]) vilka funktioner som ska visas i tabellen.

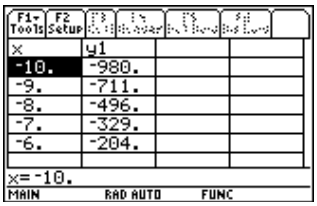
Ange tabellens parametrar ([◆] [TblSet]).

Visa tabellen ([◆] [TABLE]).









Utforska tabellen

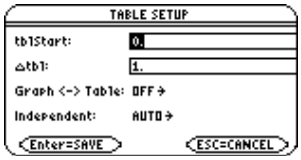
Från tabellfönstret kan du göra följande:

- Rulla igenom tabellen för att visa värdena på andra sidor.
- Markera en cell för att visa dess fullständiga värde.
- Ändra tabellens inställningsparametrar. Genom att ändra startvärdet eller steglängden som används för den oberoende variabeln kan du zooma in eller ut på tabellen för att visa olika detaljnivåer.
- Ändra cellbredden.
- Redigera markerade funktioner.
- Skapa eller redigera en manuell tabell som visar endast markerade värden för den oberoende variabeln.

Du matar in de ursprungliga parametrarna för en tabell genom att använda dialogrutan TABLE SETUP. När tabellen visas kan du även använda denna dialogruta för att ändra parametrarna.

Visa dialogrutan TABLE SETUP

Visa dialogrutan TABLE SETUP genom att trycka på ◆ [TblSet].
I tabellfönstret kan du också trycka på [F2].



***Obs!** Tabellen startar från början vid tblStart, men du kan använda ⊕ för att rulla till tidigare värden.*

Inställnings-parameter	Beskrivning
tblStart	Om Independent = AUTO och Graph < - > Table = OFF, anger denna parameter startvärdet för den oberoende variabeln.
Δtbl	Om Independent = AUTO och Graph < - > Table = OFF, anger denna parameter steglängden för den oberoende variabeln. Δtbl kan vara positiv eller negativ, men inte noll.
Graph < - > Table	Om Independent = AUTO: OFF – Tabellen baseras på de värden du matar in för tblStart och Δtbl. ON – Tabellen baseras på samma oberoende variabelvärden som används för att plotta funktionerna i graffönstret. Dessa värden är beroende av Window-variablerna som matas in i Window Editor (kapitel 6) och storleken för det delade fönstret (kapitel 14).
Independent	AUTO –En serie värden genereras automatiskt för den oberoende variabeln som baseras på tblStart, Δtbl och Graph < - > Table.
	ASK – Du kan skapa en tabell manuellt genom att ange särskilda värden för den oberoende variabeln.

Vilka inställnings-parametrar ska jag använda

Generera:	tblStart	Δtbl	Graph < – > Table	Independent
En automatisk tabell				
• Baserad på ursprungliga värden	värde	värde	OFF	AUTO
• Som matchar diagramfönstret	—	—	ON	AUTO
En manuell tabell	—	—	—	ASK

"—" betyder att värden som anges för denna parameter ignoreras för den angivna typen av tabell.

I grafläget SEQUENCE (kapitel 9) använder du heltal för tblStart och Δtbl.

Ändra inställnings-parametrarna

Gör följande i dialogrutan TABLE SETUP:

1. Använd ⊖ och ⊕ för att markera värdet eller inställningen som du ska ändra.
2. Ange det nya värdet eller den nya inställningen.

Om du vill ändra:	Gör du följande:
tblStart eller Δtbl	Skriv det nya värdet. Det befintliga värdet tas bort när du börjar skriva. – eller – Tryck på ◀ eller ▶ för att ta bort markeringen. Redigera sedan det befintliga värdet.
Graph < – > Table eller Independent	Tryck på ◀ eller ▶ för att visa en meny med giltiga inställningar. Gör sedan något av följande: • Flytta markören för att markera inställningen och tryck på [ENTER]. – eller – • Tryck på numret som motsvarar inställningen.

Tips! Du stänger en meny eller dialogruta utan att spara några ändringar genom att trycka på [ESC] i stället för [ENTER].

3. När du har ändrat värdena eller inställningarna trycker du på [ENTER] för att spara ändringarna och stänga dialogrutan.

Från grundfönstret eller ett program

Du kan ställa in parametrarna för en tabell från grundfönstret eller ett program. Du kan göra följande:

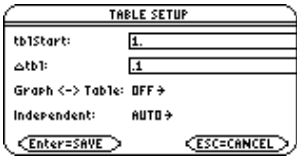
- Spara värden direkt till systemvariablerna tblStart och Δtbl. Se "Spara och hämta variabelvärden" i kapitel 2.
- Ställ in Graph < – > Table och Independent med hjälp av funktionen **setTable**. Se appendix A.

Om **Independent = AUTO** i dialogrutan TABLE SETUP genereras automatiskt en tabell när du visar tabellfönstret. Om **Graph <–> Table = ON** matchar tabellen spårningsvärdena från graffönstret. Om **Graph <–> Table = OFF** baseras tabellen på de värden du anger för **tblStart** och **Δtbl**.

Innan du börjar

Definiera och välj funktioner i Y= Editor ([◆] [Y=]). I detta exempel används y1(x) = $x^3 - x/3$.

Mata därefter in de ursprungliga tabellparametrarna ([◆] [TblSet]).



Visa tabellfönstret

Du visar tabellfönstret genom att trycka på [◆] [TABLE] eller [APPS] 5.

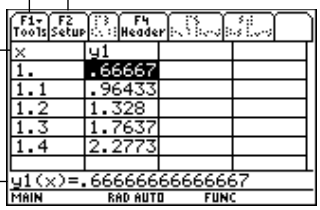
Från början markeras cellen som innehåller startvärdet för den oberoende variabeln. Du kan flytta markören till en cell som innehåller ett värde.

I första kolumnen visas värden för den oberoende variabeln.

I andra kolumner visas motsvarande värden för funktionenerna som markerats i Y= Editor.

I rubrikraden visas namnet på den oberoende variabeln (x) och på de markerade funktionerna (y1).

På inmatningsraden visas det fullständiga värdet i den markerade cellen.



***Tips!** Du kan rulla tillbaka från startvärdet genom att trycka på [▲] eller [2nd] [▲].*

Om du vill flytta markören:	Trycker du på:
En cell i taget	[▼], [▲], [▶] eller [◀]
En sida i taget	[2nd] och sedan [▼], [▲], [▶] eller [◀]

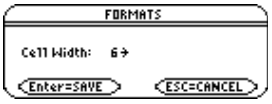
Rubrikraden och den första kolumnen låses så att de inte rullar ut från fönstret.

- När du bläddrar uppåt eller nedåt är variabel- och funktionsnamnen alltid synliga överst i fönstret.
- När du bläddrar till höger eller vänster är värdena för den oberoende variabeln alltid synliga längs fönstrets vänstra sida.

Ändra cellbredden

Cellbredden anger maximalt antal siffror och symboler (decimalkomma, minustecken och "E" för grundpotensformat) som kan visas i en cell. Alla celler i samma tabell har samma bredd.

Så här ändrar du cellbredden från tabellfönstret:



__Obs!__ Cellbredden är som standard 6.

1. Tryck på [F1] 9
   — eller —
   TI-89: [◆] [|]
   TI-92 Plus: [◆] F
2. Tryck på ⊙ eller ⊙ för att visa en meny med giltiga bredder (3 – 12).
3. Flytta markören för att markera ett antal och tryck på [ENTER]. (För enskilda siffror kan du skriva siffran och sedan trycka på [ENTER].)
4. Tryck på [ENTER] för att stänga dialogrutan och uppdatera tabellen.

Hur tal visas i en cell

Så långt det är möjligt visas ett tal som motsvarar de för tillfället markerade visningslägena (Display Digits, Exponential Format osv.). Talet kan avrundas. Däremot:

__Obs!__ Om en funktion är odefinierad vid ett visst värde visas undef *i cellen.*

- Om talet är för stort för den aktuella cellen avrundas talet och visas i grundpotensformat.
- Om cellbredden är för liten även för grundpotensformat visas "...".

__Tips!__ Använd [MODE] för att ange visningslägen.

Som standard är Display Digits = FLOAT 6. Med denna lägesinställning visas ett tal med upp till sex siffror, även om cellen är tillräckligt bred för att visa fler. Andra inställningar påverkar på liknande sätt ett visat tal.

__Tips!__ För att kunna se ett tal i fullständig precision markerar du cellen och tittar på inmatningsraden.

	Om cellbredden är:			
Fullständig precision	**3**	**6**	**9**	**12**
1.2345678901	1.2	1.2346	1.23457	1.23457
⁻123456.78	...	⁻1.2E5	⁻123457.	⁻123457.
.000005	...	5.E⁻6	.000005	.000005
1.2345678E19	...	1.2E19	1.2346E19	1.23457E19
⁻1.23456789012E⁻200	...	...	⁻1.2E⁻200	⁻1.2346E⁻200

Obs! Beroende på visningsläge, visas inte en del värden med alla siffror även om cellen är tillräckligt bred.

Om resultatet är i komplex form

I en cell visas så mycket som möjligt av ett komplext tal (enligt aktuella visningslägen) och i slutet visas "...".

När du markerar en cell som innehåller ett komplext tal visas på inmatningsraden de verkliga och imaginära delarna med maximalt fyra siffror vardera (FLOAT 4).

Redigera en markerad funktion

Du kan ändra en markerad funktion från en tabell utan att behöva använda Y= Editor.

1. Flytta markören till en cell i kolumnen för funktionen. I rubrikraden i tabellen visas funktionsnamnen (y1 osv.).
2. Tryck på [F4] för att flytta markören till inmatningsraden, där funktionen visas och är markerad.
3. Gör de ändringar som behövs.
   - Skriv den nya funktionen. Den gamla funktionen tas bort när du börjar skriva.

     – eller –
   - Tryck på [CLEAR] för att ta bort den gamla funktionen och skriv sedan den nya.

     – eller –
   - Tryck på ◀ eller ▶ för att ta bort markeringen. Redigera sedan funktionen.
4. Tryck på [ENTER] för att spara den redigerade funktionen och uppdatera tabellen. Den redigerade funktionen sparas även i Y= Editor.

Tips! *Du kan använda denna funktion för att visa en funktion utan att behöva lämna tabellen.*

Tips! *Om du vill avbryta ändringarna och återigen placera markören i tabellen trycker du på [ESC] i stället för [ENTER].*

Om du vill ändra inställningsparametrarna

När du har genererat en automatisk tabell kan du ändra inställningsparametrarna om det behövs.

Tryck på [F2] eller [◆] [TblSet] för att visa dialogrutan TABLE SETUP. Gör ändringarna enligt beskrivningen på sidan 224 och 225.

Skapa en manuell tabell

Om **Independent = ASK** i dialogrutan TABLE SETUP kan du skapa en tabell manuellt genom att ange särskilda värden för den oberoende variabeln.

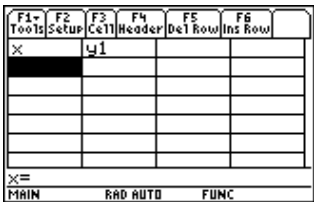
Visa tabellfönstret

Visa tabellfönstret genom att trycka på ◆ [TABLE] eller [APPS] 5.

Om du ställer in Independent = ASK (med ◆ [TblSet]) innan du visar en tabell för första gången, visas en tom tabell. Den första cellen i den oberoende variabelkolumnen markeras.

I rubrikraden visas namnet på den oberoende variabeln (x) och på de markerade funktionerna (y1).

Mata in ett värde här.



Om du visar en automatisk tabell och sedan ändrar den till Independent = ASK, fortsätter tabellen att visa samma värden. Du kan däremot inte längre se ytterligare värden genom att rulla uppåt eller nedåt i fönstret.

Mata in eller redigera ett oberoende variabelvärde

Du kan endast mata in ett värde i kolumn 1 (oberoende variabel).

1. Flytta markören för att markera cellen som du vill skriva i eller redigera.
   - Om du börjar med en tom tabell kan du endast mata in ett värde i efterföljande celler (rad 1, rad 2 osv.). Du kan inte hoppa över celler (rad1, rad 3).
   - Om en cell i kolumn 1 innehåller ett värde kan du redigera det värdet.
2. Tryck på [F3] för att flytta markören till inmatningsraden.
3. Skriv ett nytt värde eller uttryck, eller redigera ett befintligt värde.
4. Tryck på [ENTER] för att flytta värdet till tabellen och uppdatera motsvarande funktionsvärden.

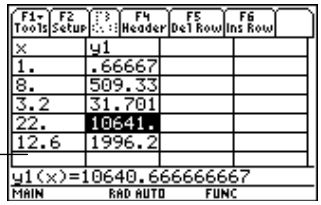
__Tips!__ Du behöver inte trycka på [F3] för att kunna mata in ett nytt värde, det räcker med att du börjar skriva.

Markören återgår till cellen du skrev i. Du kan använda ⊖ för att flytta till nästa rad.

Mata in värden i vilken numerisk ordning som helst.

Mata in nya värden här.

Visa det fullständiga värdet i den markerade cellen.



__Obs!__ I detta exempel kan du flytta markören till kolumn 2, men du kan bara skriva värden i kolumn 1.

Mata in en lista i den oberoende variabelkolumnen

1. Flytta markören för att markera en cell i den oberoende variabelkolumnen.
2. Tryck på [F4] för att flytta markören till inmatningsraden.
3. Skriv en serie värden inom klammerparenteser { } avgränsade av komman. Exempel:

```
x={1,1.5,1.75,2}
```

   Du kan också mata in en listvariabel eller ett uttryck som beräknas till en lista.

4. Tryck på [ENTER] för att flytta värdena till den oberoende variabelkolumnen. Tabellen uppdateras så att motsvarande funktionsvärden visas.

Obs! *Om den oberoende variabelkolumnen innehåller befintliga värden visas de som en lista (som du kan redigera).*

Lägga till, ta bort eller rensa

Om du vill:	Gör du följande:
Infoga en ny rad ovanför en angiven rad	Markera en cell i den angivna raden och tryck: **TI-89:** [2nd] [F6] **TI-92 Plus:** [F6] Den nya raden är odefinierad (undef) tills du anger ett värde för den oberoende variabeln.
Ta bort en rad	Markera en cell i raden och tryck på [F5]. Om du markerar en cell i den oberoende variabelkolumnen kan du även trycka på [←].
Rensa hela tabellen (*förutom* de markerade Y=-funktionerna)	Tryck på [F1] 8. När du uppmanas att bekräfta trycker du på [ENTER].

Cellbredd och visningsformat

Det är flera faktorer som påverkar hur tal visas i en tabell. Se "Ändra cellbredden" och "Hur tal visas i en cell" på sidan 227.

Från grundfönstret eller ett program

Systemvariabeln tblInput innehåller en lista över alla oberoende variabelvärden som har matats in i tabellen, även de som inte visas för tillfället. Variabeln tblInput används också för automatiska tabeller men den innehåller bara de oberoende variabelvärden som visas för tillfället.

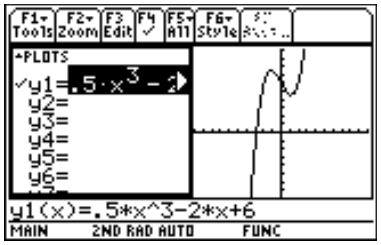
Innan du visar en tabell kan du spara en lista med värden direkt till systemvariabeln tblInput.

Delade fönster

14

Förhandsvisning av funktioner för delade fönster 232
Aktivera och avaktivera läget för delat fönster..... 233
Markera vilken applikation som ska vara aktiv 235

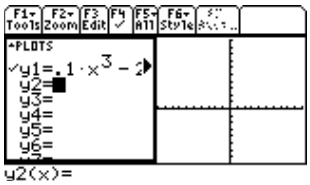
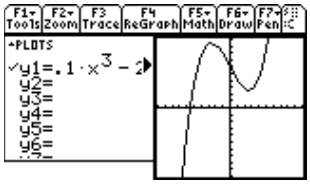
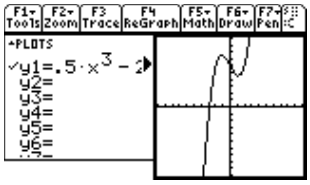
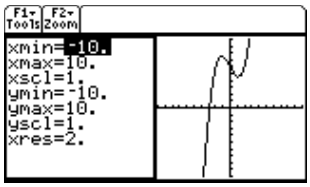
I TI-89 / TI-92 Plus kan du dela fönstret för att visa två applikationer på samma gång.



Det kan exempelvis vara användbart att visa både Y= Editor och graffönstret så att du kan visa en lista över funktioner och se hur de avbildas som grafer.

Förhandsvisning av funktioner för delade fönster

Dela fönstret så Y= Editor och graffönstret visas samtidigt. Utforska sedan hur ett polynom ändras när dess koefficient ändras.

Steg	TI-89 Tangenter	TI-92 Plus Tangenter	Teckenfönster
1. Visa dialogrutan MODE. För Graph väljer du FUNCTION. För Split Screen väljer du LEFT-RIGHT. För Split 1 App väljer du Y= Editor. För Split 2 App väljer du Graph.	[MODE] ▶ 1 [F2] ▶ 3 ▼ ▶ 2 ▼ ▶ 4 [ENTER]	[MODE] ▷ 1 [F2] ▷ 3 ▽ ▷ 2 ▽ ▷ 4 [ENTER]	
2. Rensa Y= Editor och stäng av eventuella plottningar av statistiska data. Definiera därefter y1(x) = .1x³–2x+6. *En tjock ram runt Y= Editor betyder att programmet är aktivt. När det är aktivt sträcker sig inmatningsraden hela vägen över fönstret.*	[F1] 8 [ENTER] [F5] 5 [ENTER] . 1 X [^] 3 [−] 2 X [+] 6 [ENTER]	[F1] 8 [ENTER] [F5] 5 [ENTER] . 1 X [^] 3 [−] 2 X [+] 6 [ENTER]	
3. Välj visningsfönstret ZoomStd, som växlar till graffönstret och plottar funktionen. *Den tjocka ramen visas nu runt graffönstret.*	[F2] 6	[F2] 6	
4. Växla till Y= Editor. Redigera därefter y1(x) och ändra.1x³ till .5x³. [2nd] [⊞] *är den sekundära funktionen av* [APPS]. *Den tjocka ramen visas nu runt Y= Editor.*	[2nd] [⊞] ▲ [ENTER] ◀ ▶ ▶ ← 5 [ENTER]	[2nd] [⊞] △ [ENTER] ◁ ▷ ▷ ← 5 [ENTER]	
5. Växla till graffönstret som ritar om den redigerade funktionen. *Den tjocka ramen visas nu runt graffönstret.*	[2nd] [⊞]	[2nd] [⊞]	
6. Växla till Y= Editor. Öppna sedan Window Editor i stället.	[2nd] [⊞] ◆ [WINDOW]	[2nd] [⊞] ◆ [WINDOW]	
7. Öppna grundfönstret. Avsluta sedan grundfönstret när det är i full storlek.	[2nd] [QUIT] [2nd] [QUIT]	[2nd] [QUIT] [2nd] [QUIT]	

Aktivera och avaktivera läget för delat fönster

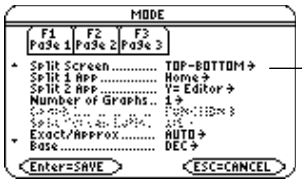
Du ställer in ett delat fönster med hjälp av dialogrutan MODE, där du anger lägesinställningar. När du har ställt in ett delat fönster förblir det så tills du ändrar det.

Ställa in Split Screen-läget

1. Tryck på [MODE] för att visa dialogrutan MODE.
2. Eftersom de lägen som har att göra med delade fönster är listade i dialogrutan MODE kan du göra något av följande:
   - Använd ⊖ för att rulla nedåt.

     – eller –
   - Tryck på [F2] för att visa sidan 2.
3. Ställ in någon av följande inställningar för Split Screen. I kapitel 2 beskrivs de steg du måste följa för att ändra en lägesinställning.

Inställningar för delat fönster
TOP-BOTTOM
LEFT-RIGHT



När du ställer in Split Screen = TOP-BOTTOM eller LEFT-RIGHT, aktiveras alternativ som förut varit inaktiva, som exempelvis Split 2 App.

Ställa in startprogrammen

Innan du trycker på [ENTER] för att stänga dialogrutan MODE kan du använda läget Split 1 App och läget Split 2 App för att välja de applikationer du vill använda.

Läge	Anger programmet i:
Split 1 App	Den översta eller vänstra delen av det delade fönstret.
Split 2 App	Den nedersta eller högra delen av det delade fönstret.

Obs! *I tvågrafsläge, som beskrivs i kapitel 12, kan samma program visas i båda delarna av ett delat fönster.*

Om du ställer in Split 1 App och Split 2 App till samma applikation avslutas läget för delat fönster och applikationer visas i ett fönster i full storlek.

Du kan öppna olika applikationer efter att du har visat det delade fönstret, enligt beskrivning på sidan 235.

Andra lägen som påverkar ett delat fönster

Läge	Beskrivning
Number of Graphs ***Obs!** Låt denna inställning vara 1 om du inte har läst tillhörande avsnitt i kapitel 12.*	Du kan ange och visa två oberoende uppsättningar grafer. Detta är en avancerad graffunktion som beskrivs i "Använda tvågrafsläget" i kapitel 12.

Delade fönster och bildpunktskoordinater

I TI-89 / TI-92 Plus finns kommandon som använder bildpunkts-koordinater för att rita linjer, cirklar osv i graffönstret. I tabellen nedan visas hur inställningarna Split Screen påverkar antalet bildpunkter som är tillgängliga i graffönstret.

***Tips!** I "Rita i graffönstret " i kapitel 17 finns en lista med ritkommandon.*

***Obs!** På grund av ramen som anger att programmet är aktivt, är det visningsbara området mindre än på ett fönster i full storlek.*

TI-89:

		Split 1App		Split 2App	
Delning	**Förh.**	**x**	**y**	**x**	**y**
FULL	N/A	0 – 158	0 – 76	N/A	N/A
TOP-BOTTOM	1:1	0 – 154	0 – 34	0 – 154	0 – 34
LEFT-RIGHT	1:1	0 – 76	0 – 72	0 – 76	0 – 72

TI-92 Plus:

		Split 1App		Split 2App	
Delning	**Förh.**	**x**	**y**	**x**	**y**
FULL	N/A	0 – 238	0 –102	N/A	N/A
TOP-BOTTOM	1:1	0 – 234	0 – 46	0 – 234	0 – 46
	1:2	0 – 234	0 – 26	0 – 234	0 – 68
	2:1	0 – 234	0 – 68	0 – 234	0 – 26
LEFT-RIGHT	1:1	0 – 116	0 – 98	0 –116	0 – 98
	1:2	0 – 76	0 – 98	0 – 156	0 – 98
	2:1	0 – 156	0 – 98	0 – 76	0 – 98

Avsluta Split Screen-läget

Metod 1: Tryck på [MODE] för att visa dialogrutan MODE. Ställ sedan in Split Screen = FULL. När du trycker på [ENTER] för att stänga dialogrutan visas i det fullständiga fönstret den applikation som anges i Split 1 App.

Metod 2: Tryck på [2nd] [QUIT] två gånger för att visa grundfönstret i full storlek.

När du stänger av din TI-89 / TI-92 Plus

Läget för delat fönster avaktiveras inte när du stänger av TI-89 / TI-92 Plus.

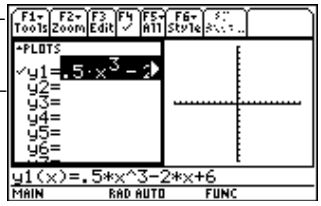
Om räknaren stängs av:	Händer följande när du slår på räknaren igen:
När du trycker på [2nd] [OFF]	Det delade fönstret är fortfarande aktivt men grundfönstret visas alltid i stället för den applikation som var aktivt när du tryckte på [2nd] [OFF].
Med funktionen APD™ (Automatic Power Down™), eller när du trycker på [◆] [OFF].	Det delade fönstret är så som det var när du stängde av räknaren.

Markera vilken applikation som ska vara aktiv

> När fönstret är delat kan endast en av de två applikationerne vara aktivt vid ett tillfälle. Du kan enkelt växla mellan befintliga program, eller så kan du öppna en annat applikation.

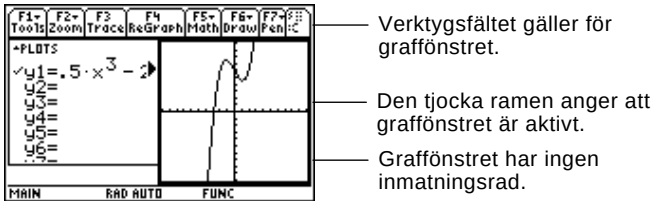
Det aktiva programmet

- Det aktiva programmet är omgivet av en tjock ram.
- Verktygsfältet och statusfältet, som alltid är lika brett som fönstret, hör alltid ihop med det aktiva applikationer.
- När det gäller program som har en inmatningsrad (exempelvis grundfönstret och Y= Editor), är inmatningsraden lika bred som fönstret *endast när det programmet är aktivt*.



Växla mellan program

Tryck på [2nd] [⊞] (den sekundära funktionen av [APPS]) för att växla från en applikation till en annat.



Öppna ett annat program

__Obs!__ Se även "Använda [2nd] [QUIT] för att visa grundfönstret" på sidan 236.

Metod 1:

1. Använd [2nd] [⊞] för att växla till applikationer du vill ersätta.
2. Använd [APPS] eller [◆] (t ex [◆] [WINDOW]) för att välja en nytt applikation.

Om du väljer ett applikation som redan visas blir den applikationer aktiv.

Metod 2:

1. Tryck på [MODE] och därefter på [F2].
2. Ändra Split 1 App och/eller Split 2 App.

Om du ställer in Split 1 App och Split 2 App till samma program, avslutas läget för delat fönster och applikationer visas i ett fönster i full storlek.

__Obs!__ I tvågrafsläget, som beskrivs i kapitel 12, kan samma applikation visas i båda delarna av ett delat fönster.

Använda 2nd [QUIT] för att visa grundfönstret

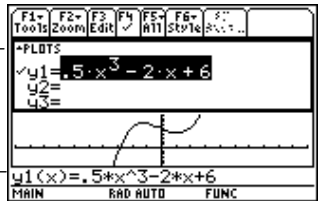
Om grundfönstret:	Händer följande när du trycker på 2nd [QUIT]:
Inte redan visas	Grundfönstret öppnas i stället för det aktiva applikationen.
Visas men inte är det aktiva programmet	Du växlar till grundfönstret som blir det aktiva applikationen.
Är den aktiva applikationen	Läget för delat fönster avslutas och grundfönstret visas i full storlek.

Tips! *Om du trycker på* 2nd [QUIT] *två gånger avslutas alltid läget för delat fönster.*

Använda en övre och undre delning

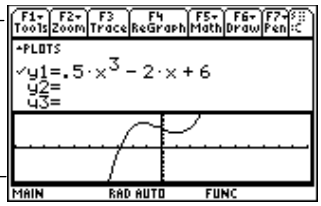
När du väljer en TOP-BOTTOM delning måste du komma ihåg att inmatningsraden och verktygsfältet alltid är relaterade till den aktiva applikationen. Exempel:

Inmatningsraden gäller för den aktiva applikationen Y= Editor, *inte* för graffönstret.



Obs! *Både vid övre/undre och vänster/höger delningar används samma metod för att markera ett program.*

Verktygsfältet gäller för det aktiva applikationens graffönster, *inte* för Y= Editor.



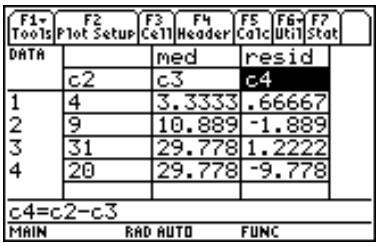
Data/Matrix Editor

15

Förhandsvisning av funktioner i Data/Matrix Editor 238
Översikt över list-, data- och matrisvariabler 239
Starta en Data/Matrix Editor-session 241
Mata in och visa cellvärden 243
Infoga och ta bort rader, kolumner och celler 246
Definiera ett kolumnhuvud med ett uttryck..... 248
Använda funktionerna Shift och CumSum i ett kolumnhuvud 250
Sortera kolumner 251
Spara en kopia av en list-, data- eller matrisvariabel..... 252

Data/Matrix Editor har två huvudsakliga funktioner.

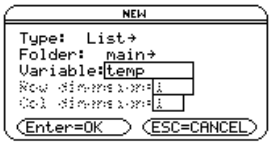
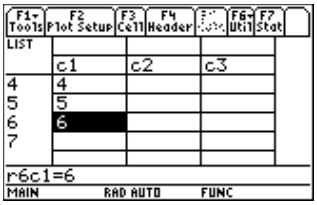
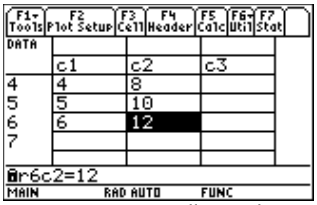
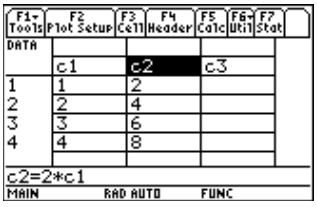
- I detta kapitel beskrivs hur du använder Data/Matrix Editor för att skapa och redigera en lista, matris eller datavariabel.



- I kapitel 16 beskrivs hur du använder Data/matrix Editor för att göra statistiska beräkningar och plotta statistiska data.

Förhandsvisning av funktioner i Data/Matrix Editor

Använd Data/Matrix Editor för att skapa en listvariabel med en kolumn. Lägg sedan till en andra kolumn med information. Lägg märke till att listvariabeln (som kan ha endast en kolumn) automatiskt konverteras till en datavariabel (som kan ha flera kolumner).

Steg	TI-89 Tangenter	TI-92 Plus Tangenter	Teckenfönster
1. Starta Data/Matrix Editor och skapa en ny listvariabel med namnet TEMP.	[APPS] 6 3 ▶ 3 ▼ ▼ T E M P [ENTER] [ENTER]	[APPS] 6 3 ▶ 3 ▼ ▼ T E M P [ENTER] [ENTER]	
2. Mata in en kolumn med tal. Flytta sedan markören uppåt en cell (för att se att värdet för den markerade cellen visas på inmatningsraden). *Ordet LIST visas i det övre vänstra hörnet för att ange att det är en listvariabel.* *Du kan använda ▼ i stället för [ENTER] för att skriva information i en cell.*	1 [ENTER] 2 [ENTER] 3 [ENTER] 4 [ENTER] 5 [ENTER] 6 [ENTER] ▲	1 [ENTER] 2 [ENTER] 3 [ENTER] 4 [ENTER] 5 [ENTER] 6 [ENTER] ▲	
3. Flytta till kolumn 2 och definiera dess kolumnrubrik så att dess värde är dubbelt så stort som det i kolumn 1. *Ordet DATA visas i det övre vänstra hörnet för att ange att listvariabeln har konverterats till en datavariabel.*	▶ [F4] 2 [×] [alpha] C 1 [ENTER]	▶ [F4] 2 [×] C 1 [ENTER]	 🔒 anger att cellen är i en definierad kolumn.
4. Flytta till kolumnrubrikcellen för kolumn 2 för att visa dess definition på inmatningsraden. *När markören är i rubrikcellen behöver du inte trycka på [F4] för att definiera den. Skriv bara uttrycket.*	[2nd] ▲ ▲	[2nd] ▲ ▲	
5. Gå till grundfönstret och återgå sedan till den aktuella variabeln.	[HOME] [APPS] 6 1	[◆] [HOME] [APPS] 6 1	
6. Rensa innehållet i variabeln. *Att endast ta bort informationen konverterar inte datavariabeln tillbaka till en listvariabel.*	[F1] 8 [ENTER]	[F1] 8 [ENTER]	

Tips! Om du inte behöver spara den aktuella variabeln kan du använda den som ett *kladdblock*. Nästa gång du behöver en variabel för tillfällig data kan du rensa den aktuella variabeln och återanvända den. På så sätt kan du mata in tillfälliga data utan att behöva skapa en ny variabel varje gång, vilket kräver mycket minne.

Översikt över list-, data- och matrisvariabler

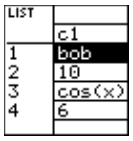
> För att kunna använda Data/Matrix Editor effektivt måste du förstå vad list-, data- och matrisvariabler är.

Listvariabel

En lista är en serie poster (tal, uttryck eller teckensträngar) som kan (men behöver inte) vara relaterade. Varje post kallas för ett element. I Data/Matrix Editor gäller följande för en listvariabel:

Obs! *Om du anger mer än en kolumn med element i en listvariabel konverteras den automatiskt till en datavariabel.*

- Den visas som en enskild kolumn med element, varje element i en separat cell.
- Den måste vara sammanhängande; tomma celler är inte tillåtna i listan.
- Den kan innehålla upp till 999 element.

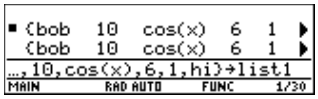


Kolumnrubrik och cellhuvud sparas inte som en del av listan.

Tips! *När du har skapat en lista i Data/Matrix Editor kan du använda listan i vilken applikation som helst (exempelvis i grundfönstret).*

I grundfönstret (eller någon annanstans där du kan använda en lista) kan du mata in en lista som en följd av element inom parenteser { } och avgränsade av komman.

Även om du måste använda komman för att avgränsa element på inmatningsraden, används mellanslag för att avgränsa element i historiklistan.



För att referera till ett element i listan använder du formatet till höger.

```
list1[1]
```

Elementnummer (eller index)

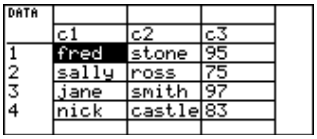
Listvariabelns namn

Datavariabel

En datavariabel är i huvudsak en samling listor som kan vara (men behöver inte vara) relaterade. I Data/Matrix Editor gäller följande för en datavariabel:

Obs! *För statistiska beräkningar måste kolumnerna vara av samma längd.*

- Den kan bestå av upp till 99 kolumner.
- Den kan bestå av upp till 999 element i varje kolumn. Beroende på typen av data behöver inte alla kolumner vara av samma längd.
- Den måste ha sammanhängande kolumner; tomma celler är inte tillåtna i en kolumn.



Datavariabel (fortsättning)

Från grundfönstret eller ett program kan du använda kommandot **NewData** för att skapa en datavariabel som består av befintliga listor.

```
NewData data1,list1,list2
```

Namn på en befintlig listvariabel

Namnet på den datavariabel som ska skapas

Även om du inte kan visa en datavariabel direkt i grundfönstret kan du visa en angiven kolumn eller ett angivet element.

Namnet på datavariabeln

Kolumnnummer

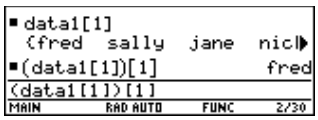
```
data1[1]
(data1[1])[1]
```

Elementnummer i kolumnen

Kolumnnummer

Exempel:

Visar kolumn 1 av variabeln data1

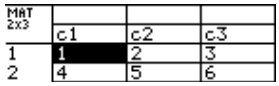


Visar element 1 i kolumn 1 av variabeln data1.

Matrisvariabel

En matris är en rektangulär uppställning av element. När du skapar en matris i Data/Matrix Editor måste du ange antalet rader och kolumner (även om du kan lägga till eller ta bort rader och kolumner senare). I Data/Matrix Editor gäller följande för en matrisvariabel:

- Den liknar en datavariabel till utseendet men alla kolumner måste vara i samma längd.
- Den skapas från början med 0 i varje cell. Du kan därefter skriva ett värde i stället för 0.



Visar matrisens storlek.

Tips! *När du har skapat en matris i Data/Matrix Editor kan du använda matrisen i vilken applikation som helst (exempelvis grundfönstret).*

Från grundfönstret eller ett program kan du använda STO▸ för att spara en matris med någon av metoderna som visas till höger.

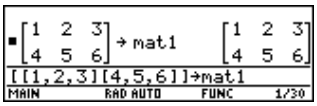
rad 1 rad 2

```
[[1,2,3][4,5,6]]→mat1
[1,2,3;4,5,6]→mat1
```

rad 1 rad 2

Obs! *Använd hakparenteser då du vill ange ett specifikt element i en matris. Ange t ex mat1[2,1] om du vill öppna det första elementet i den andra raden.*

Även om du matar in en matris enligt ovan visas den i pretty print-läget i historiklistan i traditionellt matrisformat.

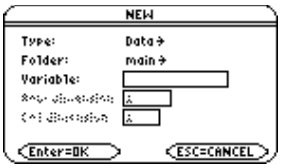


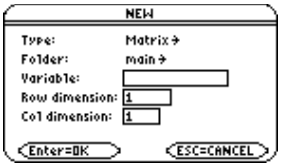
Varje gång du startar Data/Matrix Editor kan du skapa en ny variabel, fortsätta att använda den aktuella variabeln (variabeln som visades sist du använde Data/Matrix Editor) eller öppna en befintlig variabel.

Skapa en ny data-, matris- eller listvariabel

1. Tryck på [APPS] och välj 6:Data/Matrix Editor.
2. Välj 3:New.

3. Välj information för den nya variabeln.



Post	Vad du kan göra:
Type	Markera vilken typ av variabel du ska skapa. Tryck på ⊙ för att visa en meny med tillgängliga typer.
Folder	Markera mappen som den nya variabeln ska sparas i. Tryck på ⊙ för att visa en meny med befintliga mappar. Information om mappar finns i kapitel 5.
Variable	Skriv ett nytt variabelnamn. Om du matar in en variabel som redan finns visas ett felmeddelande när du trycker på [ENTER]. När du trycker på [ESC] eller [ENTER] för att bekräfta felmeddelandet visas dialogrutan NEW på nytt.
Row dimension och Col dimension	Om Type = Matrix skriver du antalet rader och kolumner i matrisen. 

Obs! *Om du inte skriver ett variabelnamn visas grundfönstret.*

4. Tryck på [ENTER] (efter att ha skrivit i en inmatningsruta, som exempelvis Variable, trycker du på [ENTER] två gånger) för att skapa och visa en tom variabel i Data/Matrix Editor.

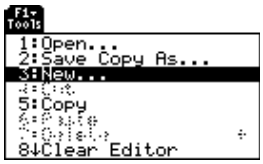
Använda den aktuella variabeln

Du kan när som helst lämna Data/Matrix Editor och gå till en annan applikation. Om du vill återgå till variabeln som visades när du lämnade Data/Matrix Editor trycker du på [APPS] 6 och väljer 1:Current.

Skapa en ny variabel från Data/Matrix Editor

Gör följande från Data/Matrix Editor:

1. Tryck på [F1] och välj 3:New.
2. Ange typ, mapp och variabelnamn. För en matris måste du också mata in antalet kolumner och rader.



Öppna en annan variabel

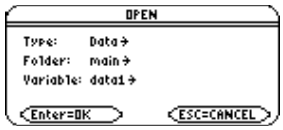
Du kan när som helst öppna en annan variabel.

1. Tryck på [F1] och välj 1:Open i Data/Matrix Editor.

   – eller –

   Tryck på [APPS] 6 och välj 2:Open från valfritt program.

2. Markera vilken typ, mapp och variabel du ska öppna.
3. Tryck på [ENTER].



Obs! *I* Variable *visas den första befintliga variabeln i alfabetisk ordning. Om det inte finns några befintliga variabler visas ingenting.*

Ta bort variabler

Eftersom alla variabler i Data/Matrix Editor sparas automatiskt kan det bli ganska många variabler, vilket tar upp mycket minne.

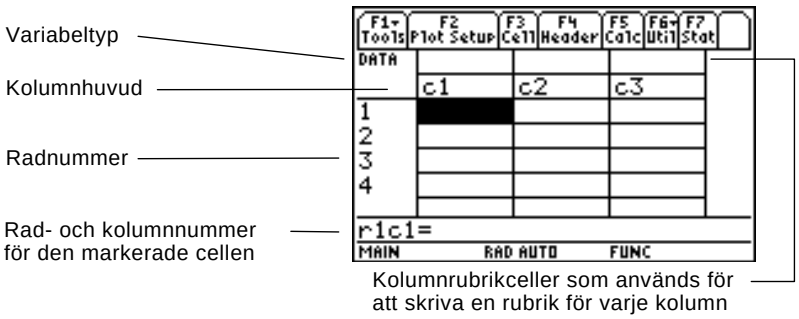
Du tar bort en variabel genom att använda fönstret VAR-LINK ([2nd] [VAR-LINK]). Information om VAR-LINK finns i kapitel 21.

Om du skapar en ny variabel är Data/Matrix Editor från början tom (för en list- eller datavariabel) eller fylld med tre nollor (för en matris). Om du öppnar en befintlig variabel visas värdena i den variabeln. Du kan därefter mata in ytterligare värden eller redigera befintliga värden.

Fönstret Data/Matrix Editor

Ett tomt Data/Matrix Editor-fönster visas nedan. När fönstret först visas markeras cellen vid rad 1, kolumn 1.

__Tips!__ Använd rubrikcellen överst vid varje kolumn för att identifiera informationen i varje kolumn.



När du matar in värden visas på inmatningsraden det fullständiga värdet av den markerade cellen.

Mata in och redigera värden i en cell

Du kan mata in vilken typ av uttryck som helst i en cell (tal, variabel, funktion, sträng m.m.).

1. Flytta markören för att markera cellen som du vill mata in eller redigera i.
2. Tryck på [ENTER] eller [F3] för att flytta markören till inmatningsraden.
3. Skriv ett nytt värde eller redigera det befintliga.
4. Tryck på [ENTER] för att mata in värdet i den markerade cellen.

__Tips!__ Du matar in ett nytt värde genom att börja skriva utan att trycka på [ENTER] eller [F3] först. Du måste däremot använda [ENTER] eller [F3] för att redigera ett befintligt värde.

När du trycker på [ENTER] flyttas markören automatiskt till nästa cell och markerar den så att du kan fortsätta mata in eller redigera värden. Variabeltypen påverkar dock i vilken riktning markören flyttas.

Variabeltyp	När du har tryckt på [ENTER] flyttas markören:
Lista eller data	Ned till cellen på nästa rad.
Matris	Till höger till cellen i nästa kolumn. Markören flyttas automatiskt från den sista cellen på en rad till den första cellen på nästa rad. På detta sätt kan du mata in värden för rad 1, rad 2 osv.

__Obs!__ Du kan också mata in ett värde från inmatningsraden genom att använda ⊖ eller ⊕.

Rulla igenom redigeraren

Om du vill flytta markören:	Trycker du på:
En cell i taget	⊖, ⊕, ⊕ eller ⊖
En sida i taget	2nd och därefter ⊖, ⊕, ⊕ eller ⊖
Flyttar markören till rad 1 i aktuell kolumn respektive till den sista rad som innehåller data i någon kolumn på skärmen. Om markören är på eller efter den sista raden flyttar ◆ ⊖ markören till rad 999.	◆ ⊕ eller ◆ ⊖
Flyttar markören till kolumn 1 respektive till den sista kolumn som innehåller data. Om markören är i eller efter den sista kolumnen flyttar ◆ ⊕ markören till kolumn 99.	◆ ⊖ eller ◆ ⊕

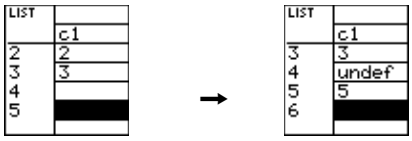
När du rullar uppåt eller nedåt förblir rubrikraden överst i fönstret för att kolumnnumren alltid ska synas. När du rullar till höger eller vänster förblir radnumren till vänster i fönstret för att de alltid ska synas.

Hur rader och kolumner fylls automatiskt

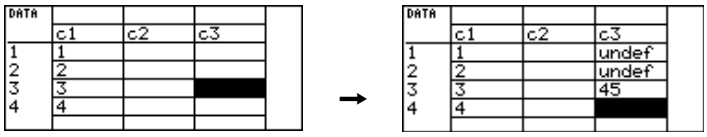
När du matar in ett värde i en cell flyttas markören till nästa cell. Du kan dock flytta markören till vilken cell som helst och mata in ett värde. Om du lämnar tomrum mellan celler hanterar räknaren tomrummen automatiskt.

- I en listvariabel är en cell i tomrummet *odefinierad* tills du matar in ett värde i cellen.

Obs! *Om du matar in mer än en kolumn av element i en listvariabel konverteras den automatiskt till en datavariabel.*



- I en datavariabel hanteras tomrum i en kolumn på samma sätt som i en lista. Om du lämnar ett tomrum mellan kolumner kommer den kolumnen dock att vara tom.



- När du i en matrisvariabel anger ett värde i en cell utanför de aktuella gränserna läggs ytterligare rader och/eller kolumner till automatiskt till matrisen för att inkludera den nya cellen. Andra celler i de nya raderna och/eller kolumnerna fylls med nollor.

Obs! *Även om du anger storleken på en matris när du skapar den, kan du enkelt lägga till ytterligare rader och/eller kolumner.*

Ändra cellbredden

Cellbredden påverkar antalet tecken som visas i varje cell. Så här ändrar du cellbredden i Data/Matrix Editor:

Tips! *Om du vill visa ett tal i full precision kan du alltid markera cellen och titta på inmatningsraden.*

1. Öppna dialogrutan FORMAT genom att trycka på [F1] 9
   — eller —
   TI-89: [◆] [|]
   TI-92 Plus: [◆] F

Cellbredden är det totala antalet tecken som kan visas i en cell.

Alla celler har samma bredd.

2. När den aktuella inställningen för Cell Width är markerad trycker du på ⊙ eller ⊙ för att visa en meny med siffror (3 till och med 12).
3. Flytta markören och markera ett tal och tryck på [ENTER]. (För tal med endast en siffra kan du skriva talet och trycka på [ENTER].)
4. Tryck på [ENTER] för att stänga dialogrutan.

Rensa en eller alla kolumner

Med den här proceduren rensar du innehållet i en kolumn. Själva kolumnen tas inte bort.

Obs! *För en list- eller datavariabel är en rensad kolumn tom. För en matris innehåller en rensad kolumn nollor.*

Om du vill rensa:	Gör du följande:
En kolumn	1. Flytta markören till en cell i kolumnen. 2. **TI-89:** [2nd] [F6] **TI-92 Plus:** [F6] och välj 5:Clear Column. (Detta alternativ är inte tillgängligt för matriser.)
Alla kolumner	Tryck på [F1] och välj 8:Clear Editor. När du blir uppmanad att bekräfta trycker du på [ENTER] (eller [ESC] för att avbryta).

Procedurerna för att infoga och ta bort en cell, rad eller kolumn är enkla. Du kan ha upp till 99 kolumner med upp till 999 element i varje kolumn.

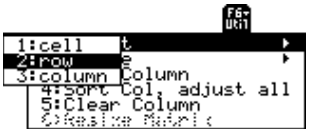
Kolumnrubriker och kolumnhuvuden

Du kan inte ta bort de rader eller celler som innehåller kolumnrubriker eller kolumnhuvuden. Du kan heller inte infoga en rad eller cell före en kolumnrubrik eller ett kolumnhuvud.

Infoga en rad eller kolumn

Den nya raden eller kolumnen infogas *före* raden eller kolumnen som innehåller den markerade cellen. Gör så här i Data/Matrix Editor:

1. Flytta markören till en cell på raden eller i kolumnen.
2. **TI-89:** [2nd] [F6]
   TI-92 Plus: [F6]
   och välj 1:Insert.

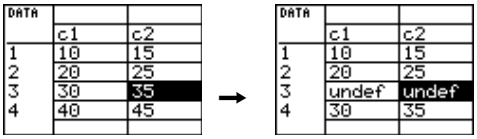


3. Välj 2:row eller 3:column.

Obs! *För en listvariabel är det samma sak att infoga en rad som att infoga en cell.*

När du infogar en rad:

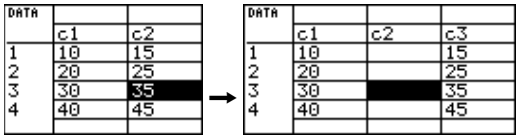
- I en list- eller datavariabel är raden *odefinierad*.
- I en matrisvariabel är raden fylld med nollor.



Obs! *För en listvariabel kan du inte infoga en kolumn, eftersom en lista bara har en kolumn.*

När du infogar en kolumn:

- I en datavariabel är kolumnen tom.
- I en matrisvariabel är kolumnen fylld av nollor.

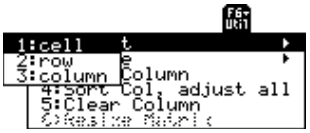


Du kan därefter mata in värden i de odefinierade eller tomma cellerna.

Infoga en cell

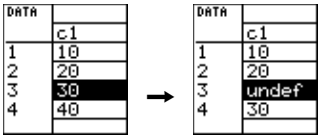
Den nya cellen infogas *före* den markerade cellen i samma kolumn. (Du kan inte infoga en cell i en låst kolumn som är definierad av en funktion i kolumnhuvudet. Se sidan 248.) Gör så här i Data/Matrix Editor:

1. Flytta markören till lämplig cell.
2. **TI-89:** [2nd] [F6]
   TI-92 Plus: [F6]
   och välj 1:Insert.
3. Välj 1:cell.



Obs! *För en matrisvariabel kan du inte infoga en cell eftersom matrisen måste behålla en rektangulär form.*

Den infogade cellen är odefinierad. Du kan därefter mata in ett värde i cellen.



Ta bort en rad eller kolumn

Gör så här i Data/Matrix Editor:

1. Flytta markören till en cell i den rad eller kolumn som du vill ta bort.
2. **TI-89:** [2nd] [F6]
   TI-92 Plus: [F6]
   och välj 2:Delete.
3. Välj 2:row eller 3:column.

Om du tar bort en rad flyttas de rader som finns nedanför den borttagna raden upp. Om du tar bort en kolumn flyttas de kolumner som finns till höger om den borttagna kolumnen till vänster.

Ta bort en cell

Gör så här i Data/Matrix Editor:

1. Flytta markören till cellen som du vill ta bort. (Du kan inte ta bort en cell i en låst kolumn, som är definierad av en funktion i kolumnhuvudet. Se sidan 248.)

Obs! *För en matrisvariabel kan du inte infoga en cell eftersom matrisen måste behålla en rektangulär form.*

2. **TI-89:** [2nd] [F6]
   TI-92 Plus: [F6]
   och välj 2:Delete.
3. Välj 1:cell.

Eventuella celler nedanför den borttagna cellen flyttas upp.

Om du vill lägga till en ny "sista" rad, kolumn eller cell

Du behöver *inte* använda menyn Util för att:

- Lägga till en ny rad eller cell längst ned i en kolumn.

  – eller –
- Lägga till en ny kolumn till höger om den sista kolumnen.

Flytta markören till cellen och mata in ett värde.

Definiera ett kolumnhuvud med ett uttryck

För en listvariabel eller en kolumn i en datavariabel kan du mata in en funktion i kolumnhuvudet som automatiskt genererar en lista med element. I en datavariabel kan du även definiera en kolumn uttryckt i en annan kolumn.

Mata in en huvud-definition

Gör så här i Data/Matrix Editor:

1. Flytta markören till en cell i kolumnen och tryck på [F4].
   – eller –
   Flytta markören till huvudcellen (c1, c2 osv) och tryck på [ENTER].

   Obs! Du behöver inte trycka på [ENTER] om du vill skriva en ny definition eller ersätta den befintliga. Om du däremot vill redigera den befintliga definitionen måste du trycka på [ENTER].

2. Skriv det nya uttrycket, som ersätter en befintlig definition.

   Om du använde [F4] eller [ENTER] i steg 1 flyttade du markören till inmatningsraden och markerade den befintliga definitionen. Du kan även göra följande:

   - Tryck på [CLEAR] för att rensa det markerade uttrycket. Skriv därefter det nya uttrycket.
     – eller –
   - Tryck på ◀ eller ▶ för att ta bort markeringen. Redigera därefter det gamla uttrycket.

Tips! Om du vill visa en befintlig definition trycker du på [F4] eller flyttar markören till huvudcellen och tittar på inmatningsraden.

Tips! Du kan avbryta alla ändringar genom att trycka på [ESC] innan du trycker på [ENTER].

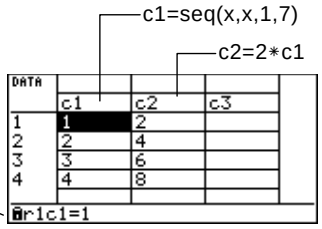
Du kan använda ett uttryck som:	Exempel:
Genererar en följd av tal.	`c1=seq(x^2,x,1,5)` `c1={1,2,3,4,5}`
Refererar till en annan kolumn.	`c2=2*c1` `c4=c1*c2-sin(c3)`

Obs! Funktionen seq *beskrivs i appendix A.*

Obs! Om du refererar till en tom kolumn visas ett felmeddelande (såvida inte Auto-calculate = OFF, enligt beskrivningen på sidan 249).

3. Tryck på [ENTER], ▼ eller ▲ för att spara definitionen och uppdatera kolumnerna.

   Du kan inte ändra i en låst cell (🔒) eftersom den definieras av kolumnhuvudet.



Obs! För en datavariabel sparas huvuddefinitioner när du stänger Data/ Matrix Editor. För en listvariabel sparas inte definitionerna (endast de resulterande cellvärdena).

Rensa en huvud-definition

1. Flytta markören till en cell i kolumnen och tryck på [F4].
   – eller –
   Flytta markören till kolumnhuvudet (c1, c2 osv.) och tryck på [ENTER].
2. Tryck på [CLEAR] för att ta bort det markerade uttrycket.
3. Tryck på [ENTER], ▼ eller ▲.

Använda en befintlig lista som en kolumn

Anta att du har en eller fler befintliga listor och du vill använda dessa som kolumner i en datavariabel.

__Obs!__ Om du har programmet CBL™ eller CBR™ (tillval) kan du använda dessa tekniker för de insamlade listorna.

__Tips!__ Använd [2nd] [VAR-LINK] för att visa befintliga listvariabler.

Från:	Gör du följande:
Data/Matrix Editor	Använd [F4] i lämplig kolumn för att definiera kolumnhuvudet. Referera till den befintliga listvariabeln. Exempel: c1=list1
Grundfönstret eller ett program	Använd kommandot **NewData** enligt beskrivningen i appendix A. Exempel: **NewData** *datavar*, *lista1* [, *lista2*] [, *lista3*] ... └ Befintliga listvariabler att kopiera till kolumner i datavariabeln. └ Datavariabeln. Om variabeln redan finns, kommer den att omdefinieras utifrån de nya listorna.

Fylla en matris med en lista

Du kan inte använda Data/Matrix Editor för att fylla en matris med en lista. Du kan däremot använda kommandot **list▸mat** från grundfönstret eller ett program. Information finns i appendix A.

Funktionen Auto-calculate

När det gäller list- och datavariabler finns det i Data/Matrix Editor en funktion som heter Auto-calculate. Som standard är Auto-calculate = ON. Om du därför gör en ändring som påverkar definitionen av ett kolumnhuvud (eller en kolumn som det refereras till i ett kolumnhuvud), räknas alla kolumnhuvuddefinitioner om automatiskt. Exempel:

- Om du ändrar definitionen för ett kolumnhuvud tillämpas den nya definitionen automatiskt.
- Om kolumnhuvudet i kolumn 2 är definierat som c2=2∗c1 kommer de ändringar du gör i kolumn 1 att visas i kolumn 2 automatiskt.

__Tips!__ Du kanske vill ställa in Auto-calculate = OFF för att:

- *Göra flera ändringar utan att värdena räknas om varje gång.*
- *Mata in en definition, exempelvis c1=c2+c3, innan du matar in kolumnerna 2 och 3.*
- *Åsidosätta fel i en definition tills du åtgärdar felen.*

Så här aktiverar och avaktiverar du Auto-calculate från Data/Matrix Editor:

1. Tryck [F1] 9
   —eller—
   TI-89: [◆] [I]
   TI-92 Plus: [◆] F
2. Ändra Auto-Calculate till OFF eller ON.
3. Tryck på [ENTER] för att stänga dialogrutan.

Om Auto-calculate = OFF och du gör ändringar som de som beskrevs ovan, räknas inte definitionerna för kolumnhuvuden om förrän du ställer in Auto-calculate = ON.

När du definierar ett kolumnhuvud kan du använda funktionerna **shift** och **cumSum**, enligt beskrivningen nedan. Dessa beskrivningar skiljer sig något från appendix A. I detta kapitel beskrivs hur du använder funktionerna i Data/Matrix Editor.
I appendix A finns en mer allmän beskrivning för grundfönstret eller ett program.

Använda funktionen Shift

Med funktionen **shift** kopierar du en baskolumn och flyttar upp eller ned den ett angivet antal element. Använd F4 för att definiera kolumnhuvudet med syntaxen:

shift (*kolumn* [,*heltal*])

Antal element att skifta (positivt värde skiftar uppåt, negativa värden skiftar nedåt). Standardvärdet är ⁻1.

Kolumn som används som baskolumn för skiftet.

Exempel: Flytta upp och ned två element:

c2=shift(c1,2)

c3=shift(c1,⁻2)

c1	c2	c3
1	3	undef
2	4	undef
3	undef	1
4	undef	2

Skiftade kolumner har samma längd som baskolumnen (c1).

De sista två värdena av c1 skiftas ut nedtill och odefinierade element skiftas in ovanifrån.

De första två elementen av c1 skiftas ut upptill och odefinierade element skiftas in underifrån.

Obs! *Du anger "shift" genom att skriva det med tangentbordet eller genom att välja det från CATALOG.*

Använda funktionen CumSum

Funktionen **cumSum** returnerar den kumulerade summan av elementen i en baskolumn. Använd F4 för att definiera ett kolumnhuvud med syntaxen:

cumSum (*kolumn*)

Kolumn som används som bas för den kumulerade summan

Exempel:

c2=cumSum(c1)

c1	c2
1	1
2	3
3	6
4	10

1+2

1+2+3+4

Obs! *Du anger "cumSum" genom att skriva det, välja det på CATALOG eller genom att trycka på* 2nd [MATH] *och välja funktionen på undermenyn List.*

Sortera kolumner

> När du har matat in information i en data-, list- eller matrisvariabel kan du enkelt sortera en angiven kolumn i numerisk eller alfabetisk ordning. Du kan också sortera alla kolumner som en enhet, baserat på "nyckelkolumnen".

Sortera en enskild kolumn

I Data/Matrix Editor:

1. Flytta markören till en cell i kolumnen.
2. **TI-89:** [2nd] [F6]
   TI-92 Plus: [F6]
   och välj 3:Sort Column.

Numren sorteras i stigande ordning.

Teckensträngar sorteras i alfabetisk ordning.

c1	→	c1
fred		75
sally	→	82
chris	→	98
jane		chris
75	→	fred
98		jane
82		sally

Sortera alla kolumner baserat på en "nyckelkolumn"

Anta att du har en databasstruktur i vilken varje kolumn längs samma rad innehåller relaterad information (som exempelvis förnamn, efternamn och poäng för en student). Om du i det fallet sorterar endast en enskild kolumn, förstörs sambandet mellan kolumnerna.

I Data/Matrix Editor:

__Obs!__ För en listvariabel är detta det samma som att sortera en enskild kolumn.

1. Flytta markören till en cell i "nyckelkolumnen".

   I det här exemplet flyttar du markören till den andra kolumnen (c2) för att sortera efter efternamn.

c1	c2	c3
fred	stone	95
sally	ross	75
jane	smith	97
nick	castle	93

__Obs!__ Detta menyalternativ är inte tillgängligt om en kolumn är låst.

2. **TI-89:** [2nd] [F6]
   TI-92 Plus: [F6]
   och välj 4:Sort Col, adjust all.

c1	c2	c3
nick	castle	93
sally	ross	75
jane	smith	97
fred	stone	95

När du använder denna procedur för en datavariabel gäller följande:

- Alla kolumner måste ha samma längd.
- Ingen av kolumnerna kan låsas (definieras av en funktion i kolumnhuvudet). När markören befinner sig i en låst kolumn visas symbolen 🔒 i början av inmatningsraden.

Spara en kopia av en list-, data- eller matrisvariabel

Du kan spara en kopia av en list-, data- eller matrisvariabel. Du kan också kopiera en lista till en datavariabel eller välja en kolumn från en datavariabel och kopiera den kolumnen till en lista.

Giltiga kopieringstyper

Du kan kopiera:	Till en:
En lista	Lista eller datavariabel
Datavariabel	Datavariabel
En datakolumn	Lista
En matris	Matris

Obs! *En lista konverteras automatiskt till en datavariabel om du matar in mer än en kolumn med information.*

Procedur

Gör följande från Data/Matrix Editor:

1. Visa variabeln som du vill kopiera.
2. Tryck på [F1] och välj 2:Save Copy As.
3. Gör följande i dialogrutan:
   - Välj Type och Folder för kopian.
   - Skriv ett variabelnamn för kopian.
   - Välj kolumnen som du ska kopiera ifrån.

Kolumnen är nedtonad om du inte kopierar en datakolumn till en lista. Kolumninformation används inte för andra typer av kopior.

Obs! *Om du skriver namnet på en befintlig variabel ersätts dess innehåll.*

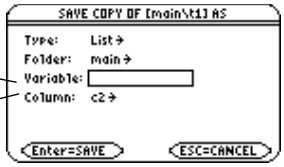
4. Tryck på [ENTER] (efter att ha skrivit i en inmatningsruta, exempelvis Variable, måste du trycka på [ENTER] två gånger).

Kopiera en datakolumn till en lista

En datavariabel kan ha flera kolumner, men en listvariabel kan endast ha en kolumn. När du därför kopierar från en datavariabel till en lista måste du välja vilken kolumn du vill kopiera.

Listvariabel att kopiera till.

Datakolumn som kommer att kopieras till listan. Som standard visas kolumnen som innehåller markören.



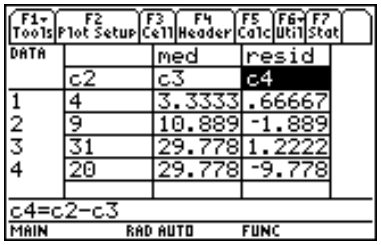
Plotta statistiska data

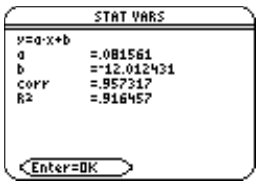
16

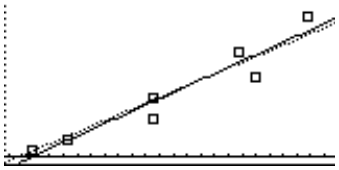
Förhandsvisning över plottning av statistiska data 254
Översikt över stegen i statistisk analys 259
Utföra en statistisk beräkning 260
Statistiska beräkningstyper 262
Statistiska variabler 264
Definiera plottning av statistiska data 265
Statistiska plottningstyper 267
Använda Y= Editor med statistiska plottningar 269
Plotta och spåra en definierad statistisk plottning 270
Använda frekvenser och kategorier 271
Om du har tillvalsprogrammet CBL och CBR 273

Data/Matrix Editor har två huvuduppgifter.

- I kapitel 15 beskrevs hur du med hjälp av Data/Matrix Editor kan skapa och underhålla en lista, matris eller datavariabel.
- I det här kapitlet beskrivs hur du använder Data/Matrix Editor för att utföra statistiska beräkningar och plotta statistiska data.

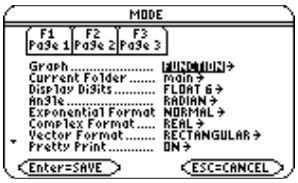
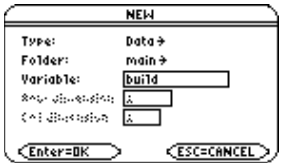
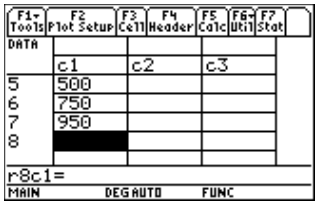
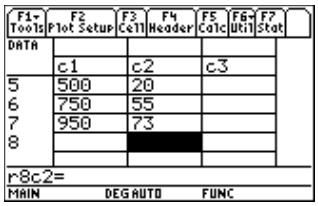


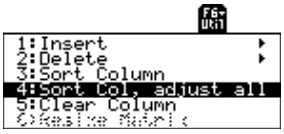
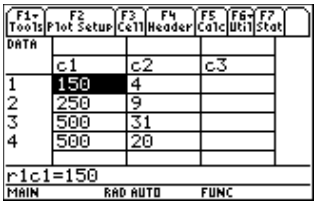
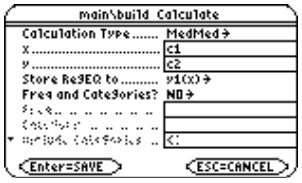
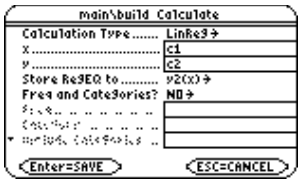
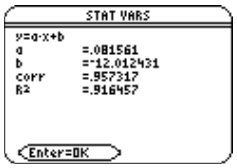
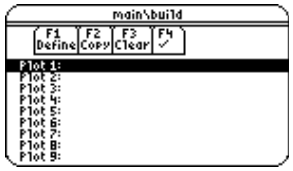


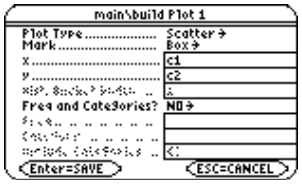
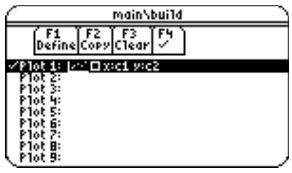
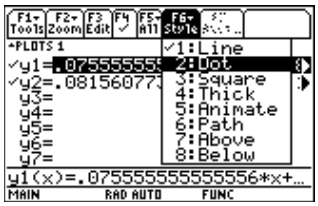
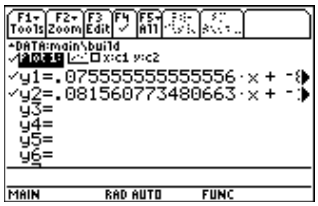
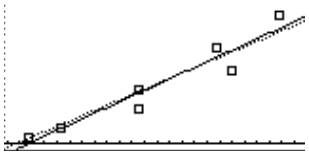


Förhandsvisning över plottning av statistiska data

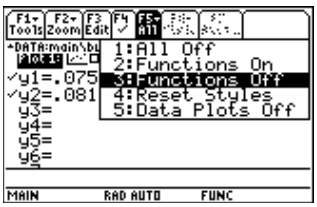
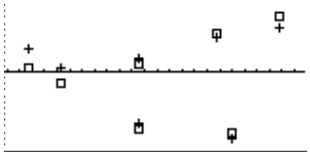
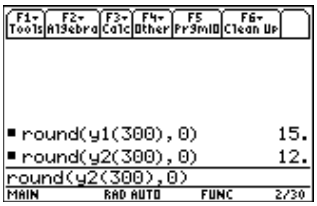
Baserat på ett exempel med sju städer matar du in data som relaterar befolkning till antalet byggnader med mer än 12 våningar. Med hjälp av median-median-linjen och linjär regression, kan du hitta och plotta ekvationer som passar informationen. För varje regressionsekvation uppskattar du hur många byggnader med fler än 12 våningar som du kan förväntas finna i en stad med 300 000 invånare.

Steg	TI-89 Tangenter	TI-92 Plus Tangenter	Teckenfönster
1. Visa dialogrutan MODE. För Graph mode, välj FUNCTION.	[MODE] [▶] 1 [ENTER]	[MODE] [▶] 1 [ENTER]	
2. Visa Data/Matrix Editor och skapa en ny datavariabel med namnet BUILD.	[APPS] 6 3 [▼] [▼] B U I L D [ENTER] [ENTER]	[APPS] 6 3 [▼] [▼] B U I L D [ENTER] [ENTER]	
3. Använd exempelinformationen nedan och mata in befolkningen i kolumn 1. Bef. (i tusental) / Byggn > 12 vån 150 / 4 500 / 31 800 / 42 250 / 9 500 / 20 750 / 55 950 / 73	1 5 0 [ENTER] 5 0 0 [ENTER] 8 0 0 [ENTER] 2 5 0 [ENTER] 5 0 0 [ENTER] 7 5 0 [ENTER] 9 5 0 [ENTER]	1 5 0 [ENTER] 5 0 0 [ENTER] 8 0 0 [ENTER] 2 5 0 [ENTER] 5 0 0 [ENTER] 7 5 0 [ENTER] 9 5 0 [ENTER]	
4. Flytta markören till rad 1 i kolumn 2 (r1c2). Mata sedan in motsvarande antal byggnader. *När du trycker på [◆] [▲] flyttas markören uppåt. När du har matat in data i en cell kan du trycka på [ENTER] eller [▼] för att mata in värdet och flytta markören nedåt en cell. Tryck på [▲] för att mata in värdet och flytta markören uppåt en cell.*	[▶] [◆] [▲] 4 [ENTER] 3 1 [ENTER] 4 2 [ENTER] 9 [ENTER] 2 0 [ENTER] 5 5 [ENTER] 7 3 [ENTER]	[▶] [2nd] [▲] 4 [ENTER] 3 1 [ENTER] 4 2 [ENTER] 9 [ENTER] 2 0 [ENTER] 5 5 [ENTER] 7 3 [ENTER]	

Steg	TI-89 Tangenter	TI-92 Plus Tangenter	Teckenfönster
5. Flytta markören till rad 1 i kolumn 1 (r1c1). Sortera informationen i stigande ordning efter befolkning. *Detta sorterar kolumn 1 och justerar sedan om alla andra kolumner så att de behåller samma ordning som kolumn 1. Detta är av största vikt för att behålla förhållandet mellan kolumner med data.* *Om du vill sortera kolumn 1 kan markören vara vart som helst i kolumn 1. I detta exempel trycker du på* ***TI-89:*** *◆ ⊖* ***TI-92 Plus:*** *2nd ⊖ så att du kan se de fyra första raderna.*	◀ ◆ ⊖ 2nd [F6] 4	◀ 2nd ⊖ [F6] 4	 
6. Visa dialogrutan Calculate. Ställ in Calculation Type = MedMed x = C1 y = C2 Store RegEQ to = y1(x)	[F5] ▶ 7 ⊖ C [alpha] 1 ⊖ [alpha] C 2 ⊖ ▶ ⊖ [ENTER]	[F5] ▶ 7 ⊖ C 1 ⊖ C 2 ⊖ ▶ ⊖ [ENTER]	
7. Utför beräkningen för att visa ekvationen för median-median-linjen. *Som visas i dialogrutan* Calculate, *sparas ekvationen i y1(x).*	[ENTER]	[ENTER]	
8. Stäng fönstret STAT VARS. Data/Matrix Editor visar.	[ENTER]	[ENTER]	
9. Visa dialogrutan Calculate. Ställ in: Calculation Type = LinReg x = C1 y = C2 Store RegEQ to = y2(x)	[F5] ▶ 5 ⊖ ⊖ ⊖ ▶ ⊖ [ENTER]	[F5] ▶ 5 ⊖ ⊖ ⊖ ▶ ⊖ [ENTER]	
10. Utför beräkningen för att visa den linjära regressionsekvationen. *Ekvationen sparas i y2(x).*	[ENTER]	[ENTER]	
11. Stäng fönstret STAT VARS. Data/Matrix Editor visas.	[ENTER]	[ENTER]	
12. Visa fönstret Plot Setup. *Plot 1 markeras som standard.* *Med* [F3] *kan du radera markerade inställningar för plottningen.*	[F2]	[F2]	

Steg	TI-89 Tangenter	TI-92 Plus Tangenter	Teckenfönster
13. Definiera Plot 1 som: Plot Type = Scatter Mark = Box x = C1 y = C2 *Lägg märke till likheterna mellan denna och dialogrutan* Calculate.	F1 ▶ 1 ▼ ▶ 1 ▼ C alpha 1 ▼ alpha C 2	F1 ▶ 1 ▼ ▶ 1 ▼ C 1 ▼ C 2	
14. Spara plottningsdefinitionen och återgå till fönstret Plot Setup. *Lägg märke till hur definitionen för Plot 1 ser ut.*	ENTER ENTER	ENTER ENTER	
15. Visa Y= Editor. Ställ in visningsformat till Dot för y1(x), ekvationen för median-median-linjen. ***Obs!*** *Beroende på det föregående innehållet i Y= Editor kanske du måste flytta markören till y1.* *När PLOTS 1 är längst upp i fönstret betyder det att Plot 1 är markerad.* *Observera att y1(x) och y2(x) markerades när regressionsekvationerna sparades.*	◆ [Y=] 2nd [F6] 2	◆ [Y=] F6 2	
16. Rulla uppåt för att markera Plot 1. *Definitionen för Plot 1 är samma som den som visas i fönstret Plot Setup.*	▲	▲	
17. Använd ZoomData för att plotta Plot 1 och regressions-ekvationerna y1(x) och y2(x). *ZoomData undersöker data för alla markerade statistiska plottningar och justerar visningsfönstret för att inkludera alla punkter.*	F2 9	F2 9	
18. Återgå till den aktuella sessionen i Data/Matrix Editor.	APPS 6 1	APPS 6 1	

Steg	TI-89 Tangenter	TI-92 Plus Tangenter	Teckenfönster
19. Mata in en rubrik för kolumn 3. Definiera huvudet för kolumn 3 som de beräknade värdena av median-median-linjen. *För att kunna mata in en rubrik måste markören markera rubrikcellen längst upp i kolumnen.* *Med knappen [F4] kan du definiera ett kolumnhuvud från vilket ställe som helst i en kolumn. När markören befinner sig på en kolumnhuvudcell behöver du inte trycka på [F4].*	▶ ▶ ▲ ▲ [2nd] [a-lock] M E D [alpha] [ENTER] [F4] Y 1 [(] [alpha] C 1 [)] [ENTER]	▶ ▶ ▲ ▲ M E D [ENTER] [F4] Y 1 [(] C 1 [)] [ENTER]	
20. Mata in en rubrik för kolumn 4. Definiera kolumnhuvudet för kolumn 4 som avvikelsen (skillnaden mellan observerat och beräknat värde) för median-median-linjen.	▶ ▲ [2nd] [a-lock] R E S I D [alpha] [ENTER] [alpha] C 2 [−] [alpha] C 3 [ENTER]	▶ ▲ R E S I D [ENTER] [F4] C 2 [−] C 3 [ENTER]	F1 Tools F2 Plot Setup F3 Cell F4 Header F5 Calc F6 Util F7 Stat DATA c2 c3 c4 med resid 1 4 3.3333 .66667 2 9 10.889 -1.889 3 31 29.778 1.2222 4 20 29.778 -9.778 c4=c2-c3 MAIN RAD AUTO FUNC
21. Mata in en rubrik för kolumn 5. Definiera kolumnhuvudet för kolumn 5 som beräknat värde för LinReg-linjen.	▶ ▲ ▲ [2nd] [a-lock] L I N [alpha] [ENTER] [F4] Y 2 [(] [alpha] C 1 [)] [ENTER]	▶ ▲ L I N [ENTER] [F4] Y 2 [(] C 1 [)] [ENTER]	
22. Mata in en rubrik för kolumn 6. Definiera kolumnhuvudet för kolumn 6 som avvikelsen för LinReg.	▶ ▲ [2nd] [a-lock] R E S I D [alpha] [ENTER] [F4] [alpha] C 2 [−] [alpha] C 5 [ENTER]	▶ ▲ R E S I D [ENTER] [F4] C 2 [−] C 5 [ENTER]	F1 Tools F2 Plot Setup F3 Cell F4 Header F5 Calc F6 Util F7 Stat DATA c4 c5 c6 resid lin resid 1 .66667 .22169 3.7783 2 -1.889 8.3778 .62224 3 1.2222 28.768 2.232 4 -9.778 28.768 -8.768 c6=c2-c5 MAIN RAD AUTO FUNC
23. Visa fönstret Plot Setup och avmarkera Plot 1.	[F2] [F4]	[F2] [F4]	
24. Markera Plot 2 och definiera inställningen som: Plot Type = Scatter Mark = Box x = C1 y = C4 (avvikelsen från median-median-linjen)	▼ [F1] ▼ ▼ C [alpha] 1 ▼ [alpha] C 4 [ENTER] [ENTER]	▼ [F1] ▼ ▼ C 1 ▼ C 4 [ENTER] [ENTER]	main\build Plot 2 Plot Type ..... Scatter → Mark ..... Box → x ..... c1 y ..... c4 Freq and Categories? NO → Enter=SAVE ESC=CANCEL
25. Markera Plot 3 och definiera inställningen som: Plot Type = Scatter Mark = Plus x = C1 y = C6 (avvikelsen från LinReg-kurvan)	▼ [F1] ▼ ▶ 3 ▼ C [alpha] 1 ▼ [alpha] C 6 [ENTER] [ENTER]	▼ [F1] ▼ ▶ 3 ▼ C 1 ▼ C 6 [ENTER] [ENTER]	main\build Plot 3 Plot Type ..... Scatter → Mark ..... Plus → x ..... c1 y ..... c6 Freq and Categories? NO → Enter=SAVE ESC=CANCEL

Steg	TI-89 Tangenter	TI-92 Plus Tangenter	Teckenfönster
26. Visa Y= Editor och stäng av alla funktioner. *Tryck på* F5 *och välj* 3:Functions Off, *inte* 1:All Off. *Plots 2 och 3 är fortfarande markerade.*	◆ [Y=] F5 3	◆ [Y=] F5 3	
27. Använd ZoomData för att visa avvikelserna som en graf. □ *markerar avvikelser från median-median-linjen,* + *markerar avvikelser från LinReg-kurvan.*	F2 9	F2 9	
28. Visa grundfönstret.	HOME	◆ [HOME]	
29. Använd regressionsekvationerna median-median-linjen (y1(x)) och LinReg-kurvan (y2(x)) för att beräkna värdet då x = 300 (300 000 invånare). *Med funktionen* **round** (2nd [MATH] 13) *garanterar du att resultaten visas som ett heltal (antal byggnader).* *När du har beräknat det första resultatet redigerar du inmatningsraden för att ändra y1 till y2.*	2nd [MATH] 1 3 Y 1 (3 0 0) , 0) ENTER ▷ ◁ ◁ ◁ ◁ ◁ ◁ ◁ ◁ ← 2 ENTER	2nd [MATH] 1 3 Y 1 (3 0 0) , 0) ENTER ▷ ◁ ◁ ◁ ◁ ◁ ◁ ◁ ◁ ← 2 ENTER	

Översikt över stegen i statistisk analys

I det här avsnittet visas en översikt över de steg du följer för att utföra en statistisk beräkning eller plotta statistiska data. Detaljerad beskrivning finns på sidorna som följer.

Beräkna och plotta statistiska data

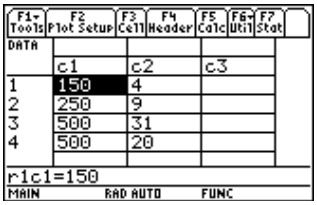
Obs! *I kapitel 15 finns information om hur du matar in data i Data/Matrix Editor.*

Tips! *Du kan också använda Y= Editor för att definiera och markera statistiska plottningar och y(x)-funktioner.*

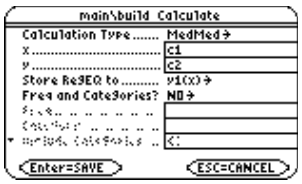
Tips! *Använd kommandot* ZoomData *för att optimera visningsfönstret för statistiska plottningar.* [F2] Zoom *är tillgängligt i Y= Editor, Window Editor och i graffönstret.*

Ställ in Graph ([MODE]) till FUNCTION.

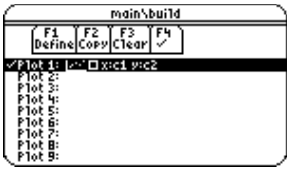
Mata in statistiska data i Data/Matrix Editor ([APPS] 6).



Utför statistiska beräkningar för att hitta statistiska variabler eller för att passa in data i en modell ([F5]).



Definiera och markera statistiska plottningar ([F2] och sedan [F1]).



Definiera visningsfönstret ([◆] [WINDOW]).

Ändra grafformat om nödvändigt.

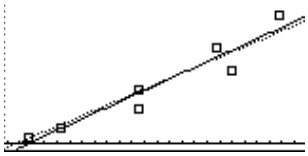
[F1] 9

— eller —

TI-89: [◆] [I]

TI-92 Plus: [◆] F

Visa de markerade statistiska plottningarna som diagram och plotta funktionerna ([◆] [GRAPH]).



Utforska plottade statistiska data

Du kan göra följande från graffönstret:

- Visa koordinaterna för en bildpunkt genom att använda den rörliga markören, eller för en plottad punkt genom att spåra plottningen.
- Använda menyn [F2] Zoom för att zooma in eller ut på en del av grafen.
- Använda menyn [F5] Math för att analysera en funktion (men inte en statistisk plottning) som kan visas som en graf.

Du kan använda menyn F5 **Calc** från Data/Matrix Editor för att utföra statistiska beräkningar. Du kan analysera statistik med en eller två variabler eller utföra flera typer av regressionanalyser.

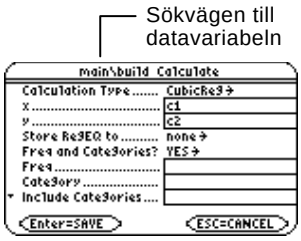
Dialogrutan Calculate

Du måste ha öppnat en datavariabel. Du kan inte utföra statistiska beräkningar med Data/Matrix Editor med en list- eller matrisvariabel.

Gör följande från Data/Matrix Editor:

1. Tryck på F5 för att visa dialogrutan Calculate.

   I detta exempel är alla poster aktiva. På räknaren är poster endast aktiva om de är giltiga för de aktuella inställningarna för Calculation Type och Use Freq and Categories.



__Obs!__ Om en post inte är giltig för de aktuella inställningarna visas posten nedtonad. Du kan inte flytta markören till en nedtonad post.

2. Ange lämpliga inställningar för de aktiva posterna.

Post	Beskrivning
Calculation Type	Välj typ av beräkning. Beskrivning finns på sidan 262.
x	Skriv kolumnnumret i Data/Matrix Editor (C1, C2 osv.) som används för x-värden, den oberoende variabeln.
y	Skriv kolumnnumret som används för y-värden, den beroende variabeln. Detta krävs för alla beräkningstyper, förutom OneVar.
Store RegEQ to	Om Calculation Type är en regressionsanalys kan du välja ett funktionsnamn (y1(x), y2(x) osv.). På detta sätt kan du spara regressions-ekvationen så att den visas i Y= Editor.
Use Freq and Categories?	Välj NO eller YES. Observera att Freq, Category och Include Categories endast är aktiva när Use Freq and Categories? = YES.

__Tips!__ Om du vill använda en befintlig listvariabel för x, y, Freq eller Category skriver du listnamnet i stället för ett kolumnnummer.

Dialogrutan Calculate
(fortsättning)

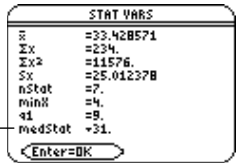
Post	Beskrivning
Freq	Skriv kolumnnumret som innehåller ett viktat värde för varje datapunkt. Om du inte anger ett kolumnnummer antas alla datapunkter ha samma vikt (1).
Category	Skriv kolumnnumret som innehåller ett kategorivärde för varje datapunkt.
Include Categories	Om du anger en Category-kolumn kan du använda denna post för att begränsa beräkningen till angivna kategorivärden. Om du exempelvis anger {1,4} används i beräkningen bara datapunkter med ett kategorivärde på 1 eller 4.

Obs! *Exempel på hur Freq, Category och Include Categories används finns på sidan 271.*

3. Tryck på [ENTER] (efter att ha skrivit i en inmatningsruta trycker du på [ENTER] två gånger).

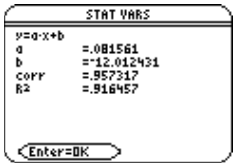
   Resultaten visas i fönstret STAT VARS. Formatet beror på Calculation Type. Exempel:

För Calculation Type = OneVar



När ▼ visas i stället för =, kan du rulla fönstret för att se fler resultat.

För Calculation Type = LinReg



Obs! *Odefinierade datapunkter (visas som* undef*) ignoreras i en statistisk beräkning.*

4. Du stänger fönstret STAT VARS genom att trycka på [ENTER].

Visa fönstret
STAT VARS igen

På menyn Stat i Data/Matrix Editor visas återigen föregående beräkningsresultat (tills du tar bort dem från minnet).
TI-89: [2nd] [F7]
TI-92 Plus: [F7]

Föregående resultat tas bort när du:

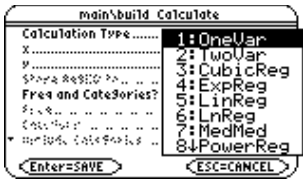
- Redigerar datapunkterna eller ändrar Calculation Type.
- Öppnar en annan datavariabel eller öppnar samma datavariabel igen (om beräkningen refererade till en kolumn i en datavariabel). Resultat tas också bort om du stänger och öppnar Data/Matrix Editor igen med en datavariabel.
- Ändrar den aktuella mappen (om beräkningen refererade till en listvariabel i föregående mapp).

Så som beskrevs i föregående avsnitt kan du i dialogrutan **Calculate** ange vilken statistisk beräkning du vill utföra. Det här avsnittet innehåller mer information om beräkningstyper.

Välja beräkningstyp

Från dialogrutan Calculate (F5) markerar du den aktuella inställningen för Calculation Type genom att först trycka på ⊙.

Du kan därefter välja från en meny av tillgängliga typer.



Om ett alternativ är nedtonat, är det inte tillgängligt för den aktuella beräkningstypen.

Beräkningstyp	Beskrivning
OneVar	Statistik med en variabel. Beräknar de statistiska variabler som beskrivs på sidan 264.
TwoVar	Statistik med två variabler. Beräknar de statistiska variabler som beskrivs på sidan 264.
CubicReg	Kubisk regression. Anpassar data till tredjegrads-polynomet $y=ax^3+bx^2+cx+d$. Du måste ha minst fyra datapunkter. • För fyra punkter är ekvationen en polynom anpassning. • För fem eller fler punkter är det en polynom regression.
ExpReg	Exponentiell regression. Anpassar data till modellen $y=ab^x$ (där a är skärningspunkten med y-axeln) med hjälp av minsta kvadratmetoden samt x och ln(y).
LinReg	Linjär regression. Anpassar data till modellen y=ax+b (där a är lutningen och b är skärningen med y-axeln) med hjälp av minsta kvadratmetoden samt x och y.
LnReg	Logaritmisk regression. Anpassar data till modellen y=a+b ln(x) med hjälp av minsta kvadratmetoden samt ln(x) och y.
Logistic	Logistisk regression — Anpassar data till modellen y=a/(1+b∗*e*^(c∗x))+d och uppdaterar alla systemets statistikvariabler.

Obs! *För TwoVar och alla regressionsberäkningar, måste de kolumner som du anger för x och y (och eventuellt Freq eller Category) ha samma längd.*

Välja beräkningstyp (fortsättning)

Beräkningstyp	Beskrivning
MedMed	Median-median-linje. Anpassar data till ekvationen y=ax+b (där a är lutningen och b är skärningen med y-axeln) med hjälp av median-median-linjen. Summapunkterna medx1, medy1, medx2, medy2, medx3 och medy3 beräknas och sparas i variabler, men visas inte i fönstret STAT VARS.
PowerReg	Potensregression. Anpassar data till modellen $y=ax^b$ med hjälp av minsta kvadratmetoden samt ln(x) och ln(y).
QuadReg	Kvadratisk regression. Anpassar data till ett andragradspolynom $y=ax^2+bx+c$. Du måste ha minst tre datapunkter. • För tre punkter är ekvationen en polynom anpassning. • För fyra eller fler punkter är det en polynom regression.
QuartReg	Kvartär regression. Anpassar data till ett fjärdegrads-polynom $y=ax^4+bx^3+cx^2+dx+e$. Du ha måste ha minst fem datapunkter. • För fem punkter är ekvationen en polynom anpassning. • För sex eller fler punkter är det en polynom regression.
SinReg	Sinusodial regression — Anpassar data till sinusfunktionen y=asin(bx+c) och uppdaterar systemets alla statistikvariabler. Resultatet är alltid i radianer, oavsett inställningarna för vinkelmått.

Från grundfönstret eller ett program

Använd lämpligt kommando för den beräkning du vill utföra. Kommandona har samma namn som motsvarande Calculation Type. I appendix A finns information om varje kommando.

Viktigt! Dessa kommandon utför statistiska beräkningar men visar inte automatiskt resultaten. Använd kommandot **ShowStat** för att visa beräkningsresultaten.

Statistiska beräkningsresultat lagras i variabler. Du får tillgång till dessa variabler genom att skriva variabelnamnet eller använda VAR-LINK-fönstret enligt beskrivningen i kapitel 21. Alla statistiska variabler rensas när du redigerar data eller ändrar beräkningstyp. Andra villkor som rensar variablerna visas på sidan 261.

Beräknade variabler

Statistiska variabler lagras som systemvariabler. Variabeln regCoef behandlas som en listvariabel och variabeln regeq behandlas som en funktionsvariabel.

Skriv in tecknet Σ genom att trycka på:

TI-89: [◆] [(] [↑] [S]
TI-92 Plus: [2nd] G [↑] S

Skriv in tecknet σ genom att trycka på:

TI-89: [◆] [(] [alpha] [S]
TI-92 Plus: [2nd] G S

Tips! *För att skriva en exponent (t ex 2 i Σx^2), $\bar{x}$ eller $\bar{y}$ trycker du på* [2nd][CHAR] *och väljer alternativet på menyn* Math.

	En Var	Två Var	Regressioner
medelvärdet av x-värden	$\bar{x}$	$\bar{x}$	
summan av x-värden	Σx	Σx	
summan av x^2 -värden	Σx^2	Σx^2	
standardavvikelse för urvalet av x	Sx	Sx	
standardavvikelse för x-populationen †	σx	σx	
antal datapunkter	nStat	nStat	
medelvärdet av y-värden		$\bar{y}$	
summan av y-värden		Σy	
summan av y^2 -värden		Σy^2	
standardavvikelse för urvalet av y		Sy	
standardavvikelse för y-populationen †		σy	
summan av x∗ y-värden		Σxy	
minimum av x-värden	minX	minX	
maximum av x-värden	maxX	maxX	
minimum av y-värden		minY	
maximum av y-värden		maxY	
nedre kvartilen	q1		
median	medStat		
övre kvartilen	q3		
regressionsekvation			regeq
regressionskoefficienter (a, b, c, d, e)			regCoef
korrelationskoefficient ††			corr
kvadratsumman ††			R^2
summapunkter (endast median-median) †			medx1, medy1, medx2, medy2, medx3, medy3

Obs! *Den nedre kvartilen är medianen av punkterna mellan minX och medStat. Den övre kvartilen är medianen av punkterna mellan medStat och maxX.*

Tips! *Om regeq är 4x + 7, är regCoef {4 7}. För att komma åt "a"-koefficienten (första elementet i listan), använder du index regCoef[1].*

† De angivna variablerna beräknas men visas inte i STAT VARS-fönstret.

†† corr definieras endast för en linjär regression; R^2 definieras för alla polynomregressioner.

Definiera plottning av statistiska data

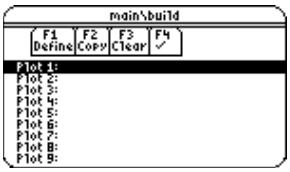
Från Data/Matrix Editor kan du använda inmatade data för att definiera flera typer av plottning av statistiska data. Du kan ange upp till nio plottningar på en och samma gång.

Procedur

Gör följande från Data/Matrix Editor:

1. Tryck på F2 för att visa Plot Setup-fönstret.

   Från början är ingen av plottningarna definierade.



2. Flytta markören för att markera vilket plottnings-nummer du vill definiera.

Sökvägen till datavariabeln

__Obs!__ Den här dialogrutan liknar dialogrutan Calculate.

3. Tryck på F1 för att definiera plottningen.

   I detta exempel är alla poster aktiva. På din räknare är posterna endast aktiva om de är giltiga för den aktuella inställningen för Plot Type och Use Freq and Categories?

__Obs!__ Om en post inte är giltig för de aktuella inställningarna är posten nedtonad. Du kan inte flytta markören till en nedtonad post.

4. Ange en inställning för de aktiva posterna.

Post	Beskrivning
Plot Type	Välj plottningstyp. Se sidan 267 för beskrivningar.
Mark	Ange vilken symbol som användes för att plotta datapunkterna: Box (□), Cross (x), Plus (+), Square (■) eller Dot (•).
x	Skriv vilket kolumnnummer i Data/Matrix Editor (C1, C2 osv.) som används för x-värden, den oberoende variabeln.
y	Skriv kolumnnumret som användes för y-värden, den beroende variabeln. Denna är endast aktiv för Plot Type = Scatter eller xyline.
Hist. Bucket Width	Anger bredden för varje stapel i ett histogram. Mer information finns på sidan 268.
Use Freq and Categories?	Välj NO eller YES. Observera att Freq, Category och Include Categories endast är aktiva när Use Freq and Categories? = YES. (Freq är endast aktiv för Plot Type = Box Plot eller Histogram.)

__Obs!__ Plottningar som har definierats med kolumn-nummer använder alltid den senaste datavariabeln i Data/Matrix Editor, även om den variabeln inte används för att skapa definitionen.

__Tips!__ Om du använda en befintlig listvariabel för x, y, Freq eller Category skriver du listnamnet i stället för kolumnnumret.

Obs! På sidan 271 finns ett exempel på hur du använder Freq, Category och Include Categories.

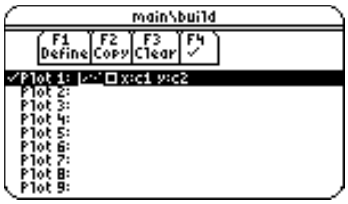
Post	Beskrivning
Freq	Skriv kolumnnumret som innehåller ett "viktat" värde för varje datapunkt. Om du inte anger ett kolumnnummer antas alla datapunkter ha samma vikt (1).
Category	Skriv kolumnnumret som innehåller ett kategorivärde för varje datapunkt.
Include Categories	Om du anger ett värde för Category kan du använda detta för att begränsa beräkningen till angivna kategorivärden. Om du exempelvis matar in {1,4} använder plottningen endast datapunkter med kategorivärdet 1 eller 4.

5. Tryck på [ENTER] (när du har skrivit i en inmatningsruta trycker du på [ENTER] två gånger).

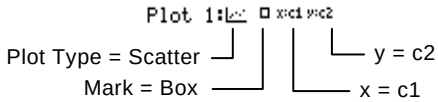
Obs! Odefinierade datapunkter (visas som undef*) ignoreras i en statistisk plottning.*

Plot Setup-fönstret visas igen.

Plottningen du just definierade markeras automatiskt för plottning.



Lägg märke till definitionen för plottningen.



Markera och avmarkera en plottning

Från Plot Setup-fönstret markerar du plottningen och trycker på [F4] för att växla mellan att markera och avmarkera den. Om en statistisk plottning är markerad förblir den markerad när du:

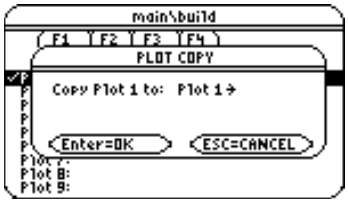
- Ändrar grafläget. (Statistiska plottningar plottas inte i 3D-läge.)
- Utför ett **Graph**-kommando.
- Öppnar en annan variabel i Data/Matrix Editor.

Kopiera en plottnings-definition

Gör följande i Plot Setup-fönstret:

1. Markera plottningen och välj [F2].
2. Tryck på ⊙ och välj det plottningsnummer som du vill kopiera till.
3. Tryck på [ENTER].

Obs! Om den ursprungliga plottningen var markerad (✓), är även kopian markerad.



Rensa en plottnings-definition

Från fönstret Plot Setup markerar du plottningen och trycker på [F3]. Om du vill omdefiniera en befintlig plottning behöver du inte nödvändigtvis rensa den först, du kan göra ändringar i den befintliga definitionen. För att förhindra en plottningsdefinition från att plottas kan du avmarkera den.

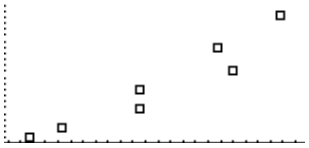
Statistiska plottningstyper

> När du definierar en plottning enligt beskrivningen i föregående avsnitt, kan du i fönstret **Plot Setup** välja plottningstyp. Det här avsnittet innehåller mer information om de tillgängliga plottningstyperna.

Spridningsdiagram, Punktdiagram

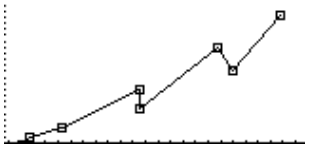
Datapunkter från x och y plottas som koordinatpar. Därför måste de kolumner eller listor som du anger för x och y ha samma längd.

- Plottade punkter visas med den symbol du angav som Mark.
- Om det behövs kan du ange samma kolumn eller lista för både x och y.



xy-linje

Detta är ett punktdiagram där datapunkter är plottade och kopplade i den ordningsföljd som de visas i x och y.

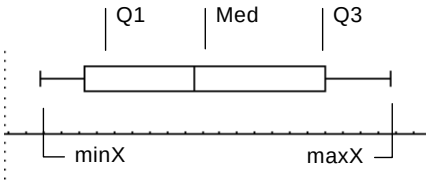


Du kan sortera alla kolumner i Data/Matrix Editor innan du plottar.
TI-89: [2nd] [F6] 3 eller [2nd] [F6] 4
TI-92 Plus: [F6] 3 eller [F6] 4

Lådagram

Detta plottar en variabel data i förhållande till minimum och maximum för datapunkterna (minX och maxX) i listan.

- En låda definieras av dess nedre kvartil (Q1), median (Med) och övre kvartil (Q3).
- Strecken går från minX till Q1 och från Q3 till maxX.



- När du väljer flera lådagram plottas de ovanpå varandra i samma ordningsföljd som deras plottningsnummer.
- Använd **NewPlot** för att visa statistiska data som ett modifierat lådagram.
- Välj Mod Box Plot för Plot Type när du definierar en plottning i Data/Matrix Editor.

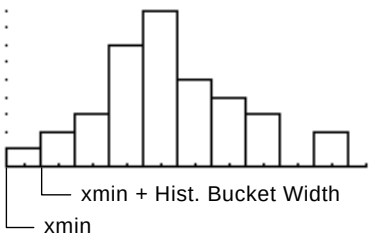
Ett modifierat lådagram inkluderar inte punkter utanför intervallet [Q1– X, Q3+X], där X är definierat som 1.5 (Q3– Q1). Dessa punkter, som kallas utliggare, plottas individuellt utanför grafen med det tecken du valt.

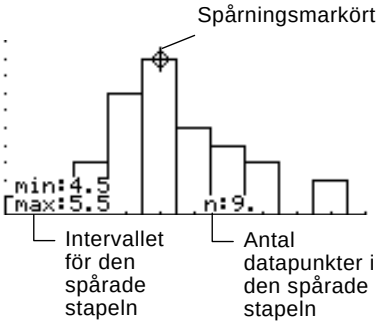
Histogram

Detta plottar data med en variabel som ett histogram. X-axeln delas upp i lika breda intervall som kallas staplar. Höjden på varje stapel (dess y-värde) anger hur många datapunkter som ingår i stapelns intervall.

$$\text{Antal staplar} = \frac{\text{xmax} - \text{xmin}}{\text{Hist. Bucket Width}}$$

- När du definierar diagrammet kan du ange Hist. Bucket Width (standardvärdet är 1) för att ställa in bredden på varje stapel.
- En datapunkt i kanten av en stapel räknas in i stapeln till höger.
- ZoomData (F2 9 från graffönstret, Y= Editor eller Window Editor) justerar xmin och xmax till att inkludera alla datapunkter, men justerar inte y-axeln.
  - Använd ◆ [WINDOW] för att ange ymin = 0 och ymax = antalet datapunkter som förväntas i den högsta stapeln.
- När du spårar (F3) ett stapeldiagram visas information i fönstret om den spårade stapeln.



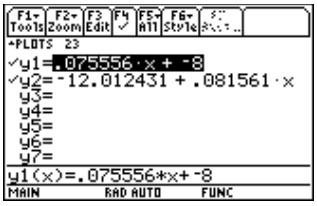


I föregående avsnitt beskrevs hur du definierar och väljer statistiska plottningar från Data/Matrix Editor. Du kan också definiera och välja statistiska plottningar från Y= Editor.

Visa listan med statistiska plottningar

Tryck på ◆ [Y=] för att visa Y= Editor. Från början befinner sig de nio statistiska plottningarna "ovanför" fönstret, dvs ovanför y(x)-funktionerna. PLOTS-indikatorn ger dock viss information.

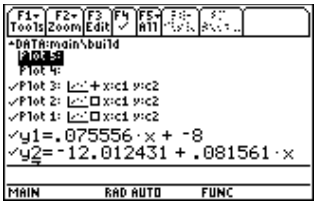
Exampel: PLOTS 23 betyder att Plots 2 och 3 är markerade.



Om du vill visa en lista över statistiska plottningar använder du ⊕ för att rulla ovanför y(x)-funktionerna.

Om en plottning är markerad, visas den datavariabel som kommer att användas till plottningen.

Om en plottning är definierad, visas samma definition som i fönstret Plot Setup.



Obs! *Plottningar som har definierats med kolumnnummer använder alltid den senaste datavariabeln i Data/Matrix Editor, även om den variabeln inte användes för att skapa definitionen.*

Från Y= Editor kan du utföra de flesta åtgärder på en statistisk plottning som på vilken övrig y(x)-funktion som helst.

Om du vill:	Gör du följande:
Redigera en plottningsdefinition	Markera plottningen och tryck på [F3]. Du ser samma definitionsfönster som visas i Data/Matrix Editor.
Markera eller avmarkera en plottning	Markera plottningen och tryck på [F4].
Stänga av alla plottningar och/eller funktioner	Tryck på [F5] och välj en post. Du kan också använda denna meny för att aktivera alla funktioner.

Obs! *Du kan inte använda* ***TI-89:*** [2nd] [F6] ***TI-92 Plus:*** [F6] *till att ställa in visningsstil för en graf. Med plottningsdefinitionen kan du däremot välja vilken symbol som ska användas vid plottningen.*

Plotta statistiska data och Y=-funktioner

Om det behövs kan du markera och plotta statistiska data och y(x)-funktioner på en och samma gång. I exemplet i början av detta kapitel plottas datapunkter och deras regressionsekvationer.

> När du har matat in datapunkterna och definierat statistiska plottningar kan du plotta de markerade punkterna genom att använda samma metoder som du använde för att plotta en funktion från Y= Editor (beskrivs i kapitel 6).

Definiera visnings-fönstret

Statistiska plottningar visas i den aktuella grafen och använder de Window-variabler som definieras i Window Editor.

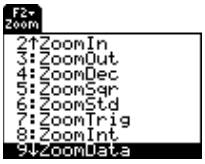
Använd [◆] [WINDOW] för att visa Window Editor. Gör något av följande:

- Ange lämpliga värden.

  – eller –
- Välj 9:ZoomData på menyn [F2] Zoom. (Även om du kan använda vilken zoomning som helst, optimeras ZoomData för statistiska plottningar.)

Tips! *[F2] Zoom är tillgänglig i Y= Editor, Window Editor och i graffönstret.*

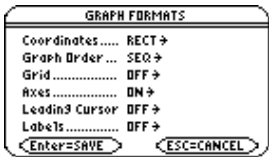
ZoomData ställer in visningsfönstret till att visa alla statistiska datapunkter.



För histogram och lådagram justeras endast xmin och xmax. Om överdelen av ett histogram inte visas kan du spåra i histogrammet för att hitta värdet för ymax.

Ändra grafformat

Tryck:
[F1] 9
— eller —
TI-89: [◆] [|]
TI-92 Plus: [◆] F
i Y= Editor, Window Editor, eller graffönstret.



Ändra inställningarna efter önskemål.

Spåra statistiska plottningar

Tryck på [F3] från graffönstret för att spåra en graf. Markörens rörelser är beroende på Plot Type.

Obs! *När en statistisk plottning visas panoreras inte graffönstret automatiskt om du spårar utanför fönstrets vänstra eller högra sida. Du kan dock fortfarande trycka på [ENTER] för att centrera fönstret vid spårningsmarkören.*

Plot Type	Beskrivning
Scatter eller xyline	Spårning börjar vid den första datapunkten.
Box plot	Spårning börjar vid medianen. Tryck på ◀ för att spåra till den nedre kvartilen (Q1) och minX. Tryck på ▶ för att spåra till den övre kvartilen (Q3) och maxX.
Histogram	Markören flyttar mellan staplarnas övre mittpunkter, med början från stapeln längst till vänster.

När du trycker på ▲ eller ▼ för att flytta till en annan plottning eller y(x)-funktion, flyttas spårningen till den aktuella punkten eller den första punkten i den plottningen (inte till den närmaste bildpunkten).

Du kan bestämma på vilket sätt datapunkter ska analyseras genom att använda frekvensvärden och/eller kategorivärden. Med frekvensvärden kan du "väga" särskilda datapunkter. Med kategorivärden kan du analysera en undergrupp av datapunkterna.

Exempel på en frekvenskolumn

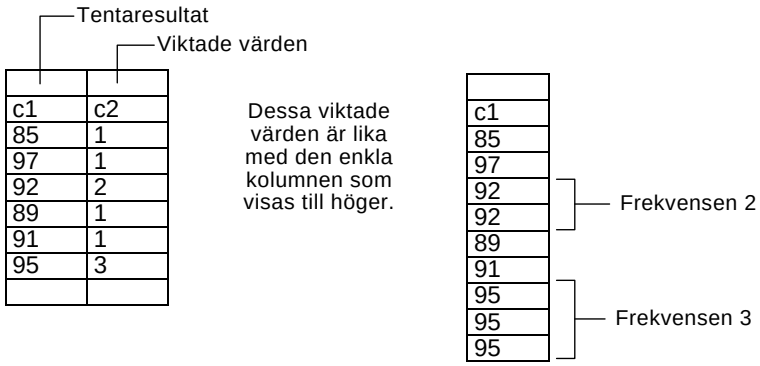
I en datavariabel kan du använda kolumner i Data/Matrix Editor för att ange ett frekvensvärde (eller vikt) för datapunkterna på varje rad. Ett frekvensvärde måste vara ett heltal ≥ 0 om Calculation Type = OneVar eller MedMed eller om Plot Type = Box Plot. För andra statistiska beräkningar eller plottningar kan frekvensvärdet vara vilket tal som helst ≥ 0.

Anta att du matar in en students resultat där:

- Hösttentans vikt är två gånger så stor som övriga tentor.
- Sluttentans vikt är tre gånger så stor som övriga tentor.

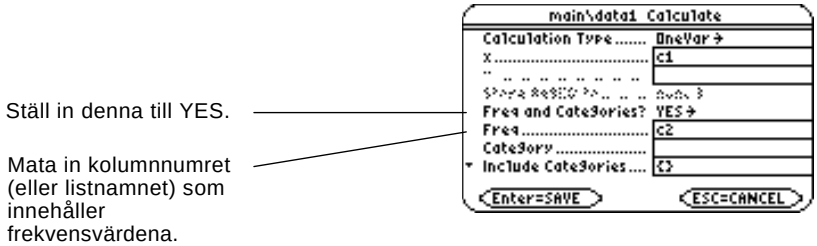
I Data/Matrix Editor kan du ange resultaten och viktningen i två kolumner.

__Tips!__ Ett frekvensvärde på 0 tar effektivt bort datapunkten från analysen.



__Obs!__ Du kan även använda frekvensvärden från en listvariabel i stället för en kolumn.

Om du vill använda frekvensvärden anger du frekvenskolumnen när du utför en statistisk beräkning eller definierar en statistisk plottning. Exempel:



Exempel på en kategorikolumn

I en datavariabel kan du använda en kolumn för att ange ett kategorivärde (eller undergrupp) för datapunkterna på varje rad. Ett kategorivärde kan vara vilket tal som helst.

Anta att du matar in tentaresultaten från en klass från linje 1 och en klass från linje 2. Du vill analysera resultaten för båda linjerna, men du vill även analysera kategorier som exempelvis flickor på linje 1, pojkar på linje 1, flickor och pojkar på linje 1 osv.

Först bestämmer du vilka kategorivärden du vill använda.

***Obs!** Du behöver inget kategorivärde för båda linjerna. Du behöver heller inga kategorivärden för alla på linje 1 eller alla på linje 2 eftersom de är en kombination av andra kategorier.*

Kategorivärde	Används för att ange:
1	flickor på linje 1
2	pojkar på linje 1
3	flickor på linje 2
4	pojkar på linje 2

I Data/Matrix Editor kan du ange resultatet och kategorivärdena i två kolumner.

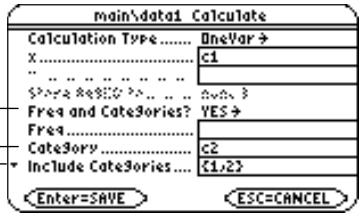
Testresultat

Kategorivärde

c1	c2
85	1
97	3
92	2
88	3
90	2
95	1
79	4
68	2
92	4
84	3
82	1

***Obs!** Du kan också använda kategorivärden från en listvariabel i stället för en kolumn.*

Om du vill använda kategorivärden anger du kategorikolumn och de kategorivärden som ska ingå i analysen när du utför en statistisk beräkning eller definierar en statistisk plottning.



Ställ in denna till YES.

Mata in kolumnnumret (eller listnamnet) som innehåller kategorivärdena.

Mata in de kategorivärde inom { } som ska användas, avgränsade med komman. (Mata inte in kolumnnummer eller listnamn.)

***Obs!** Om du vill analysera hela klassen låter du inmatningsrutan för Category vara tom. Eventuella kategorivärden ignoreras.*

Om du vill analysera:	Inkluderar du kategorierna:
flickor på linje 1	{1}
pojkar på linje 1	{2}
pojkar och flickor på linje 1	{1,2}
flickor på linje 2	{3}
pojkar på linje 2	{4}
pojkar och flickor på linje 2	{3,4}
alla flickor (linje 1 och 2)	{1,3}
alla pojkar (linje 1 och 2)	{2,4}

Om du har tillvalsprogrammet CBL och CBR

Systemet CBL™ (Calculator-Based Laboratory™) och CBR™ (Calculator-Based Ranger™) är ett tillvalsprogram som du kan köpa separat och med vilket du kan samla in data från olika experiment i verkligheten. CBL- och CBR-program för TI-89 / TI-92 Plus finns på TI:s hemsidor: http://www.ti.com/calc/cbl och http://www.ti.com/calc/cbr

Hur CBL-data lagras

När du samlar in data med CBL lagras dessa från början i själva CBL-enheten. Du måste därefter ta fram dessa data (överföra den till TI-89 / TI-92 Plus) genom att använda kommandot **Get** som beskrivs i appendix A.

Även om varje uppsättning överförd data kan lagras i flera variabeltyper (list-, reell-, matris-, bild-), är det lättare utföra statistiska beräkningar med hjälp av listvariabler.

Obs! *Mer specifik information om hur du använder CBL och överför data till TI-89 / TI-92 Plus finns i den bruksanvisning som levereras med CBL-enheten.*

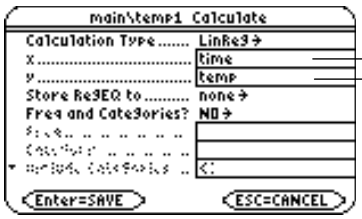
När du överför den insamlade informationen till TI-89 / TI-92 Plus kan du ange namnen på de listvariabler du vill använda. Du kan exempelvis använda CBL för att insamla temperaturdata över en tidsperiod. När du överför data kan vi till exempel anta att du lagrar:

- Temperaturdata i en listvariabel med namnet temp.
- Tidsdata i en listvariabel med namnet time.

När du har lagrat CBL-informationen i TI-89 / TI-92 Plus kan du använda CBL-listvariablerna på två sätt.

Referera till CBL-listorna

När du utför en statistisk beräkning eller definierar en plottning kan du referera direkt till CBL-listvariablerna. Exempel:



Mata in CBL-listvariabel-namnet i stället för ett kolumnnummer.

Skapa en datavariabel med CBL-listorna

Du kan skapa en ny datavariabel som består av nödvändiga CBL™-listvariabler.

- Från grundfönstret eller ett program använder du kommandot **NewData**.

  NewData *dataVar*, *lista1* [,*lista2*] [,*lista3*] ...

  CBL-listvariabelnamnen. I den nya datavariabeln, kommer lista1 att kopieras till kolumn 1, lista2 till kolumn 2 osv.

  Namnge den nya datavariabeln som du vill skapa.

  Exempel:

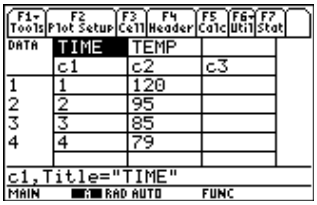
  NewData temp1, time, temp

  skapar en datavariabel med namnet temp1 i vilken time är i kolumn 1 och temp är i kolumn 2.

__Tips!__ Om du vill definiera eller rensa ett kolumnhuvud använder du F4. Mer information finns i kapitel 15.

- Från Data/Matrix Editor skapar du en ny, tom datavariabel med lämpligt namn. För varje CBL-lista som du vill inkludera definierar du ett kolumnhuvud som listnamn.

Exempel: Definiera kolumn 1 som time och kolumn 2 som temp.



Vid det här tillfället är kolumnerna länkade till CBL-listorna. Om listorna ändras uppdateras kolumnerna automatiskt. Om listorna däremot tas bort förloras alla data.

Om du vill göra datavariabeln oberoende av CBL-listorna rensar du kolumnhuvudet för varje kolumn. Informationen förblir i kolumnen men kolumnen är inte längre länkad till CBL-listan.

CBR

Du kan även använda Calculator-Based Ranger™ (CBR™) när du vill utforska sambanden mellan sträcka, hastighet, acceleration och tid med data insamlade från dina aktiviteter.

Programmering

17

Förhandsvisning av programmeringsfunktioner 276
Köra befintliga program 278
Starta en session i Program Editor 280
Översikt över inmatning av program 282
Översikt över inmatning av funktioner 285
Anropa ett program från ett annat 287
Använda variabler i ett program 288
Använda lokala variabler i funktioner och program 290
Strängoperationer 292
Villkorliga tester 294
Använda If, Lbl och Goto för att styra programflödet 295
Använda slingor för att repetera en grupp kommandon 297
Konfigurera TI-89 / TI-92 Plus 300
Låta användaren göra en inmatning och sedan visa utmatningen 301
Skapa en anpassad meny 303
Skapa en tabell eller graf 305
Rita i graffönstret 307
Använda en annan TI-89 / TI-92 Plus, en CBL eller en CBR 309
Felsöka program och hantera fel 310
Exempel: Använda alternativa metoder 311
Assemblerprogram 313

I det här kapitlet beskrivs hur du använder Program Editor för att skapa egna program och funktioner.

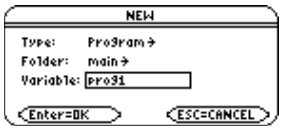
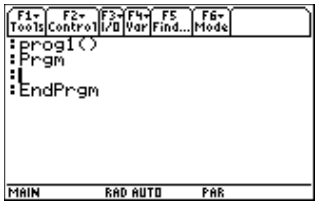
Kapitlet innehåller:

- Specifika instruktioner för hur du använder Program Editor och hur du kör befintliga program.
- En översikt över grundläggande programmeringstekniker som exempelvis **If...EndIf**-strukturer och olika typer av slingor.
- Referensinformation som delar upp de tillgängliga programkommandona i kategorier.
- Hämta och köra assemblerprogram.

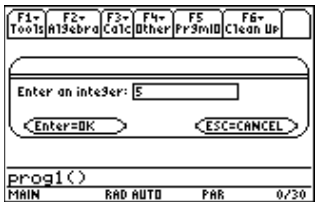
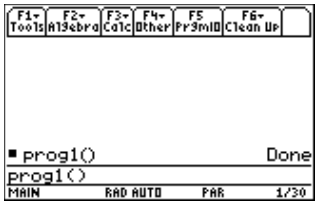
Obs! *Detaljerad information och exempel på programkommandon som nämns i detta kapitel finns i appendix A.*

Förhandsvisning av programmeringsfunktioner

Skriv ett program som uppmanar användaren att ange ett heltal, summerar alla heltal från 1 till det angivna heltalet och slutligen visar resultatet.

Steg	TI-89 Tangenter	TI-92 Plus Tangenter	Teckenfönster
1. Starta ett nytt program i Program Editor.	APPS 7 3	APPS 7 3	
2. Skriv PROG1 (utan mellanslag) som namnet på den nya programvariabeln.	▼ ▼ P R O G alpha 1	▼ ▼ P R O G 1	
3. Visa "mallen" för ett nytt program. Programnamnet, **Prgm** och **EndPrgm** visas automatiskt. *När du har skrivit i en inmatningsruta, exempelvis Variable, måste du trycka på ENTER två gånger.*	ENTER ENTER	ENTER ENTER	

4. Skriv följande programrader.
`Request "Enter an integer",n`
Visar en dialogruta som visar meddelandet "Enter an integer" och väntar på att användaren ska ange ett värde och sparar sedan värdet (som en sträng) till variabeln n.
`expr(n)→n`
Konverterar strängen till ett numeriskt uttryck.
`0→temp`
Skapar en variabel med namnet temp och initialiserar den till 0.
`For i,1,n,1`
Startar en For-slinga baserat på variabeln i. Första gången genom slingan är i = 1. I slutet av slingan ökas i med 1. Slingan fortsätter tills i > n.
`temp+i→temp`
Lägger till det aktuella värdet av i till temp.
`EndFor`
Markerar slutet av For-slingan.
`Disp temp`
Visar det slutliga värdet av temp.
	Skriv programraderna så som de visas. Tryck på ENTER i slutet av varje rad.	Skriv programraderna så som de visas. Tryck på ENTER i slutet av varje rad.	`:prog1()`
			`:Prgm`
			`:Request "Enter an integer",n`
			`:expr(n)→n`
			`:0→temp`
			`:For i,1,n,1`
			`:  temp+i→temp`
			`:EndFor`
			`:Disp temp`
			`:`
			`:EndPrgm`

Steg	TI-89 Tangenter	TI-92 Plus Tangenter	Teckenfönster
5. Visa grundfönstret. Mata in programnamnet, följt av två parenteser. *Du måste inkludera ( ), även om det inte finns några argument för programmet.* *Programmet visar en dialogruta med meddelandet angivet i programmet.*	[HOME] [2nd] [a-lock] P R O G [alpha] 1 [(] [)] [ENTER]	[♦] [HOME] P R O G 1 [(] [)] [ENTER]	prog1()
6. Skriv 5 i dialogrutan som visas.	5	5	
7. Fortsätt med programmet. Kommandot **Disp** visar resultatet i Program I/O-fönstret. *Resultatet är summan av heltalen från 1 till och med 5.* *Även om Program I/O-fönstret liknar grundfönstret, är det endast till för programinmatning och programresultat. Du kan därför inte utföra beräkningar i Program I/O-fönstret.*	[ENTER] [ENTER]	[ENTER] [ENTER]	
8. Växla från Program I/O-fönstret till grundfönstret. *Du kan också trycka på [ESC], [2nd] [QUIT] eller* ***TI-89:*** *[HOME]* ***TI-92 Plus:*** *[♦] [HOME] för att återgå till grundfönstret.*	[F5]	[F5]	

Köra befintliga program

När du har skapat ett program (beskrivs i de återstående avsnitten i detta kapitel) kan du köra det från grundfönstret. Eventuella programresultat visas i Program I/O-fönstret, i en dialogruta eller i graffönstret.

Köra ett program

Gör följande från grundfönstret:

1. Skriv namnet på programmet.
2. Du måste *alltid* skriva två parenteser efter namnet.

   Vissa program kräver att du överför argument till programmet.

```
prog1()
```

Om inga argument krävs

```
prog1(x,y)
```

Om argument krävs

3. Tryck på [ENTER].

Tips! *Använd [2nd] [VAR-LINK] för att visa befintliga PRGM-variabler. Markera en variabel och tryck på [ENTER] för att klistra in dess namn på inmatningsraden.*

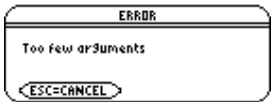
Obs! *Argument anger särskilda startvärden för ett program. Se sidan 283.*

När du kör ett program kontrollerar räknaren automatiskt om det finns några fel. Följande meddelanden kan exempelvis visas om du:

- Inte matade in ( ) efter programnamnet.

Detta felmeddelande visas om du:

- Inte matade in tillräckligt många argument.



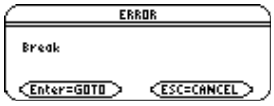
Obs! *Räknaren kontrollerar också om det finns fel som uppstår under körning inuti själva programmet. Se sidan 310.*

Du kan avbryta programkörningen om ett fel uppstår, genom att trycka på [ESC]. Du kan därefter rätta till felen och köra programmet igen.

Stoppa ett program

När ett program körs visas indikatorn BUSY i statusfältet.

Tryck på [ON] för att stoppa programkörningen. Ett meddelande visas.



- Om du vill visa programmet i Program Editor trycker du på [ENTER]. Markören visas vid kommandot där programmet stoppades.
- Du avbryter en programkörning genom att trycka på [ESC].

Var visas resultatet?

Beroende på kommandona i programmet visas automatiskt information i lämpligt fönster.

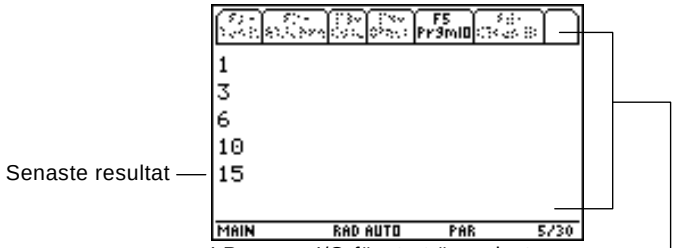
- De flesta in- och utmatningar använder Program I/O-fönstret. (Inmatningskommandon uppmanar användaren att ange information.)
- Grafrelaterade kommandon använder oftast graffönstret.

När programmet har stoppats visas det sista fönstret som visades.

Program I/O-fönstret

I Program I/O-fönstret visas nya resultat under föregående resultat (som kan ha visats tidigare i samma program eller i annat program). När Program I/O-fönstret är fullt rullar det översta resultatet ut från fönstrets överdel.

Tips! *Du rensar föregående resultat genom att ange kommandot* **ClrIO** *i programmet. Du kan också köra* **ClrIO** *från grundfönstret.*



I Program I/O-fönstret är endast menyn:
- F5 tillgänglig, alla andra är nedtonade.
- Det finns ingen inmatningsrad.

Tips! *Om beräkningar i grundfönstret inte fungerar efter att du har kört ett program kan det vara för att du befinner dig i Program I/O-fönstret.*

När ett program stannar i Program I/O-fönstret måste du kunna se att det *inte* är grundfönstret (även om de två fönstren är lika varandra). Program I/O-fönstret används endast för att visa resultat eller för att uppmana användaren att ange något. Du kan inte utföra beräkningar i detta fönster.

Växla från Program I/O-fönstret

Gör följande från Program I/O-fönstret:

- Tryck på F5 för att visa grundfönstret och Program I/O-fönstret.

  – eller –
- Tryck på ESC, 2nd [QUIT], eller
  TI-89: HOME
  TI-92 Plus: ♦ [HOME]
  för att öppna grundfönstret.

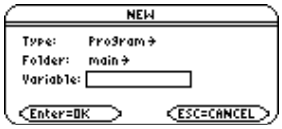
  – eller –
- Visa ett annat applikationsfönster (med APPS, ♦ [Y=] osv).

Varje gång du startar Program Editor kan du fortsätta med det aktuella programmet eller den aktuella funktionen (som visades förra gången du använde Program Editor), öppna ett befintligt program eller befintlig funktion eller starta ett nytt program eller en ny funktion.

Starta ett nytt program eller ny funktion

1. Tryck på [APPS] och välj 7:Program Editor.
2. Välj 3:New.

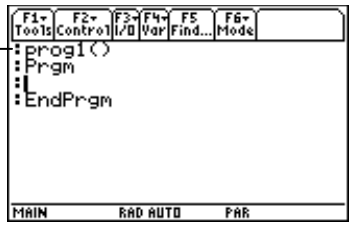
3. Ange lämplig information för det nya programmet eller den nya funktionen.



Med alternativet	Kan du göra följande:
Type	Välja om du vill skapa ett nytt program eller en ny funktion.
Folder	Välja vilken mapp som det nya programmet eller den nya funktionen ska sparas i. Information om mappar finns i kapitel 5.
Variable	Skriva ett variabelnamn för programmet eller funktionen. Om du anger en variabel som redan finns visas ett felmeddelande när du trycker på [ENTER]. När du trycker på [ESC] eller [ENTER] för att bekräfta felet visas dialogrutan NEW igen.

4. Tryck på [ENTER] (när du har skrivit i en inmatningsruta, som exempelvis Variable, måste du trycka på [ENTER] två gånger) för att visa en tom mall.

Detta är mallen för ett program. Funktioner har en liknande mall.



Du kan nu använda Program Editor på det sätt som beskrivs i de återstående avsnitten i detta kapitel.

Obs! *Ett program (eller en funktion) sparas automatiskt när du skriver. Du behöver inte spara det manuellt innan du växlar från Program Editor, startar ett nytt program eller öppnar ett tidigare program.*

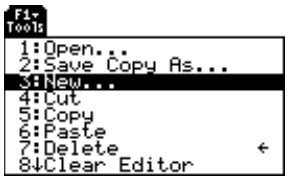
Fortsätta det aktuella programmet

Du kan växla från Program Editor och gå till ett annat program när som helst. Du återgår till programmet eller funktionen som visades när du växlade från Program Editor genom att trycka på [APPS] 7 och välja 1:Current.

Starta ett nytt program från Program Editor

Så här växlar du från det aktuella programmet eller den aktuella funktionen och startar ett nytt program eller en ny funktion:

1. Tryck på [F1] och välj 3:New.
2. Ange typ, mapp och variabel för det nya programmet eller den nya funktionen.
3. Tryck på [ENTER] två gånger.



Öppna ett föregående program

Du kan när som helst öppna ett program eller funktion som skapades tidigare.

1. Från Program Editor trycker du på [F1] och väljer 1:Open.

   – eller –

   Tryck på [APPS] 7 från ett annat program och välj 2:Open.

2. Välj typ, mapp och variabel.
3. Tryck på [ENTER].

Obs! *Som standard visar Variable det första befintliga programmet eller den första befintliga funktionen i alfabetisk ordning.*

Kopiera ett program

Ibland kanske du vill kopiera ett program eller funktion så att du kan redigera kopian utan att originalet ändras.

1. Visa programmet eller funktionen som du vill kopiera.
2. Tryck på [F1] och välj 2:Save Copy As.
3. Ange mappen och variabeln för kopian.
4. Tryck på [ENTER] två gånger.

Anmärkning om att ta bort program

Eftersom alla sessioner i Program Editor sparas automatiskt kan det bli ganska många program och funktioner, som tar upp onödigt mycket utrymme i minnet.

Du tar bort program och funktioner med hjälp av VAR-LINK-fönstret ([2nd] [VAR-LINK]). Information om VAR-LINK finns i kapitel 21.

Översikt över inmatning av program

Ett program är en serie kommandon som körs efter varandra (även om vissa kommandon ändrar programflödet). Allmänt sett kan allt som kan köras från grundfönstret inkluderas i ett program. Programkörning fortsätter till slutet av programmet eller till kommandot **Stop**.

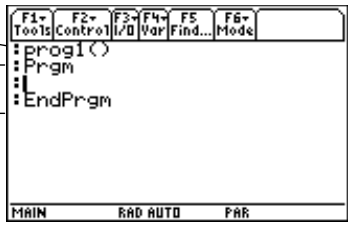
Mata in och redigera programrader

Du kan börja mata in kommandot för det nya programmet i en tom mall.

Programnamn, som du anger när du skapar ett nytt program.

Mata in programkommandona mellan **Prgm** och **EndPrgm**.

Alla programrader börjar med ett kolon.



Obs! *Använd markörknappen för att rulla genom programmet och mata in eller redigera kommandona. Använd [◆] [⊖] respektive [◆] [⊕] för att flytta till början eller slutet av ett program.*

Du matar in och redigerar programkommandon i Program Editor på samma sätt som du matar in och redigerar text i Text Editor.
Se "Skriva och redigera text" i kapitel 18.

Obs! *Kommandot körs inte när du matar in det. Det körs inte förrän du kör programmet.*

Efter varje programrad som du skriver trycker du på [ENTER].
Då infogas en ny tom rad och du kan mata in en till programrad.
En programrad kan vara längre än en rad i fönstret. Om så är fallet radbryts den automatiskt.

Mata in rader med flera kommandon

Om du vill mata in mer än ett kommando på samma rad, avgränsar du dem med ett kolon genom att trycka på [2nd] [:].

Mata in kommentarer

När kommentarsymbolen (◉) visas kan du mata in en kommentar i programmet. När du kör programmet ignoreras alla tecken till höger om symbolen ◉.

Tips! *Använd kommentarer för att mata in information som är användbar för någon annan som läser programkoden.*

Beskrivning av programmet.

Beskrivning av **expr**.

```
:prog1()
:Prgm
:◉Displays sum of 1 thru n
:Request "Enter an integer",n
:expr(n)→n:◉Convert to numeric expression
:------
```

Så här matar du in en kommentarsymbol, tryck på:

- **TI-89:** [◆] [)]
  TI-92 Plus: [2nd] X

  – eller –
- Tryck på [F2] och välj 9:◉.

Styra programflödet

Tips! Mer information finns på sidan 295 och 297.

När du kör ett program utförs programraderna efter varandra. Vissa kommandon kan dock ändra programflödet. Exempel:

- Villkorsstrukturer som exempelvis kommandona **If...EndIf** använder ett villkorligt test för att bestämma vilken del av programmet som ska köras.
- Slingkommandon som exempelvis **For...EndFor** repeterar en grupp av kommandon.

Använda indrag

För mer komplexa program som använder **If...EndIf** och slingstrukturer som exempelvis **For...EndFor** kan du göra programmen lättare att läsa och förstå genom att använda indrag.

```
:If x>5 Then
:  Disp "x is > 5"
:Else
:  Disp "x is < or = 5"
:EndIf
```

Visa beräknade resultat

I ett program visas inte beräknade resultat såvida du inte använder ett utmatningskommando. Detta är viktig skillnad mellan att utföra en beräkning i grundfönstret och i ett program.

Dessa beräkningar visar inte ett resultat i ett program (även om de gör det i grundfönstret).

```
:12*6
cos(π/4)
:solve(x^2-x-2=0,x)
```

Tips! En lista över tillgängliga utmatningskommandon finns på sidan 302.

Utmatningskommandon, som exempelvis **Disp**, visar ett resultat i ett program.

```
:Disp 12*6
:Disp cos(π/4)
:Disp solve(x^2-x-2=0,x)
```

Genom att visa ett beräkningsresultat lagras inte resultatet. Om du vill använda ett resultat senare kan du spara det till en variabel.

```
:cos(π/4)→maximum
:Disp maximum
```

Överföra värden till ett program

För att överföra inmatningsvärden till ett program kan du göra följande:

- Kräva att användarna lagrar ett värde (med [STO▸]) till de obligatoriska variablerna innan de kör programmet. Programmet kan sedan använda dessa värden.
- Mata in värdena direkt i programmet.

```
:Disp 12*6
:cos(π/4)→maximum
```

- Ta med inmatningskommandon som uppmanar användarna att ange de obligatoriska värdena när de kör programmet.

```
:Input "Enter a value",i
:Request "Enter an integer",n
```

Tips! En lista över tillgängliga inmatningskommandon finns på sidan 301.

- Kräva att användarna överför ett eller fler värden till programmet när de kör det.

```
prog1(3,5)
```

Exempel på att överföra värden till ett program

Följande program ritar en cirkel i graffönstret och ritar sedan en horisontell linje ovanpå cirkeln. Tre värden måste överföras till programmet: x- och y-koordinaterna för cirkelns medelpunkt och radien r.

- När du skriver ett program i Program Editor:

*__Obs!__ I det här exemplet kan du inte använda **circle** som programnamn eftersom det är i konflikt med ett kommandonamn.*

I parenteserna ( ) bredvid programnamnet anger du vilka variabler som ska användas för att lagra de överförda värdena.

Lägg märke till att programmet även innehåller kommandon som ställer in graffönstret.

```
:circ(x,y,r)
:Prgm
:FnOff
:ZoomStd
:ZoomSqr
:Circle x,y,r
:LineHorz y+r
:EndPrgm
```

Det är endast **circ( )** som visas från början i den tomma mallen; kom ihåg att redigera denna rad.

Innan cirkeln ritas stänger programmet av eventuella markerade funktioner i Y= Editor, visar ett standardvisningsfönster och visar fönstret på ett sådant sätt att 1 längdenhet på x-axeln är lika lång som 1 längdenhet på y-axeln.

- Så här kör du programmet från grundfönstret:

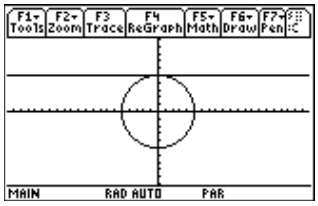
__Obs!__ I detta exempel antas att användaren matar in värden som kan visas av visningsfönstret, som ställs in av ZoomStd och ZoomSqr.

Användaren måste ange lämpliga värden som argument inom ( ).

Argumenten överförs efter varandra till programmet.

```
circ(0,0,5)
```

Överfört till r.
Överfört till y.
Överfört till x.



Översikt över inmatning av funktioner

En funktion som skapas i Program Editor påminner mycket om de funktioner och instruktioner som du ofta använder från grundfönstret.

Varför skapa en användardefinierad funktion?

__Obs!__ Du kan skapa en funktion från grundfönstret (se kapitel 5), men Program Editor är bättre att använda för komplexa funktioner som är längre en rad.

Funktioner (liksom program) är idealiska för upprepade beräkningar och uppgifter. Du behöver bara skriva funktionen en gång. Därefter kan du återanvända den hur många gånger som helst. Det finns dock vissa fördelar med funktioner, jämfört med program.

- Du kan skapa funktioner som utvidgar antalet inbyggda funktioner i TI-89 / TI-92 Plus. Du kan därefter använda de nya funktionerna på samma sätt som vilka funktioner som helst.
- Funktioner, till skillnad från program, returnerar värden som kan plottas eller matas in i en tabell.
- Du kan använda en funktion, till skillnad från ett program, inuti ett uttryck. Exempelvis är 3∗ func1(3) giltigt, men inte 3∗ prog1(3).
- Eftersom du överför parametrar till en funktion kan du skriva generiska funktioner som inte är bundna till vissa variabelnamn.

Skillnader mellan funktioner och program

I denna handbok används ibland termen *kommando* som en allmän referens till instruktioner och funktioner. När du skriver en funktion måste du dock skilja på instruktioner och funktioner.

En användardefinierad funktion:

- Kan endast använda följande instruktioner. Alla andra är ogiltiga.

Cycle	Define	Exit
For...EndFor	Goto	If...EndIf (alla former)
Lbl	Local	Loop...EndLoop
Return	While...EndWhile	→ (tangenten STO▸)

- Kan använda alla inbyggda funktioner i TI-89 / TI-92 Plus förutom:

setFold	setGraph	setMode
setTable	switch	

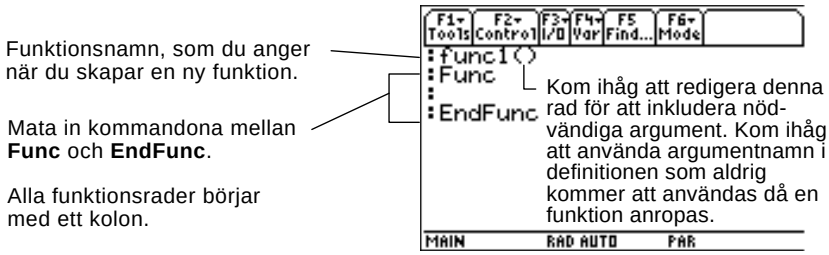
__Tips!__ Information om lokala variabler finns på sidan 288 och 290.

- Kan referera till vilken variabel som helst, men den kan bara spara ett värde till en lokal variabel.
  - Argumenten som används för att överföra värden till en funktion behandlas automatiskt som lokala variabler. Om du sparar till någon annan variabel *måste* du uppge den som lokal, inifrån funktionen.
- Kan inte anropa ett program som en subrutin, men den kan anropa en annan användardefinierad funktion.
- Kan inte definiera ett program.
- Kan inte definiera en global funktion, men den kan definiera en lokal funktion.

Mata in en funktion

Obs! Använd markörknappen för att rulla igenom funktionen och mata in och redigera kommandon.

När du skapar en ny funktion i Program Editor visas en tom mall.



Om funktionen kräver inmatning måste ett eller fler värden överföras till funktionen. (En användardefinierad funktion kan endast använda lokala variabler och den kan inte använda instruktioner som uppmanar användaren att göra en inmatning.)

Returnera ett värde från en funktion

Det finns två sätt att returnera ett värde från en funktion:

- På funktionens sista rad (före **EndFunc**) beräknar du värdet som ska returneras.

```
:cube(x)
:Func
:x^3
:EndFunc
```

Obs! I detta exempel beräknas kuben om x≥0; annars returneras 0.

- Använd **Return**. Detta är användbart för att avsluta en funktion och returnera ett värde vid något annat ställe än i slutet av funktionen.

```
:cube(x)
:Func
:If x<0
:  Return 0
:x^3
:EndFunc
```

Argumentet x behandlas automatiskt som en lokal variabel.
Om exemplet däremot skulle behöva en annan variabel måste funktionen deklarera den som lokal med hjälp av kommandot **Local** (sidan 288 och 290).

Det finns en underförstådd **Return** i slutet av funktionen. Om den sista raden inte är ett uttryck uppstår ett fel.

Exempel på en funktion

Följande funktion returnerar den x:te roten av ett värde y ($\sqrt[x]{y}$).
Två värden måste överföras till funktionen: x och y.

Obs! Eftersom x och y i funktionen är lokala påverkas de inte av en befintlig x- eller y-variabel.

Funktionen som den anropas från grundfönstret	Funktionen som den definierades i Program Editor
3→x: 125→y	
4*xroot(3,125) 20	:xroot(x,y) :Func :x^(1/y) :EndFunc
5	

Anropa ett program från ett annat

Ett program kan anropa ett annat program som en subrutin. Subrutinen kan vara extern (ett separat program) eller intern (inkluderad i huvudprogrammet). Subrutiner är användbara när ett program behöver upprepa samma grupp av kommandon på olika ställen.

Anropa ett separat program

När du anropar ett separat program använder du samma syntax som används för att köra programmet från grundfönstret.

```
:subtest1()
:Prgm
:For i,1,4,1
:  subtest2(i,i*1000)
:EndFor
:EndPrgm
```

```
:subtest2(x,y)
:Prgm
:  Disp x,y
:EndPrgm
```

Anropa en intern subrutin

Om du vill definiera en intern subrutin använder du kommandot **Define** med **Prgm...EndPrgm**. Eftersom en subrutin måste definieras innan den kan anropas är det en god idé att definiera subrutiner i början av programmet.

En intern subrutin anropas och körs på samma sätt som ett separat program.

Tips! *Använd menyn* F4 Var *i Program Editor för att ange kommandona* **Define** *och* **Prgm...EndPrgm**.

Deklarerar subrutinen som en lokal variabel.

Definierar subrutinen.

Anropar subrutinen.

```
:subtest1()
:Prgm
:local subtest2
:Define subtest2(x,y)=Prgm
:  Disp x,y
:EndPrgm
:● Beginning of main program
:For i,1,4,1
:  subtest2(i,i*1000)
:EndFor
:EndPrgm
```

Använda subrutiner

I slutet av en subrutin återgår körningen till det anropande programmet. Om du vill avsluta subrutinen vid något annat tillfälle använder du kommandot **Return**.

En subrutin kan inte använda lokala variabler som har deklarerats i det anropande programmet. På samma sätt kan inte det anropande programmet använda lokala variabler som har deklarerats i en subrutin.

Kommandona **Lbl** är lokala för de program där de finns. Därför kan inte ett **Goto**-kommando förgrena sig till en nivå i en subrutin eller tvärtom.

Program använder variabler på samma sätt som du använder dem från grundfönstret. Olika variabler sparas och hämtas på olika sätt.

Olika typer av variabler

Variabeltyp	Beskrivning
System-variabler (globala)	Variabler med reserverade namn som skapas automatiskt för att lagra data om räknarens status. Exempelvis är Window-variabler (xmin, xmax, ymin, ymax osv.) globalt tillgängliga från vilken mapp som helst. • Du kan alltid hänvisa till dessa variabler genom att endast använda variabelnamnet, oavsett den aktuella mappen. • Ett program kan inte skapa systemvariabler, men det kan använda värdena och (i de flesta fall) lagra nya värden.
Mapp-variabler	Variabler som är lagrade i en viss mapp. • Om du endast lagrar till ett variabelnamn lagras den i den aktuella mappen. Exempel: 5→start • Om du refererar till endast ett variabelnamn måste den variabeln finnas i den aktuella mappen. Annars kan den inte hittas (även om variabeln finns i en annan mapp). • Om du vill lagra eller referera till en variabel som är lagrad i en annan mapp måste du ange hela sökvägen. Exempel: 5→class\start Variabelnamn Mappnamn Efter att programmet har avslutats finns eventuella mappvariabler som skapades av programmet fortfarande kvar och tar upp minne.
Lokala variabler	Temporära variabler som endast finns under tiden programmet körs. När programmet avslutas tas lokala variabler bort automatiskt. • Om du vill skapa en lokal variabel i ett program använder du kommandot **Local** för att deklarera variabeln. • En lokal variabel behandlas som unik även om det finns en befintlig mappvariabel med samma namn. • Lokala variabler är idealiska för att tillfälligt lagra värden som du inte vill spara permanent.

Obs! *Information om mappar finns i kapitel 5.*

Obs! *Om ett program har lokala variabler kan en plottad funktion inte använda dem. Exempel:*
Local a
5→a
Graph a∗cos(x)
kan visas som ett fel eller oförväntat resultat (om a är en befintlig variabel i den aktuella mappen).

Fel av typen Circular Definition

När du utvärderar en användardefinierad funktion eller program kan du ange ett argument som innehåller samma variabel som användes för att definiera funktionen eller programmet. För att undvika cirkelreferenser måste du dock ange ett värde för variablerna x eller i som används vid beräkningen av funktionen eller programmet. Till exempel:

```
x+1→x
– eller –
For i,i,10,1
  Disp i
EndFor
```

Orsakar ett fel av typen **Circular definition** om x eller i saknar värde. Felet uppstår inte om x eller i redan har tilldelats ett värde.

Variabelrelaterade kommandon

Obs! *Kommandona* **Define**, **DelVar** *och* **Local** *är tillgängliga från menyn* F4 Var *i Program Editor.*

Kommando	Beskrivning
STO▸	Lagrar ett värde till en variabel. Precis som i grundfönstret trycker du på STO▸ för att mata in symbolen →.
Archive	Flyttar angivna variabler från RAM-minnet till användarminnet.
BldData	Med detta kan du skapa en datavariabel utifrån den plottningsinformation du angett i Y=Editor, Window Editor, etc.
CopyVar	Kopierar innehållet i en variabel.
Define	Definierar ett program (subrutin) eller en funktions-variabel inuti programmet.
DelFold	Tar bort en mapp. Alla variabler i den mappen måste först tas bort.
DelVar	Tar bort en variabel.
getFold	Returnerar namnet på den aktuella mappen.
getType	Returnerar en sträng som anger datatypen (EXPR, LIST osv) av en variabel.
Local	Deklarerar en eller fler variabler som lokala variabler.
Lock	Låser en variabel så att den inte av misstag kan ändras eller tas bort utan att först låsas upp.
MoveVar	Flyttar en variabel från en mapp till en annan.
NewData	Skapar en datavariabel vars kolumner består av en serie angivna listor.
NewFold	Skapar en ny mapp.
NewPic	Skapar en bildvariabel baserat på en matris.
Rename	Ändrar namn på en variabel.
Unarchiv	Flyttar angivna variabler från användarminnet till RAM-minnet.
Unlock	Låser upp en låst variabel.

Om en användardefinierad funktion eller program innehåller en lokal variabel som inte är initierad ges ett felmeddelande av typen **Undefined variable** när variabeln refereras.

Exempel på en lokal variabel

I följande programsegment visas en **For...EndFor**-slinga (beskrivs senare i detta kapitel). Variabeln i är slingans räknare. I de flesta fall används endast variabeln i under tiden programmet körs.

__Tips!__ När du kan bör du använda lokala variabler för variabler som endast används inuti ett program och inte behöver sparas efter att programmet har avslutats.

Deklarerar variabeln i som lokal.

```
:Local i
:For i,0,5,1
:  Disp i
:EndFor
:Disp i
```

Om du deklarerar variabel i som lokal tas den automatiskt bort när programmet stannar, så att den inte tar upp minne.

Vad orsakar felmeddelandet?

På TI-89 / TI-92 Plus kan användardefinierade funktioner och program använda lokala variabler som inte tilldelats något värde. Med TI-89 / TI-92 Plus visas felmeddelande av typen Undefined variable när du försöker utvärdera funktionen eller köra programmet.

Detta exempel består av en funktion med flera programsatser i stället för av ett program. I exemplet visas radbrytningar men du måste skriva in texten på inmatningsraden som en kontinuerlig text i formen: Define fact(n)=Func:Local… där de tre punkterna visar att texten fortsätter utanför skärmen.

Till exempel:

```
Define fact(n)=Func:
Local m:
While n>1:
  n*m→m: n-1→n:
EndWhile:
Return m:
EndFunc
```

Den lokala variabeln m ges inte något startvärde.

I exemplet ovan existerar den lokala variabeln m oberoende av variabler m som existerar utanför funktionen.

Måste du initiera lokala variabler

På en TI-89 / TI-92 Plus måste alla lokala variabler ges ett värde innan de används.

```
Define fact(n)=Func:
Local m: 1→m:
While n>1:
  n*m→m: n-1→n:
EndWhile:
Return m:
EndFunc
```

1 lagras som startvärde för m.

Ett resultat av denna förändring är att funktioner och program på en TI-89 / TI-92 Plus inte kan använda lokala variabler för symboliska beräkningar.

Utföra symboliska beräkningar

Om du vill att en funktion eller ett program ska utföra symboliska beräkningar måste du använda globala variabler i stället för lokala. Kontrollera dock noga att den globala variabeln inte redan finns definierad utanför programmet. Följande metoder kan vara till hjälp.

- Referera till ett globalt variabelnamn, med två eller fler tecken, som inte troligen används utanför funktionen eller programmet.
- Inkludera **DelVar** i funktionen eller programmet för att ta bort den globala variabeln innan du refererar till den. (**DelVar** tar inte bort låsta eller sparade variabler.)

Strängoperationer

Du använder strängar för att mata in och visa tecken. Du kan skriva en sträng direkt eller spara en sträng till en variabel.

Hur strängar används

En sträng är en följd av tecken inom citattecken (""). I programmering gör strängar så att programmet kan visa information eller uppmanar användaren att utföra en funktion. Exempel:

```
Disp "Resultatet är",answer
```

– eller –

```
Input "Ange vinkeln i grader",ang1
```

– eller –

```
"Ange vinkeln i grader"→str1
Input str1,ang1
```

Vissa inmatningskommandon (exempelvis **InputStr**) sparar automatiskt användarinmatningar som en sträng och användaren behöver då inte skriva citattecken.

En sträng kan inte beräknas matematiskt, även om den ser ut att vara ett numeriskt uttryck. Strängen "61" motsvarar exempelvis tecknen "6" och "1", inte talet 61.

Även om du inte kan använda en sträng som "61" eller "2x+4" i en beräkning kan du konvertera en sträng till ett numeriskt uttryck genom att använda kommandot **expr**.

Strängkommandon

Obs! *I appendix A hittar du syntaxen för alla kommandon och funktioner i din TI-89 / TI-92 Plus.*

Kommando	Beskrivning
#	Konverterar en sträng till ett variabelnamn.
&	Sätter ihop (länkar ihop) två strängar till en sträng.
char	Returnerar tecknet som motsvarar en angiven teckenkod. Detta är motsatsen till kommandot **ord**.
dim	Returnerar antalet tecken i en sträng.
expr	Konverterar en sträng till ett uttryck och kör uttrycket. Detta är motsatsen till kommandot **string**. **Viktigt!** Vissa användarinmatningskommandon sparar det inmatade värdet som en sträng. Innan du kan utföra en matematisk operation på värdet måste du konvertera det till ett numeriskt uttryck.
format	Returnerar ett uttryck som en teckenbaserad sträng formaterad efter aktuell mall (fixed, scientific, engineering, etc.).
inString	Söker efter en sträng för att se om den innehåller en angiven sträng. Om den gör det returnerar **inString** teckenpositionen där den första förekomsten av den andra strängen börjar.
left	Returnerar ett angivet antal tecken från den vänstra sidan (början) av en sträng.
mid	Returnerar en angivet antal tecken från en position inuti en sträng.
ord	Returnerar teckenkoden för det första tecknet i en sträng. Detta är motsatsen till kommandot **char**.
right	Returnerar ett angivet antal tecken från den högra sidan (slutet) av en sträng.
rotate	Roterar tecknen i en sträng. Förvald inställning är -1 (rotera ett steg till höger).
shift	Skiftar tecknen i en sträng och ersätter dem med blanksteg. Förvald inställning är -1 (skifta ett tecken till höger och ersätt med ett blanksteg). Exempel: shift("abcde",2)⇒"cde " och shift("abcde")⇒" abcd"
string	Konverterar ett numeriskt uttryck till en sträng. Detta är motsatsen till kommandot **expr**.

Villkorliga tester

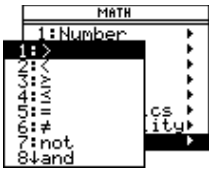
Med villkorliga tester kan program ta beslut. Ett program kan exempelvis, beroende på om ett test är sant eller falskt, bestämma vilken av de två åtgärderna som ska utföras. Villkorliga tester används tillsammans med villkorsstrukturer, som exempelvis **If...EndIf**, och slingor, som exempelvis **While...EndWhile** (beskrivs senare i detta kapitel).

Ange en testoperator

- Skriv operatorn direkt från tangentbordet.

  – eller –
- Tryck på [2nd][MATH] och välj 8:Test. Välj sedan operatorn på menyn.

  – eller –
- Visa de inbyggda funktionerna. Tryck:
  TI-89: [CATALOG]
  TI-92 Plus: [2nd] [CATALOG]
  Test-operatorerna listas i slutet av menyn [F2] Built-in.



Relationstester

Med relationsoperatorer kan du definiera ett villkorlig test som jämför två värden. Värdena kan vara tal, uttryck, listor eller matriser (men de måste matcha typ och dimension).

__Tips!__ Du kan skriva följande från tangentbordet:
>= för ≥
<= för ≤
/= för ≠
(För tecknet / trycker du på [÷].)

Operator	Sant om:	Exempel
>	Större än	a>8
<	Mindre än	a<0
≥	Större än eller lika med	a+b≥100
≤	Mindre än eller lika med	a+6≤b+1
=	Lika med	list1=list2
≠	Inte lika med	mat1≠mat2

Booleska tester

Med booleska operatorer kan du kombinera resultaten av två separata tester.

Operator	Sant om:	Exempel
and	Båda tester är sanna	a>0 and a≤10
or	Minst ett test är sant	a≤0 or b+c>10
xor	Ett test är sant och det andra är falskt	a+6<b+1 xor c<d

Funktionen Not

Med funktionen **not** ändras resultatet från ett test från sant till falskt och tvärtom. Exempel:

not x>2 är sant om x≤2
falskt om x>2

Obs! Om du använder **not** från grundfönstret visas det som **~** i historiklistan. Exempelvis visas not x>2 som ~(x>2).

Använda If, Lbl och Goto för att styra programflödet

En **If...EndIf**-struktur använder ett villkorligt test för att avgöra om ett eller flera kommandon ska köras eller inte. Kommandona **Lbl** (etikett) och **Goto** kan också användas för att förgrena sig (eller hoppa till) från en plats till en annan i ett program.

Menyn F2 Control

Om du vill använda **If...EndIf**-strukturer använder du menyn F2 Control i Program Editor.

Kommandot **If** är tillgängligt direkt på menyn F2.

Visa en undermeny med andra **If**-strukturer genom att välja 2:If...Then.

När du väljer en struktur som exempelvis **If...Then...EndIf** infogas en mall vid markören.

```
:If | Then

:EndIf
```

Markören placeras så att du kan ange ett villkorligt test.

Kommandot If

__Tips!__ Använd indrag för att göra programmen lättare att förstå och läsa.

När du vill köra endast ett kommando om ett villkorligt test är sant, använder du följande struktur:

Körs endast om x>5; annars ignoreras det.

Visar alltid x-värdet.

```
:If x>5
:   Disp "x is greater than 5"
:Disp x
```

I detta exempel måste du lagra ett värde till x innan du kör kommandot **If**.

If...Then...EndIf-strukturer

*__Obs!__ **EndIf** markerar slutet av **Then**-blocket som körs om villkoret är sant.*

När du vill köra flera kommandon om ett villkorligt test är sant, använder du följande struktur:

Körs endast om x>5.

Visar värdet av:
- 2x om x>5.
- x om x≤5.

```
:If x>5 Then
:  Disp "x is greater than 5"
:  2* x→ x
:EndIf
:Disp x
```

If...Then...Else... EndIf-strukturer

När du vill köra en grupp kommandon om ett villkorligt test är sant och en annan grupp om villkoret är falskt, använder du följande struktur:

```
:If x>5 Then
:  Disp "x is greater than 5"
:  2* x→ x
:Else
:  Disp "x is less than or
   equal to 5"
:  5* x→ x
:EndIf
:Disp x
```

Körs endast om x>5.

Körs endast om x≤5.

Visar värdet av:
- 2x om x>5.
- 5x om x≤5.

If...Then...Elself... EndIf-strukturer

Du kan använda en mer komplex form av kommandot **If** för att testa en serie av villkor. Anta att programmet uppmanar användaren att uppge ett tal som motsvarar ett av fyra alternativ. Om du vill testa för varje alternativ (If Choice=1, If Choice = 2 osv.), använder du strukturen **If...Then...Elself...EndIf**.

I appendix A finns mer information och ett exempel.

Kommandona Lbl och Goto

Du kan också styra programflödet genom att använda kommandona **Lbl** (etikett) och **Goto**.

Använd kommandot **Lbl** för att etikettera (tilldela ett namn till) en viss plats i programmet.

Lbl *etikettnamn*

Namn som ska tilldelas till denna plats (använd samma namnkonventioner som för variabelnamn)

Du kan därefter använda kommandot **Goto** vid ett ställe i programmet för att förgrena till platsen som motsvarar den angivna etiketten.

Goto *etikettnamn*

Anger vilket **Lbl**-kommando som förgreningen ska ske till

Eftersom ett **Goto**-kommando är ovillkorligt (det förgrenas alltid till den angivna etiketten), används det ofta tillsammans med ett **If**-kommando så att du kan ange ett villkorligt test. Exempel:

```
:If x>5
:  Goto GT5
:Disp x
:--------
:--------
:Lbl GT5
:Disp "The number was > 5"
```

Om x>5, hoppar programmet till etiketten GT5.

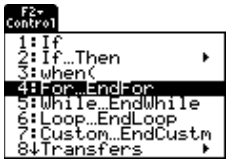
I detta exempel måste programmet inkludera kommandon (exempelvis **Stop**) som förhindrar att Lbl GT5 utförs om x≤5.

Använda slingor för att repetera en grupp kommandon

Om du vill repetera samma grupp kommandon efter varandra kan du använda en slinga. Det finns flera olika typer av slingor tillgängliga. Varje typ erbjuder ett sätt att avsluta slingan, baserat på ett villkorligt test.

Menyn F2 Control

Du anger de flesta slingrelaterade kommandon med menyn F2 Control i Program Editor.



__Obs!__ Ett slingkommando markerar starten på slingan. Motsvarande __End__-kommando markerar slutet av slingan.

När du väljer en slinga infogas slingkommandot och dess motsvarande **End**-kommando vid markören.

```
:For |
:EndFor
```

Om slingan kräver argument placeras markören efter kommandot.

Du kan därefter börja mata in de kommandon som ska utföras i slingan.

For...EndFor-slingor

En **For...EndFor**-slinga använder en räknare för att styra hur många gånger slingan ska repeteras. Syntaxen för kommandot **For** är följande:

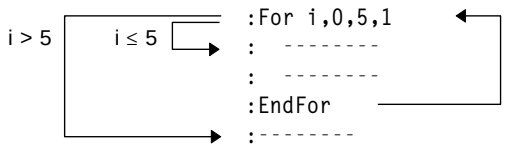
For(*variabel*, *start*, *slut*[, *steglängd*])

__Obs!__ Startvärdet kan vara mindre än slutvärdet, men ökningen måste då vara negativ.

- *steglängd* — Läggs till räknaren varje efterföljande gång som **For** körs. (Om detta valfria värde utesluts är ökningen 1.)
- *slut* — Avslutar slingan när *variabeln* överskrider detta värde
- *start* — Räknevärde som används den första gången **For** körs
- *variabel* — Variabel som används som räknare

När **For** körs jämförs *variabelvärdet* med *slutvärdet*. Om *variabeln* inte överskrider *slut* körs slingan, annars styr programmet hopp till kommandot som kommer efter **EndFor**.

__Obs!__ Kommandot __For__ ökar automatiskt räknevariabeln så att programmet kan avsluta slingan efter ett visst antal repetitioner.



I slutet av slingan (**EndFor**) hoppar programkontrollen tillbaka till **For**-kommandot och *variabel* ökas och jämförs med *end*.

Tips! Du kan uppge räknevariabeln som lokal (sidan 288 och 290) om den inte behöver sparas när programmet har avslutats.

Exempel:

```
:For i,0,5,1
:  Disp i
:EndFor
:Disp i
```

Visar 0, 1, 2, 3, 4 och 5.

Visar 6. När *variabeln* ökas till 6, utförs inte slingan.

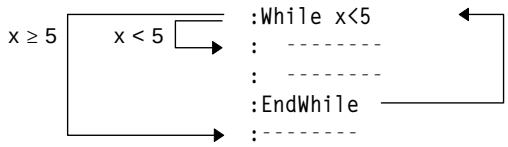
While...EndWhile-slingor

En **While...EndWhile**-slinga repeterar en grupp kommandon så länge som ett angivet villkor är sant. Syntaxen för kommandot **While** är följande:

While *villkor*

När **While** körs, beräknas villkoret. Om *villkor* är sant körs slingan, annars hoppar programkontrollen till kommandot efter **EndWhile**.

Obs! Kommandot **While** *ändrar inte automatiskt villkoret. Du måste ta med kommandon som låter programmet avsluta slingan.*



I slutet av slingan (**EndWhile**), hoppar programkontrollen tillbaka till **While**-kommandot och *villkoret* testas igen.

När du kör slingan första gången måste *villkoret* vara sant.

- Variabler som refereras i *villkoret* måste anges före kommandot **While**. (Du kan inkludera värdena i programmet eller uppmana användaren att ange värdena.)
- Slingan måste innehålla kommandon som ändrar värdena i *villkoret* och till slut gör det falskt. Annars är alltid *villkoret* sant och programmet kan inte avsluta slingan (en så kallad oändlig slinga).

Exempel:

```
:0→x
:While x<5
:  Disp x
:  x+1→x
:EndWhile
:Disp x
```

Ställer från början in x.

Visar 0, 1, 2, 3 och 4.

Ökar x.

Visar 5. När x ökas till 5, körs inte slingan.

Loop...EndLoop-slingor

Programsatsen **Loop...EndLoop** skapar en oändlig slinga, som upprepas i all oändlighet. Kommandot **Loop** har inga argument.

```
:Loop
:  --------
:  --------
:EndLoop
:--------
```

Slingan innehåller ofta kommandon som låter programmet avsluta från slingan. Kommandon som ofta används är: **If**, **Exit**, **Goto** och **Lbl** (etikett). Exempel:

```
:0→x
:Loop
:  Disp x
:  x+1→x
:  If x>5
:    Exit
:EndLoop
:Disp x
```

Ett **If**-kommando kontrollerar villkoret.

Avslutar slingan och hoppar hit när x ökas till 6.

Obs! *Kommandot* **Exit** *avslutar den aktuella slingan och hoppar till kommandot efter slingan.*

I detta exempel kan kommandot **If** vara var som helst i slingan.

När kommandot If är:	Körs slingan:
I början av slingan	Endast om villkoret är sant.
I slutet av slingan	Minst en gång och upprepas endast om villkoret är sant.

Kommandot **If** kan också använda ett **Goto**-kommando för att överföra programkontrollen till ett angivet **Lbl**- (etikett) kommando.

Omedelbart upprepa en slinga

Kommandot **Cycle** överför omedelbart programkontrollen till nästa upprepning av slingan (innan nästa upprepning är klar). Detta kommando fungerar med **For...EndFor**, **While...EndWhile** och **Loop...EndLoop**.

Lbl- och Goto-slingor

Även om kommandona **Lbl** (etikett) och **Goto** inte är strikta slingkommandon, kan du använda de för att skapa en oändlig slinga. Exempel:

```
:Lbl START
:  --------
:  --------
:Goto START
:--------
```

Precis som med **Loop...EndLoop** bör slingan innehålla kommandon som låter programmet avsluta slingan och hoppa till kommandot efter slingan.

Program kan innehålla kommandon som ändrar konfigureringen av din TI–89 / TI–92 Plus. Eftersom ändringar av läget är särskilt användbara kan du med hjälp av menyn **Mode** i Program Editor enkelt mata in korrekt syntax för kommandot **setMode**.

Konfigurations-kommandon

Obs! *Parametrar/lägen som används med setMode( ), getMode( ), setGraph( ) och setTable( ) översätts inte till andra språk när de används i program. Se appendix D.*

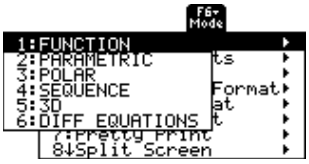
Kommando	Beskrivning
getConfg	Returnerar en lista med information om räknaren.
getFold	Returnerar namnet på den aktuella mappen.
getMode	Returnerar den aktuella inställningen för ett angivet läge.
getUnits	Visar en lista med förvalda måttenheter.
setFold	Ställer in den aktuella mappen.
setGraph	Ställer in ett angivet grafformat (Coordinates, Graph Order osv.).
setMode	Ställer in vilket läge som helst förutom Current Folder.
setTable	Ställer in en angiven inställningsparameter för tabeller (tblStart, Δtbl osv.)
setUnits	Ställer in förvalda måttenheter för visning av resultat.
switch	Ställer in det aktiva fönstret i ett delat fönster, eller returnerar numret på det aktiva fönstret.

Ange kommandot SetMode

Obs! *På menyn* Läge *kan du inte ange läget* Aktuell mapp. *Det gör du med kommandot* **setFold**.

Gör följande i Program Editor:

1. Placera markören där du vill infoga kommandot **setMode**.
2. Tryck:
   TI-89: [2nd] [F6]
   TI-92 Plus: [F6]
   för att öppna en lista med lägen.



3. Välj ett läge för att visa en meny med giltiga inställningar.
4. Välj en inställning.

Den korrekta syntaxen infogas i programmet.

```
:setMode("Graph","FUNCTION")
```

Låta användaren göra en inmatning och sedan visa utmatningen

Även om värden kan inkluderas i ett program (eller sparas till variabler i förväg), kan ett program uppmana användaren att mata in information när programmet körs. På samma sätt kan ett program visa information, som exempelvis resultatet av en beräkning.

Menyn F3 I/O

Du matar in de vanligaste inmatnings- och utmatningskommandona med hjälp av menyn F3 I/O i Program Editor.

Visa en undermeny med ytterligare kommandon genom att välja 1:Dialog.

Inmatningskommandon

Kommandot	Beskrivning
getKey	Returnerar tangentkoden för nästa tangent som trycks ned.
Input	Uppmanar användaren att mata in ett uttryck. Uttrycket behandlas efter hur det matas in. Exempel: • Ett numeriskt uttryck behandlas som ett uttryck. • Ett uttryck som är omgivet av citattecken behandlas som en sträng. **Input** kan också visa graffönstret och låta användaren uppdatera variablerna xc och yc (rc och θc i polärläge) genom att placera grafmarkören.
InputStr	Uppmanar användaren att mata in ett uttryck. Uttrycket behandlas alltid som en sträng, användaren behöver inte omge uttrycket med citattecken.
PopUp	Visar en snabbmeny där användaren kan välja ett alternativ.
Prompt	Uppmanar användaren att mata in en serie uttryck. Liksom med **Input** behandlas varje uttryck efter hur det matas in.
Request	Visar en dialogruta som uppmanar användaren att mata in ett uttryck. **Request** behandlar alltid det inmatade uttrycket som en sträng.

Tips! *Stränginmatning kan inte användas i en beräkning. Om du vill konvertera en sträng till ett numeriskt uttryck använder du kommandot* ***expr****.*

Utmatningskommandon

Obs! *Bara genom att utföra en beräkning i ett program visas inte resultatet. Du måste använda ett utmatningskommando.*

Tips! *Efter* **Disp** *och* **Output** *fortsätter programmet omedelbart. Du kan lägga till ett* **Pause***-kommando om du vill.*

Kommando	Beskrivning
ClrIO	Rensar Program I/O-fönstret.
Disp	Visar ett uttryck eller en sträng i Program I/O-fönstret. **Disp** kan också visa det aktuella innehållet i Program I/O-fönstret utan att visa ytterligare information.
DispG	Visar det aktuella innehållet i graffönstret.
DispHome	Visar det aktuella innehållet i grundfönstret.
DispTbl	Visar det aktuella innehållet i tabellfönstret.
Output	Visar ett uttryck eller en sträng som startar vid angivna koordinater i Program I/O-fönstret.
Format	Formaterar det sätt på vilket numerisk information visas.
Pause	Stoppar programkörning tills användaren trycker på [ENTER]. Om du vill kan du visa ett uttryck under paus. En paus gör att användaren kan läsa utmatningen och själv bestämma när han/hon ska fortsätta.
Text	Visar en dialogruta som innehåller en angiven teckensträng.

Användargränssnitts-kommandon

Tips! *När du kör ett program som ställer in ett anpassat verktygsfält, kommer det verktygsfältet fortfarande vara tillgängligt när programmet har avslutats.*

Obs! **Request** *och* **Text** *är fristående kommandon som också kan användas utanför en dialogruta eller ett programblock för ett verktygsfält.*

Kommando	Beskrivning
Dialog... endDlog	Definierar ett programblock (består av kommandona **Title**, **Request** osv) som visar en dialogruta.
Toolbar... EndTbar	Definierar ett programblock (består av kommandona **Title**, **Item** osv) som ersätter menyer i verktygsfältet. Det omdefinierade verktygsfältet visas endast under tiden programmet körs och bara tills användaren väljer ett alternativ. Därefter visas det ursprungliga verktygsfältet igen.
CustomOn... CustomOff	Aktiverar eller gömmer en anpassad menyrad.
Custom... EndCustm	Definierar ett programblock som visar ett anpassat verktygsfält när användaren trycker på [2nd] [CUSTOM]. Det verktygsfältet visas tills användaren trycker på [2nd] [CUSTOM] igen eller byter program.
DropDown	Visar en snabbmeny i en dialogruta.
Item	Visar ett menyalternativ för det omdefinierade verktygsfältet.
Request	Skapar en inmatningsruta i en dialogruta.
Text	Visar en teckensträng i en dialogruta.
Title	Visar namnet på en dialogruta eller ett menynamn i ett verktygsfält.

Skapa en anpassad meny

Med den anpassade menyn i din TI-89 / TI-92 Plus kan du skapa egna menyer. En anpassad meny kan innehålla funktioner, instruktioner eller en uppsättning tecken. I din TI-89 / TI-92 Plus finns en förvald anpassad meny som du kan ändra eller definiera om.

Aktivera och deaktivera den anpassade menyn

__Obs!__ När den anpassade menyn är aktiverad ersätts den vanliga menyn. Om ingen anpassad meny finns tillgänglig används standardmenyn i stället.

När du skapar en anpassad meny kan du låta användaren aktivera/deaktivera den manuellt, eller så kan du låta programmet göra detta automatiskt.

Om du vill:	Gör så här:
Aktivera den anpassade menyn	I grundfönstret eller i något annat program: • Tryck [2nd] [CUSTOM]. I grundfönstret eller programmatiskt: • Kör kommandot CustmOn.
Deaktivera den anpassade menyn	Från ett program: • Tryck på [2nd] [CUSTOM] igen. — eller — • Öppna ett annat program. Använda den förvalda menyn i grundfönstret: 1. Välj menyn Tools: **TI-89:** [2nd] [F7] **TI-92 Plus:** [F7] och sedan 3:CustmOff. Kommandot CustmOff klistras in på inmatningsraden. 2. Tryck på [ENTER]. Du kan även använda CustmOff i ett program.

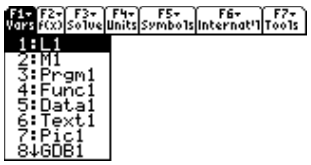
```
CustmOff
```

Definiera en anpassad meny

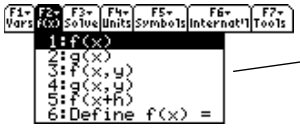
__Obs!__ När användaren väljer ett alternativ på menyn kommer texten som är definierad av det **Item** *att klistras in vid den aktuella markörpositionen.*

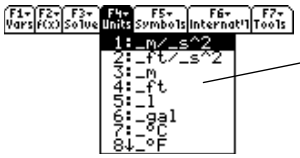
När du vill skapa en anpassad meny följer du beskrivningen nedan.

```
:Custom
: Title titel på meny F1
:   Item alternativ 1
:   Item alternativ 2
:   . . .
: Title titel på meny F2
:   . . .
: Title titel på meny F3
:   . . .
:EndCustm
```



***Obs!** Följande text kan skilja sig något från den aktuella texten på din räknare.*





***Obs!** Se hur "_\o\C" och "_\o\F" visas som °C och °F i menyn. Observera även de internationella tecknen.*

Till exempel:

```
:Custom
:Title "Vars"
:Item "L1":Item "M1":Item "Prgm1":Item "Func1":Item "Data1"
:Item "Text1":Item "Pic1":Item "GDB1":Item "Str1"
:Title "f(x)"
:Item "f(x)":Item "g(x)":Item "f(x,y)":Item "g(x,y)"
:Item "f(x+h)":Item "Define f(x) ="
:Title "Solve"
:Item "Solve(":Item " and ":Item "{x,y}"
:Item "Solve( and ,{x,y})"
:Title "Units"
:Item "_m/_s^2":Item "_ft/_s^2":Item "_m":Item "_ft":Item "_l"
:Item "_gal":Item "_\o\C":Item "_\o\F":Item "_kph":Item "_mph"
:Title "Symbols"
:Item "#":Item "\beta\":Item "?":Item "~":Item "&"
:Title "Internat'l"
:Item "\e`\":Item "\e'\":Item "\e^\":Item "\a`\"
:Item "\u`\":Item "\u^\":Item "\o^\":Item "\c,\":Item "\u..\"
:Title "Tools"
:Item "ClrHome":Item "NewProb":Item "CustmOff"
:EndCustm
:CustmOn
```

***Obs!** Detta sätter in alla kommandon på en rad. Du behöver **inte** dela upp dem på separata rader.*

Modifiera den förvalda menyn genom att välja 3:Restore custom default (enligt beskrivningen nedan) och återställ menyn till ursprungsformen. Kopiera kommandona, använd Program Editor för att skapa ett nytt program, och klistra in kommandona i det tomma programmet. Modifiera därefter kommando efter önskemål.

Du kan bara skapa och använda en anpassad meny i taget. Om du behöver fler kan du skriva ett separat program för varje meny. Starta sedan programmet med den meny du vill använda.

Återställa den förvalda menyn

Gör så här när du vill återställa den förvalda menyn:

1. I grundfönstrets normala meny (inte den anpassade) väljer du Clean Up:
   TI-89: [2nd] [F6]
   TI-92 Plus: [F6]
2. Välj 3:Restore custom default.

   Detta klistrar in kommandot för återställning av menyn på kommandoraden.
3. Tryck på [ENTER] för att köra kommandot och återställa menyn.

När du återställer den förvalda menyn raderas tidigare anpassade menyer. Om menyn skapats med ett program kan du senare köra programmet igen när du vill använda din anpassade meny.

Skapa en tabell eller graf

Du skapar en tabell eller graf baserat på en eller flera funktioner eller ekvationer genom att använda kommandona som beskrivs i detta avsnitt.

Tabellkommandon

Kommando	Beskrivning
DispTbl	Visar det aktuella innehållet i tabellfönstret.
setTable	Ställer in tabellparametern Graph <–> Table eller Independent. (Om du vill ställa in de andra två tabellparametrarna kan du spara de lämpliga värdena till systemvariablerna tblStart och Δtbl.)
Table	Skapar och visar en tabell baserat på ett eller flera uttryck eller en eller flera funktioner.

Grafkommandon

Kommando	Beskrivning
ClrGraph	Tar bort funktioner och uttryck som plottades med kommandot **Graph**.
Define	Skapar en användardefinierad funktion.
DispG	Visar det aktuella innehållet i graffönstret.
FnOff	Avmarkerar alla (eller endast angivna) Y=-funktioner.
FnOn	Markerar alla (eller endast angivna) Y=-funktioner.
Graph	Plottar ett eller fler angivna uttryck med hjälp av det aktuella grafläget.
Input	Visar diagramfönstret och låter användaren uppdatera variablerna xc och yc (rc och θc i polärläge) genom att placera grafmarkören.
NewPlot	Skapar en ny definition av en statistisk plottning.
PlotsOff	Avmarkerar alla (eller endast angivna) statistiska plottningar.
PlotsOn	Markerar alla (eller endast angivna) statistiska plottningar.
setGraph	Ändrar inställningar för de olika grafformaten (Coordinates, Graph Order osv).
setMode	Ställer in grafläget samt övriga lägen.
Style	Ställer in visningsformatet för en funktion.
Trace	Låter ett program spåra en graf.
ZoomBox – till – ZoomTrig	Utför alla Zoom-åtgärder som finns på menyn F2 i Y= Editor, Window Editor och graffönstret.

Obs! *Mer information om hur du använder* **setMode** *finns på sidan 300.*

Grafbilds- och databaskommandon

Obs! *Information om grafbilder och databaser finns även i kapitel 12.*

Kommando	Beskrivning
AndPic	Visar graffönstret och lägger en sparad grafbild ovanpå genom att använda AND-logik.
CyclePic	Animerar en serie sparade grafbilder.
NewPic	Skapar en grafbildsvariabel baserad på en matris.
RclGDB	Återställer alla inställningar som är sparade i en grafdatabas.
RclPic	Visar graffönstret och lägger en sparad grafbild ovanpå genom att använda OR-logik.
RplcPic	Rensar graffönstret och visar en sparad grafbild.
StoGDB	Sparar de aktuella grafinställningarna till en grafdatabasvariabel.
StoPic	Kopierar graffönstret (eller en angiven rektangulär del) till en grafbildsvariabel.
XorPic	Visar graffönstret och lägger en sparad grafbild ovanpå genom att använda XOR-logik.

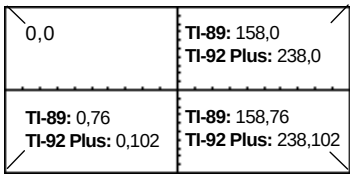
Du skapar ett ritobjekt i graffönstret genom att använda kommandona som beskrivs i detta avsnitt.

Bildpunktskoordinater och punktkoordinater

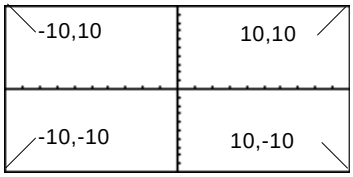
När du ritar ett objekt kan du använda ett av två koordinatsystem för att ange en plats i fönstret.

- **Bildpunktskoordinater** – Avser de bildpunkter som fysiskt utgör fönstret. Dessa är oberoende av visningsfönstret eftersom fönstret alltid är: :
  TI-89: 159 (0 till 158) pixlar bred och 77 (0 till 76) pixlar hög.
  TI-92 Plus: 239 (0 till 238) pixlar bred och 103 (0 till 102) pixlar hög.
- **Punktkoordinater** – Avser de koordinater som gäller för det aktuella visningsfönstret (som definieras i Window Editor).

Tips! Information om bildpunktskoordinater i delade fönster finns i kapitel 14.



Bildpunktskoordinater (oberoende av visningsfönstret)



Punktkoordinater (för standardvisningsfönstret)

***Obs!** Bildpunktskommandon startar med* Pxl*, exempelvis* **PxlChg**.

Många ritkommandon har två former: en för bildpunktskoordinater och en för punktkoordinater.

Ta bort ritade objekt

Kommando	Beskrivning
ClrDraw	Tar bort alla ritade objekt från graffönstret.

Rita en punkt eller en bildpunkt

Kommando	Beskrivning
PtChg eller PxlChg	Växlar (inverterar) en bildpunkt vid de angivna koordinaterna. **PtChg**, som använder punktkoordinater, påverkar bildpunkten som är närmast den angivna punkten. Om bildpunkten är avstängd slås den på. Om bildpunkten är på stängs den av.
PtOff eller PxlOff	Stänger av (tar bort) en bildpunkt vid de angivna koordinaterna. **PtOff**, som använder punktkoordinater, påverkar den bildpunkt som är närmast den angivna punkten.
PtOn eller PxlOn	Slår på (visar) en bildpunkt vid de angivna koordinaterna. **PtOn**, som använder punktkoordinater, påverkar den bildpunkt som är närmast den angivna punkten.
PtTest eller PxlTest	Returnerar sant eller falskt för att ange om den angivna koordinaten är på eller av.
PtText eller PxlText	Visar en teckensträng vid de angivna koordinaterna.

Rita linjer och cirklar

Kommando	Beskrivning
Circle eller PxlCrcl	Ritar, tar bort eller inverterar en cirkel med en angiven medelpunkt och radie.
DrawSlp	Ritar en linje med en angiven lutning genom en viss punkt.
Line eller PxlLine	Ritar, tar bort eller inverterar en linje mellan två koordinatpar.
LineHorz eller PxlHorz	Ritar, tar bort eller inverterar en horisontell linje vid en angiven radkoordinat.
LineTan	Ritar en tangent för ett angivet uttryck vid en angiven punkt. (Detta ritar endast tangenten, inte uttrycket.)
LineVert eller PxlVert	Ritar, tar bort eller inverterar en vertikal linje vid en angiven kolumnkoordinat.

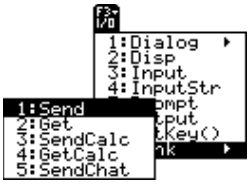
Plotta uttryck

Kommando	Beskrivning
DrawFunc	Plottar ett uttryck.
DrawInv	Plottar inversen av ett uttryck.
DrawParm	Plottar en parameterekvation genom att använda uttryck som dess x- och y-komponenter.
DrawPol	Plottar ett polärt uttryck.
DrwCtour	Ritar konturer i grafläget 3D.
Shade	Plottar två uttryck och skuggar ytorna där *uttryck1* < *uttryck2*.

Om du länkar två TI-89 / TI-92 Plus-räknare (beskrivs i kapitel 22), kan program på båda räknarna överföra variabler mellan sig. Om du länkar en TI-89 / TI-92 Plus till en CBL™ (Calculator-Based Laboratory™) eller en CBR™ (Calculator-Based Ranger™), kan ett program på TI-89 / TI-92 Plus använda CBL eller CBR.

Menyn F3 I/O

Använd menyn F3 I/O i Program Editor för att mata in kommandona i detta avsnitt.



1. Tryck på F3 och välj 8:Link.
2. Välj ett kommando.

Använda ytterligare en TI–89 / TI–92 Plus

När två TI-89 / TI-92 Plus-räknare är länkade fungerar en som en mottagande enhet och den andra som en sändande enhet.

__Obs!__ Programmen på de mottagande och sändande räknarna måste vara synkroniserade så att **GetCalc** *och* **SendCalc** *körs i rätt ordning, se "Överföra variabler med program-kontroll" i kapitel 22.*

Kommando	Beskrivning
GetCalc	Körs på den mottagande enheten. Ställer in enheten att ta emot en variabel via I/O-porten. • När den mottagande enheten har kört **GetCalc** måste den sändande enheten köra **SendCalc**. • När den sändande enheten har kört **SendCalc** sparas den överförda variabeln på den mottagande enheten (i det variabelnamn som anges av **GetCalc**).
SendCalc	Körs på den sändande enheten. Skickar en variabel till den mottagande enheten via I/O-porten. • Innan den sändande enheten kör **SendCalc** måste den mottagande enheten köra **GetCalc**.
SendChat	Utförs på den sändande enheten som ett alternativ till **SendCalc**. Användbart om den mottagande enheten är en TI-92 (eller för allmänna "chat"-program som tillåter att antingen en TI-92 eller TI-92 Plus används).

Använda CBL eller CBR

Mer information finns i den bruksanvisning som levereras med CBL- eller CBR-enheten.

Kommando	Beskrivning
Get	Hämtar en variabel från en ansluten CBL eller CBR och sparar den på TI-89 / TI-92 Plus.
Send	Skickar en listvariabel från TI-89 / TI-92 Plus till CBL eller CBR.

Felsöka program och hantera fel

> När du har skrivit ett program kan du använda flera metoder för att hitta och rätta till fel. Du kan också skapa ett felhanteringskommando i själva programmet.

Fel som uppstår vid programkörning

Det första steget i att felsöka programmet är att köra det. TI-89 / TI-92 Plus kontrollerar automatiskt varje kommando som körs för att se om det finns några fel. Om det finns ett fel visas ett meddelande som anger vilken typ av fel det är.

- Du visar programmet i Program Editor genom att trycka på [ENTER]. Markören visas ungefär där felet är.

- Om du vill avbryta programkörningen och återgå till grundfönstret trycker du på [ESC].

Om programmet låter användaren välja bland flera alternativ bör du köra programmet och testa alla alternativ.

Felsökningsmetoder

Felmeddelanden som uppstår vid programkörning kan lokalisera syntaxfel men inte fel i programlogiken (programuppbyggnaden). Följande metoder kan vara användbara.

- När du testar ska du inte använda lokala variabler. Annars kan du inte kontrollera variabelvärdena efter att programmet har avslutats. När programmet är felsökt deklarerar du de lämpliga variablerna som lokala.
- Infoga temporärt kommandona **Disp** och **Pause** i ett program för att visa värdena av kritiska variabler.
  - **Disp** och **Pause** kan inte användas i en användardefinierad funktion. Om du tillfälligt vill ändra funktionen i ett program ändrar du **Func** och **EndFunc** till **Prgm** och **EndPrgm**. Använd **Disp** och **Pause** för att felsöka programmet. Ta sedan bort **Disp** och **Pause** och ändra programmet tillbaka till en funktion.
- För att bekräfta att en slinga körs rätt antal gånger visar du räknevariabeln eller värdena i det villkorliga testet.
- För att bekräfta att en subrutin körs visar du meddelanden som exempelvis "Entering subroutine" och "Exiting subroutine" i början och slutet av en subrutin.

Felhanteringskommando

Kommando	Beskrivning
Try...EndTry	Definierar ett programblock som låter programmet köra ett kommando och, ifall det behövs, återställas från ett fel som orsakas av det kommandot.
ClrErr	Tar bort en felstatus genom att ställa in felnumret till noll.
PassErr	Överför ett fel till nästa nivå av **Try...EndTry**-programsatsen.

Exempel: Använda alternativa metoder

I översikten i början av detta kapitel visas ett program som uppmanar användaren att mata in ett heltal, summerar alla heltal från 1 till det angivna heltalet och sedan visar resultatet. Detta avsnitt handlar om hur du med alternativa metoder kan uppnå samma mål.

Exempel 1

I detta exempel visas samma program som visades i översikten i början av kapitlet. Se översikten för mer detaljerad information.

Uppmanar användaren att göra en inmatning i en dialogruta.

Konverterar en sträng som angavs med **Request** till ett uttryck.

Beräkning i slingan.

Visar resultat i Program I/O-fönstret

```
:prog1()
:Prgm
:Request "Enter an integer",n
:expr(n)→n
:0→temp
:For i,1,n,1
:  temp+i→temp
:EndFor
:Disp temp
:EndPrgm
```

Exempel 2

I detta exempel används **InputStr** för inmatning, en **While...EndWhile**-slinga för att beräkna resultatet och **Text** för att visa resultatet.

Uppmanar användaren att göra en inmatning i Program I/O-fönstret.

Konverterar en sträng som angavs med **InputStr** till ett uttryck.

Beräkning i slingan.

Visar resultat i en dialogruta.

```
:prog2()
:Prgm
:InputStr "Enter an integer",n
:expr(n)→n
:0→temp:1→i
:While i≤n
:  temp+i→temp
:  i+1→i
:EndWhile
:Text "The answer is "&string(temp)
:EndPrgm
```

Tips! *För ≤, skriv* ◆ 0 *(noll).*
För &, tryck:
TI-89: ◆ × *(gånger)*
TI-92 Plus: 2nd H

Exempel 3

I detta exempel används **Prompt** för inmatning, **Lbl** och **Goto** för att skapa en slinga och **Disp** för att visa resultatet.

Uppmanar användaren att göra en inmatning i Program I/O-fönstret.

Beräkning i slingan.

Visar resultat i Program I/O-fönstret.

```
:prog3()
:Prgm
:Prompt n
:0→temp:1→i
:Lbl top
:  temp+i→temp
:  i+1→i
:  If i≤n
:    Goto top
:Disp temp
:EndPrgm
```

Obs! *Eftersom* **Prompt** *returnerar n som ett tal behöver du inte använda* **expr** *för att konvertera n.*

Exempel 4

I detta exempel används **Dialog...EndDlog** för att skapa dialogrutor för inmatning och utmatning. **Loop...EndLoop** används för att beräkna resultatet.

```
:prog4()
:Prgm
:Dialog
:  Title "Enter an integer"
:  Request "Integer",n
:EndDlog
:expr(n)→n
:0→temp:0→i
:Loop
:  temp+i→temp
:  i+1→i
:  If i>n
:    Exit
:EndLoop
:Dialog
:  Title "The answer is"
:  Text string(temp)
:EndDlog
:EndPrgm
```

Definierar en dialogruta för inmatning.

Konverterar en sträng som matas in med **Request** till ett uttryck.

Beräkning i slingan.

Definierar en dialogruta för resultat.

Exempel 5

I detta exempel används de inbyggda funktionerna för att beräkna resultatet utan att använda en slinga.

```
:prog5()
:Prgm
:Input "Enter an integer",n
:sum(seq(i,i,1,n))→temp
:Disp temp
:EndPrgm
```

Uppmanar användaren att göra en inmatning i Program I/O-fönstret.

Beräknar summan.

Visar resultat i Program I/O-fönstret.

Obs! *Eftersom* **Input** *returnerar n som ett tal behöver du inte använda* **expr** *för att konvertera n.*

Funktion	Används i följande exempel:
seq	Generera talföljden av heltal från 1 till n. **seq**(*uttryck*, *variabel*, *start*, *slut*[,*steg*]) Ökning för *variabeln*; om den utelämnas används 1. Start- och slutvärden för *variabeln* Variabel som kommer att ökas Uttryck som används för att generera talföljden
sum	Summerar heltalen i listan som har genererats av **seq**.

Assemblerprogram

Du kan köra program skrivna i assembler med TI-89 / TI-92 Plus. Normalt är assemblerprogram snabbare och ger mer kontroll över körningen än program skrivna med Program Editor.

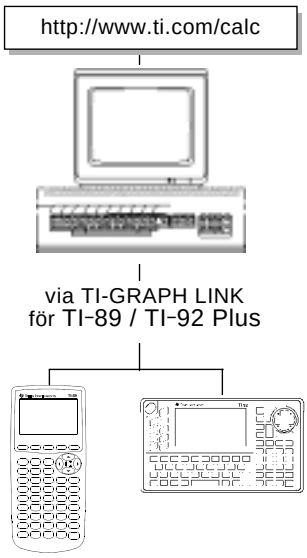
Hitta nya assemblerprogram

Assemblerprogram, liksom andra program, hittar du på TI:s hemsidor:

http://www.ti.com/calc

Programmen på denna webb-server ger dig extra funktioner som inte finns inbyggda i din TI-89 / TI-92 Plus. På webb-servern hittar du alltid den senaste informationen.

Efter du har laddat ner ett program från webben till din dator använder du TI-GRAPH LINK™ (säljs separat) för att sända programmet till din TI-89 / TI-92 Plus. I handboken som medföljer TI-GRAPH LINK finns mer information om hur du gör.



Observera för TI-GRAPH LINK

Om du har en kabel (dator till räknare) för TI-GRAPH LINK och programvara för TI-92, bör du observera att TI-GRAPH LINK *programvara* inte är kompatibel med TI-89 / TI-92 Plus. Kabeln fungerar dock med båda enheterna. Mer information om var du kan köpa TI-GRAPH LINK programvara, eller en kabel för att koppla räknaren till en dator, finns på Texas Instruments hemsidor: **http://www.ti.com/calc/docs/link.htm**, eller så kan du kontakta Texas Instruments enligt informationen i appendix C till denna handbok.

Köra ett assemblerprogram

När du har installerat ett assemblerprogram för TI-89 / TI-92 Plus kan du köra programmet från grundfönstret på samma sätt som du kör andra program.

Om programmet kräver ett eller flera argument skriver du dem inom parenteserna. Se programmets dokumentation för mer information om argument.

Tips! *Om programmet inte är i aktuell katalog måste du ange hela sökvägen.*

Du kan anropa ett assemblerprogram som en subrutin från andra program, ta bort det eller använda det på samma sätt som andra program.

Genvägar för att starta program

I grundfönstret kan du använda snabbtangenter för att starta upp till nio användardefinerade program eller assemblerprogram. Namngivningen av programmen måste följa nedanstående regler.

__Obs!__ Programmen måste vara lagrade i katalogen MAIN. Du kan inte köra program som kräver argument med snabb-tangenter.

I grundfönstret tryck på:	Om du vill köra programmet:
◆ 1	kbdprgm1()
⋮	⋮
◆ 9	kbdprgm9()

Om du har ett eget program som du vill starta via snabbtangenterna, och som inte är namngivet enligt ovanstående, måste du ändra programmets namn till kbdprgm1() etc.

Du kan inte redigera assemblerprogram

Du kan inte använda din TI-89 / TI-92 Plus för att redigera assemblerprogram. Program Editor kan inte öppna assemblerprogram.

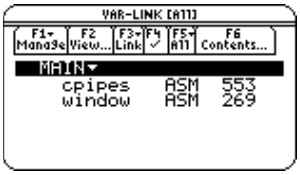
Visa en lista med assemblerprogram

Gör så här om du vill lista alla assemblerprogram som finns lagrade i minnet:

1. Öppna fönstret VAR-LINK (2nd [VAR-LINK]).
2. Tryck på F2 View.
3. Välj önskad katalog (eller All folders) och ange Var Type = Assembly.

__Obs!__ Assemblerprogram är av typen ASM.

4. Tryck på ENTER för att visa en lista med assemblerprogram.



Information om att skriva assemblerprogram

Information för nybörjare om hur man skriver program i assembler finns inte i denna handbok. Om du är insatt i assemblerprogrammering kan du hitta mer information på TI:s hemsidor (**http://www.ti.com/calc**) om hur du kommer åt olika funktioner i din TI-89 / TI-92 Plus.

__Obs!__ Du måste ha en dator för att skriva assemblerprogram. Du kan inte skriva in assemblerprogram direkt på räknarens tangentbord.

Din TI-89 / TI-92 Plus kan också använda kommandot **Exec** för att exekvera en sträng bestående av en serie 68000 op-koder. Dessa koder fungerar som en annan sorts assemblerprogram. Mer information hittar du på TI:s webbserver.

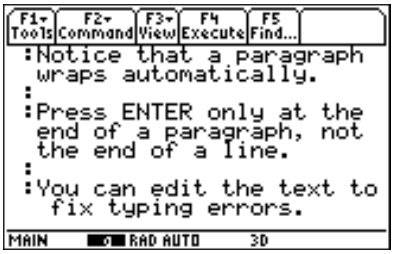
Varning: Med **Exec** får du full tillgång till räknarens mikroprocessor. Observera att du enkelt kan göra ett misstag som låser räknaren och gör att du förlorar all data. Vi rekommenderar att du gör en säkerhetskopia av räknaren innan du använder kommandot **Exec**.

Text Editor

18

Översikt över textfunktioner 316
Starta en session i Text Editor 317
Skriva och redigera text 319
Skriva specialtecken 324
Mata in och köra ett kommandoskript 328
Skapa en labbrapport 330

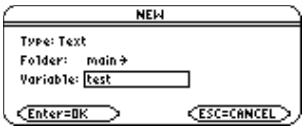
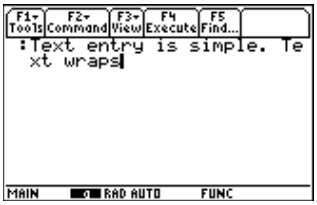
I detta kapitel beskrivs hur du använder Text Editor för att skriva och redigera text. Det är lätt att mata in text, du börjar bara skriva. När du redigerar text kan du använda samma tekniker som du använder för att redigera information i grundfönstret.



Varje gång du startar en ny textsession måste du ange namnet på en textvariabel. När du har påbörjat en session sparas text som du skriver automatiskt i den tillhörande textvariabeln. Du behöver inte spara en session manuellt innan du stänger Text Editor.

Förhandsvisning av textfunktioner

Starta en ny session i Text Editor. Öva sedan på att använda Text Editor genom att skriva vilken text som helst. När du skriver kan du öva på att flytta textmarkören och rätta till eventuella fel som du gör.

Steg	TI-89 Tangenter	TI-92 Plus Tangenter	Teckenfönster
1. Starta en ny session i Text Editor.	APPS 8 3	APPS 8 3	
2. Skapa en textvariabel med namnet TEST, som automatiskt kommer att spara text som du skriver i den nya sessionen. *Använd mappen* MAIN, *som visas som standardmapp i dialogrutan* NEW. *När du har skrivit i en inmatningsruta, som exempelvis* Variable, *måste du trycka på* ENTER *två gånger.*	⊖ T E S T ENTER ENTER	⊖ T E S T ENTER ENTER	
3. Skriv en exempeltext. • *När du vill skriva in ett enstaka tecken som versal trycker du på* ⇧ *och sedan önskat tecken.* ***TI-89** endast:* – *Skriv in ett blanksteg genom att trycka på* ␣ *(tangentens* (-) *alfa-funktion).* – *Skriv in en punkt genom trycka på* alpha *för att deaktivera alfalåsningen, tryck* ·, *och därefter* 2nd [a-lock] *för att aktivera alfa-låsningen igen.* *Öva på att redigera text genom att använda:* • *Markörknapparna för att flytta textmarkören.* • ← *respektive* ◆ [DEL] *för att ta bort tecken till vänster eller till höger om markören.*	2nd [a-lock] skriv valfri text	skriv valfri text	
4. Stäng Text Editor och visa grundfönstret. *Textsessionen sparades automatiskt när du skrev. Du behöver därför inte spara sessionen manuellt innan du avslutar Text Editor.*	HOME	◆ [HOME]	
5. Återgå till den aktuella sessionen i Text Editor.	APPS 8 1	APPS 8 1	
6. Lägg märke till att sessionen som visas är exakt så som den var när du stängde den.			

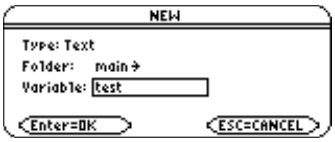
Starta en session i Text Editor

Varje gång du startar Text Editor kan du starta en ny text-session, fortsätta med den aktuella sessionen (den session som visades sist du använde Text Editor) eller öppna en föregående session.

Starta en ny session

1. Tryck på [APPS] och välj 8:Text Editor.
2. Välj 3:New.

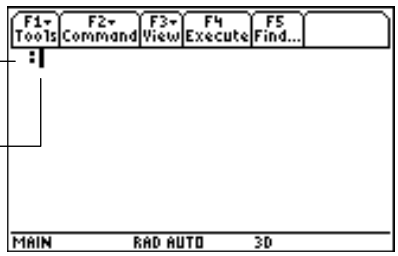
   Dialogrutan NEW visas.
3. Ange en mapp och en textvariabel som du vill använda för att spara den nya sessionen.



Alternativ	Beskrivning
Type	Är automatiskt inställd till Text och kan inte ändras.
Folder	Visar i vilken mapp textvariabeln kommer att sparas. Information om mappar finns i kapitel 5. Om du vill använda en annan mapp trycker du på ▶ för att visa en meny med tillgängliga mappar. Välj en mapp.
Variable	Skriv ett variabelnamn. Om du anger en variabel som redan finns visas ett felmeddelande när du trycker på [ENTER]. När du trycker på [ESC] eller [ENTER] för att bekräfta felet visas dialogrutan NEW igen.

4. Tryck på [ENTER] (när du har skrivit i en inmatningsruta, som exempelvis Variable, måste du trycka på [ENTER] två gånger) för att visa ett tomt Text Editor-fönster.

Obs! *Sessionen sparas automatiskt när du skriver. Du behöver inte spara en session manuellt innan du avslutar Text Editor, startar en ny session eller öppnar en föregående session.*



Ett kolon markerar början av ett stycke.

Den blinkande markören visas var den skrivna texten kommer att visas.

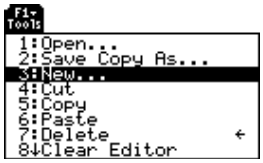
Du kan nu använda Text Editor enligt beskrivningen i de återstående avsnitten i detta kapitel.

Fortsätta med den aktuella sessionen

Du kan växla från Text Editor och gå till ett annat program när som helst. När du vill återgå till den session som visades när du växlade från Text Editor trycker du på [APPS] 8 och väljer 1:Current.

Starta en ny session från Text Editor

Så här växlar du från den aktuella sessionen i Text Editor och startar en ny session:



1. Tryck på [F1] och välj 3:New.
2. Ange en mapp och en textvariabel för den nya sessionen.
3. Tryck på [ENTER] två gånger.

Öppna en föregående session

Du kan när som helst öppna en föregående Text Editor-session.

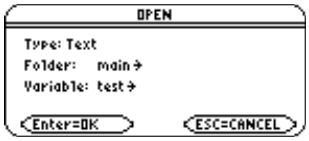
1. Från Text Editor trycker du på [F1] och väljer 1:Open.

   – eller –

   Från vilket program som helst trycker du på [APPS] 8 och väljer 2:Open.

Obs! *Som standard visar* Variable *den första befintliga textvariabeln i alfabetisk ordning.*

2. Välj mappen och textvariabeln.
3. Tryck på [ENTER].



Kopiera en session

Ibland kanske du vill kopiera en session så att du kan redigera kopian och samtidigt behålla originalet.

1. Visa sessionen som du vill kopiera.
2. Tryck på [F1] och välj 2:Save Copy As.
3. Ange mappen och textvariabeln för den kopierade sessionen.
4. Tryck två gånger på [ENTER].

Anmärkning om att ta bort en session

Eftersom alla sessioner i Text Editor sparas automatiskt kan det bli ganska många sessioner, vilka tar upp onödigt mycket minne.

Du tar bort en session genom att använda VAR-LINK-fönstret ([2nd] [VAR-LINK]) för att ta bort textvariabeln för den sessionen. Information om VAR-LINK finns i kapitel 21.

Skriva och redigera text

När du har påbörjat en session i Text Editor kan du skriva och redigera text. Allmänt sett använder du samma tekniker som du använder för att mata in och redigera information på inmatningsraden i grundfönstret.

Skriva text

__Obs!__ Använd markörknappen för att rulla igenom en session eller placera textmarkören.

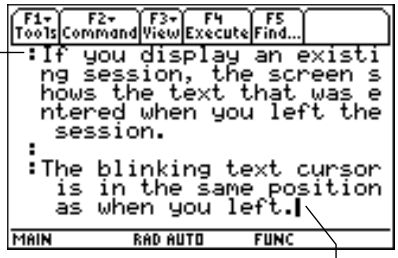
__Tips!__ Tryck på [2nd] ⊕ eller [2nd] ⊖ när du vill bläddra uppåt eller nedåt en rad i taget och [◆] ⊕ eller [◆] ⊖ när du vill gå till textens början eller slut.

__Tips!__ Om du har tillbehöret TI-GRAPH LINK kan du använda ett dator-tangentbord för att skriva in längre texter och sedan skicka texten till räknaren. Se sidan 322.

När du skapar en ny session i Text Editor visas ett tomt fönster. När du öppna en föregående session eller återgår till den aktuella sessionen visas den befintliga texten för den sessionen.

Alla textstycken börjar med ett mellanslag och ett kolon.

Mellanslaget i början används i kommandoskript och labbrapporter.



Blinkande textmarkör

Du behöver inte trycka på [ENTER] vid slutet av varje rad. I slutet av en rad flyttas nästa tecken till raden nedanför. Använd [ENTER] bara när du vill påbörja ett nytt stycke.

När du når till skärmens nederkant kommer tidigare rader att bläddras ut ut skärmen upptill.

Skriva in alfabetiska tecken

__Obs!__ På din TI-89, du behöver inte använda [alpha] eller alfalåsning när du vill skriva x, y, z eller t. Du måste dock använda [↑] eller versalt ALFALÅS för X, Y, Z eller T.

__Obs!__ På din TI-89, deaktiveras skiftlås alltid när du byter applikation, t ex om du går från Text Editor till grundfönstret.

Om du vill:	På din TI-89, tryck:	På din TI-92 Plus, tryck:
Skriva in ett enstaka tecken i gemen form.	[alpha] och sedan bokstavstangenten (statusraden visar ◨)	bokstavstangenten
Mata in ett enskilt alfabetiskt tecken i som versaler.	[↑] och sedan bokstavstangenten (statusraden visar ⬆)	[↑] och sedan bokstavstangenten (statusraden visar ⬆)
Mata in ett blanksteg.	[alpha] [␣] (alfafunktionen hos tangenten [(-)])	blanksteg
Aktivera skiftlås för gemener.	[2nd] [a-lock] (statusraden visar ■◨)	(ingen åtgärd krävs)
Aktivera skiftlås för versaler.	[↑] [a-lock] (statusraden visar ■⬆)	[2nd] [CAPS]
Deaktivera skiftlås.	[alpha] (för gemener och för versaler stänger av skiftlås)	[2nd] [CAPS] (stänger av skiftlås för versaler)

Skriva in alfabetiska tecken (fortsättning)

På TI-89 gäller, när någon form av skiftlås är aktiverat:

- Om du vill skriva in en punkt, ett komma, eller något annat tecken som är primärfunktion hos en tangent måste du deaktivera skiftlåset.
- Om du vill skriva in ett tecken som är en andra funktion, t ex [2nd] [{], behöver du inte deaktivera skiftlåset. När du har skrivit in tecknet är skiftlås fortfarande aktiverat.

Ta bort tecken

Om du vill ta bort:	Trycker du på:
Tecknet till vänster om markören	[←] eller [F1] 7
Tecknet till höger om markören	[◆] [DEL] (samma som [◆] [←])
Alla tecken som finns till höger om markören och till slutet av stycket	[CLEAR]
Alla tecken i stycket (oavsett var i stycket markören befinner sig)	[CLEAR] [CLEAR]

Obs! *Om det inte finns några tecken till höger om markören tas hela stycket bort när du trycker på* [CLEAR].

Markera text

Om du vill:	Gör du följande:
Markera text	1. Flytta markören till början eller slutet av texten. 2. Håll ned [↑] och tryck på: • ◀ eller ▶ för att markera tecken till vänster eller höger om markören. • ▼ eller ▲ för att markera alla tecken upp till markörpositionen på nästa eller föregående rad.

Tips! *Om du vill ta bort en markering utan att ersätta eller ta bort informationen flyttar du markören.*

Ersätta eller ta bort markerad text

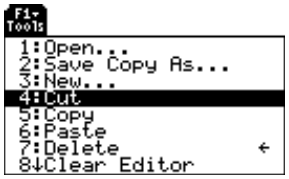
Om du vill:	Gör du följande:
Ersätta markerad text	Skriv den nya texten.
Ta bort markerad text	Tryck på [←].

Klippa ut, kopiera och klistra in text

Både när du klipper ut och kopierar placeras den markerade texten i Urklipp. När du klipper ut tas texten bort från den aktuella platsen (används för att flytta text) och när du kopierar finns texten kvar.

1. Markera texten som du vill flytta eller kopiera.
2. Tryck på [F1].
3. Välj ett menyalternativ.
   - Flytta texten genom att välja 4:Cut.

     – eller –
   - Kopiera texten genom att välja 5:Copy.
4. Flytta textmarkören dit du vill infoga texten.
5. Tryck på [F1] och välj 6:Paste.



Tips! *Du kan trycka på:*
TI-89:
[◆] [CUT], [◆] [COPY], [◆] [PASTE]
TI-92 Plus:
[◆] X, [◆] C, [◆] V
för att klippa ut, kopiera och klistra in utan att behöva använda menyn [F1].

Du kan använda följande allmänna procedur för att klippa ut, kopiera och klistra in text:

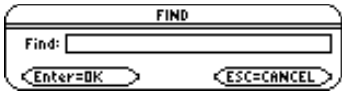
- Inom samma textsession.
- Från en textsession till en annan. När du har klippt ut eller kopierat text i en session, öppnar du den andra sessionen och klistrar in texten.
- Från en textsession till en annan applikation. Du kan exempelvis klistra in text på inmatningsraden i grundfönstret.

Söka efter text

Gör följande från Text Editor:

1. Placera textmarkören precis före den text du vill söka efter. Alla sökningar startar vid den aktuella markörpositionen.
2. Tryck på [F5].
3. Skriv söktexten.

   Sökningen är inte skiftlägeskänslig. Exempelvis har SKIFT, skift och Skift samma effekt.
4. Tryck på [ENTER] två gånger.



Tips! *I dialogrutan FIND behålls den senaste söktexten som du skrev. Du kan skriva över den eller redigera den.*

Om söktexten:	Händer följande med markören:
Hittas	Den flyttas till början av söktexten.
Inte hittas	Den flyttas inte.

Infoga eller skriva över ett tecken

Som standard är TI-89 / TI-92 Plus i infogningsläge. Du växlar mellan infognings- och överskrivningsläge genom att trycka på [2nd] [INS].

Tips! Du ser på markörens form om du är i infognings- eller överskrivningsläge.

Om det aktiva läget är:	Kommer nästa tecken som du skriver att:
Insert mode Tunn markör mellan tecken	Infogas vid markören.
Overtype mode Markören markerar ett tecken	Ersätta det markerade tecknet.

Rensa Text Editor

Om du vill ta bort alla befintliga stycken och visa ett tomt textfönster trycker du på [F1] och väljer 8:Clear Editor.

Använda en dator och TI-GRAPH LINK för att mata in text

Om du har tillbehörskabeln TI-GRAPH LINK™ och programvara för TI-89 / TI-92 Plus kan du använda din dator till att skriva in text och sedan skicka texten som en fil till din TI-89 / TI-92 Plus. Detta är ett enkelt sätt att skapa långa textavsnitt

Information om var du kan köpa TI-GRAPH LINK, eller uppgradera din TI-GRAPH LINK programvara för användning med TI-89 / TI-92 Plus, finns på TI:s hemsidor:

http://www.ti.com/calc

Du kan även kontakta Texas Instruments, se appendix C.

Utförliga instruktioner för hur du skapar en textfil på din dator och skickar den till en räknare finns i handboken som medföljer TI-GRAPH LINK. Nedan finner du en kort sammanfattning av de olika stegen:

1. Använd programvaran TI-GRAPH LINK till att skapa en ny textfil.
   a. Välj New på menyn File. Markera en TI-89 Data File eller TI-92 Plus Data File och klicka på OK. Ett namnlöst redigeringsfönster öppnas.
   b. Skriv in det namn du vill använda för textvariabeln i fältet Name, längst upp i redigeringsfönstret på din TI-89 / TI-92 Plus. Skriv in önskad text.
   c. Välj Save As på menyn File. Skriv in ett filnamn i dialogrutan som öppnas och välj Text som File Type. Välj en katalog och klicka på OK.

Obs! På räknaren kommer textvariabeln att få det namn du skrev in under steg 1b, inte filnamnet i steg 1c.

2. Använd TI-GRAPH LINK™ för att skicka filen från din dator till din TI-89 / TI-92 Plus.
   a. Anslut datorn och räknaren med kabeln som medföljer TI-GRAPH LINK.
   b. Kontrollera att grundfönstret är aktivt på din TI-89 / TI-92 Plus.
   c. Välj Send på menyn Link. Markera textfilen och klicka på Add för att lägga till det i listan Files Selected. Klicka på OK.
   d. När ett meddelande visar att överföringen är klar, klicka på OK.
3. Använd Text Editor på din TI-89 / TI-92 Plus och öppna textvariabeln.

Du kan använda menyn **CHAR** för att välja specialtecken från en lista. Du kan också skriva vissa vanligt använda specialtecken som sekundärfunktioner på tangentbordet. Du kan visa vilka specialtecken som är tillgängliga från tangentbordet genom att visa en tabell med tecknen och deras motsvarande tangenter.

Använda menyn CHAR

1. Tryck på [2nd] [CHAR].
2. Välj en kategori.

   En meny visar tecknen som finns i den kategorin.
3. Välj ett tecken. Du kanske måste rulla ned i menyn.

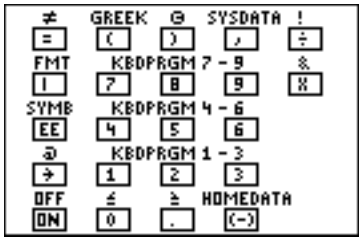
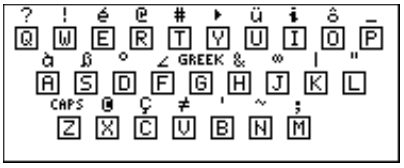
↓ anger att du kan rulla.

Välj international när du vill skriva in accentuerade tecken. De vanligaste internationella tecknen är också tillgängliga på den fövalda menyn Custom ([2nd] [CUSTOM]).

Visa tangentbordskartan

I tangentbordskartan visas genvägar för inmatning av speciella tecken och grekiska tecken via tangentbordet. Här visas även genvägar för andra funktioner i räknaren.

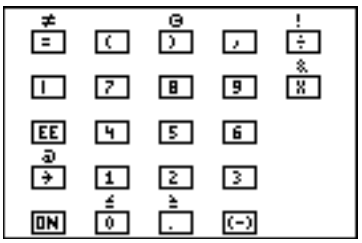
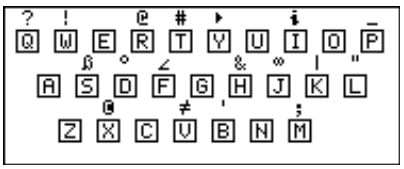
Tangentbordskartan visar inte alla tillgängliga genvägar. På insidan av främre och bakre omslaget på den här handboken finns en komplett lista över snabbtangenter.

På TI-89:	**På TI-92 Plus:**
Tryck på [◆] [EE] för att öppna tangentbordskartan.	Tryck på [◆] [KEY] för att öppna tangentbordskartan.
Tryck [ESC] för att stänga kartan.	Tryck [ESC] för att stänga kartan.
 **TI-89 tangenbordskartan**	 **TI-92 Plus tangenbordskartan**
När du vill använda någon av genvägarna trycker du först på tangenten [◆].	När du vill använda någon av genvägarna trycker du först på tangenten [2nd]. Vissa är markerade på tangentbordet men de flesta är det inte.

Räknarfunktioner som du kan komma åt via tangentbordskartan diskuteras vidare på nästa sida.

Genvägar på TI-89 tangentbordskarta:	Genvägar på TI-92 Plus tangentbordskarta:
GREEK ([◆] [(]) — Öppnar uppsättningen med grekiska tecken (beskrivs senare i detta avsnitt).	GREEK ([2nd] G) — Öppnar uppsättningen med grekiska tecken (beskrivs senare i detta avsnitt).
SYSDATA ([◆] [;]) — Kopierar aktuella grafkoordinater till systemvariabeln sysdata.	CAPS ([2nd] [CAPS])— Använder för att aktivera/avaktivera Caps Lock.
FMT ([◆] [I]) — Öppnar dialogrutan FORMATS.	Accent marks — (é, ü, ô, à, ç, och ~) läggs till *nästa* tangent du trycker på (beskrivs senare i detta avsnitt).
KBDPRGM1 – 9 ([◆]1 till [◆]9) — Om du har användardefinierade program eller assemblerprogram med namnen kbdprgm1() till kbdprgm9() startar dessa genvägar respektive program.	
OFF ([◆] [OFF]) — Liknande [2nd] [OFF] förutom: • Om du använder [◆] [OFF], startar räknaren alltid med grundfönstret. • Stänger av din TI-89 så att den återgår till senast använda program då du slår på den igen.	
HOMEDATA ([◆] [(-)]) — Kopierar aktuella grafkoordinater till grundfönstrets historiklista.	

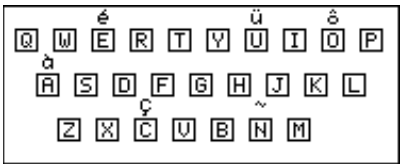
Skriva in specialtecken via tangentbordet

På TI-89:	På TI-92 Plus:
Tryck på [◆] och därefter på tangenten för den symbolen. Exemple: [◆] [×] (gånger) visar &.	Tryck på [2nd] och därefter på tangenten för den symbolen. Exemple: [2nd] H visar &.
	
Dessa specialsymboler påverkas inte av inställningen för alpha-lock.	Dessa specialsymboler påverkas inte av inställningen för Caps Lock.

Obs! *I denna teckentabell visas endast specialsymbolerna.*

Skriva accentmarkeringstecken från tangentbordet på TI-92 Plus

När du trycker på en accentmarkeringstangent visas inte ett tecken med accent. Accentmarkeringen läggs till *nästa* bokstav du trycker på.



Obs! I denna teckentabell visas endast tangenterna för accentmarkeringar.

1. Tryck på [2nd] och sedan på tangenten för accentmarkeringen.
2. Tryck på tangenten för den bokstav du vill ha accent för.
   - Du kan göra både gemena och versala bokstäver med accent.
   - Du kan bara lägga till en accentmarkering till de bokstäver som är giltiga för den markeringen.

Accent-markering	Giltiga bokstäver (gemener eller versaler)	Exempel
´	A, E, I, O, U, Y	é, É
¨	A, E, I, O, U, y (men inte Y)	ü, Ü
ˆ	A, E, I, O, U	ô, Ô
`	A, E, I, O, U	à, À
ç	C	ç, Ç
~	A, O, N	ñ, Ñ

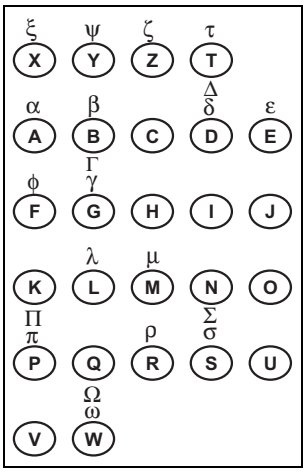
Skriva in grekiska tecken via tangentbordet

Tryck för att öppna den grekiska teckenuppsättningen. Välj därefter motsvarande alfatecken på tangentbordet för att skriva in ett grekiskt tecken.

***Obs!** Ingen av räknarna visar en karta med grekiska tecken. Kartorna här är bara för referens.*

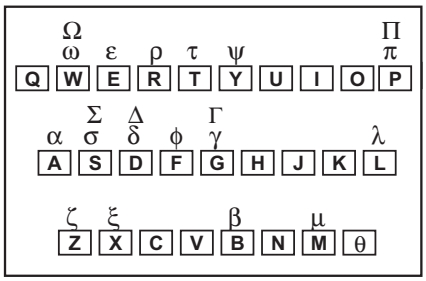
På TI-89:

Tryck på [◆][(] för att öppna den grekiska teckenuppsättningen.



På TI-92 Plus:

Tryck på [2nd] G för att öppna den grekiska teckenuppsättningen.



Om du trycker ned en tangent som inte har något grekiskt tecken skrivs det normala tecknet för den tangenten ut.

Du kan skriva versala och gemena grekiska bokstäver med flera tangenter. Exempel:

På TI-89:	**På TI-92 Plus:**
1. Tryck på [◆] [(] när du vill använda grekiska tecken.	1. Tryck på [2nd] G när du vill använda grekiska tecken.
2. Tryck på [◆] [(] [alpha] + bokstav när du vill använda gemena grekiska tecken. Exempel: [◆] [(] [alpha] [W] ger ω	2. Tryck på [2nd] G + bokstav när du vill använda gemena grekiska tecken. Exempel: [2nd] G W ger ω
3. Tryck på [◆] [(] [↑] + bokstav när du vill använda versala grekiska tecken. Exempel: [◆] [(] [↑] [W] ger Ω	3. Tryck på [2nd] G [↑] + bokstav när du vill använda versala grekiska tecken. Exempel: [2nd] G [↑] W ger Ω

Exakt vilken tangent du ska trycka ned beror på om alfalåsningen är av eller på. Till exempel:

På TI-89, om:	**Tryck på:**
Alfalåsning är av.	[◆] [(] X eller [◆] [(] [alpha] X ger ξ. [alpha] behövs inte för X, Y, Z eller T. [◆] [(] [alpha] W ger ω. [◆] [(] [↑] W ger Ω. [↑] används för versaler.
Alfalåsning för gemener ([2nd] [a-lock]) är på.	[◆] [(] X ger ξ. [◆] [(] W ger ω. [◆] [(] [↑] W ger Ω.
Alfalåsning för versaler ([↑] [a-lock]) är på.	[◆] [(] X ger ξ. [◆] [(] W ger Ω. [◆] [(] [↑] W ger Ω.

***Viktigt!** Om du trycker ned [alpha] för att skriva in ett grekiskt tecken när alfalåsning är på så stängs alfalåsningen av.*

Lista över alla specialtecken

I appendix B finns en lista över alla specialtecken.

Mata in och köra ett kommandoskript

Genom att använda ett kommandoskript kan du använda Text Editor för att skriva en serie kommandorader som kan köras närhelst du vill i grundfönstret. Det gör att du kan skapa interaktiva exempelskript där du definierar en serie kommandon och sedan kör dem individuellt.

Infoga ett kommandomärke

Gör följande i Text Editor:

1. Placera markören på raden för kommandot.
2. Tryck på [F2] för att visa menyn Command.

3. Välj 1:Command.

   "C" visas i början av textraden (till vänster om kolonet).

4. Skriv ett kommando på samma sätt som i grundfönstret.

   Raden kan bara innehålla kommandot, ingen extra text.

   Du kan skriva flera kommandon på samma rad om du använder kolon för att avgränsa kommandona.

Obs! *Detta infogar inte en ny rad för kommandot, det markerar bara en befintlig rad som en kommandorad.*

Tips! *Du kan markera en rad som ett kommando innan eller efter det att du skriver kommandot på den raden.*

Ta bort ett kommandomärke

Detta tar bara bort märket "C", inte själva kommandotexten.

1. Placera markören någonstans på den markerade raden.
2. Tryck på [F2] och välj 4:Clear command.

Köra ett kommando

För att kunna köra ett kommando måste du först markera raden med ett "C". Om du kör en rad som inte är markerad med "C", kommer den att ignoreras.

1. Placera markören någonstans på kommandoraden.
2. Tryck på [F4].

Kommandot kopieras till inmatningsraden i grundfönstret och körs. Grundfönstret visas tillfälligt under körningen och därefter visas Text Editor igen.

Efter körningen flyttar markören till nästa rad i skriptet så att du kan fortsätta köra en serie med kommandon.

Tips! *Om du vill visa resultaten i grundfönstret trycker du på:*
TI-89: [HOME]
TI-92 Plus: [♦] [HOME]
eller använder ett delat fönster.

Dela Text Editor och grundfönstret

Med ett delat fönster kan du visa kommandoskriptet och se resultatet av ett utfört kommando på samma gång.

Om du vill:	Trycker du på:
Dela fönstret	F3 och väljer 1:Script view.
Återgå till Text Editor i full storlek	F3 och väljer 2:Clear split.

Du kan också använda MODE för att ställa in ett delat fönster manuellt. Det är dock mycket enklare att ställa in Text Editor och grundfönstret i delat fönster med F3 än med MODE.

- Det aktiva programmet anges av en tjock ram. (Som standard är Text Editor det aktiva programmet.)
- Du växlar mellan Text Editor och grundfönstret genom att trycka på 2nd [+⊡] (sekundärfunktion av APPS).

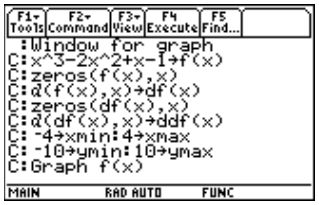
Skapa ett skript från inmatningarna i grundfönstret

Från grundfönstret kan du spara alla inmatningar i historiklistan till en textvariabel. Inmatningarna sparas automatiskt i ett skriptformat så att du kan öppna textvariabeln i Text Editor och köra inmatningarna som kommandon.

Mer information finns i "Spara inmatningar i grundfönstret som ett Text Editor-skript" i kapitel 5.

Exempel

1. Skriv skriptet. Tryck på F2 och välj 1:Command för att markera kommandoraderna.
2. Tryck på F3 och välj 1:Script view.



3. Flytta markören till den första kommandoraden. Tryck därefter på F4 för att köra kommandot.

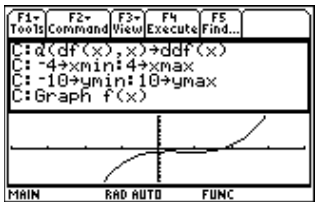
__Obs!__ Vissa kommandon tar längre tid att köra. Vänta tills indikatorn Busy försvinner innan du trycker på F4 igen.

4. Fortsätt att använda F4 för att köra varje kommando, men stanna innan du kör kommandot **Graph**.

5. Kör kommandot **Graph**.

*__Obs!__ I detta exempel visar kommandot **Graph** graffönstret i stället för grundfönstret.*

6. Tryck på F3 och välj 2:Clear split för att återgå till Text Editor i fullständig storlek.



Skapa en labbrapport

Om du har TI-GRAPH LINK™, ett tilläggsprogram som gör att räknaren kan kommunicera med en persondator, kan du skapa labbrapporter. Använd Text Editor för att skriva en rapport, som kan innehålla utskriftsobjekt. Använd därefter TI-GRAPH LINK för att skriva ut rapporten på den skrivare som är ansluten till datorn.

Utskriftsobjekt

I Text Editor kan du ange ett variabelnamn som ett utskriftsobjekt. När du skriver ut rapporten med hjälp av TI-GRAPH LINK, ersätts innehållet i variabeln (ett uttryck, en bild, en lista osv) av variabelnamnet.

Infoga ett utskriftsmärke

Gör följande i Text Editor:

1. Placera markören på raden för utskriftsobjektet.
2. Tryck på [F2] för att visa menyn Command.

3. Välj 3:PrintObj.

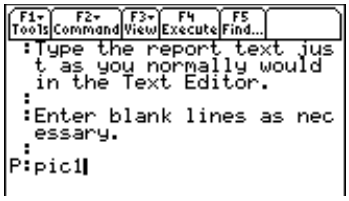
   "P" visas i början av textraden (till vänster om kolonet).

__Obs!__ Detta infogar inte en ny rad för utskriftsobjektet, det markerar bara en befintlig rad som ett utskriftsobjekt.

__Tips!__ Du kan markera en rad som ett utskriftsobjekt innan eller efter du skriver ett variabelnamn på den raden.

4. Skriv namnet på variabeln som innehåller utskriftsobjektet.

   Raden kan endast innehålla variabelnamnet, ingen extra text.



Infoga ett sidbrytnings-märke

När du skriver ut en labbrapport bryts sidorna automatiskt längst ned på varje utskriven sida. Du kan dock manuellt göra en sidbrytning var som helst.

1. Placera markören på raden som du vill skriva ut överst på nästa sida. (Raden kan vara tom eller så kan du skriva text på den.)
2. Tryck på [F2] och välj 2:Page break.

Symbolen "⤒" visas i början av raden (till vänster om kolonet).

Ta bort ett utskrifts- eller ett sidbrytningsmärke

Detta tar endast bort märket "P" eller "⤒", inte text som finns på raden.

1. Placera markören någonstans på den markerade raden.
2. Tryck på [F2] och välj 4:Clear command.

Skriva ut rapporten

Allmänna steg	Om du behöver mer detaljerad Information
1. Anslut TI-89 / TI-92 Plus till datorn via TI-GRAPH LINK.	Se bruksanvisningen som levererades med TI-GRAPH LINK.
2. Använd TI-GRAPH LINK för att hämta labbrapporten från räknaren och skriv sedan ut rapporten.	

Exempel

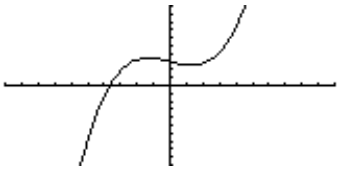
Anta att du har sparat:

- En funktion som y1(x) (ange y1, inte y1(x)).
- En grafbild som pic1.
- Lämplig information i variablerna der och sol.

När du skriver ut labbrapporten skrivs innehållet i utskriftsobjekten ut i stället för deras variabelnamn.

```
 :My assignment was to stu
  dy the function:
 :
P:y1
 :
 :The three parts were:
 :
 :1. Graph the function.
 :
P:pic1
 :
 :2. Find its derivative.
 :
P:der
 :
 :3. Look for critical poi
  nts.
 :
P:sol
```

```
My assignment was to study the function:
.1*x^3-.5*x+3
The three parts were:
1. Graph the function.
```



```
2. Find its derivative.
.3*x^2-.5
3. Look for critical points.
x=1.29099 or x=-1.29099
```

Obs! *Om du vill spara derivatan till variabeln der skriver du:* ***d***(y1(x),x)→ der

Obs! *Om du vill spara derivatans nollställen till variabeln sol skriver du:* **solve**(der=0,x)→ sol

Om grafbilden inte får plats på den aktuella sidan flyttas hela bilden överst till nästa sida.

Ekvationslösaren

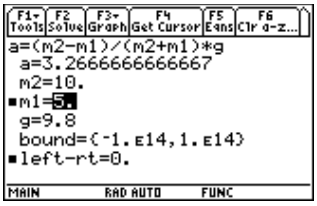
19

Förhandsvisning av den numeriska ekvationslösaren 334
Öppna Solver och skriv in en ekvation 335
Definiera kända variabler 337
Lösa för den okända variabeln 339
Plotta lösningen..... 340

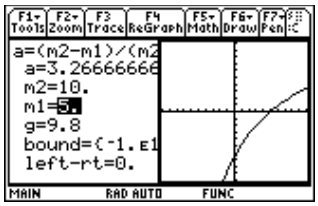
Obs! *Om du vill lösa en ekvation för en okänd variabel från grundfönstret eller ett program använder du* **nSolve()** *enligt beskrivningen i appendix A.*

I ekvationslösaren kan du skriva in ett uttryck eller en ekvation, definiera värden för alla variabler utom en och sedan lösa ekvationen för den okända variabeln.

När du har skrivit in en ekvation och dess kända variabler flyttar du markören till den okända variabeln och trycker på F2.



Du kan även plotta lösningen.



X-axeln är den okända variabeln. Y-axeln visar värdet för left–rt vilket bestämmer lösningens noggrannhet.

Lösningen är exakt där kurvan korsar x-axeln.

Som i exemplet ovan används ekvationslösaren oftast för att lösa ekvationer i sluten form, men du kan också enkelt lösa t ex transcendentala ekvationer som inte är slutna.

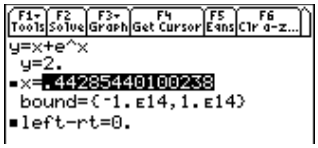
Du kan t ex manuellt arrangera om nedanstående ekvation för att lösa den för valfri variabel.

a = (m2 − m1) / (m2 + m1) ∗ g ⟶ m1 = (g − a) / (g + a) ∗ m2

En ekvation som nedanstående är inte lika lätt att manuellt lösa för x.

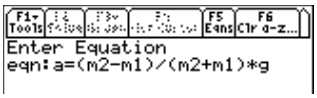
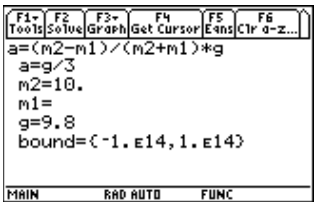
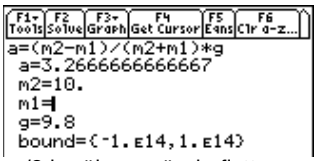
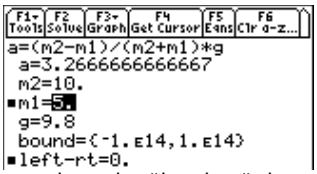
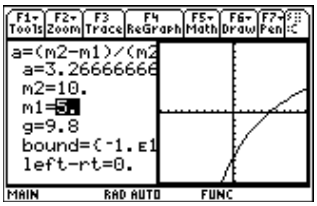
$y = x + e^x$

Ekvationslösaren är särskilt användbar för liknande ekvationer.



Förhandsvisning av den numeriska ekvationslösaren

Anta att vi har ekvationen a=(m2− m1)/(m2+m1)∗ g där de kända värdena är m2=10 och g=9.8. Beräkna värdet av m1 om a=1/3 g.

Steg	TI-89 Tangenter	TI-92 Plus Tangenter	Teckenfönster
1. Öppna Ekvationslösaren.	APPS 9	APPS 9	
2. Skriv in ekvationen. *När du trycker* ENTER *eller* ⊖ *listas variablerna som används i ekvationen.*	alpha A = (alpha M 2 − alpha M 1) ÷ (alpha M 2 + alpha M 1) × alpha G ENTER	A = (M 2 − M 1) ÷ (M 2 + M 1) × G ENTER	
3. Skriv in värden för varje variabel utom den vi vill lösa ekvationen för (m1). *Definiera* m2 *och* g *först. Definiera sedan* a. *(Du måste definiera* g *innan du kan definiera* a *i termer av* g.*) Godkänn förvalt värde för* bound. *Om en variabel har definierats tidigare så visas dess värde som förval.*	⊖ 1 0 ⊖ ⊖ 9 . 8 ⊕ ⊕ ⊕ alpha G ÷ 3	⊖ 1 0 ⊖ ⊖ 9 . 8 ⊕ ⊕ ⊕ G ÷ 3	
4. Flytta markören till den okända variabeln m1. *Du kan i förväg ange en gissning för värdet av* m1. *Även om du anger värden för alla variabler så löser Ekvationslösaren ekvationen för den variabel som markeras av markören.*	⊖ ⊖	⊖ ⊖	 g/3 beräknas när du flyttar markören från linjen.
5. Lös ekvationen för den okända variabeln. *För att kontrollera lösningens noggrannhet utvärderas varje sida av ekvationen separat. Differensen visas som* left− rt. *Om lösningen är exakt visas* left− rt=0.	F2	F2	 ▪ markerar beräknade värden.
6. Plotta lösningen i visningsfönstret ZoomStd. *Grafen visas i ett delat fönster. Du kan utforska grafen vidare med spårnng, zoomning etc.*	F3 3	F3 3	
7. Återgå till Ekvationslösaren och stäng det delade fönstret. *Tryck* ENTER *eller* ⊖ *om du vill se en lista över variablerna.*	2nd [⊞] F3 2	2nd [⊞] F3 2	Variabeln som markeras av markören (okänd variabel m1) visas på x-axeln medan left−rt visas på y-axeln.

Öppna Solver och skriv in en ekvation

Efter du har öppnat Ekvationslösaren börjar du med att skriva in den ekvation du vill lösa.

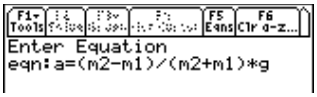
Öppna Ekvationslösaren

Öppna Ekvationslösaren genom att trycka på [APPS] 9.

I fönstret för Ekvationslösaren visas den senast inskrivna ekvationen - om det finns någon sådan.

Skriva in ekvationen

Skriv in ekvationen på raden **eqn:**.



Tips! *I din ekvation:*

- *Använd inte namn för systemfunktioner* (t ex y1(x) eller r1(θ)) *för enkla variabler* (y1 eller r1).
- *Var försiktig med implicit multiplikation. Till exempel betraktas* a(m2+m1) *som en funktionsreferens, inte som* a∗ (m2+m1).

Obs! *När du definierar variabler kan du antingen definiera* exp *eller lösa ekvationen för det.*

Obs! *När du tryckt* [ENTER] *sparas den aktuella ekvationen automatiskt till systemvariabeln* eqn.

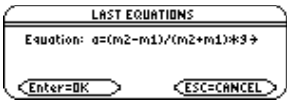
Du kan:	Till exempel:
Skriv in en ekvation.	a=(m2− m1)/(m2+m1)∗ g a+b=c+sin(d)
Referera till en funktion eller ekvation som är definierad på annan plats.	Anta att du definierat y1(x) i antingen: • Y= Editor: y1(x)=1.25x∗ cos(x) – eller – • Grundfönstret: Define y1(x)=1.25x∗ cos(x) I Ekvationslösaren skriver du sedan: y1(x)=0 eller y1(t)=0, etc. Argumenten behöver inte matcha de som användes vid definitionen av funktionen eller ekvationen.
Skriv in en ekvation utan tecknet =.	e+f− ln(g) Efter du trycker [ENTER] sätts ekvationen lika med systemvariabeln exp och skrivs som: exp=e+f− ln(g)
Öppna en tidigare inskriven ekvation eller en sparad ekvation.	Se mer information senare i detta kapitel.

Öppna tidigare inskrivna ekvationer

De senaste ekvationer du skrivit in (upp till 11 med förvald inställning) sparas i minnet. Om du vill öppna en av dessa ekvationer:

1. Tryck [F5] i fönstret Ekvationslösaren.

   En dialogruta öppnas med de senast inskrivna ekvationerna.



2. Välj en ekvation.
   - Välj markerad ekvation genom att trycka [ENTER].
   - Om du vill välja en annan ekvation trycker du ⊙ för att öppna en lista där du kan välja tidigare ekvationer.

Bara unika ekvationer visas. Om du skriver in samma ekvation 5 gånger så kommer den ändå bara att listas en gång.

3. Tryck [ENTER].

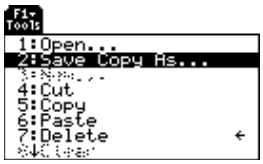
Tips! *Du kan ange hur många ekvationer som ska sparas. I Ekvationslösaren, tryck på* [F1] *och välj* 9:Format (*eller använd* ***TI-89:*** [◆] [|] ***TI-92 Plus:*** [◆] F). *Välj antal mellan 1 och 11.*

Spara ekvationer för framtida bruk

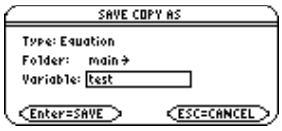
Eftersom antalet ekvationer som du kan öppna med [F5] Eqns är begränsat kan det inträffa att tidigare ekvationer inte finns tillgängliga.

När du vill spara aktuell ekvation till senare sparar du den som en variabel.

1. I fönstret Ekvationslösaren trycker du [F1] och väljer 2:Save Copy As.



2. Välj en katalog och ett variabelnamn för ekvationen.
3. Tryck [ENTER] två gånger.

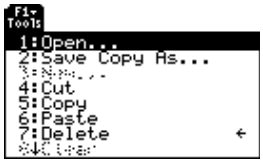


Obs! *En ekvation i formen av en variabel har datatypen EXPR som visas i fönstren MEMORY och VAR-LINK.*

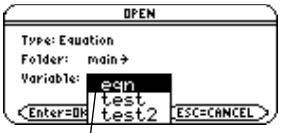
Öppna en sparad ekvation

När du vill öppna en tidigare sparad ekvation:

1. I fönstret Ekvationslösaren trycker du [F1] och väljer 1:Open.



2. Välj katalog och ekvationsvariabel.
3. Tryck [ENTER].



Variabeln eqn innehåller den aktuella ekvationen. Den listas alltid i bokstavordning i listan.

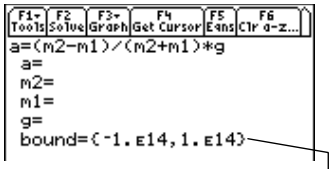
Definiera kända variabler

Efter du har skrivit in en ekvation i Ekvationslösaren anger du värdet för alla variabler utom den okända.

Definiera lista med variabler

Efter du har skrivit in ekvationen på raden **eqn:**, tryck [ENTER] eller ⊖.

På skärmen listas alla variabler i den ordning de ingår i ekvationen. Om en variabel redan är definierad visas dess värde i listan. Du kan redigera även dessa variabelvärden.



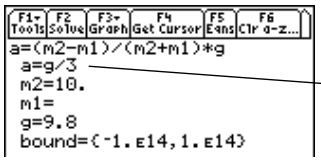
Lösningen måste vara inom angivna gränser. Du kan ändra dessa efter behov.

Obs! *Om en existerande variabel är låst eller sparad till användarminnet kan du inte redigera dess värde.*

Skriv in ett värde eller ett uttryck för alla variabler utom den du vill lösa för.

Anteckningar och vanliga fel

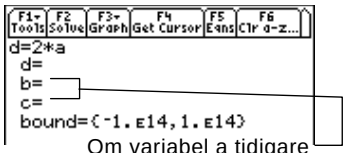
- Om du definierar en variabel:
  - I termer av en annan variabel i ekvationen måste denna variabel först vara definierad.
  - I termer av en annan variabel som inte återfinns i ekvationen så måste denna variabel redan ha ett värde. Den kan inte vara odefinierad.
  - Uttryck utvärderas när du flyttar markören till nästa rad. Uttrycket måste lösas till ett reellt tal.



Eftersom a är definierad i termer av g måste g definieras före a. När du flyttar markören från raden utvärderas g/3.

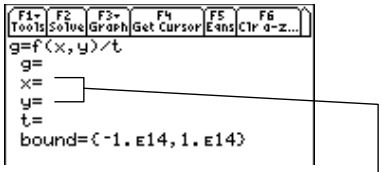
Obs! *När du anger ett värde för en variabel i Ekvationslösaren definieras variabeln globalt. Den existerar även efter det att du har lämnat Ekvationslösaren.*

- Om en ekvation innehåller en variabel som redan definierats i termer av andra variabler listas dessa variabler.



Om variabel a tidigare definierats som b+c→a så listas b och c i stället för a.

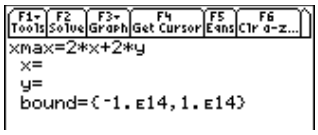
- Om du refererar till en tidigare definierad funktion listas alla variabler för funktionsanropet, inte variablerna som definierar funktionen.



Om f(a,b) tidigare definierats som √(a^2+b^2) och din ekvation innehåller f(x,y) så listas x och y i stället för a och b.

Obs! *Du kan inte lösa ekvationer för andra systemvariabler än* exp. *Om ekvationen innehåller en systemvariabel kan du inte heller använda* F3 *för att plotta den.*

- Om ekvationen innehåller en systemvariabel (xmin, xmax, etc.) listas inte den variabeln. Ekvationslösaren använder systemvariabelns aktuella värde.

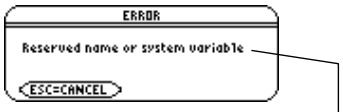


I det vanliga visningsfönstret, xmax=10.

- Även om du kan använda systemvariabler i ekvationer så uppstår ett fel om du använder F3 för att plotta lösningen.

Obs! *Detta fel uppstår om du använder ett reseverat namn felaktigt eller refererar till en odefinierad systemfunktion som en enkel variabel utan parenteser.*

- Om felmeddelandet till höger visas tar du bort värdet ur variabeln och redigerar sedan ekvationen så att den använder en annan variabel.



Till exempel y1(x) är odefinerat och du använder y1.

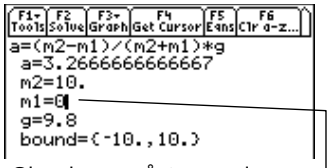
Redigera en ekvation

I fönstret Ekvationslösaren trycker du först ⊕ tills markören befinner sig över ekvationen. Skärmen ändras automatiskt så att bara raden **eqn:** visas. Genomför dina ändringar och tryck sedan ENTER eller ⊖ för att återgå till listan med variabler.

Ange en gissning och/eller gränser (valfritt)

Om du vill hitta en lösning snabbare eller om du vill hitta en särskild lösning där det kan finnas flera alternativ kan du:

- Ange en gissning för värdet av den okända variabeln. Gissningen måste ligga inom angivna gränser.
- Ange under och övre gräns nära den förväntade lösningen.



Gissningar måste vara inom gällande gränser.

Tips! *Information om hur du anger en gissning grafiskt finns på sidorna 340 och 341.*

Som gränser kan du även ange variabler eller uttryck som utvecklas till giltiga värden (bound={*lower*,*upper*}), eller listvariabler med två element (bound=*list*). Gränserna måste vara två flyttal med det första värdet mindre än, eller lika med, det andra värdet.

Lösa för den okända variabeln

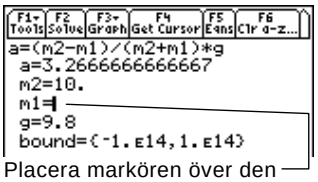
Efter du har skrivit in en ekvation i Ekvationslösaren och angett värden för de kända variablerna är det klart för att beräkna den okända variabeln.

Hitta lösningen

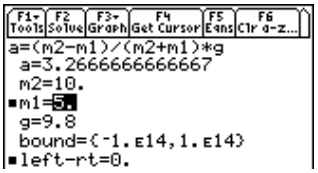
Med alla kända variabler definierade:

1. Flytta markören till den okända variabeln.
2. Tryck [F2] Solve.

   Tecknet ▪ markerar lösningen och left–rt. Tecknet ▪ försvinner då du ändrar ett värde, flyttar markören till ekvationen eller lämnar Ekvationslösaren.

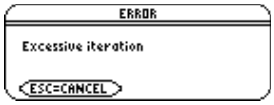


Placera markören över den variabel du vill lösa.



Obs! *Om du vill avbryta en beräkning trycker du [ON]. Den okända variabeln visar det värde som testades då beräkningen stoppades.*

Utifrån lösningen och de värden du angett utvärderas vänster och höger sida av ekvationen separat. left– rt visar differensen, vilket indikerar lösningens noggrannhet. Ju lägre värde desto nogrannare lösning. Om en lösning är exakt är left–rt=0.

Om du vill:	Gör så här:
Lösa ekvationen för andra värden.	Ändra ekvationen eller värden för variablerna.
Hitta en annan lösning för en ekvation med flera lösningar.	Ange en gissning och/eller gränser som ligger nära den förväntade lösningen.
Visa meddelandet: 	Tryck [ESC]. Den okända variabeln visar det värde som testades då felet uppstod. • Värdet för left– rt kan vara tillräckligt litet för att du ska acceptera lösningen. • Om värdet inte är tillräckligt litet anger du nya gränser.

Obs! *En iterativ process används för att lösa en ekvation. Om den iterativa processen inte kan hitta en lösning visas följande felmeddelande.*

Plotta lösningen

Du kan plotta en ekvations lösningar när som helst efter det att du har angett de kända variablerna och före eller efter att du beräknat den okända variabeln. Genom att plotta lösningarna kan du se hur många lösningar som existerar och använda markören för att ange en gissning eller gränser.

Visa grafen

Flytta markören till den okända variabeln i Ekvationslösaren. Tryck [F3] och välj:

1:Graph View
– eller –
3:ZoomStd
– eller –
4:ZoomFit

Graph View använder aktuella Window-variabler.

Mer information om ZoomStd och ZoomFit finns i kapitel 6.

***Tips!** Med delat fönster:*

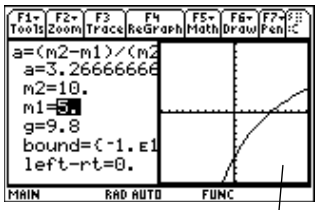
- *Använd [2nd][⊡] för att växla mellan sidor.*
- *Den aktiva sidan har en tjock ram.*
- *Verktygsfältet hör till den aktiva sidan.*

Mer information hittar du i kapitel 14.

Grafen visas i ett delat fönster där:

- Den okända variabeln plottas längs x-axeln.
- left– rt plottas längs y-axeln.

Lösningar för ekvationen finns vid left–rt=0 dvs när grafen skär x-axeln.



Aktuella inställningar för grafformat används.

Du kan utforska grafen vidare med markören, spårning, zoomning etc. Mer information hittar du i kapitel 6.

Hur grafen påverkar olika inställningar

När du använder Ekvationslösaren för att plotta en graf:

- Följande lägen ändras automatiskt till nedanstående inställningar:

***Obs!** Om du tidigare använde andra lägesinställningar så måste du återställa dessa manuellt.*

Läge	Inställning
Graph	FUNCTION
Split Screen	LEFT-RIGHT
Number of Graphs	1

Funktioner valda i Y= Editor plottas inte.

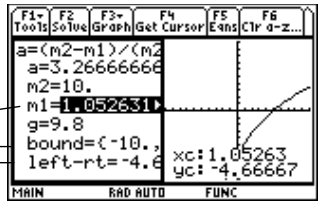
- Alla statistik-plottningar avmarkeras.
- Efter du har lämnat Ekvationslösaren kan plottningsfönstret fortsätta att visa ekvationens lösning och ignorera valda funktioner i Y=Editor. Om detta inträffar öppnar du Y= Editor och växlar sedan till plottningsfönstret. Grafen återställs också om du ändrar läge för Graph eller anropar **ClrGraph** från grundfönstret ([F4] 5) eller från ett program.

Ange en ny gissning från grafen

__Obs!__ Markörkoordinaten xc *är värdet hos den okända variabeln och* yc *är det vänstra* left– rt *värdet.*

När du vill använda grafmarkören för att ange en gissning:

1. Flytta markören (i fritt läge eller spårning) till den punkt som du vill använda värdet för en ny gissning.
2. Använd [2nd] [⊞] för att aktivera fönstret Ekvationslösaren.
3. Kontrollera att markören är över den okända variabeln och tryck [F4].
4. Tryck [F2] för att lösa ekvationen igen.



[F4] ställer in grafmarkörens xc-värde som gissning och yc-värdet som left–rt. Grafens värden för xmin och xmax anges som gränsvärden.

Återgå till fullskärm

Från den delade skärmen:

- Om du vill använda Ekvationslösaren med fullskärm använder du först [2nd] [⊞] för att aktivera Solver-fönstret. Tryck sedan [F3] och välj slutligen 2:Clear Graph View.

  – eller –

- Om du vill visa grundfönstret trycker du [2nd][QUIT] två gånger.

Radera variabler innan du lämnar Ekvationslösaren

__Tips!__ När du vill radera variabelnamn med ett tecken som listas i Ekvationslösaren kan du använda:
__TI-89:__ [2nd] [F6]
__TI-92 Plus:__ [F6]

När du har löst en ekvation kommer dess variabler att finnas kvar även när du har lämnat Ekvationslösaren. Om ekvationen innehåller variabler med namn som består av bara ett tecken kan dessa variabler påverka senare symboliska beräkningar. Innan du lämnar Ekvationslösaren vill du därför kanske radera alla variabler med namn som bara innehåller ett tecken från den aktuella katalogen:

1. Tryck:
   TI-89: [2nd] [F6]
   TI-92 Plus: [F6]
2. Tryck [ENTER] för att bekräfta åtgärden.

Skärmen återgår till raden **eqn:** i Ekvationslösaren.

Talsystem

20

Talsystem 344
Skriva in och konvertera talsystem 345
Beräkningar med binära och hexadecimala tal..... 346
Jämföra eller manipulera enskilda bitar 347

Obs! *Från den nya menyn* MATH/Base *kan du välja ett antal operationer som är relaterade till olika talsystem.*

När du skriver in ett heltal på din TI-89 / TI-92 Plus kan du skriva in det i decimal, binär eller hexadecimal form. Du kan också ange ett läge för Base för att ange i vilket talsystem resultat ska visas. Bråktal och decimaltal visas alltid i decimal form.

Binära värden består av 0 och 1 med basen 2:

100

$2^0 * 0 = +0$
$2^1 * 0 = +0$
$2^2 * 1 = +4$

Hexadecimala värden består av 0 – 9 och A – F med basen 16:

A8F

$16^0 * F = +15$
$16^1 * 8 = +128$
$16^2 * A = +2560$

Dec Bas 10	Bin Bas 2	Hex Bas 16
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F
16	10000	10

Du kan använda din TI-89 / TI-92 Plus för att konvertera ett värde från ett talsystem till ett annat. Till exempel, 100 binärt = 4 decimalt och A8F hex = 2703 decimalt.

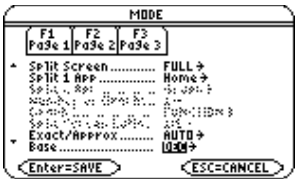
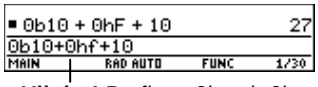
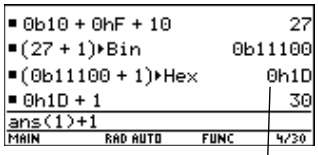
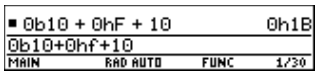
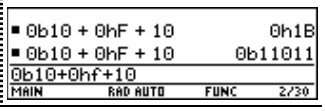
Hexadecimala tal används ofta för att presentera längre binära värden, som är svåra att komma ihåg, som t ex:

1010 1111 0011 0111
A F 3 7

Den hexadecimala formen AF37 är lättare att arbeta med än motsvarande binära 1010111100110111.

Med TI-89 / TI-92 Plus kan du också jämföra eller manipulera binära värden bit för bit.

Beräkna 10 binärt (bas 2) + F hexadecimalt (bas 16) + 10 decimalt (bas 10). Använd sedan operatorn ▸ för att konvertera ett heltal från ett talsystem till ett annat. Avsluta med att se hur förändringar av Base påverkar de visade resultaten.

Steg	TI-89 Tangenter	TI-92 Plus Tangenter	Teckenfönster
1. Öpnna dialogrutan MODE (sidan 2). I inställningen för Base väljer du DEC som aktuellt talsystem. *Heltalsresultat visas i aktuellt läge för* Base. *Bråktal och decimaltal visas alltid i decimal form.*	[MODE] [F2] (använd ⊖ när du vill flytta till läget Base) ⊕ 1 [ENTER]	[MODE] [F2] (använd ⊖ när du vill flytta till läget Base) ⊕ 1 [ENTER]	
2. Beräkna 0b10+0hF+10. *När du vill skriva in ett binärt eller hexadecimalt tal måste du alltid ange* 0b *eller* 0h *framför talet (nolla samt bokstaven* B *alternativt* H*). Om du inte gör detta betraktas talet som ett decimalt tal.*	0 [alpha] B 1 0 [+] 0 [2nd] [a-lock] H F [alpha] [+] 1 0 [ENTER]	0 B 1 0 [+] 0 H F [+] 1 0 [ENTER]	 **Viktigt!** Prefixen 0b och 0h börjar med en nolla, inte bokstaven O och följs sedan av B eller H.
3. Lägg till 1 till resultatet och konvertera det till ett binärt värde. [2nd] [▸] *öppnar konverterings-operatorn* ▸.	[+] 1 [2nd] [▸] [2nd] [a-lock] B I N [alpha] [ENTER]	[+] 1 [2nd] [▸] B I N [ENTER]	
4. Lägg till 1 till resultatet och konvertera det till ett hexadecimalt värde.	[+] 1 [2nd] [▸] [2nd] [a-lock] H E X [alpha] [ENTER]	[+] 1 [2nd] [▸] H E X [ENTER]	
5. Lägg till 1 till resultatet och visa det i decimal form utan konvertering.	[+] 1 [ENTER]	[+] 1 [ENTER]	 Resultaten börjar med 0b eller 0h för att ange aktuellt talsystem.
6. Ändra inställningen för Base till HEX. *När* Base = HEX *eller* BIN *gäller vissa begränsningar för storleken på resultatet. Se mer på sidan 346.*	[MODE] [F2] (använd ⊖ för att flytta till Base) ⊕ 2 [ENTER]	[MODE] [F2] (använd ⊖ för att flytta till Base) ⊕ 2 [ENTER]	
7. Beräkna 0b10+0hF+10.	0 [alpha] B 1 0 [+] 0 [2nd] [a-lock] H F [alpha] [+] 1 0 [ENTER]	0 B 1 0 [+] 0 H F [+] 1 0 [ENTER]	
8. Ändra inställningen för Base till BIN.	[MODE] [F2] (använd ⊖ för att flytta till Base) ⊕ 3 [ENTER]	[MODE] [F2] (använd ⊖ för att flytta till Base) ⊕ 3 [ENTER]	
9. Skriv in 0b10+0hF+10 igen.	[ENTER]	[ENTER]	

Oberoende av inställningen för Base måste du alltid ange prefix när du skriver in tal i binär och hexadecimal form.

Skriva in ett binärt eller hexadecimalt tal

När du vill skriva in ett binärt tal använder du formen:

0b*binärtTal* (till exempel: 0b11100110)

- *binärtTal*: Binärt tal med upp till 32 tecken
- 0b: Noll, inte bokstaven O följt av bokstaven b

Obs! *Du kan skriva in b alt. h i prefixet, samt hexadecimala tecknen* A – F, *med gemener eller versaler. Resultatet är detsamma.*

När du vill skriva in ett hexadecimalt tal använder du formen:

0h*hexadecimaltTal* (till exempel: 0h89F2C)

- *hexadecimaltTal*: Hexadecimalt tal med upp till 8 tecken
- 0h: Noll, inte bokstaven O följt av bokstaven h

Om du skriver in ett tal utan prefixet 0b eller 0h, som t ex 11, antas det alltid vara ett decimalt tal. Om du utesluter prefixet 0h men använder de hexadecimala tecknen A – F hanteras hela talet som ett variabelnamn.

Konvertera mellan talsystem

Använd konverteringsoperatorn ▸.

heltalsUttryck ▸Bin
heltalsUttryck ▸Dec
heltalsUttryck ▸Hex

När du vill skriva ▸, tryck [2nd] [▸]. Du kan också välja konvertering mellan talsystem via menyn MATH/Base.

Obs! *Om talet inte är ett heltal visas ett felmeddelande.*

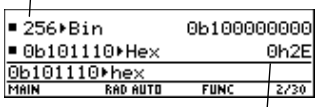
Konvertera 256 från decimalt till binärt talsystem:

256▸Bin

Konvertera 101110 från binärt till hexadecimalt talsystem:

0b101110▸Hex

För binära och hexadecimala tal måste du ange prefixet 0b alt. 0h.



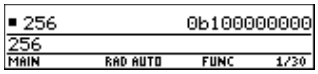
Resultaten börjar med 0b eller 0h för att ange aktuellt talsystem.

Alternativa metoder för konverteringar

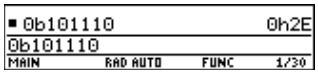
I stället för att använda ▸ kan du:

1. Använd [MODE] (sidan 346) för att ange inställning för Base till det talsystem du vill konvertera till.
2. Skriv in det värde du vill konvertera (med rätt prefix) i grundfönstret och tryck [ENTER].

Om Base = BIN:



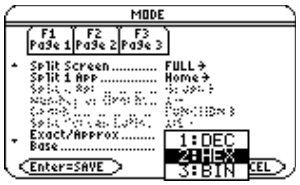
Om Base = HEX:



Alla operationer som använder heltal kan även användas med hexadecimala och binära tal. Resultaten visas med det talsystem som är inställt i Base. Resultaten är begränsade till vissa maxvärden då Base = HEX eller BIN.

Ställa in Base för visning av resultat

1. Tryck [MODE] [F2] för att öppna sidan 2 i fönstret MODE.
2. Bläddra till inställningen för Base, tryck ⊙, och välj önskad inställning.
3. Tryck [ENTER] för att stänga fönstret MODE.

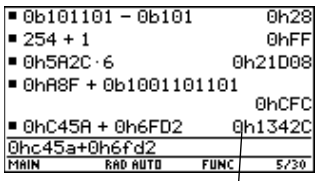


***Obs!** Inställningen för* Base *påverkar bara visade resultat. Du måste alltid ange prefixet* 0h *alt.* 0b *när du skriver in hexadecimala och binära tal.*

Inställningarna för Base påverkar bara visningen av heltal.

Bråk och decimaltal visas alltid i decimal form.

Om Base = HEX:



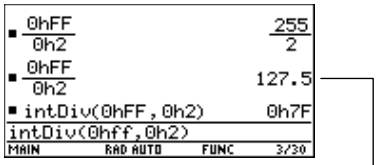
Resultaten börjar med 0h för att ange aktuellt.

Division när Base = HEX alt. BIN

Med inställningen Base=HEX eller BIN visas resultaten av en division i hexadecimal eller binär form bara om resultatet är ett heltal.

Om du vill försäkra dig om att resultatet alltid är ett heltal kan du använda **intDiv()** i stället för ÷.

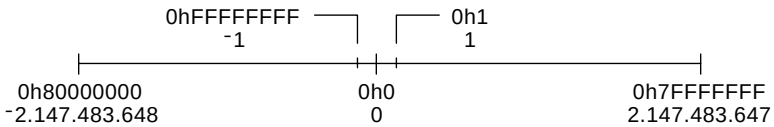
Om Base = HEX:



Tryck ◆ [ENTER] om du vill visa resultatet i formen APPROXIMATE.

Begränsningar när Base = HEX alt. BIN

När Base=HEX alt. BIN sparas heltal internt som ett 32-bitars binärt tal vilket kan använda talområdet som visas nedan (visas i hexadecimal och decimal form):



Om ett resultat är för stort för att lagras i 32 bitar används en symmetrisk modulo-operation för att bringa resultatet inom givna gränser. Alla tal större än 0h7FFFFFFF påverkas av detta. Till exempel ändras 0h80000000 till 0hFFFFFFFF till negativa värden.

Jämföra eller manipulera enskilda bitar

Med följande operatorer och funktioner kan du jämföra och manipulera enskilda bitar i binära tal. Du kan skriva in heltal i valfritt talsystem. Talen konverteras automatiskt till binära tal för operationen och resultaten visas i aktuellt talsystem inställt i Base.

Booleska operationer

Obs! *Du kan välja dessa operatorer från menyn* MATH/Base*. Exempel på hur du använder respektive operator finns i appendix A längst bak i denna bok.*

Operator med syntax	Beskrivning
not *heltal*	Returnerar förstakomplementet där varje bit är omvänd.
(-) *heltal*	Returnerar andrakomplementet vilket är samma som förstakomplementet +1.
heltal1 **and** *heltal2*	Vid en bitvis jämförelse med **and** är resultatet 1 om båda bitarna är 1, annars är resultatet 0. Det returnerade värdet motsvarar det resulterande binära värdet.
heltal1 **or** *heltal2*	Vid en bitvis jämförelse med **or** är resultatet 1 om en av bitarna är 1 och 0 bara om båda bitarna är 0. Det returnerade värdet motsvarar det resulterande binära värdet.
heltal1 **xor** *heltal2*	Vid en bitvis jämförelse med **xor** är resultatet 1 om en av bitarna är 1 (men inte båda). Resultatet är 0 om båda bitarna är 0 eller om båda bitarna är 1. Det returnerade värdet motsvarar det resulterande binära värdet.

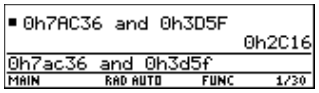
Anta att du skriver in:

0h7AC36 **and** 0h3D5F

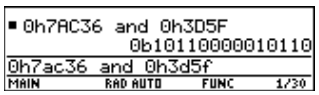
Internt konverteras de hexadecimala talen till 32-bitars binära tal.

Bitarna i de båda talen jämförs sedan bit för bit.

Om Base = HEX:



Om Base = BIN:



Obs! *Om ett resultat är för stort för att lagras i 32 bitar används en symmetrisk modulo-operation för att bringa resultatet inom givna gränser (se sidan 346).*

0h7AC36 = 0b00000000000000111101011000011011 0
and **and**
0h3D5F = 0b00000000000000000001111010101111 1
0b00000000000000000001011000001011 0 = 0h2C16

Inledande nollor visas inte i resultatet.

Resultatet visas i det talsystem som är inställt i Base.

Rotera och skifta bitar

Funktion med syntax	Beskrivning
rotate(*heltal***)** – eller – **rotate(***heltal***,#***antalSteg***)**	Om #*antalSteg* är: • uteslutet — alla bitar roteras ett steg till höger (förval är ⁻1). • negativt — alla bitar roteras angivet antal steg till höger. • positivt — alla bitar roteras angivet antal steg till vänster. Vid en rotering till höger roteras biten längst till höger till den lediga platsen längst till vänster och tvärt om för en rotering till vänster.
shift(*heltal***)** – eller – **shift(***heltal***,#***antalSteg***)**	Om #*antalSteg* är: • uteslutet — alla bitar skiftas ett steg till höger (förval är ⁻1). • negativt — alla bitar skiftas angivet antal steg till höger. • positivt — alla bitar skiftas angivet antal steg till vänster. Vid en skiftning till höger försvinner biten längst till höger och 0 alt 1 sätts in för att matcha biten längst till vänster. Vid en skiftning till vänster försvinner biten längst till vänster och 0 sätts in längst till höger.

Obs! *Du kan välja dessa operatorer från menyn* MATH/Base*. Exempel på hur du använder respektive operator finns i appendix A längst bak i denna bok.*

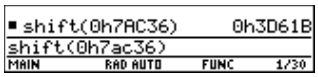
Anta att du skriver in:

shift(0h7AC36**)**

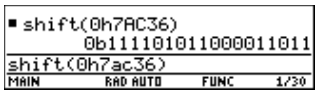
Internt konverteras de hexadecimala talen till 32-bitars binära tal.

Bitarna i de båda talen jämförs sedan bit för bit.

Om Base = HEX:



Om Base = BIN:



Obs! *Om ett resultat är för stort för att lagras i 32 bitar används en symmetrisk modulo-operation för att bringa resultatet inom givna gränser (se sidan 346).*

Varje bit skiftas till höger.

0h7AC36 = 0b00000000000000111101011000011011 0

Lägger till 0 om den vänstra biten är 0, eller 1 om den vänstra biten är 1.

Försvinner

0b00000000000000111101011000011011 = 0h3D61B

Inledande nollor visas inte i resultatet.

Resultatet visas i det talsystem som är inställt i Base.

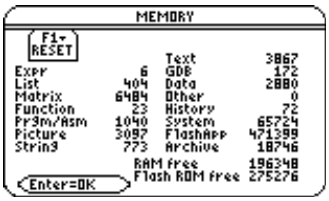
Minnes- och variabelhantering

21

Översikt över minnes- och variabelhantering 350
Kontrollera och nollställa minnet 353
Visa VAR-LINK-fönstret..... 354
Hantera variabler och mappar med VAR-LINK..... 357
Klistra in ett variabelnamn i ett program..... 359
Spara och hämta variabler..... 360
Om ett meddelande om Garbage Collection visas..... 362
Minnesfel vid läsning av sparad variabel 364

I detta kapitel beskrivs hur du administrerar variabler som är lagrade i minnet på din TI-89 / TI-92 Plus.

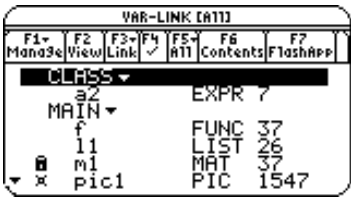
***Obs!** Kom ihåg att variabler kan inkludera program, funktioner, geometriska figurer, grafer mm.*



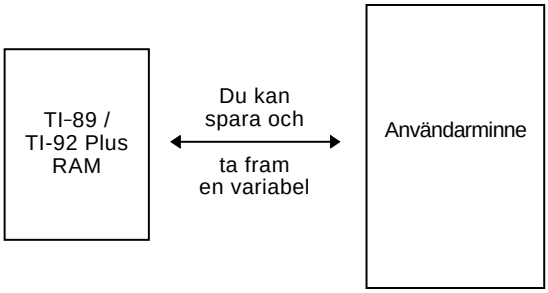
I MEMORY-fönstret visas hur minnet används för tillfället.

***Obs!** Du kan även använda VAR-LINK när du överföra variabler mellan två länkade TI-89, en TI-92, eller en TI-92 Plus. Se kapitel 22.*

I VAR-LINK-fönstret visas en lista med definierade variabler och mappar. För information om mappen, se kapitel 5.



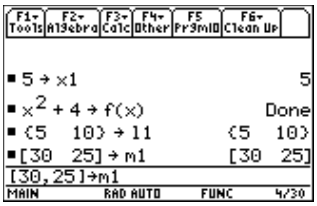
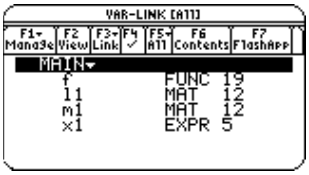
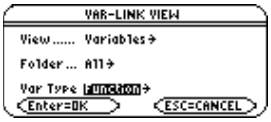
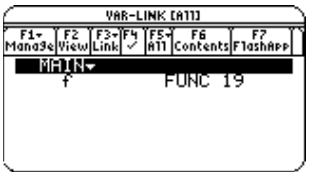
Du kan lagra variabler i användarminnet på din TI-89 / TI-92 Plus Användarminnet är ett separat minnesområde utanför RAM (Random Access Memory).



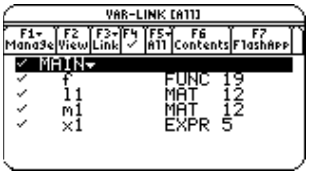
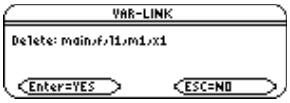
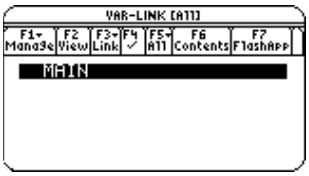
Att spara variabler i användarminnet kan vara mycket praktiskt i vissa fall, se sidan 360. Om du inte har någon användning av användarminnet behöver du inte använda det.

Översikt över minnes- och variabelhantering

Tilldelar värden till en mängd olika datatyper. Använd fönstret för VAR-LINK när du vill se en lista med definierade variabler. Flytta en variabel till användarminnet och undersök på vilka sätt du kan, och inte kan, använda variabeln. (Lagrade variabler låses automatiskt.) Ta tillbaks variabeln från användarminnet och radera oanvända variabler så de inte tar upp utrymme i minnet.

Steg	TI-89 Tangenter	TI-92 Plus Tangenter	Teckenfönster
1. Tilldela variabler från grundfönstret med följande variabeltyper. Uttryck: 5 → x1 Funktion: x^2 +4 → f(x) Lista: {5,10} → l1 Matris: [30,25] → m1	[HOME] [CLEAR] 5 [STO▸] X 1 [ENTER] X [^] 2 [+] 4 [STO▸] [alpha] F [(] X [)] [ENTER] [2nd] [{] 5 [,] 1 0 [2nd] [}] [STO▸] [alpha] L 1 [ENTER] [2nd] [[] 3 0 [,] 2 5 [2nd] []] [STO▸] [alpha] M 1 [ENTER]	[♦] [HOME] [CLEAR] 5 [STO▸] X 1 [ENTER] X [^] 2 [+] 4 [STO▸] F [(] X [)] [ENTER] [2nd] [{] 5 [,] 1 0 [2nd] [}] [STO▸] L 1 [ENTER] [2nd] [[] 3 0 [,] 2 5 [2nd] []] [STO▸] M 1 [ENTER]	
2. Anta att du börjar utföra en beräkning med hjälp av en funktionsvariabel men inte kan komma ihåg dess namn.	5 [×]	5 [×]	5*
3. Visa VAR-LINK-fönstret. *I detta exempel antas att de variabler som tilldelades ovan är de enda variabler som är definierade.*	[2nd] [VAR-LINK]	[2nd] [VAR-LINK]	
4. Ändra fönstervyn för att endast visa funktionsvariablerna. *Även om det inte verkar så användbart i ett exempel med fyra variabler kan du tänka dig hur användbart det kan vara om det fanns många variabler av olika typ.*	[F2] [⊖] [⊖] [⊕] 5 [ENTER]	[F2] [↓] [↓] [→] 5 [ENTER]	 

Steg	TI-89 Tangenter	TI-92 Plus Tangenter	Teckenfönster
5. Markera funktionsvariabeln f och visa dess innehåll. *Observera att funktionen tilldelades genom att använda f(x) men listas som f i fönstret.*	⊖ [2nd] [F6]	⊖ [F6]	x^2+4
6. Stäng innehållsfönstret.	[ESC]	[ESC]	
7. Låt variabeln f fortfarande vara markerad, stäng VAR-LINK-fönstret och klistra in variabelnamnet på inmatningsraden.	[ENTER]	[ENTER]	5* f(Observera att " (" klistrats in.
8. Slutför beräkningen.	2 [)] [ENTER]	2 [)] [ENTER]	5* f(2)
<u>Arkivera en variabel:</u>			
9. Använd fönstret VAR-LINK för att visa en lista med variabler. Markera önskad variabel. *Den tidigare visningsändringen gäller inte längre. Alla definierade variabler visas på skärmen.*	[2nd] [VAR-LINK] (använd ⊖ för att markera variabeln x1)	[2nd] [VAR-LINK] (använd ⊖ för att markera variabeln x1)	VAR-LINK [A11] F1 Manage F2 View F3 Link F4 ✓ F5 All F6 Contents F7 FlashApp MAIN f FUNC 19 l1 LIST 10 m1 MAT 12 t1 FIG 26 x1 EXPR 5
10. Använd menyn [F1] Manage för att spara variabeln.	[F1] 8	[F1] 8	F1 Manage 2:Copy 3:Rename 4:Move 5:Create Folder 6:Lock 7:UnLock 8:Archive Variable 9:Unarchive Variable VAR-LINK [A11] F1 Manage F2 View F3 Link F4 ✓ F5 All F6 Contents F7 FlashApp MAIN f FUNC 19 l1 MAT 12 m1 MAT 12 ¤ x1 EXPR 5 ¤ visar att variabeln är sparad i användarminnet.
11. Återgå till grundfönstret och använd den sparade variabeln i beräkningen.	[HOME] 6 [×] X 1 [ENTER]	[♦] [HOME] 6 [×] X 1 [ENTER]	■ 5·f(2) 40 ■ 6·x1 30 6*x1 MAIN RAD AUTO FUNC 6/30
12. Försök att spara ett annat värde i den lagrade variabeln.	1 0 [STO▸] X 1 [ENTER]	1 0 [STO▸] X 1 [ENTER]	ERROR Variable is locked, protected, or archived ESC=CANCEL
13. Stäng felmeddelandet.	[ESC]	[ESC]	

Steg	TI-89 Tangenter	TI-92 Plus Tangenter	Teckenfönster
14. Använd fönstret VAR-LINK för att flytta variabeln til RAM.	2nd [VAR-LINK] (använd ⊖ för att markera variabeln x1) F1 9	2nd [VAR-LINK] (använd ⊖ för att markera variabeln x1) F1 9	
15. Återgå till grundfönstret och spara det ändrade värdet till variabeln som nu finns i RAM.	HOME ENTER	♦ [HOME] ENTER	
<u>Ta bort variabler:</u>			
16. Använd menyn F5 All för att markera variabler. *Symbolen ✓ anger poster som är markerade. Lägg märke till att detta också markerar mappen MAIN* ***Obs!** I stället för att använda F5 (om du inte vill ta bort alla variabler), kan du markera enskilda variabler. Markera varje post som ska tas bort och tryck på F4.* *Mer information om hur du tar bort enskilda variabler finns på sidan 357.*	2nd [VAR-LINK] F5 1	2nd [VAR-LINK] F5 1	 
17. Använd menyn F1 för att ta bort. ***Obs!** Du kan trycka på ← (i stället för F1 1) när du vill ta bort de markerade variablerna.*	F1 1	F1 1	
18. Bekräfta borttagningen.	ENTER	ENTER	
19. Eftersom F5 1 även markerade mappen MAIN talar ett felmeddelande om att du inte kan ta bort mappen MAIN. Bekräfta meddelandet. *När VAR-LINK visas igen listas inte de borttagna variablerna.*	ENTER	ENTER	
20. Stäng VAR-LINK-fönstret och återgå till den aktuella applikationen (Grundfönstret, i detta exempel). *När du använder ESC (i stället för ENTER) för att stänga VAR-LINK klistras inte det markerade namnet in på inmatningsraden.*	ESC	ESC	

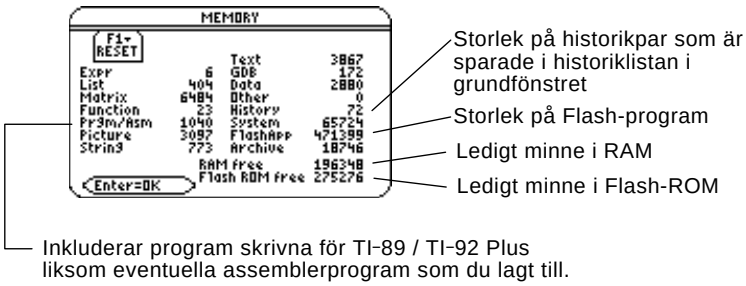
Kontrollera och nollställa minnet

I fönstret MEMORY visas hur mycket minne (i byte) som används av alla variabler av varje datatyp, oavsett om variabeln finns i RAM eller i användarminnet. Du kan även använda detta fönster om du vill återställa minnet.

Visa MEMORY-fönstret

Tryck på [2nd] [MEM].

Tips! Om du vill se storleken hos enskilda variabler eller kontrollera om de är lagrade i användarminnet öppnar du fönstret VAR-LINK.



Stäng fönstret genom att trycka på [ENTER]. Om du vill återställa minnet följer du beskrivningen nedan.

Nollställa minnet

Gör följande från MEMORY-fönstret:

1. Tryck på [F1].
2. Välj ett alternativ.

Viktigt: Om du vill ta bort en enstaka variabel, i stället för alla, kan du använda VAR-LINK enligt beskrivningen på sidan 357.

Alternativ	Beskrivning
RAM	1:All RAM: Återställning av RAM raderar alla data och alla program från RAM. 2:Default: Återställer alla systemvariabler och lägen till fabriksinställningarna. Detta påverkar inte användardefinierade variabler, funktioner eller mappar.
Flash ROM	1:Archive: Återställning av användarminnet raderar alla data och alla program från Flash-ROM. 2:Flash Apps: Återställning av Flashprogram raderar alla Flash-program från Flash-ROM. 3:Both: Återställning av båda raderar all data, program, och Flash-program från Flash-ROM.
All Memory	Återställning av allt raderar alla data, program, och Flash-program från RAM och Flash-ROM.

Tips! Avbryt nollställnings-processen genom att trycka på [ESC] i stället för [ENTER].

3. När du blir uppmanad att bekräfta trycker du på [ENTER].

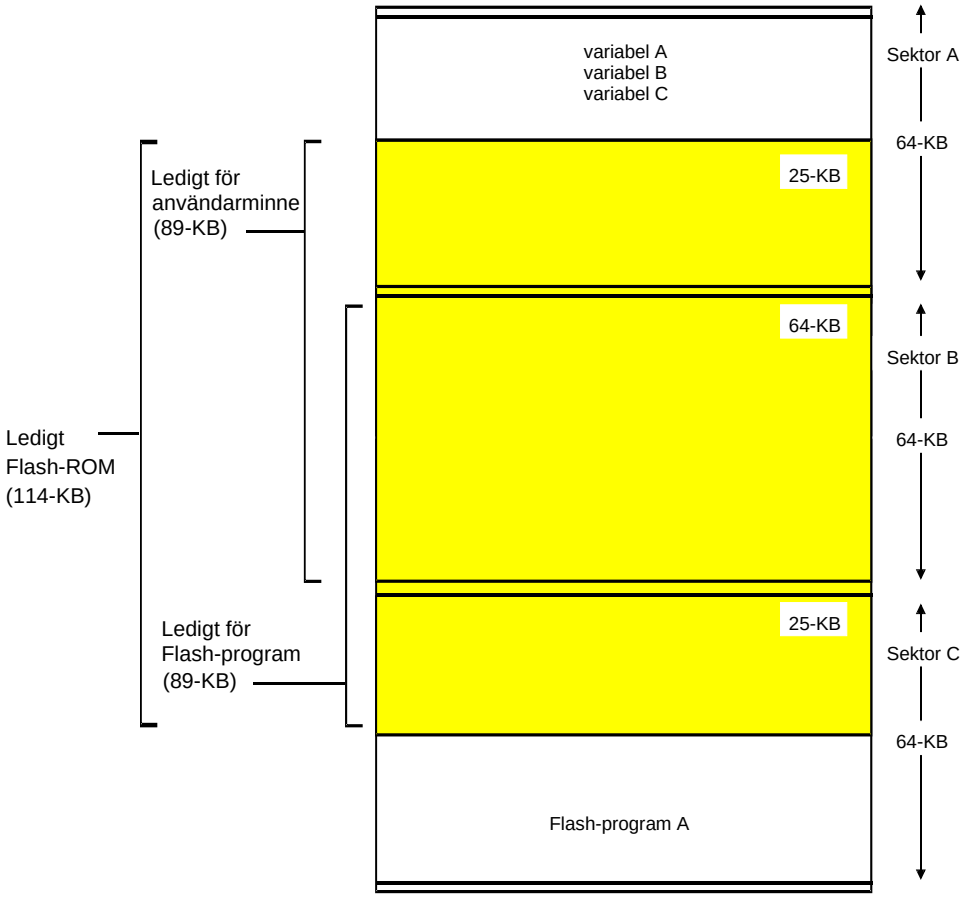
   Ett meddelande visas när nollställningen är klar.
4. Tryck på [ENTER] för att bekräfta meddelandet.

Ledigt Flash-ROM på skärmen MEMORY

__Obs!__ För TI-92 Plus moduler och en del TI-89 användare är maximalt användarminne 384-KB, oavsett hur mycket ledigt Flash-ROM som finns tillgängligt.

Ledigt minne i Flash-ROM som visas i fönstret Minne [2nd] [MEM] delas av användarminne och Flash-program. Räknarens Flash-ROM är indelat i sektorer om 64-KB. Varje enskild sektor kan innehålla antingen användarminne eller Flash-program, men inte båda samtidigt. På grund av detta kan aktuellt ledigt minne för användarminne eller Flash-program vara mindre än det totalt lediga minnet i Flash-ROM som visas i fönstret.

Visar ledigt Flash-ROM



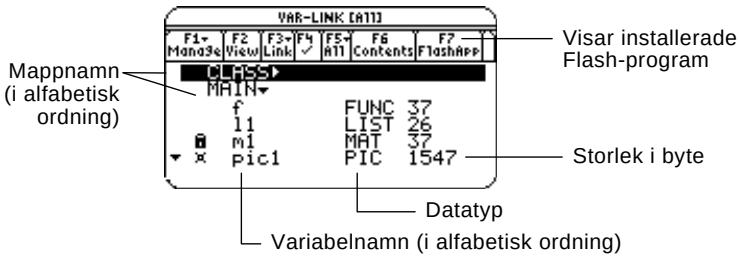
Visa VAR-LINK-fönstret

I VAR-LINK-fönstret visas de variabler och mappar som för tillfället är definierade. När du har visat fönstret kan du hantera variablerna och/eller mapparna enligt beskrivningarna i de återstående avsnitten i detta kapitel.

Visa VAR-LINK-fönstret

Tryck på [2nd] [VAR-LINK]. Som standard visas i VAR-LINK-fönstret alla användardefinierade variabler i alla mappar och med alla datatyper.

Obs! Information om hur du använder mappar finns i kapitel 5.



Detta...	indikerar detta...
[F3] Link	Enheten är klar för att överföra variabler och Flash-program mellan enheter och för att uppdatera programvara till din TI-89 / TI-92 Plus. Se kapitel 22.
▸	Stängd mapp.
▾	Öppen mapp (till höger om mappens namn).
▾	Du kan bläddra för att visa fler variabler/mappar.
✓	Om markerad med [F4].
🔒	Låst
⌘	I användarminnet

Så här rullar du igenom listan:

- Tryck på ⊖ eller ⊕. (Använd [2nd] ⊖ eller [2nd] ⊕ för att rulla en sida i taget.)

  — eller —

- Skriv en bokstav. Om det finns något variabelnamn som börjar med den bokstaven markeras den första förekomsten av dessa variabelnamn.

Tips! Tryck på en bokstav flera gånger för att bläddra igenom namnen som börjar med den bokstaven.

Variabeltyper så som de listas i VAR-LINK

Typ	Beskrivning
ASM	Assemblerprogram
DATA	Data
EXPR	Uttryck (inkluderar numeriska värden)
FUNC	Funktion
GDB	Grafdatabas
LIST	Lista
MAT	Matris
PIC	Bild av en graf
PRGM	Program
STR	Sträng
TEXT	Text Editor-session

Visa endast en angiven mapp och/eller variabeltyp

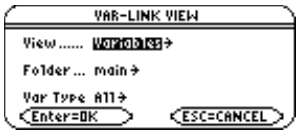
Om du har flera variabler och/eller mappar kan det vara svårt att hitta en viss variabel. Genom att ändra vyn i VAR-LINK-fönstret kan du ange vilken information du vill visa.

Gör följande från VAR-LINK-fönstret:

__Tips!__ Om du vill stänga en meny trycker du på [ESC].

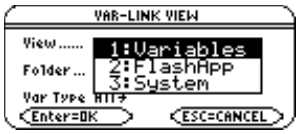
__Tips!__ Om du vill lista systemvariabler (window-variabler, etc.), välj 3:System.

1. Tryck på [F2] View.



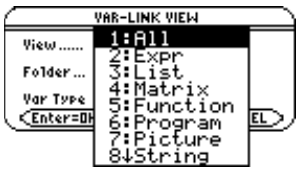
2. Markera inställningen du vill ändra och tryck på ⊙. En meny visas med tillgängliga alternativ.

   View — Här kan du välja variabler, Flash-program, eller systemvariabler som du vill visa.



   Folder — Visar alltid 1:All och 2:main och listar övriga mappar om du har skapat några.

   Var Type — Visar de giltiga variabeltyperna.



Symbolen ↓ anger att du kan rulla för att visa ytterligare variabeltyper.

3. Välj den nya inställningen.
4. När du är tillbaka i VAR-LINK VIEW-fönstret trycker du på [ENTER].

VAR-LINK-fönstret uppdateras för att visa endast den angivna mappen och/eller variabeltypen.

Stänga VAR-LINK-fönstret

__Tips!__ Mer information om hur du använder [ENTER] *- klistra in-funktionen finns på sidan 359.*

När du vill stänga VAR-LINK-fönstret och återgå till det aktuella programmet använder du [ENTER] eller [ESC] enligt beskrivningen nedan.

Tryck på:	Om du vill:
[ENTER]	Klistra in den markerade variabeln eller mappen vid markören i den aktuella applikationen.
[ESC]	Återgå till det aktuella applikationen utan att klistra in det markerade namnet.

Du visar innehållet i en variabel med hjälp av VAR-LINK-fönstret. Du kan också välja ett eller flera listade alternativ och hantera dem med hjälp av instruktionerna i detta avsnitt.

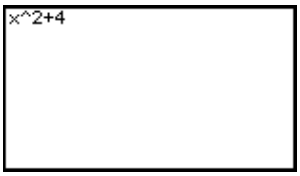
Visa innehållet i en variabel

Du kan visa alla variabeltyper förutom ASM, DATA eller GDB. Du måste till exempel öppna en DATA-variabel med Data/Matrix Editor.

1. Flytta, i VAR-LINK, markören så att variabeln markeras.

Obs! Du kan inte redigera innehållet i detta fönster.

2. Tryck på:
   TI-89: [2nd] [F6]
   TI-92 Plus: [F6]
   Om du markerar en mapp visas i fönstret antal variabler som finns i den mappen.



3. Du återgår till VAR-LINK genom att trycka på en tangent.

Välja alternativ från listan

För andra åtgärder väljer du en eller flera variabler och/eller mappar.

Om du vill markera:	Gör du följande:
En enstaka variabel eller mapp	Flytta markören till alternativet för att markera det.
En grupp variabler eller mappar	Markera varje alternativ och tryck på [F4]. Symbolen ✓ visas till vänster om varje markerat alternativ. (Om du markerar en mapp, markeras alla variabler som finns i den mappen.) Använd [F4] för att markera eller avmarkera ett alternativ.
Alla mappar och variabler	Öppna mappen ⊙, tryck på [F5] All och välj 1:Select All.

Obs! Om du använder [F4] för att markera (✓) ett eller flera alternativ och därefter markera ett annat alternativ, påverkar följande åtgärder endast de alternativ som är markerade med symbolen ✓.

Tips! Tryck på ⊙ eller ⊙ för att öppna/stänga en mapp som du markerat.

Om du väljer 4:Expand All eller 5:Collapse All öppnas/stängs alla mappar eller Flash-program.

Markerar den senaste uppsättningen poster som överförts till din enhet under den aktuella VAR-LINK sessionen. Mer information finns i kapitel 22.

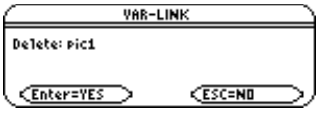
Ta bort variabler och mappar

När du tar bort en mapp måste du ta bort alla variabler som finns i den mappen. Du kan dock inte ta bort mappen MAIN, även om den är tom.

Tips! När du använder [F4] för att markera en öppnad mapp så markeras automatiskt alla variabler i mappen så att du kan ta bort mappen och innehållet samtidigt.

1. Välj, i VAR-LINK, variablerna och/eller mapparna.
2. Tryck på [F1] Manage och välj 1:Delete. (Du kan trycka på [←] i stället för [F1] 1.)

3. Bekräfta borttagningen genom att trycka på [ENTER].



Skapa en ny mapp

Information om mappar finns i kapitel 5.

1. Tryck på [F1] Manage och välj 5:Create Folder i VAR-LINK-fönstret.
2. Skriv ett unikt namn och tryck på [ENTER] två gånger.

Kopiera eller flytta variabler från en mapp till en annan

Du måste ha minst en mapp förutom MAIN. Du kan inte använda VAR-LINK för att kopiera variabler inom samma mapp.

1. Markera variablerna i VAR-LINK-fönstret.
2. Tryck på [F1] Manage och välj 2:Copy eller 4:Move.
3. Markera målmappen.
4. Tryck på [ENTER].

   De kopierade eller flyttade variablerna behåller sina ursprungliga namn.

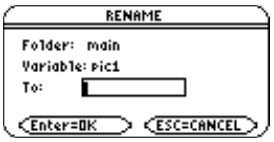
***Tips!** Om du vill kopiera en variabel till ett annat namn i samma mapp använder du [STO▸] (exempelvis* a1→a2*) eller kommandot* **CopyVar** *från grundfönstret.*

Ändra namn på variabler och mappar

Kom ihåg att om du använder [F4] för att markera en mapp, markeras automatiskt variablerna som finns i den mappen. Använd [F4] för att avmarkera enskilda variabler som inte ska vara markerade.

1. Markera variablerna och/eller mapparna i VAR-LINK-fönstret.
2. Tryck på [F1] Manage och välj 3:Rename.
3. Skriv ett unikt namn och tryck på [ENTER] två gånger.

   Om du markerade flera alternativ blir du uppmanad att ange ett nytt namn för var och en.



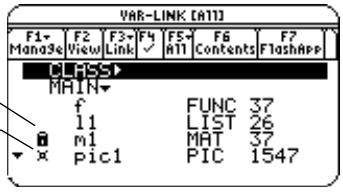
Låsa och låsa upp variabelmappar och Flash-program

När en variabel är låst kan du inte ta bort, ändra namn på eller spara den. Du kan däremot kopiera, flytta eller visa innehållet. När en mapp är låst kan du hantera variablerna i mappen (om variablerna inte är låsta), men du kan inte ta bort mappen.

1. Markera önskade variabler och/eller mappar samt eventuella Flash-program i fönstret VAR-LINK.
2. Tryck på [F1] Manage och välj 6:Lock Variable eller 7:UnLock Variable.

Symbolen 🔒 anger en låst variabel eller mapp.

⌘ anger att variabeln är lagrad i användarminnet och automatiskt låst.



Klistra in ett variabelnamn i ett program

Anta att du skriver ett uttryck i grundfönstret och inte kan komma ihåg vilken variabel du ska använda. Du kan då visa VAR-LINK-fönstret, markera en variabel från listan och klistra in det variabelnamnet direkt på inmatningsraden i grundfönstret.

Program som du kan använda

Du kan klistra in ett variabelnamn till den aktuella markörpositionen från följande applikationer.

- Home screen, Y= Editor, Table Editor eller Data/Matrix Editor — Markören måste vara på inmatningsraden.
- Text Editor, Window Editor, Numeric Solver eller Program Editor — Markören kan vara var som helst på skärmen.

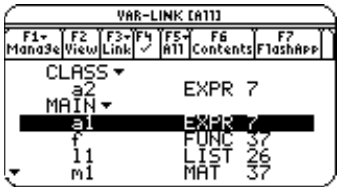
Procedur

Så här startar du från en av applikationerna som nämndes ovan:

1. Placera markören där du vill infoga variabelnamnet.

```
sin(|
```

2. Tryck på [2nd] [VAR-LINK].
3. Markera variabeln i fråga.



Obs! *Du kan också markera och klistra in mappnamn.*

4. Tryck på [ENTER] för att klistra in variabelnamnet.

```
sin(a1|
```

5. Skriv klart uttrycket.

```
sin(a1)|
```

Obs! *Detta klistrar in variabelns namn, inte dess innehåll. (Använd* [2nd] [RCL]*, i stället för* [2nd] [VAR-LINK] *för att hämta en variabels innehåll.)*

Om du klistrar in ett variabelnamn som inte finns i den aktuella mappen, klistras variabelns sökvägsnamn in.

```
sin(class\a2|
```

Om CLASS *inte* är den aktuella mappen, klistras detta in om du markerar variabeln a2 i CLASS.

Spara och hämta variabler

> När du vill spara eller hämta en eller flera variabler interaktivt använder du fönstret VAR-LINK. Du kan också utföra dessa operationer från grundfönstret eller från ett program.

Varför ska man lagra variabler i användarminnet?

__Obs!__ Du kan inte lagra variabler med reserverade namn eller systemvariabler i användarminnet.

Med användarminnet kan du:

- Lagra data, program eller olika variabler på en säker plats där de inte kan redigeras eller tas bort oavsiktligt.
- Frigöra extra RAM-minne. T ex:
  - Lagra variabler som du behöver komma åt, men inte ändra, eller variabler som du inte använder för tillfället, men som du kommer att använda igen senare.
  - Om du tänker installera extra program på din TI-89 / TI-92 Plus, särskilt minneskrävande sådana, kan du vara tvungen att frigöra extra RAM-minne innan du installerar programmen.

Extra ledigt minne kan öka prestandan för vissa typer av beräkningar.

Kontrollera tillgängligt minne

__Obs!__ Om det inte finns tillräckligt med minne kan du ta bort eller flytta variabler efter behov.

Innan du sparar eller hämtar variabler, speciellt omfattande variabler som program:

1. Använd fönstret VAR-LINK för att avläsa storleken på variabeln.
2. Använd fönstret MEMORY för att kontrollera om det finns tillräckligt med tillgängligt minne.

Om du vill:	Måste storleken vara:
Spara till användarminnet	Archive free size > variabelns storlek
Hämta till RAM	RAM free size > variabelns storlek

Även om det verkar finnas tillräckligt med minne kan du få upp en varning om Garbage Collection (se sidan 362) när du försöker att lagra en variabel. Beroende på användbarheten av lediga minnesblock kan du vara tvungen att ta bort flera variabler från användarminnet innan minnet räcker till.

Från fönstret VAR-LINK

Tips!: Om du vill välja en enstaka variabel markerar du den. Om du vill välja flera variabler kan du markera varje variabel och trycka [F4] ✓.

Obs! Om du får ett meddelande om Garbage Collection, se vidare på sidan 362.

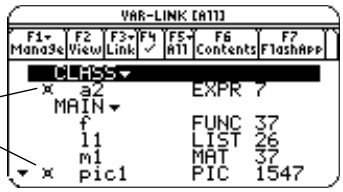
Obs! Sparade variabler låses automatiskt. Du kan läsa variabeln men du kan inte ta bort eller redigera den. Se mer information på sidan 364.

När du vill spara eller hämta variabler:

1. Tryck [2nd] [VAR-LINK] för att öppna fönstret VAR-LINK.
2. Välj en eller flera variabler. Variablerna kan finnas i olika kataloger (du kan markera en hel katalog genom att markera katalognamnet).
3. Tryck [F1] och välj antingen:

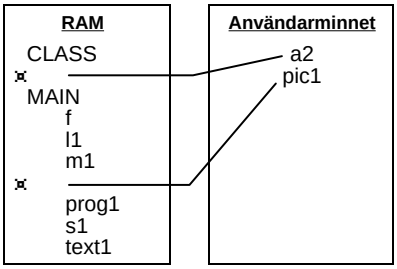
   8:Archive Variable
   – eller –
   9:Unarchive Variable

Om du väljer 8:Archive Variable flyttas variablerna till användarminnet.



Variabler i användarminnet

Du kan läsa sparade variabler på samma sätt som du läser låsta variabler. Variabeln finns fortfarande kvar i sin katalog. Den är bara lagrad i användarminnet i stället för i RAM.



Från grundfönstret eller ett program

Använd kommandona **Archive** och **Unarchiv** (appendix A).

Archive *variable1*, *variable2*, …

Unarchiv *variable1*, *variable2*, …

Om ett meddelande om Garbage Collection visas

Om du använder användarminnet i stor utsträckning kan du få ett meddelande om Garbage Collection. Detta inträffar om du försöker spara en variabel som är större än tillgängligt minne. TI-89 / TI-92 Plus försöker i dessa fall flytta runt variabler för att skapa mer ledigt utrymme.

Besvara meddelande om Garbage Collection

När du ser meddelandet till höger:

- Om du vill fortsätta spara, tryck [ENTER].

  – eller –
- Om du vill avbryta, tryck [ESC].

Efter garbage collection, beroende på hur mycket extra minne som skapats, kan variabeln ha sparats eller inte. Om variabeln inte sparats kan du ta bort några variabler och försöka igen.

Varför utförs inte Garbage Collection automatiskt utan meddelande?

Meddelandet:

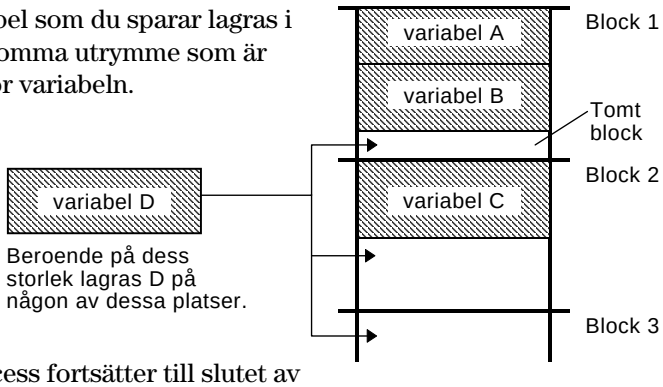
- Ger dig information om att spara-kommandot kommer att ta längre tid än normalt. Dessutom ger det information om att kommandot kanske inte kan slutföras pga för lite tillgängligt minne.
- Kan ge en indikation om att ett program fastnat i en slinga där användarminnet fylls av nya variabler. Avbryt spara-kommandot och undersök orsaken till meddelandet.

Varför behövs Garbage Collection?

Användarminnet är indelat i block. När du först börjar spara så sparas variabler efter varandra i block 1. Detta fortsätter tills blocket är fyllt. När det inte längre finns nog mycket minne kvar i blocket för aktuell variabel sparas den i början på nästa block. Eftersom ett block oftast inte passar exakt för aktuella variabler blir det därför lite ledigt minne sist i föregående block.

Obs! *En sparad variabel lagras i ett kontinuerligt minnesutrymme inom ett block. Enskilda variabler kan inte omfatta flera block.*

Varje variabel som du sparar lagras i det första tomma utrymme som är stort nog för variabeln.

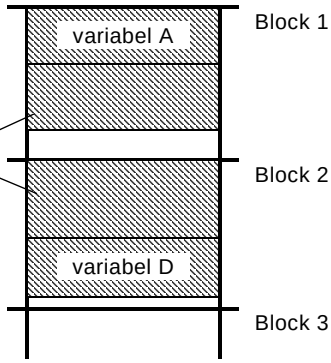


Obs! *Garbage collection startas när variabeln du sparar är större än det största tillgängliga minnesutrymmet.*

Denna process fortsätter till slutet av det sista blocket. Beroende på storleken av de enskilda variablerna kan tomma block ta upp en hel del av det totala minnesutrymmet.

Hur processen påverkas av att ta bort variabler

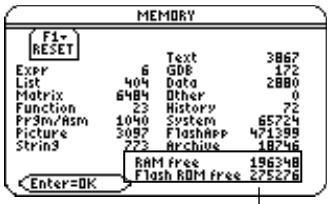
När du flyttar en variabel till RAM kopieras den till RAM men tas inte bort från användarminnet.



Variabler som är flyttade till RAM markeras för borttagning och tas bort vid nästa garbage collection.

Om fönstret MEMORY visar att det finns ledigt minne

Även om fönstret MEMORY visar att det finns tillräckligt med minne för en variabel kan du få meddelande om Garbage Collection.



Visar hur stort ledigt utrymme som kommer att finnas efter att markerade variabler tagit bort.

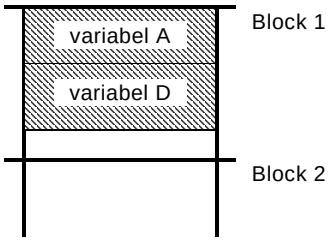
När du flyttar en variabel till RAM ökar kommandot Archive free ledigt minne omedelbart men minnesområdet finns inte tillgängligt förrän efter nästa garbage collection.

Om kommandot RAM free visar att det finns tillräckligt med minne för din variabel är det troligt att det kommer att finnas tillräckligt med minne efter nästa garbage collection (beror på hur fragmenteringen av lediga minnesområden ser ut).

Processen Garbage Collection

Processen garbage collection:

- Tar bort variabler som flyttats till RAM från användarminnet.
- Flyttar runt kvarvarande variabler till efter varandra följande block.



Minnesfel vid läsning av sparad variabel

En sparad variabel hanteras på samma sätt som låsta variabler. Du kan läsa variabeln men du kan inte ta bort eller ändra den. I vissa fall kan du dock få ett Memory Error då du försöker läsa en sparad variabel.

Vad orsakar minnesfelet?

Felmeddelandet Memory Error visas om det inte finns tillräckligt med ledigt utrymme i RAM för att läsa en variabel. Detta kan leda till frågan "Om variabeln är lagrad i användarminnet vad har tillgängligt RAM för betydelse?" Svaret är att följande operationer bara kan utföras då variabeln finns i RAM.

Obs! *Som beskrivits ovan kan en temporär kopia användas för att exekvera och läsa sparade variabler. Du kan i dessa fall inte spara eventuella ändringar till variabeln.*

- Öppna en textvariabel med Text Editor.
- Öppna en datavariabel, lista eller matris med Data/Matrix Editor.
- Öpna ett program eller en funktion med Program Editor.
- Exekvera ett program eller anropa en funktion.

TI-89 / TI-92 Plus utför kopieringar av variabler i bakgrunden för att du inte själv ska behöva flytta variabler till RAM i onödan. Om du t ex exekverar ett program som är lagrat i användarminnet så utförs följande i TI-89 / TI-92 Plus:

Obs! *Förutom för program och funktioner så skapar referenser till en sparad variabel ingen kopia. Om en variabel* ab *är sparad så skapas ingen kopia om du anger* 6*ab.

1. Kopierar programmet till RAM.
2. Exekverar programmet.
3. Tar bort kopian från RAM när programmet avslutas.

Felmeddelandet visas om det inte finns tillräckligt med RAM för den temporära kopian av variabeln.

Åtgärda felet

Frigöra RAM för att använda variabeln:

1. Använd fönstret VAR-LINK ([2nd] [VAR-LINK]) för att kontrollera storleken på den sparade variabel som du vill använda.
2. Använd fönstret MEMORY ([2nd] [MEM]) för att kontrollera tillgängligt RAM.
3. Frigör ledigt minne genom att:
   - Ta bort variabler du inte använder från RAM.
   - Spara undan stora variabler, som t ex program (flytta dem från RAM till användarminnet).

Obs! *Tillgängligt* RAM *måste vara större än variabelns storlek.*

Överföring av variabler och uppgradering

22

Länka två enheter 366
Överföra variabler, Flash-program och mappar 367
Överföra variabler under från ett program 371
Uppgradera programkod (baskod) 373
Samla in och överföra ID-listor 378
Kompatibilitet mellan TI-89, TI-92 Plus, och TI-92 380

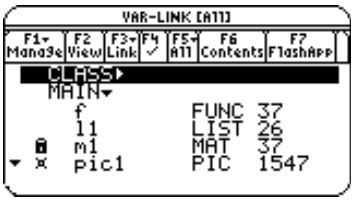
I detta kapitel beskriv hur du kan använda VAR-LINK till att:

- Överföra variabler, Flash-program, och mappar mellan två enheter
- Uppgradera programkod (baskod)
- Samla in ID-listor

Kapitlet innefattar även information om hur du överför variabler via programkörningar samt olika kompatibilitetsfrågor.

Variabler innefattar program, funktioner, grafer, etc.

I fönstret VAR-LINK visas en lista med definierade variabler, Flash-program och mappar. Mer information om hur du använder mappar finns i kapitel 5.

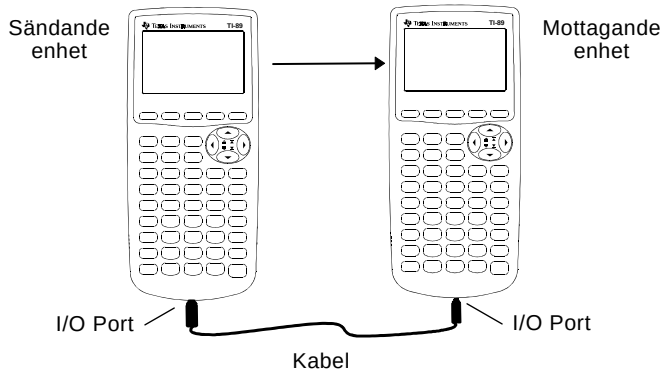


TI-89 och TI-92 Plus levereras med en kabel som du kan använda för att koppla samman två enheter. När enheterna är sammankopplade kan du överföra data mellan dem.

Anslutning före sändning eller mottagning

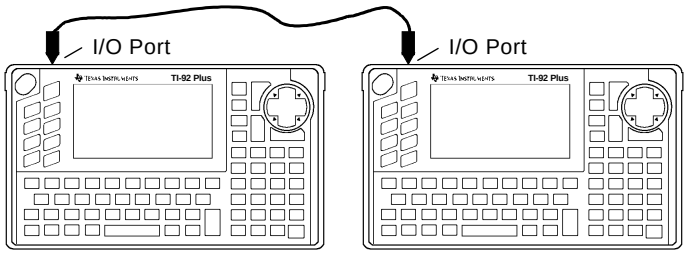
Tryck i sladden ordentligt i I/O-porten på respektive enhet. Respektive enhet kan sedan sända eller ta emot data beroende på inställningarna i fönstret VAR-LINK.

Nedanstående bild visar hur du kopplar samman två TI-89:

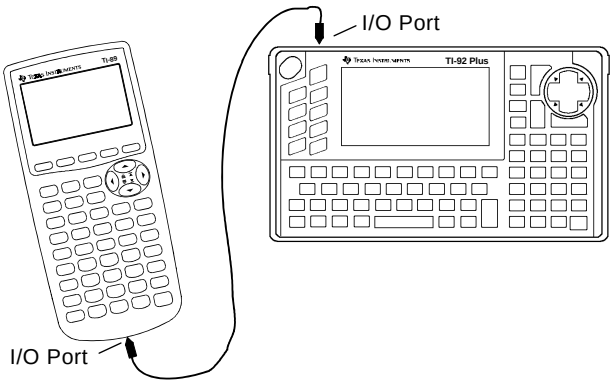


Obs! *Du kan länka en TI-89 eller TI-92 Plus till en annan TI-89, en TI-92 Plus, eller en TI-92, men inte till grafritande räknare som TI-81, TI-82, TI-83, TI-83 Plus, TI-85, eller TI-86.*

Nedanstående bild visar hur du kopplar samman två TI-92 Plus:



Du kan även använda kabeln för TI-GRAPH LINK som levererades med din räknare för att koppla samman en TI-89 och en TI-92 Plus.



Överföra variabler, Flash-program och mappar

Överföring av variabler är ett enkelt sätt att dela alla objekt som listas i fönstret VAR-LINK — funktioner, program, etc. Du kan även överföra Flash-applikationer och mappar.

Ställa in enheterna

De flesta Flash-program kan bara överföras från en TI-89 till en TI-89 eller från en TI-92 Plus till en TI-92 Plus. Du kan inte skicka Flash-applikationer till en TI-92 såvida den inte har en Plus-modul och Advanced Mathematics 2.x programvara (baskod). Mer information om olika kompatibilitetsfrågor finns på sidan 380.

1. Länka samman två enheter enligt beskrivningen på sidan 366.
2. På den **sändande** enheten, tryck på [2nd] [VAR-LINK] för att öppna fönstret VAR-LINK.
3. På den **sändande** enheten, markera de variabler, mappar eller Flash-program som du vill skicka. Stängda mappar expanderas automatiskt då du markerar dem.
   - Om du vill markera en enstaka variabel, eller ett Flash-applikation, flyttar du markören för att markera den.
   - Om du vill markera en enstaka mapp markerar du den och trycker på [F4] för att lägga till en bock (✓) bredvid den. Detta väljer mappen och dess innehåll.
   - Om du vill markera flera variabler, Flash-applikation, eller mappar så markerar du en i taget och trycker på [F4] för att bocka (✓) för dem.
   - Om du vill markera alla variabler, Flash-applikation, eller mappar kan du använda [F5] All 1:Select All.

__Obs!__ Använd [F4] om du vill markera flera variabler, Flash-program eller mappar. Tryck på [F4] igen för att avmarkera objekt du inte vill föra över.

4. På den **mottagande** enheten, tryck på [2nd] [VAR-LINK] för att öppna fönstret VAR-LINK. (Den sändande enheten är kvar i fönstret VAR-LINK.)
5. På både den mottagande *och* den sändande enheten, tryck på [F3] Link för att visa menyalternativen.
6. På den **mottagande** enheten, välj 2:Receive.

   Meddelandet VAR-LINK: WAITING TO RECEIVE och BUSY-indikatorn visas på statusraden på den mottagande enheten.
7. På den **sändande** enheten, välj antingen:
   - 1:Send to TI-89/92 Plus
   — eller —
   - 3:Send to TI-92

   Överföringen påbörjas.

Under överföringen visas en mätare på statusraden på den mottagande enheten. När överföringen är klar uppdateras skärmen VAR-LINK på den mottagande enheten.

Regler för överföring av variabler, Flash-program och mappar

Variabler som inte är låsta eller i användarminnet, och har samma namn på båda enheterna, kommer att skrivas över med data från den sändande enheten.

Låsta variabler, eller variabler i användarminnet, som har samma namn på båda enheterna måste låsas upp, eller hämtas från användarminnet, innan de kan skrivas över med nya data från den sändande enheten.

Du kan låsa Flash-program och mappar, men inte spara dem i användarminnet.

Obs! *Du kan inte skicka en variabel i användarminnet till en TI-92. Du måste först hämta den till arbetsminnet.*

Obs! *Du måste öppna mappar innan du kan överföra dem och deras innehåll.*

Om du väljer:	Sker detta:
Olåst variabel	Variabeln förs över till aktuell mapp och förblir olåst på den mottagande enheten.
Låst variabel	Variabeln förs över till aktuell mapp och förblir låst på den mottagande enheten.
Variabel i användarminnet	Variabeln förs över till aktuell mapp och förblir i användarminnet på den mottagande enheten.
Olåst Flash-program	Om den mottagande enheten har rätt certifikat överförs Flash-programmet. Programmet blir olåst på den mottagande enheten.
Låst Flash-program	Om den mottagande enheten har rätt certifikat överförs Flash-programmet. Programmet blir låst på den mottagande enheten.
Olåst mapp	Mappen och markerat innehåll förs över. Mappen förblir olåst på den mottagande enheten.
Låst mapp	Mappen och markerat innehåll förs över. Mappen förblir låst på den mottagande enheten.

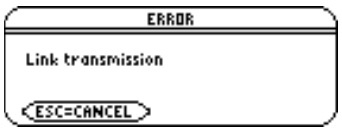
Avbryta en överföring

Gör så här på den sändande eller mottagande enheten:

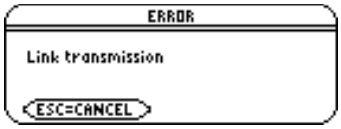
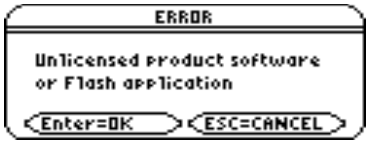
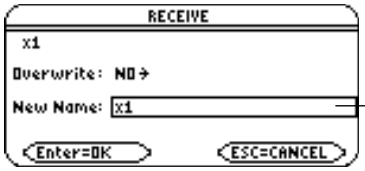
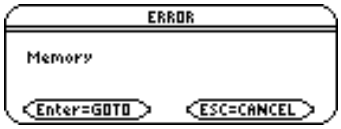
1. Tryck [ON].

   Ett felmeddelande visas.

2. Tryck [ESC] eller [ENTER].



Vanliga felmeddelanden

Visas på den:	Meddelande och beskrivning
Sändande enheten	 Detta visas efter flera sekunder om: • En sladd inte ansluts till den sändande enhetens I/O-port. – eller – • En mottagande enhet inte ansluts till den andra änden av sladden. – eller – • Den mottagande enheten inte är inställd för att ta emot. Tryck på [ESC] eller [ENTER] för att avbryta överföringen.
Sändande enhet	 Den mottagande enheten har inte rätt certifikat för programkoden (baskoden) eller Flash-applikationen som ska föras över.
Mottagande enheten	 New Name är bara aktiverat om du ändrar Overwrite till NO. Den mottagande enheten har en variabel med samma namn som den angivna variabeln som sändes. • Skriv över den befintliga variabeln genom att trycka på [ENTER]. (Som standard är Overwrite = YES.) • Om du vill spara variabeln under ett annat namn ställer du in Overwrite = NO. I inmatningsrutan New Name skriver du ett variabelnamn som inte finns på den mottagande enheten. Tryck därefter på [ENTER] två gånger. • Om du vill hoppa över den här variabeln och fortsätta med nästa ställer du in variabeln Overwrite = SKIP och trycker på [ENTER]. • Du avbryter överföringen genom att trycka på [ESC].
Mottagande enhet	 Den mottagande enheten har inte tillräckligt med ledigt minne. Tryck [ESC] eller [ENTER] för att avbryta överföringen.

Obs! *Den sändande enheten visar kanske inte alltid detta meddelande. I stället visas kanske BUSY-indikatorn tills du avbryter överföringen.*

Ta bort variabler, Flash-applikationen och mappar

1. Tryck på [2nd] [VAR-LINK] för att öppna fönstret VAR-LINK.
2. Markera de variabler, mappar och Flash-applikationer som du vill ta bort.
   - Om du vill markera en enstaka variabel, eller ett Flash-applikation, flyttar du markören för att markera den.
   - Om du vill markera en enstaka mapp markerar du den och trycker på [F4] för att lägga till en bock (✓) bredvid den. Detta väljer mappen och dess innehåll.
   - Om du vill markera flera variabler, Flash-applikationer, eller mappar så markerar du en i taget och trycker på [F4] för att bocka (✓) för dem.
   - Om du vill markera alla variabler, Flash-applikationer, eller mappar kan du använda [F5] All 1:Select All.
3. Tryck på [F1] och välj 1:Delete.
   — eller —
   Tryck på [←]. Ett meddelande visas.
4. Tryck på [ENTER] för att bekräfta borttagningen.

Obs! *You cannot delete the Main folder.*

Obs! *Använd [F4] om du vill markera flera variabler, Flash-program eller mappar. Tryck på [F4] igen för att avmarkera objekt du inte vill ta bort.*

Hämta hem Flash- applikationer

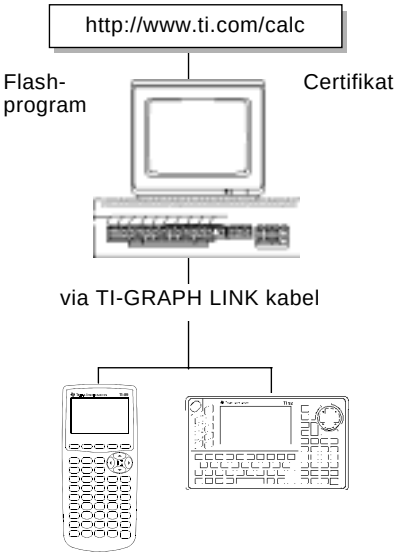
Den senaste informationen om tillgängliga Flash-program finns på Texas Instruments hemsidor:

http://www.ti.com/calc

Du kan även kontakta Texas Instruments enligt instruktionerna i appendix C.

Du kan ladda ner Flash-applikationen och/eller certifikat från Texas Instruments hemsidor till en dator och sedan använda TI-GRAPH LINK för att installera applikationen eller certifikatet på din TI-89 / TI-92 Plus.

Information om installationer finns i början av denna handbok (Flash- applikationen) och i handboken till TI-GRAPH LINK.



Överföra variabler under från ett program

Du kan använda program med kommandona **GetCalc** och **SendCalc** eller **SendChat** när du vill överföra en variabel från en räknare till en annan.

Kommandon – översikt

SendCalc skickar en variabel till utporten där en ansluten räknare kan ta emot den. Den sammankopplade räknaren måste ha grundfönstret aktivt eller använda **GetCalc** från ett program. Om du skickar till en TI-92 kan du dock inte använda **GetCalc** från ett program utan att ett fel uppstår. I dessa fall använder du **SendChat** i stället.

SendChat är ett mer allmänt alternativ till **SendCalc** som kan användas när du vill skicka data till en TI-92 (eller som ett allmänt chat-program som kan kommunicera med en TI-89, TI-92, eller TI-92 Plus). **SendChat** skickar endast variabler som är kompatibla med en TI-92, vilket normalt alla är i ett chatprogram. Du kan dock inte använda **SendChat** om du vill skicka en variabel i användarminnet, en grafdatabas i en TI-89 eller TI-92 Plus, etc.

Programmet "Chat"

I följande program visas hur du kan använda **GetCalc** och **SendCalc**. Programmet skapar två slingor (en för var och en av de två länkade räknarna) så att enheterna kan turas om att ta emot och visa en variabel med namnet msg. Instruktionen **InputStr** låter varje användare mata in ett meddelande i variabeln msg.

```
:Chat()
:Prgm
:ClrIO
:Disp "On first unit to send"," enter 1;","On first to receive,"
:InputStr " enter 0",msg
:If msg="0" Then
:  While true
:    GetCalc msg
:    Disp msg
:    InputStr msg
:    SendChat msg
:  EndWhile
:Else
:  While true
:    InputStr msg
:    SendChat msg
:    GetCalc msg
:    Disp msg
:  EndWhile
:EndIf
:EndPrgm
```

Ställer in denna enhet för att ta emot och visa variabeln msg.

Låter sedan denna användare mata in ett meddelande i msg och sända det.

Slinga utförd av enheten som tar emot det första meddelandet.

Låter denna användare mata in ett meddelande i msg och sända det.

Ställer sedan in denna enhet för att ta emot och visa msg.

Slinga utförd av enheten som sänder det första meddelandet.

För att **GetCalc** och **SendChat** ska synkroniseras ordnas slingorna så att den mottagande enheten kör **GetCalc** samtidigt som den sändande enheten väntar på att användaren ska mata in ett meddelande.

Köra programmet

Denna procedur förutsätter att:

- Två räknare är länkade med anslutningssladden, enligt beskrivningen på sidan 366.
- Chatprogrammet laddas på båda räknarna. (Ett program som laddats i en TI-92 måste använda **SendCalc** i stället för **SendChat**.)

__Obs!__ Information om hur du använder Program Editor finns i kapitel 17.

  – Använd Program Editor på varje räknare för att ange programmet.

  – eller –

  – Starta programmet på en av räknarna och använd sedan VAR-LINK för att överföra programvariabeln till den andra räknaren enligt beskrivningen på sidan 367.

Så här kör du programmet på båda enheter:

1. Ange följande från grundfönstret på båda enheter:

   chat()

2. När varje enhet visar den inledande texten, matar du in 1 eller 0 enligt tabellen nedan.

På den:	**Matar du in:**
Enhet som ska sända det första meddelandet	1 och tryck på [ENTER].
Enhet som ska ta emot det första meddelandet.	0 och tryck på [ENTER].

3. Turas om att skriva meddelandet och trycka på [ENTER] för att sända variabeln msg till den andra enheten.

Stoppa programmet

Eftersom Chat program startar upp oändliga slingor på båda räknarna måste du trycka på [ON] (på båda räknarna) när du vill avbryta programmet. Om du trycker på [ESC] när felmeddelandet visas, stannar programmet i I/O-fönstret. Tryck på [F5] eller [ESC] för att återgå till grundfönstret.

Uppgradera programkod (baskod)

Du kan uppgradera programmen, eller koden, i din TI-89 / TI-92 Plus. Du kan även flytta programkod från en TI-89 eller TI-92 Plus till en annan, förutsatt att den mottagande enheten har nödvänga rättigheter för att använda programvaran.

Uppgraderingar

Uppgraderingar av *programkod* kan vara t ex:

- Underhållsuppgraderingar för tidigare installerad programkod (dessa kan du använda utan kostnad).
- Större uppgraderingar eller nya program/applikationer (dessa måste du normalt köpa). När du vill ladda ner ny programkod från TI:s hemsidor måste du ange ett ID-nummer för din TI-89/TI-92 Plus. (Du hittar information om detta på hemsidorna.) Informationen som du lämnar används för att skapa ett elektroniskt certifikat som specificerar vilken programkod din enhet får använda.

Installation av en uppgradering eller funktionsuppdatering återställer räknarens minne till fabriksinställningarna. Detta innebär att alla användardefinierade variabler, program, listor och Flash-applikationer raderas. Se informationen om batterier (nedan) och “Säkerhetskopiera din enhet före installationen” på sidan 374 innan du uppgraderar programkoden i din räknare.

Viktig information före nedladdning av programkod

Du bör alltid installera nya batterier innan du påbörjar nedladdning av baskod (underhålls eller uppgraderings).

När du är i nedladdningsläge fungerar inte Automatic Power Down™ (APD™). Om du lämnar din räknare i nedladdningsläge en längre tid innan du påbörjar nedladdningen kan batterierna ta slut. Du måste i sådana fall byta batterierna innan du påbörjar nedladdningen.

Du kan även överföra baskod mellan två enheter med hjälp av den speciella kabeln som levererades med din räknare. Om du av misstag avbryter överföringen måste du dock installera ny baskod från en dator. Glöm aldrig att byta batterier före nedladdning.

Kontakta Texas Instruments enligt instruktionerna i appendix C om problem uppstår.

Säkerhetskopiera din enhet före (baskod) installationen

När du installerar ny programkod:

- Tas alla användardefinierade variabler (i både RAM och användarminnet) bort tillsammans med alla användardefinierade funktioner, program och kataloger.
- Kan alla Flash-applikation raderas.
- Återställs alla systemvariabler till sina fabriksinställningar. Detta är samma som att använda fönstret MEMORY för att återställa minnet.

Viktigt: *Sätt alltid i nya batterier före installationen.*

Om du vill behålla några variabler gör du så här *innan du installerar uppdateringen:*

- Flytta variablerna till en annan räknare enligt beskrivningen på sidan 367.

  — eller —

- Använd en TI-GRAPH LINK™ kabel för att koppla samman en dator med räknaren (säljs separat) och programvaran TI-GRAPH LINK (finns utan kostnad på Texas Instruments hemsidor) för att överföra variabler och/eller Flash-program till en dator.

Obs! *Kabeln för att koppla samman en dator med räknaren är inte samma som den som följde med räknaren.*

Om du har en kabel (dator till räknare) för TI-GRAPH LINK och programvara för TI-92, bör du observera att TI-GRAPH LINK *programvara* inte är kompatibel med TI-89 / TI-92 Plus. Kabeln fungerar dock med båda enheterna. Mer information om var du kan köpa TI-GRAPH LINK programvara, eller en kabel för att koppla räknaren till en dator, finns på Texas Instruments hemsidor:

http://www.ti.com/calc/docs/link.htm

eller så kan du kontakta Texas Instruments enligt informationen i appendix C till denna handbok.

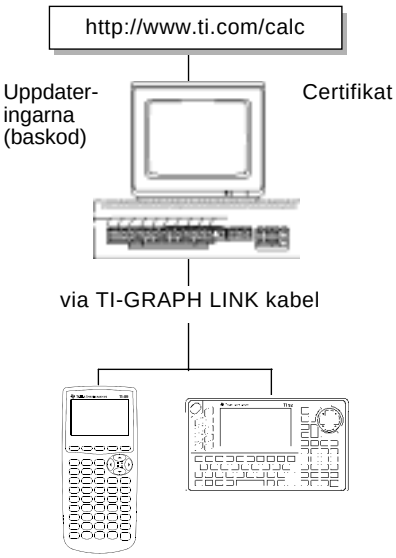
Var finns uppdateringarna? (baskod)

Aktuell information om tillgängliga uppdateringar och hur du installerar dem finns på TI:s hemsidor:

http://www.ti.com/calc

Du kan också kontakta Texas Instruments enligt uppgifterna i appendix C. Du hämtar ny programvara och produktcertifikat från TI:s hemsidor. Använd sedan TI-GRAPH LINK att installera programkoden på din TI-89 / TI-92 Plus.

Mer information finns på hemsidorna.



Överföra programkod (baskod)

Om en sändande TI-89 eller TI-92 Plus har orginalkoden (baskoden) eller en gratis uppgradering behöver en mottagande TI-89 eller TI-92 Plus inget nytt certifikat. Det aktuella certifikatet gäller och det räcker att föra över programkoden.

Om en sändande TI-89 eller TI-92 Plus har en köpt uppgradering måste den mottagande enheten också köpa uppgraderingen. Du kan sedan ladda ner certifikatet och installera det på den mottagande enheten tillsammans med programvaran.

Du kan se vilken version av programkoden du har i din TI-89 / TI-92 Plus genom att trycka på [F1] och välja A:About i grundfönstret.

Programvara (baskod) kan bara överföras från en TI-89 till en TI-89 eller från en TI-92 Plus till en TI-92 Plus. Du kan inte överföra Advanced Mathematics 2.x programkod (baskod) till en TI-92 såvida den inte innehåller en Plus-modul. Mer information om kompatibilitetsfrågor finns på sidan 380.

När du vill överföra programkod (baskod) från en enhet till en annan:

Viktigt: *Kom ihåg att säkerhetskopiera informationen och sätta i nya batterier i varje räknare du uppgraderar.*

1. Koppla samman de två enheterna enligt beskrivningen på sidan 366.
2. På både den mottagande *och* den sändande enheten, tryck på [2nd] [VAR-LINK] för att öppna fönstret VAR-LINK.
3. På både den mottagande *och* den sändande enheten, tryck på [F3] Link för att visa menyalternativen.

Viktigt: *Kontrollera att båda räknarna har fönstret VAR-LINK öppet.*

4. På den **mottagande** enheten, välj 5:Receive Product SW.

   Ett varningsmeddelande visas. Tryck på [ESC] om du vill avbryta eller på [ENTER] för att fortsätta. Om du trycker på [ENTER] visas meddelandet VAR-LINK: WAITING TO RECEIVE och BUSY-indikatorn på statusraden på den mottagande enheten.
5. På den **sändande** enheten, välj 4:Send Product SW.

   Ett varningsmeddelande visas. Tryck på [ESC] om du vill avbryta eller på [ENTER] för att påbörja överföringen.

Överföra programkod (fortsättning)

Under överföringen visas hur mycket som är kvar att kopiera på den mottagande enheten. När överföringen är klar:

- Den sändande enheten går tillbaks till skärmen VAR-LINK.
- Den mottagande enheten går tillbaks till grundfönstret. Skärmens kontrast återställs till fabriksinställningen så du kan behöva använda ◆ − (ljusare) eller ◆ + (mörkare) för att justera kontrasten.

Avbryt aldrig överföringen

När överföringen påbörjas raderas all programkod på den mottagande enheten. Om du avbryter överföringen innan den är klar kommer den mottagande enheten inte att fungera riktigt. I dessa fall måste du installera om programkoden från en dator.

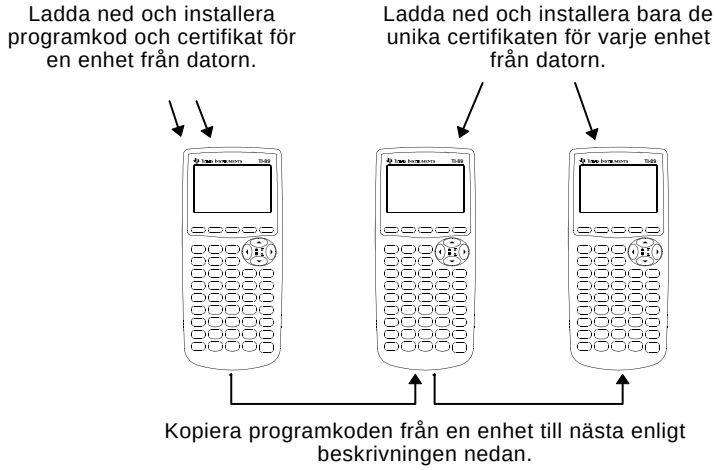
Om du uppgraderar programkod på flera enheter

Om du ska uppgradera flera enheter kan du överföra en underhållsuppgradering från en enhet till en annan i stället för att installera det på var och en via en dator. Underhållsuppgraderingar är gratis och du behöver inget nytt certifikat för att ladda ner eller installera dem.

Obs! *Det finns även gruppcertifikat. Se sidan 378.*

Innan du installerar en köpt funktionsuppgradering måste varje TI-89 eller TI-92 Plus ha ett eget, unikt, certifikat. Under nedladdningen och installationen kan du välja både programkod och certifikat eller bara certifikat. I bilden nedan visas det effektivaste sättet att förbereda flera enheter för uppgradering.

Tips! *Normalt är det mycket snabbare att skicka programkod från en enhet till en annan än det är att installera programkoden från en dator.*



Förberedelse av flera TI-92 Plus för en köpt funktionsuppgradering fungerar på samma sätt som ovan.

Felmeddelanden

De flesta felmeddelanden visas på den sändande enheten. Beroende på när i överföringsprocessen ett fel uppstår kan du ibland även få upp felmeddelanden på den mottagande enheten.

Felmeddelande	Beskrivning
ERROR Link transmission ESC=CANCEL	Enheterna är inte riktigt ihopkopplade, eller så är den mottagande enheten inte inställd för mottagning.
ERROR Unlicensed product code ESC=CANCEL	Certifikatet på den mottagande enheten är inte giltigt för den aktuella produkten på den sändande enheten. Du måste hämta ett certifikat och installera det enligt den tidigare beskrivningen.
ERROR Signature error ESC=CANCEL	Ett fel uppstod under överföringen. Programkoden på den mottagande enheten är skadad. Du måste installera om programmet från en dator.
ERROR Batteries too low for sending/receiving product code ESC=CANCEL	Byt ut batterierna på den enhet som visar detta felmeddelande.

I fönstret VAR-LINK kan du använda F3 6:Send ID List för att samla in ID-nummer från olika TI-89 / TI-92 Plus räknare.

ID-listor och gruppcertifikat

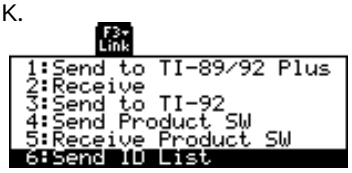
Funktionen för ID-listor ger dig ett enkelt sätt att samla in ID-nummer för flera räknare vid inköp av program för flera användare. När du har samlat in ID-numren skickar du dem till Texas Instruments som utfärdar ett gruppcertifikat.

Med ett gruppcertifikat kan du distribuera inköpt programvara till flera TI-89 / TI-92 Plus. Program kan laddas, tas bort och laddas tillbaks till alla räknare som finns listade i gruppcertifikatet. Du kan när du vill lägga till nya ID-nummer och/eller nya programvaror till gruppcertifikatet.

Samla in ID-listor

Du kan använda en räknare för att samla in alla ID-nummer, eller flera räknare och sedan samla ihop listorna på en räknare.

När du vill överföra ett ID-nummer från en räknare till en annan måste du först koppla samman dem med kabeln för räknare till räknare som medföljde din TI-89 / TI-92 Plus. Se bilden på sidan 366.

Steg:	På:	Gör så här:
1.	Insamlande enhet (Mottagande enhet)	Öppna grundfönstret. Tryck på: **TI-89**: HOME **TI-92 Plus**: ♦ [HOME]
2.	Sändande enhet	a. Tryck på 2nd[VAR-LINK]för att öppna fönstret VAR-LINK. b. Tryck på F3 Link och välj 6:Send ID List.  Den sändande enheten lägger till en *kopia* av sitt unika ID-nummer till den insamlande enhetens ID-lista. Den sändande enheten behåller alltid sitt eget ID-nummer som inte kan tas bort från räknaren.
3.	Extra enheter	Upprepa steg 1 och 2 tills alla ID-nummer är samlade på en räknare. Beroende på tillgängligt minne hos den insamlande räknaren kan du samla in mer än 4000 ID-nummer.

Obs! *Du kan inte visa ID-listan på sändande eller mottagande enhet.*

Obs! *Varje gång en ID-lista överförs från en enhet till en annan så raderas den från den sändande enheten.*

Obs! *Om ett ID-nummer hämtas från en räknare två gånger tas dubletten automatiskt bort från listan.*

Överföra ID-listan till en dator

När alla ID-nummer är ihopsamlade på en räknare använder du TI-GRAPH LINK™ och en sladd för att koppla samman räknaren med en dator (säljs separat) för att lagra ID-listan på en dator. ID-listan kan sedan skickas med e-post, eller skrivas ut och faxas/skickas, till Texas Instruments.

Utförliga instruktioner om hur du överför en ID-lista från en TI-89 / TI-92 Plus till en dator finns i handboken för TI-GRAPH LINK. Kortfattat gör du som följer:

1. Anslut kabeln till datorn och till den räknare som innehåller ID-listan.
2. Starta TI-GRAPH LINK på datorn.
3. Öppna grundfönstret på räknaren. Tryck på:
   TI-89: [HOME]
   TI-92 Plus: [♦] [HOME]
4. Välj Get ID List på menyn Link i TI-GRAPH LINK.
5. Markera en katalog på datorn där du vill spara ID-listan och gör en notering så att du hittar igen den senare.
6. Klicka på OK för att spara ID-listan på datorns hårddisk.

ID-listan finns även kvar på räknaren tills du tar bort den eller överför den till en annan TI-89 / TI-92 Plus.

Radera ID-listan

ID-listan finns kvar på räknaren även efter du har överfört den till en dator. Du kan använda räknaren till att ladda upp ID-listan till flera datorer.

Gör så här om du vill radera ID-listan från räknaren:

1. Tryck på [2nd] [VAR-LINK] för att öppna fönstret VAR-LINK.
2. Tryck på [F1] Manage och välj A:Clear ID List.

Eftersom en del funktioner ändrats och förbättrats i TI-89 och TI-92 Plus finns det viss inkompatibilitet med TI-92. När så har varit möjligt fungerar dataöverföringar mellan enheterna utan problem men vissa skillnader återstår.

Huvudsakliga typer av inkompatibilitet

Alla data är kompatibla mellan TI-89 och TI-92 Plus, men en del program som fungerar på den ena kanske inte fungerar på den andra pga olika skärmstorlekar och tangentbord. TI-89 har inte heller inbyggda funktioner för Geometry som TI-92 Plus. Geometriska figurer och makron kan överföras till en TI-89, men de kan inte användas för vidare beräkningar.

Jämfört med en TI-92 har TI-89 och TI-92 Plus:

- funktioner, instruktioner och systemvariabler som inte finns på TI-92.
- möjligheten att använda samma variabler för att definiera och sedan köra användardefinierade program och funktioner. Du kan t ex definiera en funktion i termer av x och sedan utvärdera funktionen med ett uttryck som innehåller x. Detta orsakar ett fel av typen Circular definition på en TI-92. Mer information finns i kapitel 17: Programmering.
- Hantering av lokala variabler skiljer från TI-92. Se kapitel 17: Programmering för mer information.

Text alt. kompilerat (token)

När du skapar en funktion eller ett program är det i textform tills du kör det. När det körs konverteras det automatiskt till kompilerad form.

- Data i textformat kan delas mellan en TI-89, TI-92 Plus, och en TI-92. När du överför data i textform kan det hända att du inte får något felmeddelande även om funktionen inte ger samma resultat på annan räknare.

Obs! *Om du redigerar en funktion eller ett program som är kompilerat återgår det till textform tills nästa gång det körs.*

- Data i kompilerad form innehåller information som beskriver inkluderad funktionalitet. TI-89 och TI-92 Plus använder samma kompilerade form, men TI-92 är annorlunda.
  - Om du försöker att överföra kompilerade funktioner, program eller andra datatyper från en TI-89 eller TI-92 Plus till en TI-92 kontrollerar TI-89 eller TI-92 Plus automatiskt om funktionen kommer att fungera på en TI-92. Om inte så avbryts överföringen. Detta är huvudsakligen för din egen säkerhet eftersom din TI-92 kan låsa sig om data med inkompatibla funktioner överförs.
  - Även om kompilerade data överförs är det inte säkert att samma resultat fås på den andra räknaren.

TI-92 till TI-89 eller TI-92 Plus

Alla användardefinierade variabler (inklusive funktioner och program) kan skickas från en TI-92 till en TI-89 eller TI-92 Plus. De kan däremot bete sig lite annorlunda. Exempel:

- Konflikter mellan nya systemvariabler i TI-89 och TI-92 Plus funktioner, instruktionsnamn i TI-89 och användardefinierade namn i TI-92.
- Program eller funktioner som använder symboliska lokala variabler. På TI-89 och TI-92 Plus, måste lokala variabler initieras med ett värde innan de kan användas (dvs en lokal variabel kan inte användas symboliskt). Om du vill använda symboliska variabler måste de vara globalt definierade. Detta inkluderar även program som utvärderar strängar liksom lokala variabler som är symboliska, t ex **expr()**.

TI-89 eller TI-92 Plus till TI-92

Funktionalitet som finns i TI-89 eller TI-92 Plus och INTE i TI-92 kommer INTE att fungera som förväntat på en TI-92. I vissa fall (textform) kan data överföras men ett fel uppstår då det körs på en TI-92. I andra fall (kompilerad form) kan data inte överföras till en TI-92.

Om data har samma funktionalitet som på en TI-92 kan det troligen överföras och köras på en TI-92 med samma resultat. Det finns dock vissa undantag:

- Grafdatabaser (GDB) kan inte överföras då TI-89 och TI-92 Plus använder en ny GDB-struktur som innehåller mer information än TI-92 GDB.
- TI-89 och TI-92 Plus har utökade möjligheter att använda variabler för att definiera och utvärdera användardefinierade funktioner eller program. Till exempel: du kan nu definiera en funktion i termer av en variabel, t ex x, och sedan utvärdera funktionen med ett uttryck som innehåller samma variabel. Detta orsakar ett fel av typen Circular definition med TI-92.
- En del existerande funktioner och instruktioner i TI-92 har utökad funktionalitet på en TI-89 och TI-92 Plus (t ex **NewData**, **setMode()** och matrisfunktioner som kan använda den alternativa feltoleransen). Dessa funktioner och instruktioner kanske inte överförs alls eller orsakar ett fel på en TI-92.
- Sparade variabler överförs inte till en TI-92. Ladda variablerna till RAM från användarminnet innan du försöker föra över dem.
- Datavariabler som innehåller en huvuddel skickas inte. De utan huvuddel skickas endast om innehållet är kompatibelt med TI-92.
- Programvaruuppgraderingar (baskod).
- Flash-applikationen.

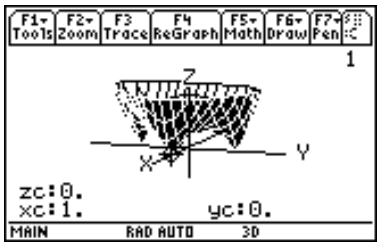
Du kan uppgradera en TI-92 till en TI-92 Plus genom att installera en TI-92 Plus-modul. Se Texas Instruments hemsidor **http://www.ti.com/calc** för mer information.

Övningar

23

Analysera problemet med en påle som ska runt ett hörn 384
Härledning av formeln för andragradsekvationens rötter 386
Utforska en matris 388
Utforska cos(x) = sin(x) 389
Hitta den minsta ytarean på en parallellepiped 390
Köra ett självstudieskript med Text Editor 392
Dela upp en rationell funktion 394
Studera statistik: Filtrera data med kategorier 396
Ett CBL program för TI-89 / TI-92 Plus 399
Analysera färden för en ivägslagen boll 400
Visualisera komplexa rötter till ett tredjegradspolynom 402
Lösa ett allmänt sparproblem 404
Beräkna betalning 405
Hitta rationella, reella och komplexa faktorer 406
Simulering av dragning utan återläggning 407

Detta kapitel innehåller övningar som beskriver hur TI-89 / TI-92 Plus kan användas för att lösa, analysera och visualisera olika matematiska problem.



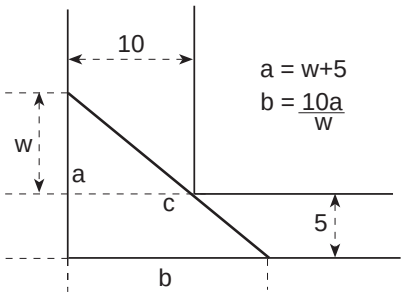
Analysera problemet med en påle som ska runt ett hörn

En tio meter bred hall möter en fem meter bred hall i hörnet av en byggnad. Hitta den maximala längden på en påle som kan flyttas runt hörnet utan att pålen lutas.

Maximala längden på pålen i hallen

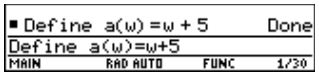
Maxlängden på en påle, c, är det kortaste linjesegment som vidrör det inre hörnet och samtidigt de motstående väggarna, så som visas i bilden nedan.

Tips! Använd proportionella sidor och Pythagoras sats för att räkna ut längden c med avseende på w. Hitta sedan nollställena för förstaderivatan av c(w). Det minsta värdet av c(w) är pålens maximala längd.

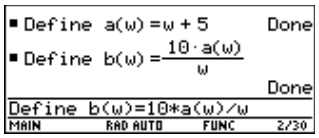


Tips: När du vill definiera en funktion använder du namn med flera tecken då du skapar definitionen.

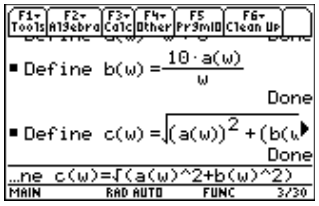
1. Skriv uttrycket för sidan a uttryckt i w och spara det i a(w).



2. Skriv uttrycket för sidan b uttryckt i w och spara det i b(w).

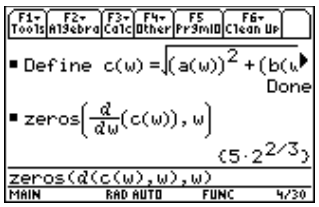


3. **Definiera** uttrycket för sidan c i termer av w och lagra det i c(w) Mata in: Define c(w)= √(a(w)^2+b(w)^2)



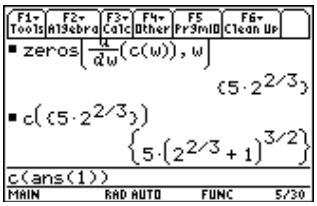
Obs! Maxlängden på pålen är minimivärdet för c(w).

4. Använd kommandot **zeros()** för att beräkna nollställena av förstaderivatan av c(w) för att hitta det minsta värdet av c(w).



5. Beräkna den exakta maxlängden av pålen.

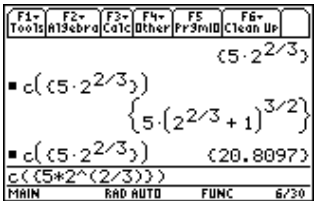
   Skriv: c([2nd][ANS])



Tips! *Klipp ut och klistra in resultatet från steg 4 till inmatningsraden inom parentes för* c( ) *och tryck på* [◆] [ENTER].

6. Beräkna den ungefärliga maxlängden för pålen.

   Resultat:
   20,8097 meter.



Härledning av formeln för andragradsekvationens rötter

Denna övning visar hur du här leder formeln:

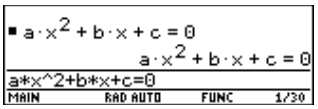
$$x = \frac{^{-}b \pm \sqrt{b^2 - 4ac}}{2a}$$

I kapitel 3: Algebra, finns detaljerad information om hur du använder kommandona i det här exemplet.

Utföra beräkningar för att härleda formeln för andragradsekvationens rötter

Följ stegen nedan för att härleda formeln för andragradsekvationens rötter genom kvadratkomplettering av den generella andragradsekvationen.

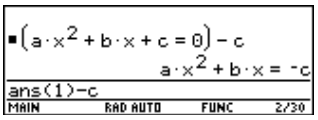
1. Töm alla variabelnamn med ett tecken i aktuell mapp.
   TI-89: [2nd] [F6]
   TI-92 Plus: [F6]
   Välj 1:Clear a-z och tryck på [ENTER] för att bekräfta.



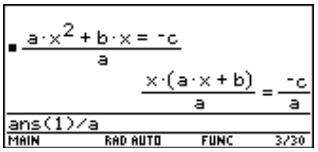
2. Skriv följande generella andragradsekvation i grundfönstret: $ax^2+bx+c=0$.

__Obs!__ I detta exempel används resultatet av det senaste svaret för att utföra beräkningarna. Med denna funktion behöver du inte trycka på så många tangenter, vilket minskar risken för fel.

3. Dra bort c från båda sidor av ekvationen.
   TI-89: [2nd] [ANS] [−] [alpha] C
   TI-92 Plus: [2nd] [ANS] [−] C

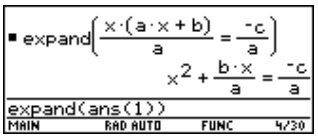


4. Dividera båda sidor av ekvationen med koefficienten a.

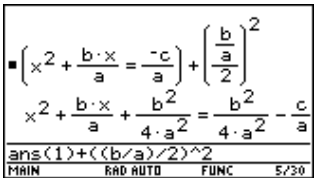


__Tips!__ Fortsätt att använda det senaste svaret ([2nd] [ANS]) på samma sätt som i steg 3 i steg 4 till och med steg 9.

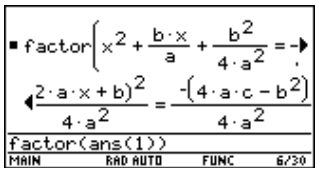
5. Använd funktion **expand()** för att utveckla det senaste svaret.



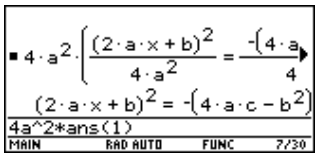
6. Kvadratkomplettera genom att lägga till $((b/a)/2)^2$ till båda sidor av ekvationen.



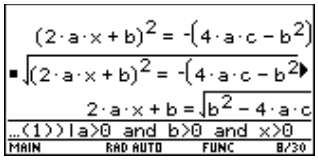
7. Faktorisera resultatet med funktionen **factor()**.



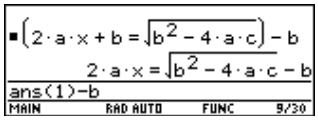
8. Multiplicera båda sidor av ekvationen med $4a^2$.



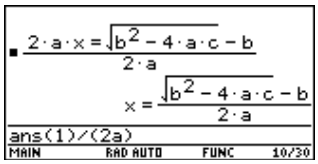
9. Dra roten ur båda sidor av ekvationen med följande begränsningar:
a>0 och b>0 och x>0.



10. Lös ut x genom att subtrahera b från båda sidor och sedan dividera med 2a.



Obs! *Detta är bara en av de två lösningarna på den generella andragrads-ekvationen beroende på begränsningarna i steg 9.*

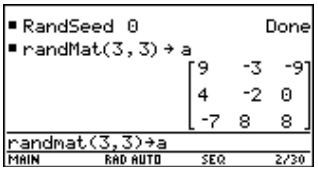


Denna övning visar hur du utför flera olika matrisoperationer.

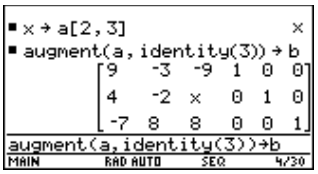
Utforska en 3x3-matris

Utför följande steg för att generera en slumpmatris, utvidga den med enhetsmatrisen och lös sedan matrisen för hitta ett ogiltigt värde på inversen.

1. Använd **RandSeed** i grundfönstret för att ställa in slumptalsgeneratorn till standard och använd därefter **randMat()** för att skapa en 3x3-slumpmatris och spara den i a.



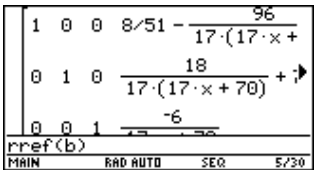
2. Byt ut matriselementet [2,3] med variabeln x och använd sedan funktion **augment()** för att utvidga a med en enhetsmatris av ordning 3. Spara resultatet i b.



__Tips!__ Använd markören i historiklistan för att rulla resultatet.

3. Använd **rref()** för att radreducera matrisen b:

   Resultatet kommer att ha enhetsmatrisen i de tre första kolumnerna och a^⁻1 i de tre sista kolumnerna.

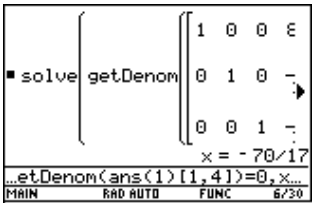


__Tips!__ Använd markören i historiklistan för att rulla resultatet.

4. Lös för det värde på x som gör så att inversen på matrisen blir ogiltig.

   Skriv: solve(getDenom([2nd] [ANS][1,4])=0,x)

   Resultat: x=⁻70/17



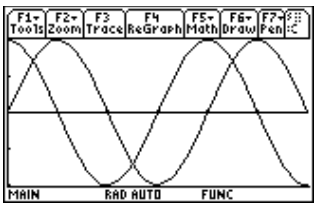
Utforska cos(x) = sin(x)

Denna övning använder två metoder för att beräkna cos(x) = sin(x) för värden hos x mellan 0 och 3π.

Metod 1: Grafplottning

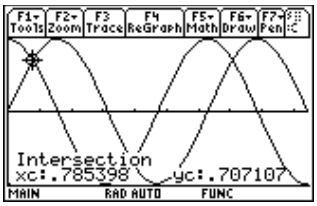
Följ stegen nedan för att se var graferna av funktionerna y1(x)=cos(x) och y2(x)=sin(x) korsar varandra.

1. Mata in y1(x)=cos(x) och y2(x)=sin(x) i Y= Editor.
2. I Window Editor ställer du in xmin=0 och xmax=3π.
3. Tryck på [F2] och välj A:ZoomFit.



Tips! Tryck på [F5] och välj 5:Intersection. *Svara på meddelandena för att välja de två kurvorna och den övre och undre gränsen för skärningspunkt* A.

4. Hitta skärningspunkterna för de två funktionerna.
5. Lägg märke till x- och y-koordinaterna. (Upprepa steg 4 och 5 för att hitta de andra skärningspunkterna.)

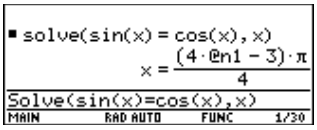


Metod 2: Algebra

Följ stegen nedan för att lösa ekvationen sin(x)=cos(x) med avseende på f(x).

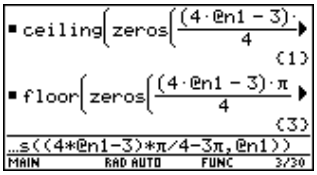
1. Mata in solve(sin(x)= cos(x),x) i grundfönstret.

   Lösningen för x är där @n1 är ett heltal.



Tips! Flytta markören till historiklistan för att markera det sista resultatet. Tryck på [ENTER] för att kopiera resultatet av den generella lösningen.

2. Använd kommandona **ceiling()** och **floor()** för att hitta det största respektive det minsta värdet på @n1 för skärningspunkterna enligt bilden.

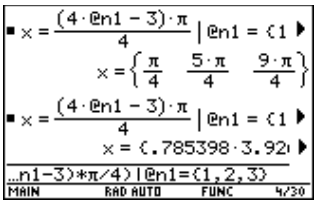


Tips! När du vill skriva in operatorn "with":

TI-89: [I]
TI-92 Plus: [2nd] ['']

3. Mata in den generella lösningen för x och ange begränsningen för @n1 enligt bilden.

   Jämför resultatet med metod 1.



Hitta den minsta ytarean på en parallellepiped

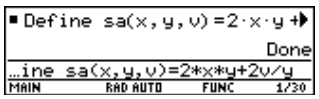
Denna övning visar hur du beräknar den minsta ytan hos en parallellepiped med en konstant volym V. I kapitel 3: Algebra och kapitel 10: 3D-plottning finns detaljerad information om de steg som används i detta exempel.

Utforska en 3D-graf av ytarean av en parallellepiped

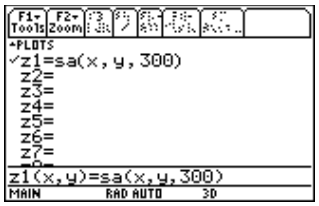
Följ stegen nedan för att definiera en funktion för ytarea av en parallellepiped, plotta en 3D-graf och använda verktyget **Trace** för att hitta en punkt nära den minsta ytarean.

1. Definiera funktionen sa(x,y,v) för ytarean av en parallellepiped, i grundfönstret.

   Skriv: define sa(x,y,v)=2∗x∗y+2∗v/x+2∗v/y

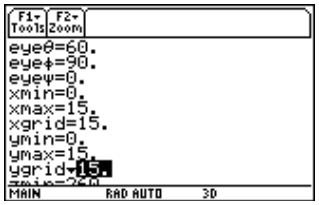


2. Välj grafläget 3D. Mata därefter in funktionen för z1(x,y), så som visas i detta exempel med volymen v=300.

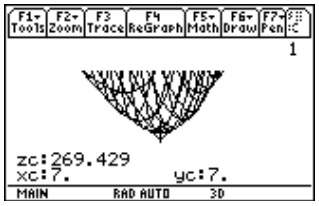


3. Ställ in Window-variablerna till:

eye=	[60,90,0]
x=	[0,15,15]
y=	[0,15,15]
z=	[260,300]
ncontour=	[5]



4. Plotta funktionen och använd **Trace** för att gå till den punkt som är närmast minimivärdet för funktionen.

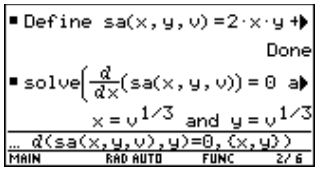


Hitta den minsta ytarean analytiskt

Följ stegen nedan för att lösa problemet analytiskt i grundfönstret.

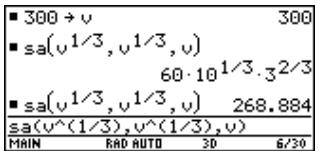
1. Lös ut x och y uttryckt i v.

   Skriv:
   solve(d(sa(x,y,v),x)=0 and (d(sa(x,y,v),y)=0, {x,y})



2. Hitta den minsta ytarean då v = 300.

   Skriv: 300→v
   Skriv: sa(v^(1/3), v^(1/3),v)



Tips! *Tryck på* [ENTER] *för att visa det exakta resultatet. Tryck på* [◆] [ENTER] *för att visa det ungefärliga resultatet i decimalform.*

Denna övning visar hur du använder textredigeraren för att köra ett övningsskript. I kapitel 18: Text Editor finns detaljerad information om textfunktioner.

Köra ett självstudieskript

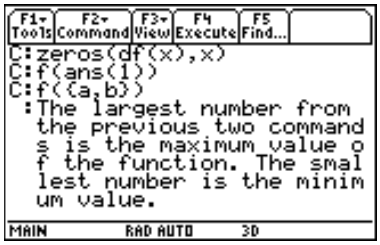
Följ stegen nedan för att skriva ett skript med hjälp av Text Editor, testa varje rad och observera resultaten i historiklistan i grundfönstret.

1. Öppna Text Editor och skapa en ny variabel med namnet demo1.

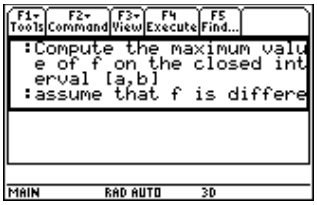
Obs! *Kommandomärket* "C" *är tillgängligt på menyn* F2 1:Command.

2. Skriv följande rader i Text Editor.

```
  : Compute the maximum value of f on the closed interval [a,b]
  : assume that f is differentiable on [a,b]
C : define f(x)=x^3- 2x^2+x- 7
C : 1→a:3.22→b
C : d(f(x),x)→df(x)
C : zeros(df(x),x)
C : f(ans(1))
C : f({a,b})
  : The largest number from the previous two commands is the maximum
  value of the function. The smallest number is the minimum value.
```

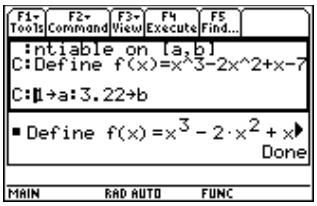


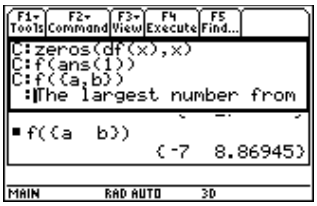
3. Tryck på F3 och välj 1:Script view för att visa Text Editor och grundfönstret i ett delat fönster. Flytta markören till den första raden i Text Editor.



Obs! *Tryck på* F3 *och välj* 2:Clear split *för att återgå till Text Editor i full storlek.*

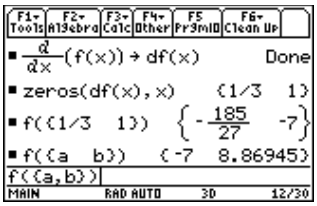
4. Tryck på F4 upprepade gånger för att köra varje rad i skriptet, en åt gången.





Tips! *Tryck på* 2nd QUIT *två gånger för att visa grundfönstret.*

5. Om du vill visa resultatet av skriptet i ett fönster i full storlek går du till grundfönstret.



Dela upp en rationell funktion

Denna övning beskriver vad som händer när en rationell funktion bryts ned till en kvot och en rest. I kapitel 6: Grundläggande funktionsplottning och kapitel 3: Algebra finns detaljerad information om stegen som används i detta exempel.

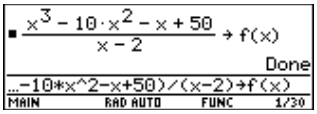
Dela upp en rationell funktion

Obs! *Verkliga inmatningar visas mot svart bakgrund i exempelfönstren.*

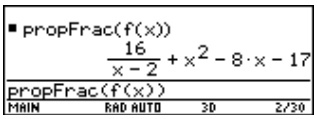
Så här undersöker du uppdelningen av den rationella funktionen $f(x)=(x^3-10x^2-x+50)/(x-2)$ på en graf:

1. Ange den rationella funktionen i grundfönstret så som visas på inmatningsraden och spara den i funktionen f(x).

   Skriv: (x^3– 10x^2– x+50)/(x– 2)➔f(x)



2. Använd funktion **propFrac** för att dela upp funktionen i en kvot och en rest.

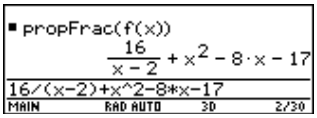


Tips! *Flytta markören till historiklistan för att markera det senaste resultatet. Tryck på* [ENTER] *för att kopiera det till inmatningsraden.*

3. Kopiera det senaste resultatet till inmatningsraden.

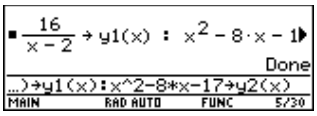
   – eller –

   Skriv: 16/(x–2)+x^2– 8*x–17

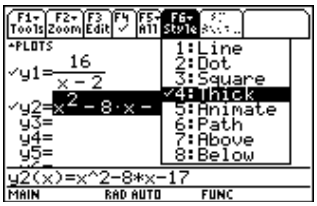


4. Redigera det senaste resultatet på inmatningsraden. Spara resten i y1(x) och kvoten i y2(x), så som visas i bilden.

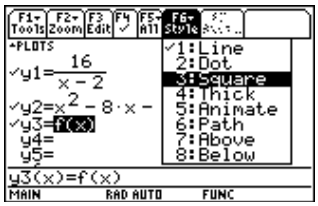
   Skriv: 16/(x–2)➔y1(x): x^2–8*x–17➔y2(x)



5. Välj det tjocka grafformatet för y2(x) i Y= Editor.

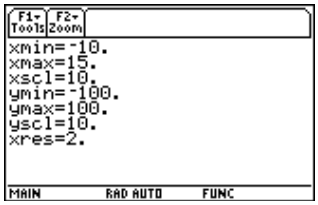


6. Lägg till den ursprungliga funktionen f(x) till y3(x) och välj grafstilen Square.



7. Ställ in Window-variablerna i Window Editor till:

   x= [⁻10,15,10]
   y= [⁻100,100,10]

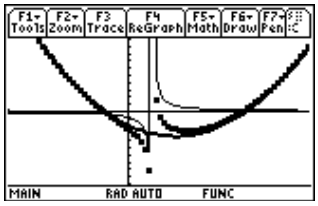


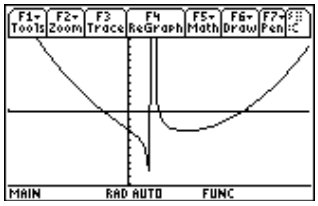
Obs! *Kontrollera att grafläget är inställt till* Function.

8. Rita grafen.

Observera att det globala uppförandet av funktionen f(x) i stora drag motsvaras av andragradsekvationen y2(x). Det rationella uttrycket är i stort sett en andragradsfunktion då x går mot stora värden, både positiva och negativa.

Den undre grafen är y3(x)=f(x) plottad separat med en tunn linje.





Studera statistik: Filtrera data med kategorier

Denna övning omfattar en statistisk studie av skolelevers vikt med filtrering av data med hjälp av kategorier. I kapitel 15: Data/Matrix Editor och kapitel 16: Plotta statistiska data finns detaljerad information om hur du använder kommandona i detta exempel.

Filtrera data med kategorier

Varje student placeras i en av åtta kategorier, beroende på kön och vilken årskurs de går i (förstaårsstudent, andraårsstudent, tredjeårsstudent eller sistaårsstudent). Informationen (vikt i pund) och respektive kategorier matas in i Data/Matrix Editor.

Tabell 1: Kategori kontra beskrivning

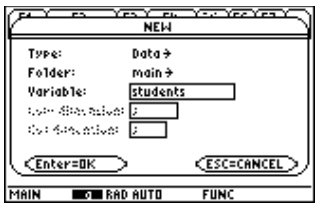
Kategori (C2)	Årskurs och kön
1	Förstaårsstudent pojkar
2	Förstaårsstudent flickor
3	Andraårsstudent pojkar
4	Andraårsstudent flickor
5	Tredjeårsstudent pojkar
6	Tredjeårsstudent flickor
7	Sistaårsstudent pojkar
8	Sistaårsstudent flickor

Tabell 2: C1 (vikten hos varje student i pund) kontra C2 (kategori)

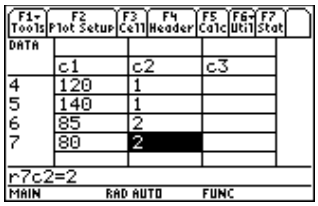
C1	C2	C1	C2	C1	C2	C1	C2
110	1	115	3	130	5	145	7
125	1	135	3	145	5	160	7
105	1	110	3	140	5	165	7
120	1	130	3	145	5	170	7
140	1	150	3	165	5	190	7
85	2	90	4	100	6	110	8
80	2	95	4	105	6	115	8
90	2	85	4	115	6	125	8
80	2	100	4	110	6	120	8
95	2	95	4	120	6	125	8

Följ stegen nedan för att jämföra studenternas vikt med deras årskurs.

1. Starta Data/Matrix Editor och skapa en ny datavariabel med namnet students.

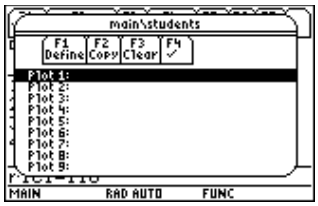


2. Mata in informationen och kategorierna från tabell 2 till kolumnerna c1 och c2.

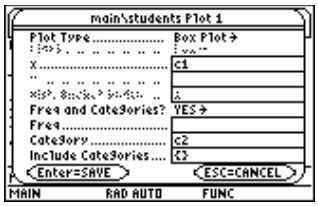


***Obs!** Ställ in flera lådagram för att jämföra olika delar av hela informationsmängden.*

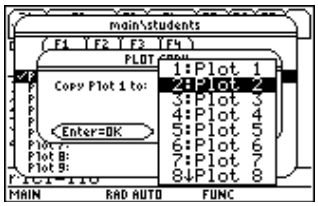
3. Öppna menyn F2 Plot Setup.



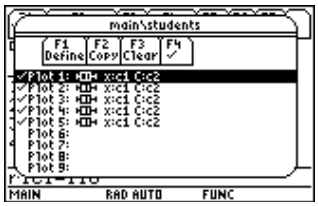
4. Definiera plottnings- och filterparametrarna för Plot 1, som visas i exemplet.



5. Kopiera Plot 1 till Plot 2.

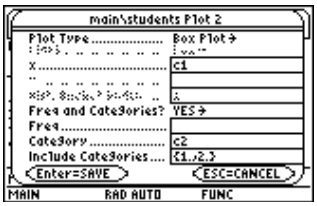


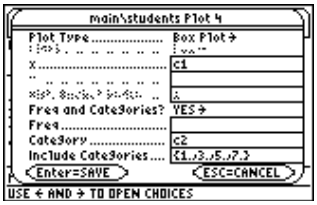
6. Upprepa steg 5 och kopiera Plot 1 till Plot 3, Plot 4 och Plot 5.



7. Tryck på F1 och ändra alternativet Include Categories för Plot 2 till och med Plot 5 till följande:

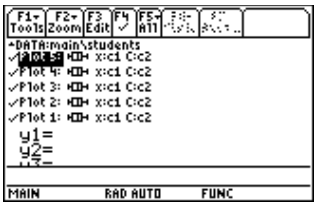
   Plot 2: {1,2}
   (förstaårsstudent pojkar och flickor)
   Plot 3: {7,8}
   (sistaårsstudent pojkar och flickor)
   Plot 4: {1,3,5,7}
   (alla pojkar)
   Plot 5: {2,4,6,8}
   (alla flickor)



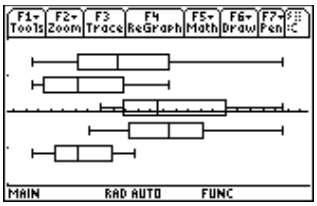


***Obs!** Du bör endast markera Plot 1 till och med Plot 5.*

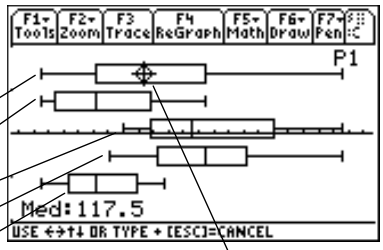
8. Avmarkera alla funktioner som kan vara markerade från tidigare övningar i Y = Editor.



9. Visa plottningarna genom att trycka på F2 och välja 9:Zoomdata.



10. Använd verktyget **Trace** för att jämföra medianvikt för olika kategorier av studenter.



Ett CBL program för TI-89 / TI-92 Plus

Denna övning omfattar ett program som kan användas när din TI-89 / TI-92 Plus är ansluten till en CBL-enhet (Calculator-Based Laboratory™). Programmet fungerar med experimentet "Newton's Law of Cooling" och, med vissa mindre ändringar, experimentet "Coffee To Go" i *CBL System Experiment Workbook*. Du kan använda din dator när du vill skriva in längre texter och sedan använda TI-GRAPH LINK för att skicka den till din TI-89 / TI-92 Plus Fler TI-89 / TI-92 Plus CBL-program hittar du på TI:s hemsidor: http://www.ti.com/calc/cbl

Programinstruktion	Beskrivning
:cooltemp()	Programnamn
:Prgm	
:Local i	Deklarerar en lokal variabel; existerar endast då programmet körs.
:setMode("Graph","FUNCTION")	Ställer in räknaren för funktionsplottning.
:PlotsOff	Stänger av eventuella tidigare plottningar.
:FnOff	Stänger av eventuella tidigare funktioner
:ClrDraw	Tar bort föregående ritobjekt från graffönstret.
:ClrGraph	Tar bort föregående grafer.
:ClrIO	Rensar Program I/O-fönstret.
:-10→xmin:99→xmax:10→xscl	Ställer in Window-variabler.
:ú20→ymin:100→ymax:10→yscl	
:{0}→data	Skapar och/eller tar bort en lista med namnet data.
:{0}→time	Skapar och/eller tar bort en lista med namnet time.
:Send{1,0}	Skickar ett kommando som rensar CBL-enheten.
:Send{1,2,1}	Ställer in kanal 2 i CBL:n till AutoID för att avläsa temperatur.
:Disp "Press ENTER to start"	Uppmanar användaren att trycka på [ENTER].
:Disp "graphingTemperature."	
:Pause	
:PtText "TEMP(C)",2,99	Väntar tills användaren är klar att börja.
:PtText "T(S)",80,-5	Etiketterar grafens y-axel.
:Send{3,1,-1,0}	Etiketterar grafens x-axel.
:	Skickar kommandot **Trigger** till CBL:n; samla in data i
:For i,1,99	realtid.
:Get data[i]	Upprepar nästa två instruktioner för 99 temperaturavläsningar.
:PtOn i,data[i]	Hämtar en temperatur från CBL:n och spara den i en lista.
:EndFor	Plottar temperaturdata som en graf.
:seq(i,i,1,99,1) →time	
:NewPlot 1,1,time,data,,,,4	Skapar en lista som ska innehålla time- eller data-avläsningsnummer.
:DispG	Plottar time och data med hjälp av NewPlot och verktyget **Trace**.
:PtText "TEMP(C)",2,99	Visar grafen.
:PtText "T(S)",80,-5	Visar åter namn (etikett) på axlarna.
:EndPrgm	
	Stoppar programmet.

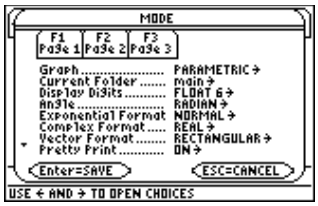
Du kan även använda Calculator-Based Ranger™ (CBR™) när du vill utforska de matematiska och fysikaliska sambanden mellan sträcka, hastighet, acceleration och tid med data insamlade från dina aktiviteter.

Denna övning använder en delad skärm för att visa en graf i parameter form tillsammans med en tabell för att beskriva rörelsen hos en boll.

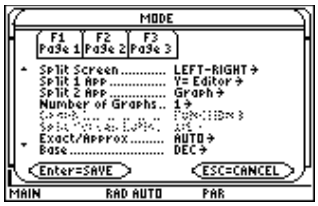
Ställa in en parametergraf och tabell

Följ stegen nedan för att granska färden för en ivägskickad boll som har en utgångshastighet på 95 fot/sek och en utgångsvinkel på 32 grader.

1. Ställ in lägena för Page 1, som i bilden.

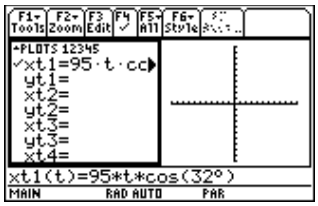


2. Ställ in lägena för Page 2, som i bilden.

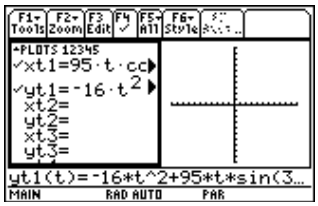


***Tips!** Tryck på* [2nd] [°] *för att skriva en gradsymbol.*

3. På vänstra sidan i Y= Editor matar du in ekvationen xt1(t) som avståndet till bollen vid en tidpunkt t.
   xt1(t)=95∗t∗cos(32°)

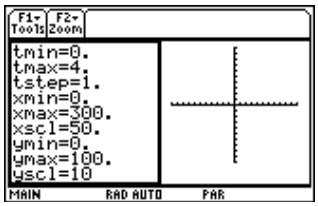


4. Mata in ekvationen yt1(t) som bollens höjd vid en tidpunkt t.
   yt1(t)=-16∗t^2+95∗t∗sin(32°)



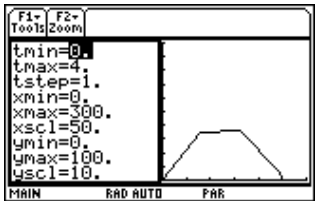
5. Ställ in Window-variablerna till:

   t values= [0,4,.1]
   x values= [0,300,50]
   y values= [0,100,10]



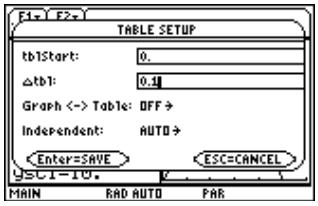
Tips! *Tryck på* [2nd] [=].

6. Växla till den högra sidan och visa grafen.



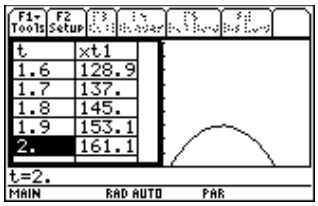
Tips! *Tryck på* [◆] [TblSet].

7. Visa dialogrutan TABLE SETUP och ändra tblStart till 0 och Δtbl till 0,1.



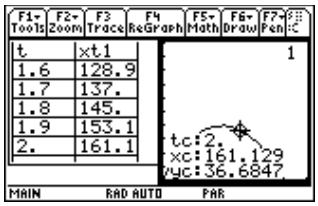
Tips! *Tryck på* [◆] [TABLE].

8. Visa tabellen på vänstra sidan och tryck på ⊖ för att markera t=2.



Obs! *När du flyttar spårningsmarkören från* tc=0,0 *till* tc=3,1 *visas bollens position vid tiden* tc.

9. Växla till den högra sidan. Tryck på [F3] och spåra grafen för att visa värdena för xc och yc när tc=2.



Valfri övning

Anta samma begynnelsehastighet på 95 fot/sek. Hitta sedan den utgångsvinkel vid vilken bollen färdas den längsta sträckan innan den faller till marken.

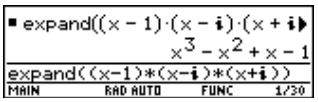
Visualisera komplexa rötter till ett tredjegradspolynom

Denna övning visar hur du plottar de komplexa nollställena hos ett kubiskt polynom. I kapitel 3: Algebra och kapitel 10: 3D-plottning finns detaljerad information om stegen som används i detta exempel.

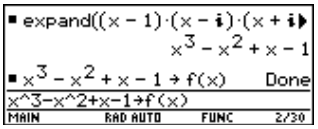
Visualisera komplexa rötter

Följ stegen nedan för att utveckla tredjegradspolynomet (x−1)(x−i)(x+i), hitta absolutvärdet av funktionen, plotta den yta som definieras av absolutbeloppet av funktionen och använda verktyget **Trace** för att utforska ytan.

1. Använd funktion **expand()** i grundfönstret för att utveckla uttrycket (x−1)(x−i) (x+i) och visa tredje gradspolynomet.

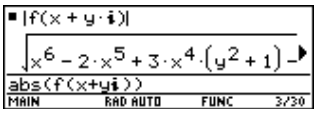


2. Kopiera och klistra in det senaste resultatet på inmatningsraden och spara det i funktionen f(x).

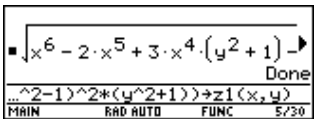


3. Använd funktion **abs()** för att hitta absolutbeloppet av f(x+yi).

   (Denna beräkning kan ta ca två minuter.)

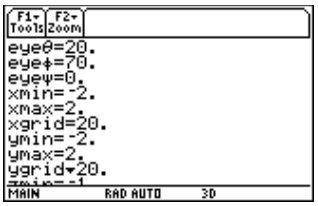


4. Kopiera och klistra in det senaste resultatet på inmatningsraden och spara det i funktionen z1(x,y).



5. Ställ in räknaren till 3D-grafläge, visa koordinataxlarna och ställ in Window-variablerna till:

eye=	[20,70,0]
x=	[⁻2,2,20]
y=	[⁻2,2,20]
z=	[⁻1,2]
ncontour=	[5]



Tips! *Flytta markören till historiklistan och markera det senaste svaret. Tryck på* [ENTER] *för att kopiera det till inmatningsraden.*

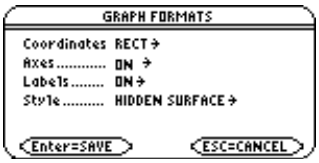
Obs! *Absolutbeloppet av en funktion tvingar alla rötter att visuellt endast nudda, i stället för att korsa x-axeln. Likaledes kommer absolutbeloppet av en funktion med två variabler att tvinga alla rötter att visuellt bara nudda xy-planet.*

Obs! *Grafen till* z1(x,y) *är ytan som definieras av absolutbeloppet av funktionen.*

***Obs!** Det tar ca tre minuter för grafen att beräknas och ritas.*

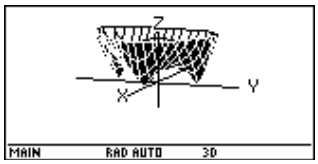
6. I Y=Editor, tryck på:
   TI-89: [◆] [|]
   TI-92 Plus: [◆] F
   och ställ in variablerna för grafformat till:

   Axes= ON
   Labels= ON
   Style= HIDDEN SURFACE

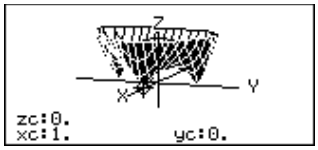


7. Plotta ytan.

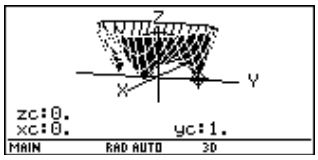
   3D-grafen används visuellt för att visa en bild av rötterna där ytan nuddar xy-planet.



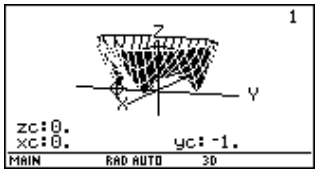
8. Använd verktyget **Trace** för att utforska funktionsvärdena då x=1 och y=0.



9. Använd verktyget **Trace** för att utforska funktionsvärdena vid x=0 och y=1.



10. Använd verktyget **Trace** för att utforska funktionsvärdena vid x=0 och y=⁻1.



Sammanfattning

Lägg märke till att zc är noll för varje funktionsvärde i punkt 7 till 9 ovan. På så sätt kan du visualisera rötterna 1, ⁻i, i till polynomet $x^3 - x - + x - 1$, där de tre punkterna för grafen av ytan nuddar xy-planet.

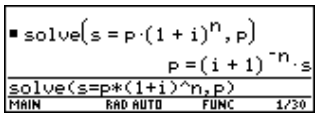
Lösa ett allmänt sparproblem

Denna övning kan användas till att beräkna räntesats, kapitalbelopp, antal betalningsperioder och framtida annuitetsvärde.

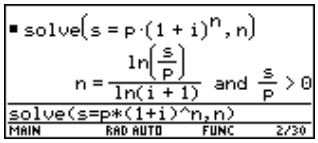
Hitta räntesatsen vid ett sparande

Följ stegen nedan för att hitta räntesatsen (i) av ett sparande där startkapitalet (p) är 1 000 kr, antal ränteterminer (n) är 6 år och det slutliga värdet (s) är 2 000 kr.

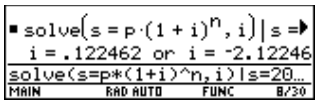
1. Mata in ekvationen i exemplet och lös den med avseende på p i grundfönstret.



2. Mata in ekvationen i exemplet och lös den med avseende på n.



3. Mata in ekvationen i exemplet och lös den med avseende på i med operatorn "with".



solve(s=p∗ (1+i)^n,i) | s=2000 and p=1000 and n=6

Resultat: Räntan är 12,246%.

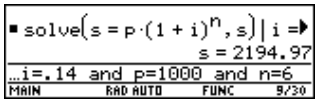
Tips! *När du vill skriva in operatorn "with" (|):*
TI-89: [|]
TI-92 Plus: [2nd] ['']

Tips! *Tryck på* [◆] [ENTER] *för att få ett decimaltalsresultat.*

Hitta det framtida värdet av ett sparande

Hitta det framtida värdet av ett sparande med värdena från föregående exempel och med en ränta på 14%.

Mata in ekvationen i exemplet och lös den med avseende på s.



solve(s=p∗ (1+i)^n,s)| i=.14 and p=1000 and n=6

Resultat: Det framtida värdet är 2 194,97 kr vid en ränta på 14%.

Beräkna betalning

Med detta övning kan du skapa en funktion som kan användas för att hitta kostnaden av att finansiera en bil. I kapitel 17: Programmering finns detaljerad information om stegen som används i detta exempel.

Funktionen betalning

I Program Editor definierar du följande betalningsfunktion (Time-Value-of-Money) där temp1= antal inbetalningar, temp2= årlig ränta, temp3= aktuellt värde, temp4= månatlig avbetalning, temp5=framtida värde och temp6= start- eller slutperiod för betalning (1=i början av månaden, 0=i slutet av månaden).

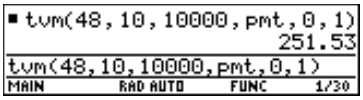
Tips! *Du kan använda din dator när du skriver in långa texter och sedan använda TI-GRAPH LINK för att skicka dessa till din TI-89 / TI-92 Plus.*

```
:tvm(temp1,temp2,temp3,temp4,temp5,temp6)
:Func
:Local tempi,tempfunc,tempstr1
: ⁻temp3+(1+temp2/1200∗temp6)∗temp4∗((1− (1+temp2/1200)^
    (⁻temp1))/(temp2/1200))− temp5∗(1+temp2/1200)^(⁻temp1)
    →tempfunc
:For tempi,1,5,1
:"temp"&exact(string(tempi))→tempstr1
:If when(#tempstr1=0,false,false,true) Then
:If tempi=2
:Return approx(nsolve(tempfunc=0,#tempstr1) | #tempstr1>0 and
    #tempstr1<100)
:Return approx(nsolve(tempfunc=0,#tempstr1))
:EndIf
:EndFor
:Return "parameter error"
:EndFunc
```

Hitta den månatliga avbetalningen

Hittar den månatliga avbetalningen på en bil som kostar 10 000 kr om du gör 48 avbetalningar med en årlig ränta på 10%.

Ange i grundfönstret tvm-värdena för att hitta pmt.

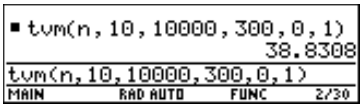


Resultat: Månadsavbetalningen är 251,53 kr.

Hitta antalet avbetalningar

Hitta antalet avbetalningar som krävs för att betala bilen om du kan betala av 300 kr per månad

Ange i grundfönstret tvm-värdena för att hitta n.



Resultat: Antalet avbetalningar är 38,8308.

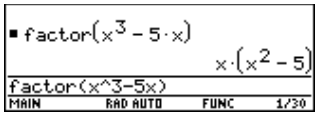
Hitta rationella, reella och komplexa faktorer

I detta övning visas hur du kan hitta rationella, reella och komplexa faktorer av uttryck. I kapitel 3: Algebra finns detaljerad information om stegen som används i detta kapitel.

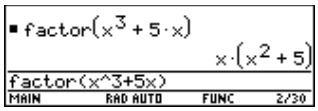
Hitta faktorer

Mata in följande uttryck i grundfönstret.

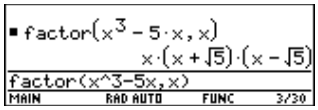
1. factor(x^3−5x) [ENTER] visar ett rationellt resultat.



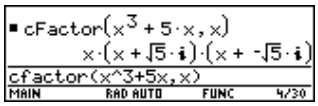
2. factor(x^3+5x) [ENTER] visar ett rationellt resultat.



3. factor(x^3−5x,x) [ENTER] visar ett reellt resultat.



4. cfactor(x^3+5x,x) [ENTER] visar ett komplext resultat.



Simulering av dragning utan återläggning

I det här övning simuleras dragning utan återläggning av olikfärgade bollar från en urna. I kapitel 17: Programmering finns detaljerad information om stegen som används i detta kapitel.

Funktionen för dragning utan återläggning

Definiera drawball() som en funktion i Program Editor som kan anropas med två parametrar. Den första parametern är en lista där varje element är antalet bollar i en viss färg. Den andra parametern är antalet bollar du kan dra. Denna funktion returnerar en lista där varje element är antalet bollar av varje färg som drogs.

```
:drawball(urnlist,drawnum)
:Func
:Local templist,drawlist,colordim,
   numballs,i,pick,urncum,j
:If drawnum>sum(urnlist)
:Return "too few balls"
:dim(urnlist)→ colordim
:urnlist→ templist
:newlist(colordim)→ drawlist
:For i,1,drawnum,1
:sum(templist)→ numballs
:rand(numballs)→ pick
:For j,1,colordim,1
:cumSum(templist)→ urncum
```

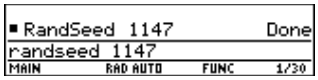
(fortsättning i nästa kolumn)

```
:If pick ≤ urncum[j] Then
:drawlist[j]+1→ drawlist[j]
:templist[j]− 1→ templist[j]
:Exit
:EndIf
:EndFor
:EndFor
:Return drawlist
:EndFunc
```

Dragning utan återläggning

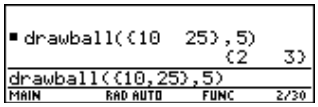
Anta att en urna innehåller *n1* bollar i en färg, *n2* bollar i en andra färg, *n3* bollar i en tredje färg osv. Dra bollar utan att lägga tillbaka dem.

1. Mata in ett slumptal med kommandot **RandSeed** för att initiera slumptalsgeneratorn.



2. Anta att en urna innehåller 10 röda och 25 vita bollar. Simulera att du drar fem bollar slumpmässigt från urnan utan återläggning. Skriv drawball({10,25},5).

   Resultat: 2 röda och 3 vita bollar.



Funktioner och instruktioner

A

Snabbguide 410
Alfabetisk lista över operationer..... 414

Den här appendix innehåller en förteckning över syntaxen och användningssättet för alla funktioner och instruktioner i TI-89 / TI-92 Plus.

Namn på funktion eller instruktion.

Tangent eller meny som används för att mata in namnet. Du kan också skriva namnet.

Exempel

Circle **CATALOG**

Circle x, y, r [, *ritläge*]

Ritar en cirkel med medelpunkten vid koordinaterna (x, y) och med en radie r.

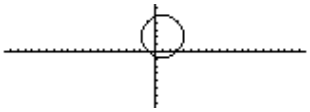
x, y och r måste vara reella värden.

Om *ritläge* = 1 ritas cirkeln (standard).
Om *ritläge* = 0 stängs cirkeln av.
Om *ritläge* = -1 inverteras bildpunkterna längs cirkeln.

Obs! Vid omritning tas alla ritobjekt bort.

I visningsfönstret ZoomSqr:

```
Circle 1,2,3 ENTER
```



Argument visas i *kursiv stil*. Argument inom hakparenteser [ ] är valfria. Skriv inte själva parenteserna.

Förklaring av funktionen eller instruktionen.

På syntaxraden visas ordningsföljden och typen av argument som du ska mata in. Avgränsa flera argument med ett komma (**,**).

Snabbguide

I det här avsnittet listas funktionerna och instruktionerna i funktionella grupper tillsammans med en sidhänvisning där de beskrivs i denna appendix.

Algebra

| ("with")	533	**cFactor()**	419	**comDenom()**	421
cSolve()	425	**cZeros()**	429	**expand()**	444
factor()	445	**getDenom()**	451	**getNum()**	452
nSolve()	472	**propFrac()**	479	**randPoly()**	485
solve()	500	**tCollect()**	509	**tExpand()**	510
zeros()	516				

Kalkyl

∫() (integrera)	528	**Π()** (produkt)	529	**Σ()** (summa)	529
arcLen()	416	**avgRC()**	417	***d()***	431
deSolve()	434	**fMax()**	447	**fMin()**	447
limit()	459	**nDeriv()**	468	**nInt()**	470
' (prime)	531	**seq()**	491	**taylor()**	509

Rita

AndPic	415	**BldData**	418	**Circle**	419
ClrDraw	420	**ClrGraph**	420	**CyclePic**	428
DrawFunc	438	**DrawInv**	439	**DrawParm**	439
DrawPol	439	**DrawSlp**	439	**DrwCtour**	440
FnOff	448	**FnOn**	448	**Graph**	454
Line	459	**LineHorz**	460	**LineTan**	460
LineVert	460	**NewPic**	469	**PtChg**	480
PtOff	480	**PtOn**	480	**ptTest()**	480
PtText	480	**PxlChg**	480	**PxlCrcl**	481
PxlHorz	481	**PxlLine**	481	**PxlOff**	481
PxlOn	481	**pxlTest()**	482	**PxlText**	482
PxlVert	482	**RclGDB**	485	**RclPic**	486
RplcPic	490	**Shade**	495	**StoGDB**	504
StoPic	504	**Style**	505	**Trace**	512
XorPic	516	**ZoomBox**	518	**ZoomData**	518
ZoomDec	519	**ZoomFit**	519	**ZoomIn**	519
ZoomInt	520	**ZoomOut**	520	**ZoomPrev**	520
ZoomRcl	520	**ZoomSqr**	521	**ZoomStd**	521
ZoomSto	521	**ZoomTrig**	522		

Listor

+ (addera)	522	**–** (subtrahera)	522	**∗** (multiplicera)	523
/ (dividera)	523	**⁻** (negation)	524	**^** (potens)	529
augment()	417	**crossP()**	424	**cumSum()**	427
dim()	437	**dotP()**	438	**exp▸list()**	443
left()	458	**list▸mat()**	461	**Δlist()**	461
mat▸list()	464	**max()**	465	**mid()**	466
min()	467	**newList()**	469	**polyEval()**	478
product()	479	**right()**	488	**rotate()**	488
shift()	496	**SortA**	502	**SortD**	502
sum()	505				

Matematiska

+ (addera)	522	− (subtrahera)	522	∗ (multiplicera)	523
/ (dividera)	523	⁻ (negation)	524	**%** (procent)	524
! (fakultet)	527	√**()** (roten ur)	529	**^** (potens)	529
° (grader)	531	∠ (vinkel)	531	**°, ', "**	531
_ (undersstryk.)	532	▸ (omvandling)	532	**10^**	533
0b, 0h	535	**▸Bin**	417	**▸Cylind**	428
▸DD	432	**▸Dec**	432	**▸DMS**	438
▸Hex	454	**▸Polar**	477	**▸Rect**	486
▸Sphere	503	**abs()**	414	**and**	414
angle()	415	**approx()**	416	**ceiling()**	418
conj()	422	**cos()**	422	**cos^{-1}()**	423
cosh()	424	**cosh^{-1}()**	424	**E**	440
e^()	441	**exact()**	443	**floor()**	447
fpart()	449	**gcd()**	450	**imag()**	455
int()	457	**intDiv()**	457	**iPart()**	457
isPrime()	457	**lcm()**	458	**ln()**	461
log()	463	**max()**	465	**min()**	467
mod()	467	**nCr()**	468	**nPr()**	472
P▸Rx()	474	**P▸Ry()**	474	**r** (radianer)	530
R▸Pθ()	484	**R▸Pr()**	484	**real()**	486
remain()	487	**rotate()**	488	**round()**	489
shift()	496	**sign()**	497	**sin()**	498
sin^{-1}()	498	**sinh()**	499	**sinh^{-1}()**	499
tan()	507	**tan^{-1}()**	508	**tanh()**	508
tanh^{-1}()	508	**tmpCnv()**	511	**ΔtmpCnv()**	511
x^{-1}	533				

Matriser

+ (addera)	522	− (subtrahera)	522	∗ (multiplicera)	523
/ (dividera)	523	⁻ (negation)	524	**.+** (elem. add.)	526
.− (elem. sub.)	527	**.∗** (elem. mult.)	527	**. /** (elem. div.)	527
.^ (elem. potens)	527	**^** (potens)	529	**augment()**	417
colDim()	421	**colNorm()**	421	**crossP()**	424
cumSum()	427	**det()**	436	**diag()**	436
dim()	437	**dotP()**	438	**eigVc()**	441
eigVl()	441	**Fill**	446	**identity()**	455
list▸mat()	461	**LU**	464	**mat▸list()**	464
max()	465	**mean()**	465	**median()**	465
min()	467	**mRow()**	467	**mRowAdd()**	468
newMat()	469	**norm()**	471	**product()**	479
QR	482	**randMat()**	485	**ref()**	487
rowAdd()	489	**rowDim()**	489	**rowNorm()**	490
rowSwap()	490	**rref()**	490	**simult()**	497
stdDev()	503	**subMat()**	505	**sum()**	505
T	506	**unitV()**	513	**variance()**	514
x^{-1}	533				

Programmering

=	525	**≠**	525	**<**	525
≤	526	**>**	526	**≥**	526
# (omdirigerare)	530	**→** (spara)	534	**◉** (kommentar)	535
and	414	**ans()**	416	**Archive**	416
ClrErr	420	**ClrGraph**	420	**ClrHome**	420
ClrIO	421	**ClrTable**	421	**CopyVar**	422
CustmOff	427	**CustmOn**	427	**Custom**	428
Cycle	428	**Define**	432	**DelFold**	433
DelVar	434	**Dialog**	436	**Disp**	437
DispG	437	**DispHome**	437	**DispTbl**	438
DropDown	440	**Else**	442	**ElseIf**	442
EndCustm	442	**EndDlog**	442	**EndFor**	442
EndFunc	442	**EndIf**	442	**EndLoop**	442
EndPrgm	442	**EndTBar**	442	**EndTry**	442
EndWhile	442	**entry()**	442	**Exec**	443
Exit	443	**For**	448	**format()**	449
Func	449	**Get**	450	**GetCalc**	450
getConfg()	451	**getFold()**	451	**getKey()**	452
getMode()	452	**getType()**	452	**getUnits()**	453
Goto	453	**If**	455	**Input**	456
InputStr	456	**Item**	458	**Lbl**	458
left()	458	**Local**	462	**Lock**	462
Loop	464	**MoveVar**	467	**NewFold**	469
NewProb	470	**not**	471	**or**	473
Output	474	**part()**	475	**PassErr**	476
Pause	477	**PopUp**	478	**Prgm**	478
Prompt	479	**Rename**	487	**Request**	487
Return	488	**right()**	488	**Send**	490
SendCalc	491	**SendChat**	491	**setFold()**	491
setGraph()	492	**setMode()**	493	**setTable()**	494
setUnits()	494	**Stop**	504	**Style**	505
switch()	506	**Table**	507	**Text**	510
Then	510	**Title**	510	**Toolbar**	512
Try	512	**Unarchiv**	513	**Unlock**	513
when()	514	**While**	515	**xor**	515

Statistik

! (fakultet)	527	**BldData**	418	**CubicReg**	427
cumSum()	427	**ExpReg**	445	**LinReg**	460
LnReg	462	**Logistic**	463	**mean()**	465
median()	465	**MedMed**	466	**nCr()**	468
NewData	469	**NewPlot**	470	**nPr()**	472
OneVar	473	**PlotsOff**	477	**PlotsOn**	477
PowerReg	478	**QuadReg**	483	**QuartReg**	484
rand()	485	**randNorm()**	485	**RandSeed**	485
ShowStat	497	**SinReg**	499	**SortA**	502
SortD	502	**stdDev()**	503	**TwoVar**	513
variance()	514				

Strängar

& (sätt ihop)	527	# (omdirigerare)	530	**char()**	419
dim()	437	**expr()**	445	**format()**	449
inString()	456	**left()**	458	**mid()**	466
ord()	474	**right()**	488	**rotate()**	488
shift()	496	**string()**	504		

Operationer vars namn inte är alfabetiska (exempelvis +, ! och >) listas i slutet av denna appendix, med början på sidan 522. Om inte annat anges har alla exempel i detta avsnitt utförts i standardläge och alla variabler antas vara odefinierade. På grund av formateringsrestriktioner visas ungefärliga resultat med tre decimaler (3.14159265359 visas som 3.141...).

abs() Menyn MATH/Number

abs(*uttryck***)** ⇒ *uttryck*
abs(*lista1***)** ⇒ *lista*
abs(*matris1***)** ⇒ *matris*

Returnerar absolutbeloppet av argumentet.

Om argumentet är ett komplext tal returneras modulus av talet

Obs! Alla odefinierade variabler behandlas som reella variabler.

```
abs({π/2,⁻π/3}) ENTER        {π/2  π/3}
abs(2−3i) ENTER              √13
abs(z) ENTER                 |z|
abs(x+yi) ENTER              √(x²+y²)
```

and Menyn MATH/Test

booleskt uttryck1 **and** *uttryck2* ⇒ *booleskt uttryck*
boolesk lista1 **and** *lista2* ⇒ *boolesk lista*
boolesk matris1 **and** *matris2* ⇒ *boolesk matris*

Returnerar sant eller falskt eller en förenklad form av den ursprungliga inmatningen.

```
x≥3 and x≥4 ENTER            x≥4
{x≥3,x≤0} and {x≥4,x≤⁻2} ENTER
                             {x ≥ 4   x ≤ ⁻2}
```

heltal1 **and** *heltal2* ⇒ *heltal*

Bitvis jämförelse av två heltal med funktionen **and**. Internt konverteras båda talen till 32-bitars binära värden. Vid jämförelse returneras 1 om båda bitarna är 1, annars returneras 0. Returnerat värde representerar resultatet och visas i det läge som tidigare valts.

Du kan skriva in heltalen i valfritt läge. Om du använder binärt eller hexadecimalt läge börjar du talet med 0b alternativt 0h. Värden utan prefix behandlas i decimalt läge (bas 10).

Om du skriver in ett decimalt tal som är för stort för 32 bitar används en symmetrisk modulooperation för att konvertera talet till gällande begränsningar.

I hexadecimalt läge:

```
0h7AC36 and 0h3D5F ENTER     0h2C16
```

Viktigt! Noll, inte bokstaven O.

I binärt läge:

```
0b100101 and 0b100 ENTER     0b100
```

I decimalt läge:

```
37 and 0b100 ENTER           4
```

Obs! Ett binärt tal kan bestå av upp till 32 tecken (förutom prefixet 0b).
Ett hexadecimalt tal kan bestå av upp till 8 tecken.

AndPic CATALOG

AndPic *bildvar*[, *rad*, *kolumn*]

Visar graffönstret och använder logiskt "AND" för bilden som sparas i *bildvar* och den aktuella bilden i graffönstret vid koordinaterna *(rad, kolumn)*.

bildvar måste vara en bildtyp.

Standardkoordinater är (0,0), vilket är det övre vänstra hörnet av fönstret.

I funktionsplottningsläge och i Y= Editor:

`y1(x) = cos(x)` ⊕

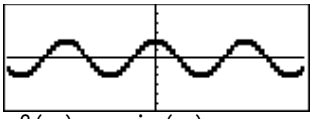
TI-89: [2nd][F6] `Style = 3:Square`

TI-92 Plus: [F6] `Style = 3:Square`

[F2] `Zoom = 7:ZoomTrig`

[F1] `= 2:Save Copy As...`

`Type = Picture, Variable = PIC1`



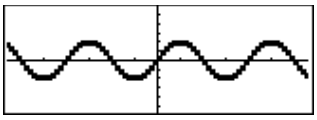
`y2(x) = sin(x)`

TI-89: [2nd][F6] `Style = 3:Square`

TI-92 Plus: [F6] `Style = 3:Square`

`y1 = no checkmark (F4 to deselect)`

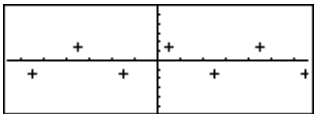
[F2] `Zoom = 7:ZoomTrig`



TI-89: [HOME]

TI-92 Plus: [♦][HOME]

`AndPic PIC1` [ENTER] `Done`



angle() Menyn MATH/Complex

angle(*uttryck1***)** ⇒ *uttryck*

Returnerar vinkeln av *uttryck1* och tolkar *uttryck1* som ett komplext tal.

Obs! Alla odefinierade variabler behandlas som reella variabler.

Med vinkelenheten grader:

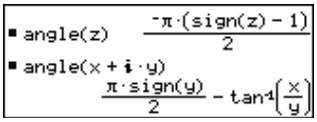
`angle(0+2i)` [ENTER] `90`

Med vinkelenheten radianer:

`angle(1+i)` [ENTER] $\frac{\pi}{4}$

`angle(z)` [ENTER]

`angle(x+iy)` [ENTER]



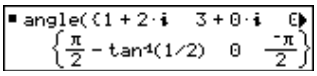
---

angle(*lista1***)** ⇒ *lista*
angle(*matris1***)** ⇒ *matris*

Returnerar en lista eller matris av vinklar av elementen i *lista1* eller *matris1* och tolkar varje element som ett komplext tal som motsvarar en tvådimensionell rektangulär koordinatpunkt.

Med vinkelenheten radianer:

`angle({1+2i,3+0i,0-4i})` [ENTER]



ans() tangenterna [2nd] [ANS]

ans() ⇒ *värde*
ans(*heltal***)** ⇒ *värde*

Returnerar ett föregående resultat från historielistan i grundfönstret.

heltal, om det är inkluderat, anger vilket föregående resultat som ska hämtas. Giltigt intervall för *heltal* är från 1 till 99 och kan inte vara ett uttryck. Standard är 1, det senaste resultatet.

Om du vill använda **ans()** för att generera Fibonacci-följden i grundfönstret, trycker du på:

1 [ENTER]	1
1 [ENTER]	1
[2nd] [ANS] [+] [2nd] [ANS] [◀] [←] 2 [ENTER]	2
[ENTER]	3
[ENTER]	5

approx() Menyn MATH/Algebra

approx(*uttryck***)** ⇒ *värde*

Returnerar beräkningen av *uttryck* som ett decimaltal när det är möjligt, oavsett vilket det aktuella Exact/Approx-läget är.

Detta är detsamma som att mata in *uttryck* och trycka på [◆] [ENTER] i grundfönstret.

approx(π) [ENTER] 3.141...

---

approx(*lista1***)** ⇒ *lista*
approx(*matris1***)** ⇒ *matris*

Returnerar en lista eller matris där varje element har beräknats till ett decimaltal där det är möjligt.

approx({sin(π),cos(π)}) [ENTER]
{0. -1.}

approx([√(2),√(3)]) [ENTER]
[1.414... 1.732...]

Archive CATALOG

Archive *var1* [, *var2*] [, *var3*] ...

Flyttar angivna variabler från RAM till användarminnet.

Du kan använda en sparad variabel på samma sätt som en variabel i RAM. Du kan däremot inte ta bort, namnändra eller ändra en variabel i användarminnet då den låses automatiskt.

Använd **Unarchiv** när du vill ta bort variabler.

10→arctest [ENTER] 10
Archive arctest [ENTER] Done
5*arctest [ENTER] 50
15→arctest [ENTER]

[ESC]
Unarchiv arctest [ENTER] Done
15→arctest [ENTER] 15

arcLen() Menyn MATH/Calculus

arcLen(*uttryck1*,*var*,*start*,*slut***)** ⇒ *uttryck*

Returnerar båglängden av *uttryck1* från *start* till *slut* med avseende på variabeln *var*.

Oavsett grafläget beräknas båglängden som en integral som antar en funktionsdefinition.

arcLen(cos(x),x,0,π) [ENTER] 3.820...

arcLen(f(x),x,a,b) [ENTER]

$$\int_a^b \sqrt{\left(\frac{d}{dx}(f(x))\right)^2+1}\ dx$$

---

arcLen(*lista1*, *var*,*start*,*slut***)** ⇒ *lista*

Returnerar en lista av båglängderna för varje element i *lista1* från *start* till *slut* med avseende på *var*.

arcLen({sin(x),cos(x)},x,0,π)
{3.820... 3.820...}

augment() Menyn MATH/Matrix

augment(*lista1*, *lista2***)** ⇒ *lista*

Returnerar en ny lista som är *lista2* tillagt till slutet av *lista1*.

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

---

augment(*matris1*; *matris2***)** ⇒ *matris*
augment(*matris1*, *matris2***)** ⇒ *matris*

Returnerar en ny matris med *matris2* tillagd till *matris1*. När tecknet "," används måste båda matriserna ha samma antal rader och *matris2* läggs till *matris1* som nya kolumner. När tecknet ";" används måste båda matriserna ha samma antal kolumner och *matris2* läggs till *matris1* som nya rader. Inga ändringar görs i *matris1* eller *matris2*.

```
[1,2;3,4]→M1 ENTER        [1 2; 3 4]
[5;6]→M2 ENTER            [5; 6]
augment(M1,M2) ENTER      [1 2 5; 3 4 6]
[5,6]→M2 ENTER            [5 6]
augment(M1;M2) ENTER      [1 2; 3 4; 5 6]
```

avgRC() CATALOG

avgRC(*uttryck1*, *var* [, *h*]**)** ⇒ *uttryck*

Returnerar differenskvoten.

uttryck1 kan vara namnet på en användardefinierad funktion (se **Func**).

h är stegvärdet. Om *h* utelämnas är standardvärdet 0,001.

Observera att den liknande funktionen **nDeriv()** använder den symmetriska differenskvoten.

```
avgRC(f(x),x,h) ENTER
                    (f(x+h) - f(x))/h
avgRC(sin(x),x,h)|x=2 ENTER
                    (sin(h+2) - sin(2))/h
avgRC(x^2-x+2,x) ENTER
                    2.·(x-.4995)
avgRC(x^2-x+2,x,.1) ENTER
                    2.·(x-.45)
avgRC(x^2-x+2,x,3) ENTER   2·(x+1)
```

▸Bin MATH/Base menu

heltal1 **▸Bin** ⇒ *heltal*

Konverterar *heltal1* till ett binärt värde. Binära och hexadecimala värden har alltid prefixet 0b respektive 0h.

Noll, inte bokstaven O, följt av b alternativt h.

0b *binärt tal*
0h *hexadecimalt tal*

Ett binärt tal kan bestå av upp till 32 tecken. Ett hexadecimalt tal kan bestå av upp till 8 tecken.

Värden utan prefix behandlas i decimalt läge (bas 10). Resultatet visas alltid i binärt läge oavsett inmatningsläge.

Om du skriver in ett decimalt tal som är för stort för 32 bitar används en symmetrisk modulo-operation för att konvertera talet till gällande begränsningar.

```
256▸Bin ENTER        0b100000000
0h1F▸Bin ENTER       0b11111
```

BldData CATALOG

BldData [*dataVar*]

Skapar datavariabeln *dataVar* utifrån den information som använts vid plottning av aktuell graf. **BldData** kan användas i alla graflägen.

Om *dataVar* inte anges sparas resultatet i systemvariabeln sysData.

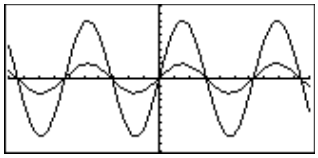
Obs! Första gången du startar Data/Matris-redigeraren efter du använt **BldData** är *dataVar* eller sysData (beroende på vilken variabel du använt med **BldData**) inställt som aktuell datavariabel.

Differensen mellan värden för enskilda variabler (x i exemplet till höger) beräknas enligt värden för Windows-variablerna.

Mer information om vilka värden som används för att plotta grafer finns i kapitlen om plottning (för aktuellt grafläge).

I funktionsplottningsläge och vinkelenheten radianer:

```
8*sin(x)→y1(x) [ENTER]          Done
2*sin(x)→y2(x) [ENTER]          Done
ZoomStd [ENTER]
```



TI-89: [HOME]

TI-92 Plus: [♦] [HOME]

```
BldData [ENTER]                 Done
[APPS] 6 [ENTER]
```

DATA	x	y1	y2
	c1	c2	c3
1	-10.	4.3522	1.088
2	-9.832	3.168	.792
3	-9.664	1.8945	.47363
4	-9.496	.56769	.14192

3D-grafläget har två fristående variabler. I exemplet till höger är x konstant medan y ökar i värde inom valt intervall.

Då y nått intervallgränsen ökas x till nästa värde och y stegar återigen genom valt intervall. Detta upprepas tills x har stegat genom hela det valda intervallet.

Obs! Följande exempeldata är från en 3D-graf.

DATA	x	y	z1
	c1	c2	c3
1	-10.	-10.	0.
2	-10.	-8.571	5.8309
3	-10.	-7.143	8.9706
4	-10.	-5.714	9.8677

ceiling() Menyn MATH/Number

ceiling(*uttryck 1***)** ⇒ *heltal*

Returnerar det närmaste heltalet som är ≥ argumentet.

Argumentet kan vara ett reellt eller komplext tal.

Obs! Se även **floor()**.

```
ceiling(0.456) [ENTER]          1.
```

ceiling(*lista1***)** ⇒ *lista*
ceiling(*matris1***)** ⇒ *matris*

Returnerar en lista eller matris av det närmaste heltalet som är ≥ varje element.

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

cFactor() Menyn MATH/Algebra/Complex

cFactor(*uttryck1***[, *var*])** ⇒ *uttryck*
cFactor(*lista1***[, *var*])** ⇒ *lista*
cFactor(*matris1***[, *var*])** ⇒ *matris*

cFactor(*uttryck1***)** returnerar *uttryck1* faktoruppdelat med avseende på alla dess variabler med en gemensam nämnare.

uttryck1 faktoruppdelas så långt det är möjligt mot linjära rationella faktorer, även om detta introducerar nya icke-reella tal. Detta alternativ är lämpligt om du vill faktoruppdela med avseende på mer än en variabel.

```
cFactor(a^3*x^2+a*x^2+a^3+a) ENTER
```

$$a\cdot(a+{}^{-}i)\cdot(a+i)\cdot(x+{}^{-}i)\cdot(x+i)$$

```
cFactor(x^2+4/9) ENTER
```

$$\frac{(3\cdot x+{}^{-}2\cdot i)\cdot(3\cdot x+2\cdot i)}{9}$$

```
cFactor(x^2+3) ENTER
```

$$x^2+3$$

```
cFactor(x^2+a) ENTER
```

$$x^2+a$$

---

cFactor(*uttryck1***,***var***)** returnerar *uttryck1* faktoruppdelat med avseende på *var*.

uttryck1 faktoruppdelas så långt det är möjligt mot linjära faktorer i *var*, även om detta introducerar nya icke-reella konstanter eller uttryck som är irrationella i andra variabler.

Faktorerna och deras termer sorteras med *var* som huvudvariabel. Likadana potenser av *var* samlas i varje faktor. Ange *var* om faktoruppdelning ska ske med avseende på endast den variabeln och du kan acceptera irrationella uttryck i andra variabler för att öka faktoruppdelningen med avseende på *var*. Det kan förekomma oavsiktlig faktoruppdelning med avseende på andra variabler.

Om du använder AUTO-inställningen i Exact/Approx-läget och inkluderar *var* tillåts även ungefärliga koefficienter på decimaltalsform där irrationella koefficienter inte uttryckligen kan uttryckas precist i termer av de inbyggda funktionerna. Även om uttrycket endast innehåller en variabel, kan du få en fullständigare faktoruppdelning om du inkluderar *var*.

Obs! Se även **factor()**.

```
cFactor(a^3*x^2+a*x^2+a^3+a,x) ENTER
```

$$a\cdot(a^2+1)\cdot(x+{}^{-}i)\cdot(x+i)$$

```
cFactor(x^2+3,x) ENTER
```

$$(x+\sqrt{3}\cdot i)\cdot(x+{}^{-}\sqrt{3}\cdot i)$$

```
cFactor(x^2+a,x) ENTER
```

$$(x+\sqrt{a}\cdot{}^{-}i)\cdot(x+\sqrt{a}\cdot i)$$

```
cFactor(x^5+4x^4+5x^3-6x-3) ENTER
```

$$x^5+4\cdot x^4+5\cdot x^3-6\cdot x-3$$

```
cFactor(ans(1),x) ENTER
```

$$(x-.965)\cdot(x+.612)\cdot(x+2.13)\cdot(x+1.11-1.07\cdot i)\cdot(x+1.11+1.07\cdot i)$$

char() Menyn MATH/String

char(*heltal***)** ⇒ *tecken*

Returnerar en teckensträng som innehåller tecknet som har koden *heltal* ur teckentabellen i TI-89 / TI-92 Plus. Se appendix B för en komplett lista med teckenkoder.

Det giltiga intervallet för *heltal* är 0–255.

```
char(38) ENTER
```

"&"

```
char(65) ENTER
```

"A"

Circle CATALOG

Circle x, y, r [, *ritläge*]

Ritar en cirkel med medelpunkten vid koordinaterna (x, y) och med radien r.

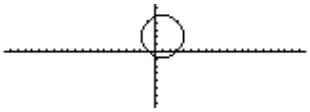
x, y och r måste vara reella värden.

Om *ritläge* = 1 ritas cirkeln (standard).
Om *ritläge* = 0 stängs cirkeln av.
Om *ritläge* = -1 inverteras bildpunkterna längs cirkeln.

Obs! Vid omritning tas alla ritobjekt bort. Se även **PxlCrcl**.

I visningsfönstret ZoomSqr:

```
ZoomSqr:Circle 1,2,3 ENTER
```



ClrDraw CATALOG

ClrDraw

Rensar graffönstret och nollställer funktionen Smart Graph så att grafen kan ritas om nästa gång graffönstret visas.

Samtidigt som du visar graffönstret kan du ta bort alla ritobjekt (exempelvis linjer och punkter) genom att trycka på [F4] (ReGraph) eller trycka på:
TI-89: [2nd] [F6]
TI-92 Plus: [F6]
och välja 1:ClrDraw.

ClrErr CATALOG

ClrErr

Rensar felstatusen. errornum ställs om till noll och de interna felvariablerna rensas.

Kommandot **Else** i programsatsen **Try...EndTry** ska använda **ClrErr** eller **PassErr**. Om felet ska bearbetas eller ignoreras använder du **ClrErr**. Om du inte vet vad du ska göra med felet använder du **PassErr** för att skicka felet vidare till nästa felhanterare. Om det inte finns några fler väntande **Try...EndTry**-felhanterare, visas dialogrutan Error som vanligt.

Obs! Se även **PassErr** och **Try**.

Programlista:

```
:clearerr()
:Prgm
:PlotsOff:FnOff:ZoomStd
:For i,0,238
:Δx*i+xmin→xcord
: Try
:  PtOn xcord,ln(xcord)
: Else
:  If errornum=800 or
     errornum=260 Then
:   ClrErr ◉clear the error
:  Else
:   PassErr ◉pass on any other
     error
:  EndIf
: EndTry
:EndFor
:EndPrgm
```

ClrGraph CATALOG

ClrGraph

Tar bort funktioner eller instruktioner som plottades med kommandot **Graph** eller skapades med kommandot **Table**.
(Se **Graph** eller **Table**.)

Föregående markerade Y=-funktioner kommer att plottas nästa gång som grafen visas.

ClrHome CATALOG

ClrHome

Tar bort alla poster som sparas i **entry()** och **ans()** (grundfönstrets historiklista).Rensar inte den aktuella inmatningsraden.

När grundfönstret är aktiverat kan du radera historiklistan genom att trycka på [F1] och välja 8:Clear Home.

För funktioner som **solve()** vilka returnerar godtyckliga konstanter eller heltal (@1, @2, osv.) sätter **ClrHome** suffixet till 1.

ClrIO CATALOG

ClrIO

Rensar Program I/O-fönstret.

ClrTable CATALOG

ClrTable

Tar bort alla tabellvärden. Gäller bara ASK-inställningen i dialogrutan Table Setup.

När du visar tabellfönstret i Ask-läge kan du ta bort värdena genom att trycka på [F1] och välja 8:Clear Table.

colDim() Menyn MATH/Matrix/Dimensions

colDim(*matris***)** ⇒ *uttryck*

Returnerar antalet kolumner som finns i *matris*.

Obs! Se även **rowDim()**.

```
colDim([0,1,2;3,4,5]) ENTER        3
```

colNorm() Menyn MATH/Matrix/Norms

colNorm(*matris***)** ⇒ *uttryck*

Returnerar det största värdet av summorna av absolutvärdet av elementen i kolumnerna i *matris*.

Obs! Odefinierade matriselement är inte tillåtna. Se även **rowNorm()**.

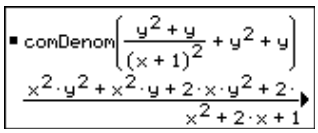
```
[1,-2,3;4,5,-6]→mat ENTER
                    1  -2   3
                  [4   5  -6]
colNorm(mat) ENTER                 9
```

comDenom() Menyn MATH/Algebra

comDenom(*uttryck1*[,*var*]**)** ⇒ *uttryck*
comDenom(*lista1*[,*var*]**)** ⇒ *lista*
comDenom(*matris1*[,*var*]**)** ⇒ *matris*

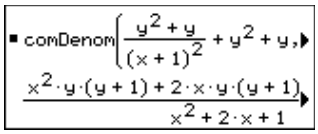
comDenom(*uttryck1***)** returnerar ett bråk med fullständigt utvecklad täljare och en fullständigt utvecklad nämnare.

```
comDenom((y^2+y)/(x+1)^2+y^2+y)
ENTER
```

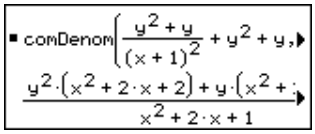


comDenom(*uttryck1*,*var***)** returnerar ett bråk med fullständigt utvecklad täljare och en fullständigt utvecklad nämnare med avseende på *var*. Termerna och deras faktorer sorteras med *var* som huvudvariabel. Likadana potenser av *var* samlas ihop. Det kan förekomma oavsiktlig faktoruppdelning av de ihopsamlade koefficienterna. Jämfört med att utelämna *var*, sparar detta ofta tid, minne och fönsterutrymme, samtidigt som uttrycket blir mer läsbart. Det gör också fortsatta åtgärder på resultatet snabbare och kör inte lika lätt slut på minnet.

```
comDenom((y^2+y)/(x+1)
^2+y^2+y,x) ENTER
```




```
comDenom((y^2+y)/(x+1)
^2+y^2+y,y) ENTER
```



Om *var* inte finns i *uttryck1*, returnerar **comDenom(***uttryck1*,*var***)** ett bråk med outvecklad täljare och nämnare. Sådana resultat sparar ofta ännu mer tid, minne och fönsterutrymme. Sådana partiellt faktoruppdelade resultat gör också fortsatta åtgärder på resultatet snabbare och kör inte lika lätt slut på minnet.

```
comDenom(exprn,abc)→comden
(exprn) [ENTER]                    Done

comden((y^2+y)/(x+1)^2+y^2+y)
[ENTER]
```

$$\blacksquare\ \text{comden}\left(\frac{y^2+y}{(x+1)^2}+y^2+y\right)$$

$$\frac{(x^2+2\cdot x+2)\cdot y\cdot(y+1)}{(x+1)^2}$$

Även om det inte finns någon nämnare är funktionen **comden** ofta ett snabbt sätt att åstadkomma partiell faktoruppdelning om **factor()** är för långsam eller kör slut på minnet.

```
comden(1234x^2*(y^3-y)+2468x
*(y^2-1)) [ENTER]
```

$$1234\cdot x\cdot(x\cdot y+2)\cdot(y^2-1)$$

Tips! Mata in funktionsdefinitionen **comden()** och använd den i stället för **comDenom()** och **factor()**.

conj() Menyn MATH/Complex

conj(*uttryck1***)** ⇒ *uttryck*
conj(*lista1***)** ⇒ *lista*
conj(*matris1***)** ⇒ *matris*

Returnerar det konjugerande komplexa talet till argumentet.

Obs! Alla odefinierade variabler behandlas som reella variabler.

```
conj(1+2i) [ENTER]                 1 - 2·i

conj([2,1-3i;-i,-7]) [ENTER]
```

$$\begin{bmatrix} 2 & 1+3\cdot i \\ i & -7 \end{bmatrix}$$

```
conj(z)                            z
conj(x+iy)                     x + -i·y
```

CopyVar CATALOG

CopyVar *var1*, *var2*

Kopierar innehållet i variabeln *var1* till *var2*. Om *var2* inte finns skapar **CopyVar** den.

Obs! CopyVar liknar lagringsfunktionen (→) när du kopierar ett uttryck, en lista, en matris eller en teckensträng, förutom att ingen förenkling sker när du använder **CopyVar**. Du måste använda **CopyVar** med icke-algebraiska variabeltyper, som exempelvis Pic- och GDB-variabler.

```
x+y→a [ENTER]                      x + y
10→x [ENTER]                        10
CopyVar a,b [ENTER]               Done
a→c [ENTER]                      y + 10
DelVar x [ENTER]                  Done
b [ENTER]                          x + y
c [ENTER]                        y + 10
```

cos() TI-89: [2nd] [COS] tangent TI-92 Plus: [COS] tangent

cos(*uttryck1***)** ⇒ *uttryck*
cos(*lista1***)** ⇒ *lista*

cos(*uttryck1***)** returnerar cosinus av argumentet som ett uttryck.

cos(*lista1***)** returnerar en lista med cosinus av alla elementen i *lista1*.

Obs! Argumentet tolkas som en vinkel i grader eller radianer, enligt den aktuella inställningen för vinkelenheter. Du kan använda ° eller ʳ för att tillfälligt åsidosätta inställningen för vinkelenhet.

Med vinkelenheten grader:

```
cos((π/4)ʳ) [ENTER]
```

$$\frac{\sqrt{2}}{2}$$

```
cos(45) [ENTER]
```

$$\frac{\sqrt{2}}{2}$$

```
cos({0,60,90}) [ENTER]   {1   1/2   0}
```

Med vinkelenheten radianer:

```
cos(π/4) [ENTER]
```

$$\frac{\sqrt{2}}{2}$$

```
cos(45°) [ENTER]
```

$$\frac{\sqrt{2}}{2}$$

cos(*kvadratiskMatris1***)** ⇒ *kvadratiskMatris*

Returnerar cosinus av matrisen *kvadratiskMatris 1*. Detta är *inte* samma sak som att beräkna cosinus för varje element.

När en skalär funktion f(A) utförs på *kvadratiskMatris1* (A) beräknas resultatet enligt nedanstående algoritm:

1. Beräkna egenvärden (λ_i) och egenvektorer (V_i) hos A.

   kvadratiskMatris1 måste vara diagonaliserbar. Den kan inte innehålla symboliska variabler som inte har tilldelats något värde.

2. Skapa matrisen:

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

3. $A = X\,B\,X^{-1}$ och $f(A) = X\,f(B)\,X^{-1}$.
   Till exempel: $\cos(A) = X\cos(B)\,X^{-1}$ där:

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

Alla beräkningar utförs med flyttalsaritmetik.

Med vinkelenheten radianer:

cos([1,2,3;4,5,6;7,8,9]) ENTER

$$\begin{bmatrix} .212\dots & .205\dots & .121\dots \\ .160\dots & .259\dots & .037\dots \\ .248\dots & -.090\dots & .218\dots \end{bmatrix}$$

$\cos^{-1}()$

TI-89: ◆ [COS⁻¹] **tangent** **TI-92 Plus:** 2nd [COS⁻¹] **tangent**

cos⁻¹(*uttryck1***)** ⇒ *uttryck*
cos⁻¹(*lista1***)** ⇒ *lista*

cos⁻¹ (*uttryck1***)** returnerar vinkeln vars cosinusvärde är *uttryck1* som ett uttryck.

cos⁻¹ (*lista1***)** returnerar en lista med de vinklar som har cosinusvärdena som element i *lista1*.

Obs! Resultatet returneras som ett vinkelmått i antingen grader eller radianer, enligt den aktuella inställningen för vinkelenhet.

Med vinkelenheten grader:

cos⁻¹(1) ENTER 0

Med vinkelenheten radianer:

cos⁻¹({0,.2,.5}) ENTER

$$\{\frac{\pi}{2} \quad 1.369\dots \quad 1.047\dots\}$$

cos⁻¹(*kvadratiskMatris1***)** ⇒ *kvadratiskMatris*

Returnerar arcuscosinus av matrisen *kvadratiskMatris1*. Detta är *inte* samma sak som att beräkna arcuscosinus för varje element. Mer information om beräkningsmetoden finns i avsnittet om **cos()**.

kvadratiskMatris1 måste vara diagonaliserbar. Resultatet anges alltid med flyttalsaritmetik.

Med vinkelenheten radianer och komplext rektangulärt läge:

cos⁻¹([1,5,3;4,2,1;6,⁻2,1]) ENTER

$$\begin{bmatrix} 1.734\dots+.064\dots\cdot i & -1.490\dots+2.105\dots\cdot i & \dots \\ -.725\dots+1.515\dots\cdot i & .623\dots+.778\dots\cdot i & \dots \\ -2.083\dots+2.632\dots\cdot i & 1.790\dots-1.271\dots\cdot i & \dots \end{bmatrix}$$

cosh() Menyn MATH/Hyperbolic

cosh(*uttryck1***)** ⇒ *uttryck*
cosh(*lista1***)** ⇒ *lista*

cosh (*uttryck1***)** returnerar cosinus hyperbolicus av argumentet som ett uttryck.

cosh (*lista1***)** returnerar en lista med cosinus hyperbolicus av alla elementen i *lista1*.

```
cosh(1.2) ENTER                    1.810...
cosh({0,1.2}) ENTER    {1   1.810...}
```

cosh(*kvadratiskMatris1***)** ⇒ *kvadratiskMatris*

Returnerar cosinus hyperbolicus av matrisen *kvadratiskMatris1*. Detta är *inte* samma sak som att beräkna cosinus hyperbolicus för varje element. Mer information om beräkningsmetoden finns i avsnittet om **cos()**.

kvadratiskMatris1 måste vara diagonaliserbar. Resultatet anges alltid med flyttalsaritmetik.

Med vinkelinställningen Radian:

```
cosh([1,5,3;4,2,1;6,-2,1])
ENTER
```

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

cosh^{-1}() Menyn MATH/Hyperbolic

cosh^{-1}(*uttryck1***)** ⇒ *uttryck*
cosh^{-1}(*lista1***)** ⇒ *lista*

cosh^{-1} (*uttryck1***)** returnerar arcuscosinus hyperbolicus av argumentet som ett uttryck.

cosh^{-1} (*lista1***)** returnerar en lista med arcuscosinus hyperbolicus av alla element i *lista1*.

```
cosh⁻¹(1) ENTER                    0
cosh⁻¹({1,2.1,3}) ENTER
            {0   1.372...   cosh⁻¹(3)}
```

cosh^{-1}(*kvadratiskMatris1***)** ⇒ *kvadratiskMatris*

Returnerar arcuscosinus hyperbolicus av matrisen *kvadratiskMatris1*. Detta är *inte* samma sak som att beräkna arcuscosinus hyperbolicus för varje element. Mer information om beräkningsmetoden finns i avsnittet om **cos()**.

kvadratiskMatris1 måste vara diagonaliserbar. Resultatet anges alltid med flyttalsaritmetik.

Med vinkelinställningen Radian och komplext rektangulärt format:

```
cosh⁻¹([1,5,3;4,2,1;6,-2,1])
ENTER
```

$$\begin{bmatrix} 2.525...+1.734...\cdot i & -.009...-1.490...\cdot i & ... \\ .486...-.725...\cdot i & 1.662...+.623...\cdot i & ... \\ -.322...-2.083...\cdot i & 1.267...+1.790...\cdot i & ... \end{bmatrix}$$

crossP() Menyn MATH/Matrix/Vector ops

crossP(*lista1***,** *lista2***)** ⇒ *lista*

Returnerar skalärprodukten av *lista1* och *lista2* som en lista.

lists1 och *lista2* måste ha samma dimension och dimensionen måste vara antingen 2 eller 3.

```
crossP({a1,b1},{a2,b2}) ENTER
                {0  0  a1·b2-a2·b1}
crossP({0.1,2.2,-5},{1,-.5,0})
ENTER
                {-2.5  -5.  -2.25}
```

crossP(*vektor1***,** *vektor2***)** ⇒ *vektor*

Returnerar en rad- eller kolumnvektor (beroende på argumenten) som är skalärprodukten av *vektor1* och *vektor2*.

Både *vektor1* och *vektor2* måste vara radvektorer, eller så måste båda vara kolumnvektorer. Båda vektorerna måste ha samma dimension och dimensionen måste vara antingen 2 eller 3.

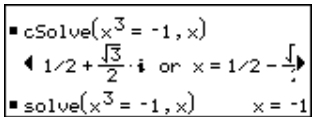
```
crossP([1,2,3],[4,5,6]) ENTER
                        [-3  6  -3]
crossP([1,2],[3,4]) ENTER
                        [0  0  -2]
```

cSolve() Menyn MATH/Algebra/Complex

cSolve(*ekvation***, *var*)** ⇒ *booleskt uttryck*

Returnerar komplexa lösningar till en ekvation för en *var*. Målet är att hitta alla reella och icke-reella lösningar. Även om *ekvation* är reell tillåter **cSolve()** icke-reella resultat i det reella läget.

```
cSolve(x^3=-1,x) ENTER
solve(x^3=-1,x) ENTER
```



Även om alla odefinierade variabler behandlas som reella, kan **cSolve()** lösa polynom för att få komplexa lösningar.

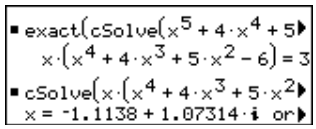
cSolve() ställer tillfälligt in räknaren i komplext format, vid lösningen, trots att räknaren kanske är i reellt format. I den komplexa inställningen används bråkpotenser med udda nämnare i huvudsak hellre än de reella. Följaktligen så är lösningar på ekvationer från **solve()** med sådana bråkpotenser inte nödvändigtvis en delmängd av de från **cSolve()**.

```
cSolve(x^(1/3)=-1,x) ENTER   false
solve(x^(1/3)=-1,x) ENTER    x = -1
```

cSolve() startar med exakta symboliska metoder. Förutom i EXACT-läge använder **cSolve()** även upprepad, approximerad, komplex faktoruppdelning av polynom, om det behövs.

Läget Display Digits = Fix 2:

```
exact(cSolve(x^5+4x^4+5x
  ^3-6x-3=0,x)) ENTER
cSolve(ans(1),x) ENTER
```



Obs! Se även **cZeros()**, **solve()** och **zeros()**.

Obs! Om *ekvation* är icke-polynom med funktioner som **abs()**, **angle()**, **conj()**, **real()**, eller **imag()** lägger du till ett streck_ (**TI-89:** ◆ [_] **TI-92 Plus:** 2nd [_]) till slutet av *var*. Genom förvald inställning hanteras variabler som reella tal. Om du anger *var*_ beräknas variabeln som komplex.

z hanteras som ett reellt tal:

```
cSolve(conj(z)=1+i,z) ENTER
                      z=1+i
```

Du bör använda *var*_ för alla variabler som kan ha icke-reella värden. Oförväntade resultat kan annars uppstå.

z_ hanteras som ett komplext tal:

```
cSolve(conj(z_)=1+i,z_) ENTER
                        z_=1−i
```

---

cSolve(*ekvation1* **and** *ekvation2* [**and** …],
{*varEllerGissning1*, *varEllerGissning2* [, …]})
⇒ *Booleskt uttryck*

Returnerar möjliga komplexa lösningar till ekvationsystem där varje *varEllerGissning* motsvarar en variabel som du vill lösa ekvationen för.

Alternativt kan du ange en gissning för en variabel. Varje *varEllerGissning* måste vara på formen:

variabel
– eller –
variabel = *reellt eller icke-reellt värde*

Till exempel är x ett giltigt värde liksom x=3+*i*.

Om alla ekvationerna är polynom och du INTE anger några gissningar använder **Solve()** Gröbners/Buchberger elimineringsmetod för att försöka bestämma **alla** komplexa lösningar.

Obs! Följande exempel använder ett understrykningstecken _ (**TI-89:** ◆ [_] **TI-92 Plus:** 2nd [_]) för att variablerna ska hanteras som komplexa tal.

Komplexa lösningar kan omfatta både reella och icke-reella lösningar som i exemplet till höger.

```
cSolve(u_*v_-u_=v_ and
v_^2=⁻u_,{u_,v_}) ENTER
```

$$u_=1/2+\frac{\sqrt{3}}{2}\cdot i \text{ and } v_=1/2-\frac{\sqrt{3}}{2}\cdot i$$
$$\text{or } u_=1/2-\frac{\sqrt{3}}{2}\cdot i \text{ and } v_=1/2+\frac{\sqrt{3}}{2}\cdot i$$
$$\text{or } u_=0 \text{ and } v_=0$$

Ekvationssystem med polynom kan ha extra variabler utan värde som representerar givna numeriska värden vilka senare kan bytas ut.

```
cSolve(u_*v_-u_=c_*v_ and
v_^2=⁻u_,{u_,v_}) ENTER
```

$$u_=\frac{-(\sqrt{1-4\cdot c_}+1)^2}{4} \text{ and } v_=\frac{\sqrt{1-4\cdot c_}+1}{2}$$
$$\text{or } u_=\frac{-(\sqrt{1-4\cdot c_}-1)^2}{4} \text{ and } v_=\frac{-(\sqrt{1-4\cdot c_}-1)}{2}$$
$$\text{or } u_=0 \text{ and } v_=0$$

Du kan också ange variabler som inte ingår i ekvationerna. Dessa lösningar visar hur grupper av lösningar kan innehålla godtyckliga konstanter av formen *@k*, där *k* är ett heltalssuffix mellan 1 och 255. Suffixet återställs till 1 när du använder **ClrHome** eller F1 8:Clear Home.

```
cSolve(u_*v_-u_=v_ and
v_^2=⁻u_,{u_,v_,w_}) ENTER
```

$$u_=1/2+\frac{\sqrt{3}}{2}\cdot i \text{ and } v_=1/2-\frac{\sqrt{3}}{2}\cdot i \text{ and } w_=@1$$
or
$$u_=1/2-\frac{\sqrt{3}}{2}\cdot i \text{ and } v_=1/2+\frac{\sqrt{3}}{2}\cdot i \text{ and } w_=@1$$
$$\text{or } u_=0 \text{ and } v_=0 \text{ and } w_=@1$$

Beräkningstid och minnesbehov för polynoma system kan i stor utsträckning bero på i vilken ordning du listar dina lösningsvariabler. Om ditt första val överskrider tillgängligt minne, eller ditt tålamod, kan du ändra ordningen på variablerna i ekvationen och/eller i listan med variablerna *varEllerGissning*.

Om du inte anger några gissningar eller om någon ekvation är icke-polynom, men alla ekvationer är linjära med alla lösningsvariabler använder **cSolve()** Gauss elimineringsmetod för att försöka bestämma alla lösningar.

```
cSolve(u_+v_=e^(w_) and u_-v_=
i, {u_,v_}) ENTER
```

$$u_=\frac{e^{w_}}{2}+1/2\cdot i \text{ and } v_=\frac{e^{w_}-i}{2}$$

Om ett system varken är polynomt i alla variabler eller linjärt i alla lösningsvariabler bestämmer **cSolve()** högst en lösning genom en ungefärlig iterativ metod. För att kunna göra detta måste antalet lösningsvariabler motsvara antalet ekvationer och alla andra variabler måste förenklas till numeriska värden.

```
cSolve(e^(z_)=w_ and w_=z_^2,
{w_,z_}) ENTER
```

$$w_=.494\ldots \text{ and } z_=⁻.703\ldots$$

En icke-reell gissning är ofta nödvändig för att bestämma en icke-reell lösning. Konvergenskraven gör att gissningen måste vara ganska nära lösningen.

```
cSolve(e^(z_)=w_ and w_=z_^2,
{w_,z_=1+ i}) ENTER
```

$$w_=.149\ldots+4.891\ldots\cdot i \text{ and } z_=1.588\ldots+1.540\ldots\cdot i$$

CubicReg — Menyn MATH/Statistics/Regressions

CubicReg *lista1*, *lista2*[, [*lista3*] [, *lista4*, *lista5*]]

Beräknar tredjegradspolynomets regressionskoefficienter och uppdaterar alla statistiska variabler.

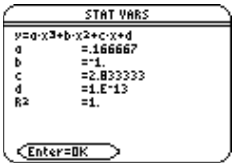
Alla listor måste ha samma dimension förutom *lista5*.

lista1 motsvarar x-listan.
lista2 motsvarar y-listan.
lista3 motsvarar frekvens.
lista4 motsvarar kategorivärden.
lista5 motsvarar de kategorivärden som ska inkluderas.

Obs! *lista1* till och med *lista4* måste vara ett variabelnamn eller c1–c99 (kolumner i den senaste datavariabeln som visades i Data/Matrix Editor). *lista5* behöver inte vara ett variabelnamn och kan inte vara någon av c1–c99.

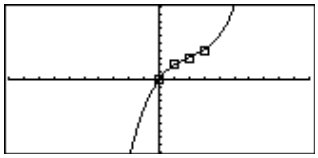
I funktionsplottningsläge.

```
{0,1,2,3}→L1 ENTER        {0 1 2 ...}
{0,2,3,4}→L2 ENTER        {0 2 3 ...}
CubicReg L1,L2 ENTER          Done
ShowStat ENTER
```




```
ENTER
regeq(x)→y1(x) ENTER          Done
NewPlot 1,1,L1,L2 ENTER       Done
```

◆ [GRAPH]



cumSum() — Menyn MATH/List

cumSum(*lista1***)** ⇒ *lista*

Returnerar en lista med den kumulativa summan av elementen i *lista1*, med början på element 1.

```
cumSum({1,2,3,4}) ENTER
                    {1 3 6 10}
```

---

cumSum(*matris1***)** ⇒ *matris*

Returnerar en matris med den kumulativa summan av elementen i *matrix1*. Varje element är den växande summan av kolumnen från topp till botten.

```
[1,2;3,4;5,6]→m1 ENTER
```

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

```
cumSum(m1) ENTER
```

$$\begin{bmatrix} 1 & 2 \\ 4 & 6 \\ 9 & 12 \end{bmatrix}$$

CustmOff — CATALOG

CustmOff

Tar bort ett anpassat verktygsfält.

CustmOn och **CustmOff** låter ett program kontrollera ett anpassat verktygsfält. Manuellt kan du använda 2nd [CUSTOM] för att ta fram och ta bort ett anpassat verktygsfält. Anpassade verktygsfält tas också bort när du växlar mellan program.

Se programlistning och exempel under **Custom**.

CustmOn — CATALOG

CustmOn

Aktiverar ett anpassat verktygsfält som tidigare angetts i blocket **Custom...EndCustm**.

CustmOn och **CustmOff** låter ett program kontrollera ett anpassat verktygsfält. Manuellt kan du använda 2nd [CUSTOM] för att ta fram och

Se programlistning och exempel under **Custom**.

ta bort ett anpassat verktygsfält.

Custom — tangenterna 2nd [CUSTOM]

Custom
block
EndCustm

Skapar ett verktygsfält som aktiveras när du trycker på 2nd [CUSTOM]. Påminner mycket om instruktionen **ToolBar**, förutom att programsatserna Title och Item inte kan ha etiketter.

block kan antingen vara en enskild programsats eller en serie programsatser som avgränsas med tecknet ":".

Obs! 2nd [CUSTOM] fungerar som en växelfunktion. Den första förekomsten öppnar menyn och den andra förekomsten tar bort menyn. Menyn tas även bort när du byter program.

Programlista:

```
:Test()
:Prgm
:Custom
:Title  "Lists"
:Item   "List1"
:Item   "Scores"
:Item   "L3"
:Title  "Fractions"
:Item   "f(x)"
:Item   "h(x)"
:Title  "Graph"
:EndCustm
:EndPrgm
```

Cycle — CATALOG

Cycle

Överför programkontrollen direkt till nästa upprepning av den aktuella slingan (**For**, **While** eller **Loop**).

Cycle är inte tillåten utanför de tre slingstrukturerna (**For**, **While** eller **Loop**).

Programlista:

```
:◙ Sum the integers from 1 to
  100 skipping 50.
:0→temp
:For i,1,100,1
:If i=50
:Cycle
:temp+i→temp
:EndFor
:Disp temp
```

Innehåll i `temp` efter körning: `5000`

CyclePic — CATALOG

CyclePic *bildnamnssträng*, *n* [, [*vänta*] , [*cykler*], [*riktning*]

Visar alla bildvariabler som är definierade och vid angivet intervall. Användaren kan styra tiden mellan bilder, hur många gånger som bilderna ska gås igenom och åt vilket håll de ska gå, cirkulärt eller framåt och bakåt.

riktning är 1 för circulär eller ⁻1 för framåt och bakåt. Standard är = 1.

1. Spara tre bilder med namnen pic1, pic2 och pic3.
2. Skriv: `CyclePic "pic",3,.5,4,⁻1`
3. De tre bilderna (3) kommer att visas automatiskt med en halv sekund (.5) mellan bilderna, i fyra cykler (4) och framåt och bakåt (⁻1).

▸Cylind — Menyn MATH/Matrix/Vector ops

vektor **▸Cylind**

Visar rad- eller kolumnvektorn i cylindrisk form [r∠θ, z].

vektor måste ha exakt tre element. Det kan vara antingen en rad eller en kolumn.

`[2,2,3] ▸Cylind` ENTER

$[2\cdot\sqrt{2}\ \angle\frac{\pi}{4}\quad 3]$

cZeros() Menyn MATH/Algebra/Complex

cZeros(*uttryck***, *var*)** ⇒ *lista*

Returnerar en lista med lösningar av reella och icke-reella värden av *var* som löser *uttryck*=0. **cZeros()** gör detta genom att beräkna **exp▸list(cSolve(***uttryck*=0,*var***),***var***)**. För övrigt påminner **cZeros()** mycket om **zeros()**.

Obs! Se även **cSolve()**, **solve()** och **zeros().**

Obs! Om *ekvation* är icke-polynom med funktioner som **abs()**, **angle()**, **conj()**, **real()**, eller **imag()**, lägger du till ett streck_ (**TI-89:** [◆] [_] **TI-92 Plus:** [2nd] [_]) till slutet av *var*. Genom förvald inställning hanteras variabler som reella tal. Om du anger *var*_ beräknas variabeln som komplex.

Du bör använda *var*_ för alla variabler som kan ha icke-reella värden. Oväntade resultat kan annars uppstå.

Läget Display Digits = Fix 3:

```
cZeros(x^5+4x^4+5x^3-6x-3,x)
ENTER
{-2.125  -.612  .965
  -1.114-1.073·i
  -1.114+1.073·i}·i}
```

z hanteras som reellt värde:

```
cZeros(conj(z)-1-i,z) ENTER
{1+i}
```

z_ hanteras som komplext värde:

```
cZeros(conj(z_)-1-i,z_) ENTER
{1-i}
```

cZeros({*uttryck1***,** *uttryck2* **[,** ... **]},** **{***varEllerGissning1***,** *varEllerGissning2* **[,** ... **]})** ⇒ *matris*

Returnerar möjliga positioner där uttrycken är noll samtidigt. Varje *varEllerGissning* motsvarar en okänd variabel vars värde du söker.

Alternativt kan du ange en gissning för en variabel. Varje *varEllerGissning* måste vara i formen:

variabel
– eller –
variabel = reellt eller icke-reellt värde

Till exempel är x ett giltigt värde liksom x=3+***i***.

Om alla ekvationerna är polynoma och du INTE anger några gissningar använder **cZeros()** Gröbners/Buchberger elimineringsmetod för att försöka bestämma **alla** komplexa nollställen.

Obs! Följande exempel använder ett understrykningstecken _ (**TI-89:** ◆ [_] **TI-92 Plus:** 2nd [_]) för att variablerna ska hanteras som komplexa tal.

Komplexa nollställen kan omfatta både reella och icke-reella lösningar som i exemplet till höger.

```
cZeros({u_*v_-u_-v_,v_^2+u_},
{u_,v_}) ENTER
```

$$\begin{bmatrix} 1/2 - \frac{\sqrt{3}}{2}\cdot i & 1/2 + \frac{\sqrt{3}}{2}\cdot i \\ 1/2 + \frac{\sqrt{3}}{2}\cdot i & 1/2 - \frac{\sqrt{3}}{2}\cdot i \\ 0 & 0 \end{bmatrix}$$

Varje rad i den resulterande matrisen motsvarar en alternativ nollställe, med komponenterna ordnade i samma ordning som listan med *varEllerGissning*. Om du vill ta ut en rad indexerar du matrisen med [*row*].

Extract row 2:

```
ans(1)[2] ENTER
```

$$\begin{bmatrix} 1/2 + \cdot i & 1/2 - \frac{\sqrt{3}}{2}\cdot i \end{bmatrix}$$

System av polynom kan ha extra variabler utan värde, som representerar givna numeriska värden vilka senare kan bytas ut.

```
cZeros({u_*v_-u_-(c_*v_),
v_^2+u_},{u_,v_}) ENTER
```

$$\begin{bmatrix} \frac{-(\sqrt{1-4\cdot c_}+1)^2}{4} & \frac{\sqrt{1-4\cdot c_}+1}{2} \\ \frac{-(\sqrt{1-4\cdot c_}-1)^2}{4} & \frac{-(\sqrt{1-4\cdot c_}-1)}{2} \\ 0 & 0 \end{bmatrix}$$

Du kan också ange variabler som inte ingår i uttrycken. Dessa nollställen visar hur grupper av nollställen kan innehålla godtyckliga konstanter av formen @*k*, där *k* är ett heltalssuffix mellan 1 och 255. Suffixet återställs till 1 när du använder **ClrHome** eller F1 8:Clear Home.

```
cZeros({u_*v_-u_-v_,v_^2+u_},
{u_,v_,w_}) ENTER
```

$$\begin{bmatrix} 1/2 - \frac{\sqrt{3}}{2}\cdot i & 1/2 + \frac{\sqrt{3}}{2}\cdot i & @1 \\ 1/2 + \frac{\sqrt{3}}{2}\cdot i & 1/2 - \frac{\sqrt{3}}{2}\cdot i & @1 \\ 0 & 0 & @1 \end{bmatrix}$$

Beräkningstid och minnesbehov för system av polynom kan i stor utsträckning bero på i vilken ordning du listar okända värden. Om ditt första val överskrider tillgängligt minne, eller ditt tålamod, kan du ändra ordningen på variablerna i uttrycken och/eller i listan med variablerna *varEllerGissning*.

Om du inte anger några gissningar och om något uttryck är icke-polynomt i alla variabler, men alla uttryck är linjära i alla okända variabler använder **cZeros()** Gauss elimineringsmetod för att försöka bestämma alla nollställen.

cZeros({u_+v_-e^(w_),u_-v_-i},{u_,v_}) ENTER

$$\left[\frac{e^{w_}}{2}+1/2\cdot i \quad \frac{e^{w_}-i}{2}\right]$$

Om ett system varken är polynomt i alla variabler eller linjärt i alla okända variabler bestämmer **cZeros()** högst ett nollställe genom en ungefärlig iterativ metod. För att kunna göra detta måste antalet okända variabler motsvara antalet uttryck och alla andra variabler i uttrycken måste förenklas till numeriska värden.

cZeros({e^(z_)-w_,w_-z_^2},{w_,z_}) ENTER

[.494… -.703…]

En icke-reell gissning är ofta nödvändig för att bestämma ett icke-reellt nollställe. Konvergenskraven gör att gissningen måste vara ganska nära ett nollställe.

cZeros({e^(z_)-w_,w_-z_^2},{w_,z_=1+i}) ENTER

[.149…+4.89…·i 1.588…+1.540…·i]

d() tangenterna 2nd [*d*] eller menyn MATH/Calculus

d*(uttryck1*, *var* [,*ordning*]**)** ⇒ *uttryck*
d*(lista1*,*var* [,*ordning*]**)** ⇒ *lista*
d*(matris1*,*var* [,*ordning*]**)** ⇒ *matris*

Returnerar den förstaderivatan av *uttryck1* med avseende på *var*. *uttryck1* kan vara en lista eller en matris.

ordning, om den är inkluderad, måste vara ett heltal. Om ordningen är mindre än noll blir resultatet en antiderivata.

***d*()** följer inte den normala beräkningsföljden att fullständigt förenkla sitt argument och sedan lägga på funktionsdefinitionen till dessa fullständigt förenklade argument. Istället utförs följande steg:

1. Det andra argumentet förenklas endast i den utsträckning att det inte leder till en icke-variabel.
2. Det första argumentet förenklas endast i den utsträckning att det hämtar sparade värden för variabeln som angavs i steg 1.
3. Den symboliska derivatan av resultatet i steg 2 bestäms med avseende på variabeln i steg 1.
4. Om variabeln i steg 1 har ett sparat värde eller ett värde som är angivet av operatorn "with" (|), substitueras det värdet i resultatet från steg 3.

d(3x^3-x+7,x) ENTER $9x^2-1$

d(3x^3-x+7,x,2) ENTER $18\cdot x$

d(f(x)*g(x),x) ENTER

$$\frac{d}{dx}(f(x))\cdot g(x)+\frac{d}{dx}(g(x))\cdot f(x)$$

d(sin(f(x)),x) ENTER

$$\cos(f(x))\frac{d}{dx}(f(x))$$

d(x^3,x)|x=5 ENTER 75

d(d(x^2*y^3,x),y) ENTER $6\cdot y^2\cdot x$

d(x^2,x,-1) ENTER $\frac{x^3}{3}$

d({x^2,x^3,x^4},x) ENTER

$\{2\cdot x \quad 3\cdot x^2 \quad 4\cdot x^3\}$

▸DD Menyn MATH/Angle

tal ▸**DD** ⇒ *värde*
lista1 ▸**DD** ⇒ *lista*
matris1 ▸**DD** ⇒ *matris*

Returnerar motsvarande decimaltal av argumentet. Argumentet är ett tal, en lista eller en matris som tolkas av vinkelenheten radianer eller grader.

Obs! ▸**DD** kan också acceptera argumentet i radianer.

Med vinkelenheten grader:

```
1.5° ▸DD ENTER                    1.5°
45° 22'14.3" ▸DD ENTER            45.370...°
{45° 22'14.3",60° 0'0"} ▸DD ENTER
                          {45.370...  60}°
```

Med vinkelenheten radianer:

```
1.5 ▸DD ENTER                     85.9°
```

▸Dec MATH/Base menu

heltal1 ▸**Dec** ⇒ *heltal*

Konverterar *heltal1* till ett decimalt värde (bas 10). Ett binärt eller hexadecimalt värde måste ha prefixet 0b respektive 0h.

Noll, inte bokstaven O, följt av b alternativt h.

0b *binärt tal*
0h *hexadecimalt tal*

Ett binärt tal kan bestå av upp till 32 tecken. Ett hexadecimalt tal kan bestå av upp till 8 tecken.

Utan prefix hanteras *heltal1* som ett decimalt tal. Resultatet visas i decimalt läge oavsett inställningar.

```
0b10011 ▸Dec ENTER                19
0h1F ▸Dec ENTER                   31
```

Define CATALOG

Define *funktionsnamn*(*arg1namn*, *arg2namn*, ...) = *uttryck*

Skapar *funktionsnamn* som en användardefinierad funktion. Du kan därefter använda *funktionsnamn*() på samma sätt som med inbyggda funktioner. Funktionen beräknar *uttryck* med hjälp av de angivna argumenten och returnerar resultatet.

funktionsnamn kan inte vara namnet på en systemvariabel eller inbyggd funktion.

Argumentnamn är platshållare, du behöver inte använda dessa namn som argument när du använder funktionen.

Obs! Denna form av **Define** är detsamma som att köra uttrycket: *uttryck*→ *funktionsnamn*(*arg1namn*, *arg2namn*). Du kan också använda detta kommando för att definiera enkla variabler, exempelvis Define a=3.

```
Define g(x,y)=2x-3y ENTER         Done
g(1,2) ENTER                      -4
1→a:2→b:g(a,b) ENTER              -4

Define h(x)=when(x<2,2x-3,
-2x+3) ENTER                      Done

h(-3) ENTER                       -9
h(4) ENTER                        -5

Define eigenvl(a)=
  cZeros(det(identity(dim(a)
  [1])-x*a),x) ENTER              Done
eigenvl([-1,2;4,3]) ENTER
```

$$\left\{\frac{2\cdot\sqrt{3}-1}{11}\quad\frac{-(2\cdot\sqrt{3}+1)}{11}\right\}$$

Define *funktionsnamn*(*arg1namn*, *arg2namn*, ...) **= Func**
block
EndFunc

Identisk med den föregående formen av **Define**, förutom att den användardefinierade funktionen *funktionsnamn*() i denna form kan köra ett block med flera programsatser.

block kan antingen vara en enstaka programsats eller en serie programsatser som avgränsas med tecknet ":". *block* kan även innehålla uttryck och instruktioner (exempelvis **If**, **Then**, **Else** och **For**). Detta gör att funktionen *funktionsnamn*() kan använda instruktionen **Return** för att returnera ett visst resultat.

Obs! Det är oftast lättare att skriva och redigera denna form av funktionen i Program Editor i stället för på inmatningsraden.

```
Define g(x,y)=Func:If x>y Then
:Return x:Else:Return y:EndIf
:EndFunc ENTER                Done

g(3,⁻7) ENTER                    3
```

Define *programnamn*(*arg1namn*, *arg2namn*, ...) **= Prgm**
block
EndPrgm

Skapar *programnamn* som ett program eller subprogram, men kan inte returnera ett resultat med hjälp av **Return**. Kan köra ett block av flera programsatser.

block kan antingen vara en enstaka programsats eller en serie programsatser som avgränsas med tecknet":". *block* kan även innehålla uttryck och instruktioner (exempelvis **If**, **Then**, **Else** och **For**) utan restriktioner.

Obs! Det är oftast lättare att skriva och redigera denna form av funktionen i Program Editor i stället för på inmatningsraden.

```
Define listinpt()=prgm:Local
  n,i,str1,num:InputStr "Enter
  name of list",str1:Input "No.
  of elements",n:For
  i,1,n,1:Input "element
  "&string(i),num:
  num→#str1[i]:EndFor:EndPrgm
  ENTER
                              Done

listinpt() ENTER Enter name of list
```

DelFold CATALOG

DelFold *mappnamn1*[,*mappnamn2*] [,*mappnamn3*] **...**

Tar bort användardefinierade mappar med namnen *mappnamn1*, *mappnamn2* osv. Ett felmeddelande visas om mapparna innehåller några variabler.

Obs! Du kan inte ta bort mappen main.

```
NewFold games ENTER           Done
```
(skapar mappen games)

```
DelFold games ENTER           Done
```
(tar bort mappen games)

DelVar CATALOG

DelVar *var1*[,*var2*] [,*var3*] **...**

Tar bort de angivna variablerna från minnet.

```
2→a ENTER                    2
(a+2)^2 ENTER               16
DelVar a ENTER            Done
(a+2)^2 ENTER           (a + 2)^2
```

deSolve() MATH/Calculus menu

deSolve(*1aEl2aOrdnOde*, *oberoendeVar*, *beroendeVar***)** ⇒ *en allmän lösning*

Returnerar en ekvation som explicit eller implicit anger en allmän lösning till en första eller andra ordningens ordinära differentialekvation (ODE). I ODE:

- Använd ett primtecken (', tryck 2nd [′]) för att ange förstaderivatan av den beroende variabeln med avseende på den oberoende variabeln.
- Använd två primtecken för att ange den motsvarande andraderivatan.

Tecknet ' används för derivata endast i funktionen **deSolve()**. Andra funktioner använder ***d*()**.

Den allmänna lösningen av en första ordningens ekvation innehåller en godtycklig konstant i formen @*k*, där *k* är ett heltalssuffix mellan 1 och 255. Suffixet återställs till 1 när du använder **ClrHome** eller F1 8: Clear Home. Lösningen till en andra ordningens ekvation innehåller två sådana konstanter.

Obs! När du vill skriva en primtalsymbol (') trycker du 2nd [′].

```
deSolve(y''+2y'+y=x^2,x,y)
ENTER
```

$$y=(@1\cdot x+@2)\cdot e^{-x}+x^2-4\cdot x+6$$

```
right(ans(1))→temp ENTER
```

$$(@1\cdot x+@2)\cdot e^{-x}+x^2-4\cdot x+6$$

```
d(temp,x,2)+2*d(temp,x)+temp-x
^2 ENTER                      0
delVar temp ENTER          Done
```

Använd **solve()** med en implicit lösning om du vill försöka konvertera den till en eller flera motsvarande explicita lösningar.

När du jämför dina resultat med lärobokens, eller egna manuella beräkningar, kan dessa skilja sig åt då olika metoder inför godtyckliga konstanter vid olika punkter i kalkylen.

```
deSolve(y'=(cos(y))^2*x,x,y)
ENTER
```

$$\tan(y)=\frac{x^2}{2}+@3$$

```
solve(ans(1),y) ENTER
```

$$y=\tan^{-1}\left(\frac{x^2+2\cdot @3}{2}\right)+@n1\cdot\pi$$

Obs! När du vill skriva tecknet @ trycker du:

TI-89: ◆ STO▸
TI-92 Plus: 2nd R

```
ans(1)|@3=c-1 and @n1=0 ENTER
```

$$y=\tan^{-1}\left(\frac{x^2+2\cdot(c-1)}{2}\right)$$

deSolve(*1aOrdnOde* **and** *begynnelseVillkor*, *oberoendeVar*, *beroendeVar***)** ⇒ *en särskild lösning*

Returnerar en särskild lösning som uppfyller kraven i *1aOrdnOde* och *begynnelseVillkor*. Detta är oftast enklare än att bestämma en allmän lösning, ersätta ursprungliga värden, lösa för den godtyckliga konstanten och sätta in detta värde i den allmänna lösningen

begynnelseVillkor är en ekvation i formen:

beroendeVar (*ursprungligtOberoendeVärde*) = *ursprungligtBeroendeVärde*

ursprungligtOberoendeVärde och *ursprungligtBeroendeVärde* kan vara variabler som x0 och y0 som inte har några lagrade värden. Implicit differentiering kan vara till hjälp vid verifiering av implicita lösningar.

```
sin(y)=(y*e^(x)+cos(y))y'→ode ENTER
```

$$\sin(y)=(e^{x}\cdot y+\cos(y))\cdot y'$$

```
deSolve(ode and y(0)=0,x,y)→soln ENTER
```

$$\frac{-(2\cdot\sin(y)+y^{2})}{2}=-(e^{x}-1)\cdot e^{-x}\cdot\sin(y)$$

```
soln|x=0 and y=0 ENTER        true
```

```
d(right(eq)-left(eq),x)/(d(left(eq)-right(eq),y))→impdif(eq,x,y) ENTER
                                  Done
ode|y'=impdif(soln,x,y) ENTER
                                  true
delVar ode,soln ENTER         Done
```

deSolve(*2OrdnOde* **and** *begynnelseVärde1* **and** *begynnelseVärde2*, *oberoendeVar*, *beroendeVar***)** ⇒ *en särskild lösning*

Returnerar en särskild lösning som uppfyller kraven i *2OrdnOde* och har ett angivet värde för den beroende variabeln och dess derivata i en punkt.

För *begynnelseVärde1* används formen:

beroendeVar (*startOberoendeVärde*) = *startBeroendeVärde*

För *begynnelseVärde2* används formen:

beroendeVar' (*startOberoendeVärde*) = *start1aDerivataVärde*

```
deSolve(y''=y^(-1/2) and y(0)=0 and y'(0)=0,t,y) ENTER
```

$$\frac{2\cdot y^{3/4}}{3}=t$$

```
solve(ans(1),y) ENTER
```

$$y=\frac{2^{2/3}\cdot(3\cdot t)^{4/3}}{4}\text{ and }t\geq 0$$

deSolve(*2OrdnOde* **and** *randVillkor1* **and** *randVillkor2*, *oberoendeVar*, *beroendeVar***)** ⇒ *en särskild lösning*

Returnerar en särskild lösning som uppfyller kraven i *2OrdnOde* med specificerade värden i två olika punkter.

```
deSolve(w''-2w'/x+(9+2/x^2)w=x*e^(x) and w(π/6)=0 and w(π/3)=0,x,w) ENTER
```

$$w=\frac{e^{\frac{\pi}{3}}\cdot x\cdot\cos(3\cdot x)}{10}-\frac{e^{\frac{\pi}{6}}\cdot x\cdot\sin(3\cdot x)}{10}+\frac{x\cdot e^{x}}{10}$$

det() Menyn MATH/Matrix

det(*kvadratiskMatris*[, *tol*]**)** ⇒ *uttryck*

Returnerar determinanten av *kvadratiskMatris*.

Varje element i en matris betraktas som noll om dess absolutvärde är mindre än *tol*. Denna tolerans används endast om matrisen innehåller decimala värden och inte innehåller symboliska variabler som saknar värde.
I andra fall ignoreras *tol*.

- Om du använder ◆ ENTER eller anger läget Exact/Approx=APPROXIMATE utförs beräkningar med flyttalsaritmetik.
- Om *tol* utesluts eller inte används beräknas standardtoleransen enligt:

  5E⁻14 ∗ **max(dim(***kvadratiskMatris***)) ∗ rowNorm(***kvadratiskMatris***)**

```
det([a,b;c,d]) ENTER        a·d−b·c
det([1,2;3,4]) ENTER            -2
det(identity(3)−x*[1,-2,3;
-2,4,1;-6,-2,7]) ENTER
        -(98·x³−55·x²+12·x−1)
[1E20,1;0,1]·mat1        [1.E20  1
                          0      1]
det(mat1) ENTER                  0
det(mat1,.1) ENTER          1.E20
```

diag() Menyn MATH/Matrix

diag(*lista***)** ⇒ *matris*
diag(*radmatris***)** ⇒ *matris*
diag(*kolumnmatris***)** ⇒ *matris*

Returnerar en matris med värdena i argumentlistan eller argumentmatrisen som sin diagonal.

```
diag({2,4,6}) ENTER
```

$$\begin{bmatrix} 2 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 6 \end{bmatrix}$$

diag(*kvadratiskMatris***)** ⇒ *radmatris*

Returnerar en radmatris som innehåller elementen från diagonalen av *kvadratiskMatris*.

kvadratiskMatris måste vara kvadratisk.

```
[4,6,8;1,2,3;5,7,9] ENTER
```

$$\begin{bmatrix} 4 & 6 & 8 \\ 1 & 2 & 3 \\ 5 & 7 & 9 \end{bmatrix}$$

```
diag(ans(1)) ENTER          [4 2 9]
```

Dialog CATALOG

Dialog
block
EndDlog

Skapar en dialogruta när programmet körs.

block kan vara antingen en enstaka programsats eller en serie av programsatser som avgränsas med tecknet ":". Giltiga alternativ för *block* i menyalternativet F3 I/O, 1:Dialog i Program Editor är 1:Text, 2:Request, 4:DropDown och 7:Title.

Variablerna i en dialogruta kan vara givna värden som kommer att visas som standardvärden. Om du trycker på ENTER uppdateras variablerna från dialogrutan och variabeln ok ställs in till 1. Om du trycker på ESC uppdateras inte variablerna och systemvariabeln ok ställs in till noll.

Programlista:

```
:Dlogtest()
:Prgm
:Dialog
:Title    "This is a dialog box"
:Request  "Your name",Str1
:Dropdown "Month you were born",
  seq(string(i),i,1,12),Var1
:EndDlog
:EndPrgm
```

dim() Menyn MATH/Matrix/Dimensions

dim(*lista***)** ⇒ *heltal*

Returnerar dimensionen för *lista*.

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

dim(*matris***)** ⇒ *lista*

Returnerar dimensionen för *matris* som en lista med två element {rader, kolumner}.

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

dim(*sträng***)** ⇒ *heltal*

Returnerar antalet tecken som finns i teckensträngen *sträng*.

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

Disp CATALOG

Disp [*uttryckEllerText1*] [, *uttryckEllerText2*] **...**

Visar aktuellt innehåll från skärmen för Program I/O. Om en eller flera *uttryckEllerText* angivits visas varje uttryck eller sträng på en egen rad i Program I/O-fönstret.

Ett uttryck kan innehålla omvandlingsoperationer som t ex **▸DD** och **▸Rect**. Du kan också använda operatorn ▸ om du vill genomföra enhets- och talomvandlingar.

Om Pretty Print = ON visas uttryck i pretty print.

Från skärmen Program I/O kan du trycka på F5 för att visa skärmen Home. Ett program kan använda funktionen **DispHome.**

```
Disp "Hello" ENTER          Hello
Disp cos(2.3) ENTER         -.666…
{1,2,3,4}→L1 ENTER
Disp L1 ENTER               {1  2  3  4}
Disp 180_min▸_hr ENTER      3.•_hr
```

Obs! När du vill skriva ett understrykningstecken (_) trycker du:
TI-89: ◆ [_]
TI-92 Plus: 2nd [_]
När du vill skriva ▸, tryck på 2nd [▸].

DispG CATALOG

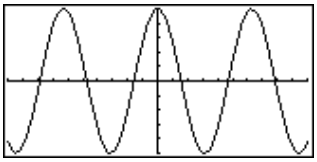
DispG

Visar det aktuella innehållet i graffönstret.

I funktionsplottningsläge:

Programsegment:

```
  ⋮
:5*cos(x)→y1(x)
:-10→xmin
:10→xmax
:-5→ymin
:5→ymax
:DispG
  ⋮
```



DispHome CATALOG

DispHome

Visar aktuellt innehåll i grundfönstret.

Programsegment:

```
  ⋮
:Disp "The result is: ",xx
:Pause "Press Enter to quit"
:DispHome
:EndPrgm
```

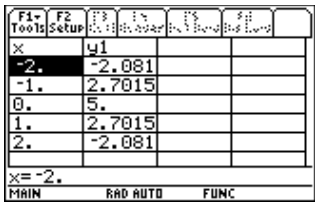
DispTbl CATALOG

DispTbl

Visar det aktuella innehållet i tabellfönstret.

Obs! Markörknappen är aktiv för rullning. Tryck på [ESC] eller [ENTER] för att fortsätta körningen om du är i ett program.

```
5*cos(x)→y1(x) [ENTER]
DispTbl [ENTER]
```



▸DMS Menyn MATH/Angle

uttryck **▸DMS**
lista **▸DMS**
matris **▸DMS**

Tolkar argumentet som en vinkel angiven som ett decimalt gradtal och visar motsvarande DMS-tal (*DDDDDD°MM′SS.ss″*). Se °, ', " på sidan 531 för formatet DMS (grader, minuter, sekunder).

Obs! ▸DMS är en visningsformatinstruktion, inte en konverteringsfunktion. Den visas endast i slutet av en inmatningsrad.

Med vinkelenheten grader:

```
45.371 ▸DMS [ENTER]        45° 22'15.6"

{45.371,60} ▸DMS [ENTER]
                 {45° 22'15.6"  60° }
```

dotP() Menyn MATH/Matrix/Vector ops

dotP(*lista1*, *lista2***)** ⇒ *uttryck*

Returnerar den "elementvisa" produkten av två listor.

```
dotP({a,b,c},{d,e,f}) [ENTER]
                        a·d+b·e+c·f

dotP({1,2},{5,6}) [ENTER]          17
```

---

dotP(*vektor1*, *vektor2***)** ⇒ *uttryck*

Returnerar den "elementvisa" produkten av två vektorer.

Båda måste vara radvektorer eller så måste båda vara kolumnvektorer.

```
dotP([a,b,c],[d,e,f]) [ENTER]
                        a·d+b·e+c·f

dotP([1,2,3],[4,5,6]) [ENTER]      32
```

DrawFunc CATALOG

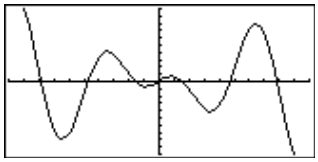
DrawFunc *uttryck*

Ritar *uttryck* som en funktion, genom att använda x som den oberoende variabeln.

Obs! Vid omritning tas alla ritobjekt bort.

I funktionsplottningsläge och ZoomStd-fönstret:

```
DrawFunc 1.25x*cos(x) [ENTER]
```



DrawInv CATALOG

DrawInv *uttryck*

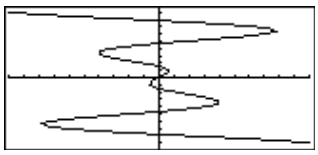
Ritar inversen av *uttryck* genom att plotta x-värden på y-axeln och y-värden på x-axeln.

x är den oberoende variabeln.

Obs! Vid omritning tas alla ritobjekt bort.

I funktionsplottningsläge och ZoomStd-fönstret:

```
DrawInv 1.25x*cos(x) ENTER
```



DrawParm CATALOG

DrawParm *uttryck1*, *uttryck 2* [, *tmin*] [, *tmax*] [, *tsteg*]

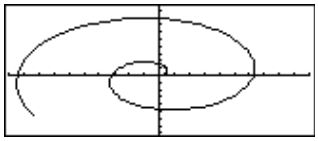
Ritar parameterekvationerna *uttryck1* och *uttryck2*, genom att använda t som den oberoende variabeln.

Standardinställningarna för *tmin*, *tmax* och *tsteg* är de aktuella inställningarna för Window-variablerna tmin, tmax och tstep.
Att ange värden ändrar inte fönster-inställningarna. Om det aktuella grafläget inte är parameterform, krävs dessa tre argument.

Obs! Vid omritning tas alla ritobjekt bort.

I funktionsplottningsläge och ZoomStd-fönstret:

```
DrawParm
t*cos(t),t*sin(t),0,10,.1 ENTER
```



DrawPol CATALOG

DrawPol *uttryck*[, θ*min*] [, θ*max*] [, θ*steg*]

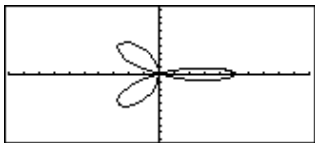
Ritar den polär grafen av *uttryck* genom att använda θ som den oberoende variabeln.

Standardinställningarna för θ*min*, θ*max* och θ*steg* är de aktuella inställningarna för Window-variablerna θmin, θmax och θstep. Att ange värden ändrar inte fönsterinställningarna. Om det aktuella grafläget inte är polärt, krävs dessa tre argument.

Obs! Vid omritning tas alla ritobjekt bort.

I funktionsplottningsläge och ZoomStd-fönstret:

```
DrawPol 5*cos(3*θ),0,3.5,.1
ENTER
```



DrawSlp CATALOG

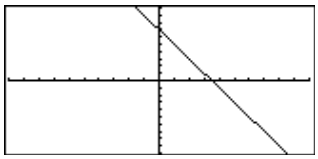
DrawSlp *x1*, *y1*, *lutning*

Visar grafen och ritar en linje genom att använda formeln y−y1=slope•(x−x1).

Obs! Vid omritning tas alla ritobjekt bort.

I funktionsplottningsläge och ZoomStd-fönstret:

```
DrawSlp 2,3,⁻2 ENTER
```



DropDown CATALOG

DropDown *titelsträng*, **{***alt1sträng*, *alt2sträng*, **...}**, *varnamn*

Visar en snabbmeny med namnet *titelsträng* som innehåller alternativen **1:***alt1sträng*, **2:***alt2sträng* osv. **DropDown** måste vara i ett **Dialog...EndDlog**-block.

Om *varnamn* redan finns och har ett värde i intervallet av alternativ visas det refererade alternativet som standardalternativ. Annars är menyns första alternativ standardalternativet.

När du väljer ett alternativ på menyn sparas motsvarande nummer på alternativet i variabeln *varnamn*. (Om det behövs skapar **DropDown** *varnamn*.)

Se programlistan för **Dialog**.

DrwCtour CATALOG

DrwCtour *uttryck*
DrwCtour *lista*

Ritar konturerna av den aktuella 3D-grafen vid de z-värden som anges av *uttryck* eller *lista*. 3D-grafläge måste vara valt i förväg. **DrwCtour** ställer automatiskt in formatet till CONTOUR LEVELS.

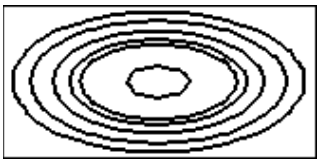
Genom förval innehåller grafen automatiskt det antal jämt distribuerade konturer som anges av Windows-variabeln ncontour. **DrwCtour** ritar upp extra konturer till de förvalda.

Om du vill stänga av förvalda konturer sätter du ncontour till noll, antingen från Windows-skärmen eller genom att lagra 0 i system-variabeln ncontour.

In 3D graphing mode:

I 3D-grafläge:

```
(1/5)x^2+(1/5)y^2-10•z1(x,y)
ENTER
                                Done
-10•xmin:10•xmax ENTER           10
-10•ymin:10•ymax ENTER           10
-10•zmin:10•zmax ENTER           10
0•ncontour ENTER                  0
DrwCtour {-9,-4.5,-3,0,4.5,9}
ENTER
```



- Använd markören för att ändra visningsvinkel. Tryck på 0 (noll) när du vill återgå till orginalvyn.
- Om du vill växla mellan olika grafformat trycker du på:
- **TI-89**: I **TI-92 Plus**: F
- Tryck på X, Y eller Z när du vill visa grafen längs motsvarande axel.

E TI-89: EE tangenten TI-92 Plus: 2nd EE tangenten

*mantissa***E***exponent*

Visar ett tal i grundpotensform. Talet tolkas som *mantissa* × $10^{exponent}$.

Tips! Om du vill göra 10 upphöjt till någonting utan att skapa ett decimaltal, använder du 10^*heltal*.

```
2.3E4 ENTER                 23000.
2.3E9+4.1E15 ENTER          4.1E15
3*10^4 ENTER                30000
```

e^( )

TI-89: [◆] [e^x] **tangenten** **TI-92 Plus:** [2nd] [e^x] **tangenten**

e*^(uttryck1***)** ⇒ *uttryck*

Returnerar *e* upphöjt till *uttryck1*.

Obs! Att trycka på [◆] [e^x] på TI-89 för att visa *e*^(är inte detsamma som att trycka på [alpha] [E]). På TI-92 Plus är [2nd][e^x] för att skriva e^ inte samma sak som att skriva e på tangentbordet.

Du kan skriva in komplexa tal i den polära formen $re^{i\theta}$. Använd detta endast med vinkelinställningen Radian. Om du använder den med vinkelinställningen Degree orsakar den ett fel av typen Domain error.

```
e^(1) ENTER                    e
e^(1.) ENTER                   2.718...
e^(3)^2 ENTER                  e^9
```

---

e*^(lista1***)** ⇒ *lista*

Returnerar *e* upphöjt till varje element i *lista1*.

```
e^({1,1.,0,.5}) ENTER
        {e   2.718...   1   1.648...}
```

---

e*^(kvadratiskMatris1***)** ⇒ *kvadratiskMatris*

Returnerar matrisexponenten av matrisen *kvadratiskMatris1*. Detta är *inte* samma sak som att beräkna *e* upphöjt till varje element. Mer information om beräkningsmetoden finns i avsnittet om **cos()**.

kvadratiskMatris1 måste vara diagonaliserbar. Resultatet anges alltid med flyttalsaritmetik.

```
e^([1,5,3;4,2,1;6,-2,1]) ENTER
```

$$\begin{bmatrix} 782.209 & 559.617 & 456.509 \\ 680.546 & 488.795 & 396.521 \\ 524.929 & 371.222 & 307.879 \end{bmatrix}$$

eigVc()

MATH/Matrix menu

eigVc(*kvadratiskMatris***)** ⇒ *matris*

Returnerar egenvektorerna för en reell eller komplex *kvadratiskMatris* där varje kolumn i resultatet motsvarar ett egenvärde. Observera att en egenvektor inte är unik. Den kan skalas av valfri konstant faktor. Egenvektorerna är normaliserade vilket innebär att om $V = [x_1, x_2, \ldots, x_n]$, ges resultatet:

$$\sqrt{x_1{}^2 + x_2{}^2 + \ldots + x_n{}^2} = 1$$

kvadratiskMatris balanseras först med likhetstransformering tills rad- och kolumnnorm är så nära samma värde som möjligt. *kvadratiskMatris* reduceras sedan till formen upper Hessenberg och egenvektorerna beräknas sedan utifrån Hessenberg-matrisen.

I komplext rektangulärt läge:

```
[-1,2,5;3,-6,9;2,-5,7]→m1 ENTER
```

$$\begin{bmatrix} -1 & 2 & 5 \\ 3 & -6 & 9 \\ 2 & -5 & 7 \end{bmatrix}$$

```
eigVc(m1) ENTER
```

$$\begin{bmatrix} -.800\ldots & .767\ldots & .767\ldots \\ .484\ldots & .573\ldots+.052\ldots\cdot i & .573\ldots-.052\ldots\cdot i \\ .352\ldots & .262\ldots+.096\ldots\cdot i & .262\ldots-.096\ldots\cdot i \end{bmatrix}$$

eigVl()

MATH/Matrix menu

eigVl(*kvadratiskMatris***)** ⇒ *lista*

Returnerar en lista med egenvärdena för en reell eller komplex *kvadratiskMatris*

kvadratiskMatris balanseras först med likhetstransformering tills rad- och kolumnnorm är så nära samma värde som möjligt. *kvadratiskMatris* reduceras sedan till formen upper Hessenberg och egenvektorerna beräknas sedan utifrån Hessenberg-matrisen.

I komplext rektangulärt läge:

```
[-1,2,5;3,-6,9;2,-5,7]→m1 ENTER
```

$$\begin{bmatrix} -1 & 2 & 5 \\ 3 & -6 & 9 \\ 2 & -5 & 7 \end{bmatrix}$$

```
eigVl(m1) ENTER
{-4.409...  2.204...+.763...·i
            2.204...-.763...·i}
```

Else

Se **If**, sidan 455.

ElseIf

CATALOG Se även **If**, sidan 455.

If *booleskt uttryck1* **Then**
block1
ElseIf *booleskt uttryck2* **Then**
block2
⋮
ElseIf *booleskt uttryckN* **Then**
blockN
EndIf
⋮

ElseIf kan användas som en programinstruktion för programuppdelning.

Programsegment:

```
  ⋮
:If choice=1 Then
:  Goto option1
:  ElseIf choice=2 Then
:  Goto option2
:  ElseIf choice=3 Then
:  Goto option3
:  ElseIf choice=4 Then
:  Disp "Exiting Program"
:  Return
:EndIf
  ⋮
```

EndCustm

Se **Custom**, sidan 428.

EndDlog

Se **Dialog**, sidan 436.

EndFor

Se **For**, sidan 448.

EndFunc

Se **Func**, sidan 449.

EndIf

Se **If**, sidan 455.

EndLoop

Se **Loop**, sidan 464.

EndPrgm

Se **Prgm**, sidan 478.

EndTBar

Se **ToolBar**, sidan 512.

EndTry

Se **Try**, sidan 512.

EndWhile

Se **While**, sidan 515.

entry()

CATALOG

entry() ⇒ *uttryck*
entry(*heltal***)** ⇒ *uttryck*

Returnerar en föregående inmatning från historiklistan i grundfönstret.

Om *heltal* är inkluderat, anger det vilket uttryck i historiklistan som ska returneras. Standardvärdet är 1, den senast beräknade inmatningen. Giltigt intervall är från 1 till 99 och kan inte vara ett uttryck.

Obs! Om den senaste inmatningen fortfarande är markerad i grundfönstret motsvarar en tryckning på [ENTER] att köra **entry(1)**.

I grundfönstret:

`1+1/x` [ENTER]	$\frac{1}{x}+1$
`1+1/entry(1)` [ENTER]	$2-\frac{1}{x+1}$
[ENTER]	$\frac{1}{2\cdot(2\cdot x+1)}+3/2$
[ENTER]	$5/3-\frac{1}{3\cdot(3\cdot x+2)}$
`entry(4)` [ENTER]	$\frac{1}{x}+1$

exact() Menyn MATH/Number

exact(*uttryck1* [, *tol*]**)** ⇒ *uttryck*
exact(*lista1* [, *tol*]**)** ⇒ *lista*
exact(*matris1* [, *tol*]**)** ⇒ *matris*

Använder läget Exact vid beräkningar, oavsett vilken inställning för Exact/Approx-läget som ska returnera (där det är möjligt) det ekvivalenta rationella talet till argumentet.

tol anger toleransen för konverteringen, standardvärdet är 0 (noll).

```
exact(.25) ENTER                    1/4
exact(.333333) ENTER                333333/1000000
exact(.33333,.001)                  1/3
exact(3.5x+y) ENTER                 7·x/2 + y
exact({.2,.33,4.125}) ENTER
                                    {1/5 33/100 33/8}
```

Exec CATALOG

Exec *sträng* [, *uttryck1*] [, *uttryck2*] **...**

Exekverar en *sträng* innehållande en rad Motorola 68000 op-koder. Dessa koder fungerar liknande ett assembler-program. Om behov finns kan *uttryck* användas för att skicka med en eller flera parametrar till programmet.

Mer information hittar du på TI:s webb-sidor: **http://www.ti.com/calc**

Varning: Med **Exec** får du full tillgång till räknarens microprocessor. Observera att du enkelt kan göra ett misstag som låser räknaren och gör att du förlorar all data. Vi rekommenderar att du gör en säkerhetskopia av räknaren innan du använder kommandot **Exec**.

Exit CATALOG

Exit

Avslutar det aktuella **For**-, **While**- eller **Loop**-blocket.

Exit är inte tillåtet utanför de tre slingstrukturerna (**For**, **While** och **Loop**).

Programlista:

```
:0→temp
:For i,1,100,1
:  temp+i→temp
:  If temp>20
:  Exit
:EndFor
:Disp temp
```

Innehåll i **temp** efter körning: 21

exp▸list() CATALOG

exp▸list(*uttryck*,*var***)** ⇒ *lista*

Undersöker *uttryck* för ekvationer som är avgränsade av ordet "or" och returnerar en lista som innehåller den högra sidan av ekvationerna på formen *var=uttryck*. Detta är ett enkelt sätt att extrahera vissa värden som är inbäddade i resultaten av funktionerna **solve()**, **cSolve()**, **fMin()** och **fMax()**.

Obs! exp▸list() är inte obligatorisk tillsammans med funktionerna **zeros** och **cZeros()** eftersom de returnerar en lista med värden direkt.

```
solve(x^2-x-2=0,x) ENTER    x=2 or
x=-1
exp▸list(solve(x^2-x-2=0,x),x)
ENTER
                            {-1  2}
```

expand() Menyn MATH/Algebra

expand(*uttryck1* [, *var*]**)** ⇒ *uttryck*
expand(*lista1* [,*var*]**)** ⇒ *lista*
expand(*matris1* [,*var*]**)** ⇒ *matris*

expand(*uttryck1***)** returnerar *uttryck1* utvecklat med avseende på alla dess variabler. Utvecklingen är en polynom utveckling för polynom och partiell utveckling av bråk för rationella uttryck.

Målet för **expand()** är att transformera *uttryck1* till en summa och/eller differens av enkla termer. I motsats är målet för **factor()** att transformera *uttryck1* till en produkt och/eller kvot av enkla faktorer.

```
expand((x+y+1)^2) ENTER
```

$$x^2 + 2 \cdot x \cdot y + 2 \cdot x + y^2 + 2 \cdot y + 1$$

```
expand((x^2-x+y^2-y)/(x^2*y^2
  -x^2*y-x*y^2+x*y)) ENTER
```

$$\blacksquare\ \text{expand}\left(\frac{x^2-x+y^2-y}{x^2\cdot y^2-x^2\cdot y-x\cdot y}\right.\blacktriangleright \frac{1}{x-1}-\frac{1}{x}+\frac{1}{y-1}-\frac{1}{y}$$

---

expand(*uttryck1*,*var***)** returnerar *uttryck* utvecklat i förhållande till *var*. Liknande potenser av *var* samlas. Termerna och tillhörande faktorer sorteras med *var* som huvudvariabel. Det kan uppstå viss oavsiktlig faktorering av de samlade koefficienterna. Jämfört med att utesluta *var* sparar detta ofta tid, minne och fönsterutrymme, samtidigt som uttrycket blir mer lättförståeligt.

```
expand((x+y+1)^2,y) ENTER
```

$$y^2 + 2 \cdot y \cdot (x+1) + (x+1)^2$$

```
expand((x+y+1)^2,x) ENTER
```

$$x^2 + 2 \cdot x \cdot (y+1) + (y+1)^2$$

```
expand((x^2-x+y^2-y)/(x^2*y^2
  -x^2*y-x*y^2+x*y),y) ENTER
```

$$\blacksquare\ \text{expand}\left(\frac{x^2-x+y^2-y}{x^2\cdot y^2-x^2\cdot y-x\cdot y}\right.\blacktriangleright \frac{1}{y-1}-\frac{1}{y}+\frac{1}{x\cdot(x-1)}$$

```
expand(ans(1),x) ENTER
```

$$\blacksquare\ \text{expand}\left(\frac{1}{y-1}-\frac{1}{y}+\frac{1}{x\cdot(x-1)}\right.\blacktriangleright \frac{1}{x-1}-\frac{1}{x}+\frac{1}{y\cdot(y-1)}$$

Även när det endast är en variabel kan faktoruppdelningen av nämnaren som används för partiell utveckling av bråk bli mer fullständig när du använder *var*.

```
expand((x^3+x^2-2)/(x^2-2))
ENTER
```

$$\frac{2\cdot x}{x^2-2}+x+1$$

Tips! För rationella uttryck är **propFrac()** ett snabbare men mindre extremt alternativ till **expand()**.

```
expand(ans(1),x) ENTER
```

$$\frac{1}{x-\sqrt{2}}+\frac{1}{x+\sqrt{2}}+x+1$$

Obs! Se även **comDenom()** för en utvecklad täljare över en utvecklad nämnare.

expand(*uttryck1*,[*var*]**)** distribuerar också logaritmer och bråkpotenser oavsett *var*. För ökad distribution av logaritmer och bråkpotenser, kan olikhetsrestriktioner vara nödvändiga för att garantera att vissa faktorer inte är icke-negativa.

```
ln(2x*y)+√(2x*y) ENTER
```

$$\ln(2\cdot x\cdot y)+\sqrt{(2\cdot x\cdot y)}$$

```
expand(ans(1)) ENTER
```

$$\ln(x\cdot y)+\sqrt{2}\cdot\sqrt{(x\cdot y)}+\ln(2)$$

expand(*uttryck1*, [*var*]**)** distribuerar även absoluta värden, **sign()** och exponenter, oavsett *var*.

```
expand(ans(1))|y>=0 ENTER
```

$$\ln(x)+\sqrt{2}\cdot\sqrt{x}\cdot\sqrt{y}+\ln(y)+\ln(2)$$

Obs! Se även **tExpand()** för trigonometriska vinkelsummor och utveckling av multipla vinklar.

```
sign(x*y)+abs(x*y)+e^(2x+y)
ENTER
```

$$e^{2\cdot x+y}+\text{sign}(x\cdot y)+|x\cdot y|$$

```
expand(ans(1)) ENTER
```

$$\text{sign}(x)\cdot\text{sign}(y)+|x|\cdot|y|+(e^x)^2\cdot e^y$$

expr() Menyn MATH/String

expr(*sträng***)** ⇒ *uttryck*

Returnerar teckensträngen i *sträng* som ett uttryck och utför det omedelbart.

```
expr("1+2+x^2+x") ENTER        x²+x+3
expr("expand((1+x)^2)") ENTER
                          x²+2·x+1
"Define cube(x)=x^3"→funcstr
ENTER
              "Define cube(x)=x^3"
expr(funcstr) ENTER            Done
cube(2) ENTER                     8
```

ExpReg Menyn MATH/Statistics/Regressions

ExpReg *lista1*, *lista2* [, [*lista3*] [, *lista4*, *lista5*]]

Beräknar den exponentiella regressionen och uppdaterar alla systemstatistikvariabler.

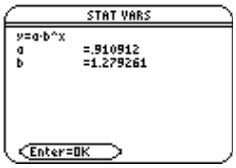
Alla listor måste ha samma dimension förutom *lista5*.

lista1 motsvarar x-listan.
lista2 motsvarar y-listan.
lista3 motsvarar frekvens.
lista4 motsvarar kategorivärden.
lista5 motsvarar de kategorivärden som ska inkluderas.

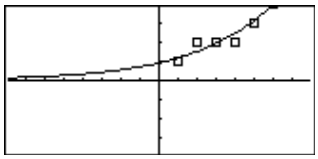
Obs! *lista1* till och med *lista4* måste vara ett variabelnamn eller c1–c99 (kolumner i den senaste datavariabeln som visades i Data/Matrix Editor). *lista5* behöver inte vara ett variabelnamn och kan inte vara någon av c1–c99.

I funktionsplottningsläge:

```
{1,2,3,4,5,6,7,8}→L1 ENTER
                          {1 2 ...}
{1,2,2,2,3,4,5,7}→L2 ENTER
                          {1 2 ...}
ExpReg L1,L2 ENTER            Done
ShowStat ENTER
```




```
ENTER
Regeq(x)→y1(x) ENTER          Done
NewPlot 1,1,L1,L2 ENTER       Done
◆ [GRAPH]
```



factor() Menyn MATH/Algebra

factor(*uttryck1***[, *var*])** ⇒ *uttryck*
factor(*lista1***[,*var*])** ⇒ *lista*
factor(*matris1***[,*var*])** ⇒ *matris*

factor(*uttryck1***)** returnerar *uttryck1* faktoruppdelat med avseende på alla dess variabler över en gemensam nämnare.

uttryck1 är faktoruppdelat så mycket som möjligt mot linjära rationella faktorer utan att introducera nya icke-reella uttryck. Detta alternativ är lämpligt om du vill faktoruppdela med avseende på mer än en variabel.

```
factor(a^3*x^2-a*x^2-a^3+a)
ENTER
  a·(a−1)·(a+1)·(x−1)·(x+1)
factor(x^2+1) ENTER           x²+1
factor(x^2-4) ENTER (x−2)·(x+2)
factor(x^2-3) ENTER           x²−3
factor(x^2-a) ENTER           x²−a
```

factor(*uttryck1*,*var***)** returnerar *uttryck1* faktoruppdelat med avseende på *var*.

uttryck1 faktoruppdelas så mycket som möjligt mot reella faktorer som är linjära i *var*, även om det producerar irrationella konstanter eller uttryck som är irrationella i andra variabler.

Faktorerna och deras termer sorteras med *var* som huvudvariabel. Liknande potenser av *var* samlas i varje faktor. Inkludera *var* om faktoruppdelning behövs med avseende på den variabeln och du är beredd att acceptera irrationella uttryck i andra variabler för att öka faktoruppdelningen med avseende på *var*. Det kan uppstå viss oavsiktlig faktoruppdelning med avseende på andra variabler.

```
factor(a^3*x^2-a*x^2-a^3+a,x)
ENTER
```

$a \cdot (a^2 - 1) \cdot (x - 1) \cdot (x + 1)$

```
factor(x^2-3,x) ENTER
```

$(x + \sqrt{3}) \cdot (x - \sqrt{3})$

```
factor(x^2-a,x) ENTER
```

$(x + \sqrt{a}) \cdot (x - \sqrt{a})a)$

Om du med AUTO-inställningen i Exact/Approx-läget inkluderar *var* tillåts approximering med decimaltalskoefficienter där irrationella koefficienter inte explicit kan uttryckas koncist i termer av de inbyggda funktionerna. Även om det bara finns en variabel kan du genom att inkludera *var* få en mer fullständig faktoruppdelning.

Obs! Se även **comDenom()** för ett snabbt sätt att göra partiell faktoruppdelning när **factor()** inte är tillräckligt snabbt eller om det använder för mycket minne.

Obs! Se även **cFactor()** för faktoruppdelning ända till komplexa koefficienter för att få linjära faktorer.

```
factor(x^5+4x^4+5x^3-6x-3)
ENTER
```

$x^5 + 4 \cdot x^4 + 5 \cdot x^3 - 6 \cdot x - 3$

```
factor(ans(1),x) ENTER
```

$(x - .964\ldots) \cdot (x + .611\ldots) \cdot (x + 2.125\ldots) \cdot (x^2 + 2.227\ldots \cdot x + 2.392\ldots)$

factor(*rationelltTal***)** returnerar det rationella talet faktoruppdelat till primtal. För sammansatta tal ökar beräkningstiden exponentiellt med antalet siffror i den näst högsta faktorn. Att faktoruppdela ett heltal med 30 siffror kan ta mer än en dag och att faktoruppdela ett tal med 100 siffror kan ta mer än ett århundrande.

Obs! Om du vill avbryta en beräkning trycker du ON.

Om du bara vill bestäma om *rationelltal* är ett primtal använder du **isPrime()** i stället för **factor()**. Denna funktion är betydligt snabbare, speciellt då *tal* inte är ett primtal och har en näst högsta faktor med mer än fem siffror.

```
factor(152417172689) ENTER
```

$123457 \cdot 1234577$

```
isPrime(152417172689) ENTER false
```

Fill

Menyn MATH/Matrix

Fill *uttryck*, *matrisvar* ⇒ *matris*

Substituerar varje element i *matrisvar* med *uttryck*.

matrisvariabel måste redan finnas.

```
[1,2;3,4]→amatrx ENTER
```

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

```
Fill 1.01,amatrx ENTER
```

Done

```
amatrx ENTER
```

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

Fill *uttryck*, *listvar* ⇒ *lista*

Substituerar varje element i *listvar* med *uttryck*.

listvar måste redan finnas.

```
{1,2,3,4,5}→alist ENTER
```

{1 2 3 4 5}

```
Fill 1.01,alist ENTER
```

Done

```
alist ENTER
```

{1.01 1.01 1.01 1.01 1.01}

floor() Menyn MATH/Number

floor(*uttryck***)** ⇒ *heltal*

Returnerar det största heltalet som är ≤ argumentet. Denna funktion är identisk med **int()**.

Argumentet kan vara ett reellt eller komplext tal.

```
floor(⁻2.14) ENTER
                    ⁻3.
```

floor(*lista1***)** ⇒ *lista*
floor(*matris1***)** ⇒ *matris*

Returnerar en lista eller en matris av det största heltalet som är ≤ varje element.

Obs! Se även **ceiling()** och **int()**.

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

fMax() Menyn MATH/Calculus

fMax(*uttryck, var***)** ⇒ *booleskt uttryck*

Returnerar ett booleskt uttryck som anger värden på *var* som maximerar *uttryck* eller hittar dess minsta övre begränsning.

Använd operatorn "|" för att begränsa lösningsintervallet och/eller ange tecken för andra odefinierade variabler.

För inställningen APPROX i Exact/Approx-läget söker **fMax()** upprepningsvis efter ett approximativt lokalt maximum. Detta är ofta snabbare, särskilt om du använder operatorn "|" för att begränsa sökningen till ett relativt litet intervall som innehåller exakt ett lokalt maximum.

Obs! Se även **fMin()** och **max()**.

```
fMax(1-(x-a)^2-(x-b)^2,x) ENTER
                    x = (a+b)/2
fMax(.5x^3-x-2,x) ENTER
                    x = ∞
fMax(.5x^3-x-2,x)|x≤1 ENTER
                    x = ⁻.816...
fMax(a*x^2,x) ENTER
  x = ∞ or x = ⁻∞ or x = 0 or a = 0
fMax(a*x^2,x)|a<0 ENTER
                    x = 0
```

fMin() Menyn MATH/Calculus

fMin(*uttryck, var***)** ⇒ *booleskt uttryck*

Returnerar ett booleskt uttryck som anger värden på *var* som minimerar *uttryck* eller hittar den största undre begränsningen.

Använd operatorn "|" för att begränsa lösningsintervallet och/eller ange tecken för andra odefinierade variabler.

För APPROX-inställningen i Exact/Approx-läget söker **fMin()** upprepningsvis efter ett approximativt lokalt minimum. Detta är ofta snabbare, särskilt om du använder operatorn "|" för att begränsa sökningen till ett relativt litet intervall som innehåller exakt ett lokalt minimum.

Obs! Se även **fMax()** och **min()**.

```
fMin(1-(x-a)^2-(x-b)^2,x) ENTER
                    x = ∞ or x = ⁻∞
fMin(.5x^3-x-2,x)|x≥1 ENTER
                    x = 1
fMin(a*x^2,x) ENTER
  x = ∞ or x = ⁻∞ or x = 0 or a = 0
fMin(a*x^2,x)|a>0 and x>1 ENTER
                    x = 1.
fMin(a*x^2,x)|a>0 ENTER
                    x = 0
```

FnOff CATALOG

FnOff

Avmarkerar alla Y=-funktioner för det aktuella grafläget.

I delat fönster, tvågrafsläge, gäller **FnOff** endast den aktiva grafen.

FnOff [1] [, 2] ... [,99]

Avmarkerar de angivna Y=-funktionerna för det aktuella grafläget.

I funktionsplottningsläge:
FnOff 1,3 [ENTER] avmarkerar y1(x) och y3(x).

I parametriskt grafläge:
FnOff 1,3 [ENTER] avmarkerar xt1(t), yt1(t), xt3(t) och yt3(t).

FnOn CATALOG

FnOn

Markerar alla Y=-funktioner som är definierade för det aktuella grafläget.

I delat fönster, tvågrafsläge, gäller **FnOn** endast den aktiva grafen.

FnOn [1] [, 2] ... [,99]

Avmarkerar de angivna Y=-funktionerna för det aktuella grafläget.

Obs! I 3D-grafläge kan endast en funktion i taget markeras. FnOn 2 markerar z2(x,y) och avmarkerar tidigare markerade funktioner. I andra graflägen påverkas inte tidigare markerade funktioner.

For CATALOG

For *var*, *start*, *slut*[, *steg*]
block
EndFor

Kör programsatserna i *block* upprepningsvis för varje värde på *var*, från *start* till *slut* i steg om *steg*.

var kan inte vara en systemvariabel.

Standardvärdet för *steg* är 1.

block kan antingen vara en enskild programsats eller en serie programsatser som avgränsas med tecknet ":".

Programsegment:

```
   ⋮
:0→tempsum : 1→step
:For i,1,100,step
:  tempsum+i→tempsum
:EndFor
:Disp tempsum
   ⋮
```

Innehåll i `tempsum` efter körning: `5050`

Innehåll i `tempsum` när `step` ändras till 2: `2500`

format() Menyn MATH/String

format(*uttryck*[, *formatsträng*]**)** ⇒ *sträng*

Returnerar *uttryck* som en teckensträng baserad på formatmallen.

uttryck måste förenklas till ett tal. *formatsträng* är en sträng som måste vara på formen: "F[*n*]", "S[*n*]", "E[*n*]", "G[*n*][*c*]" där[ ] anger tillvalsdelar.

F[*n*]: Fast decimalinställning. *n* är antalet siffror som visas efter decimalkommat.

S[*n*]: Grundpotensform. *n* är antalet siffror som visas efter decimalkommat.

E[*n*]: Potensform. *n* är antalet siffror som visas efter den första signifikanta siffran. Exponenten justeras till en multipel av tre och decimalkommat flyttas åt höger med noll, en eller två siffror.

G[*n*][*c*]: Samma som fast decimalinställning men avgränsar även siffror till vänster om decimalkommat i grupper om tre. *c* anger tusentalsavgränsaren och är som standard ett kommatecken.

[R*c*]: Alla av de ovan nämnda notationerna kan få ett tillägg med R*c* decimalflagga, där *c* är ett enstaka tecken som anger vad som ska substituera decimalkommat. Decimalkommat är som standard en punkt men ändras automatiskt till ett komma om G-formatet anges.

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

fpart() Menyn MATH/Number

fpart(*uttryck1***)** ⇒ *uttryck*
fpart(*lista1***)** ⇒ *lista*
fpart(*matris1***)** ⇒ *matris*

Returnerar decimaldelen av argumentet.

För en lista eller en matris returneras decimaldelen av varje element.

Argumentet kan vara ett reellt eller ett komplext tal.

```
fpart(-1.234) ENTER          -.234
fpart({1, -2.3, 7.003}) ENTER
                   {0  -.3  .003}
```

Func CATALOG

Func
block
EndFunc

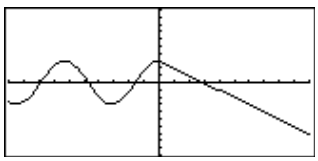
Krävs som den första programsatsen i en funktionsdefinition med flera programsatser.

block kan vara antingen en enstaka programsats eller en serie av programsatser som är avgränsade med tecknet ":".

Obs! when() kan också användas för att definiera och plotta funktioner som är definierade av flera delar.

I funktionsplottningsläge definierar du en funktion i flera delar:

```
Define g(x)=Func:If x<0 Then
  :Return 3*cos(x):Else:Return
  3-x :EndIf:EndFunc ENTER    Done

Graph g(x) ENTER
```



gcd() Menyn MATH/Number

gcd(*tal1*, *tal2***)** ⇒ *uttryck*

Returnerar den största gemensamma delaren till de två argumenten. Den största gemensamma delaren till två bråk är den största gemensamma delaren till deras täljare delat med minsta gemensamma multipel till deras nämnare.

Största gemensamma delare till decimaltal på bråkform är 1,0.

```
gcd(18,33) ENTER                    3
```

gcd(*lista1*, *lista2***)** ⇒ *lista*

Returnerar de största gemensamma delaren för motsvarande element i *lista1* och *lista2*.

```
gcd({12,14,16},{9,7,5}) ENTER
                          {3 7 1}
```

gcd(*matris1*, *matris2***)** ⇒ *matris*

Returnerar de största gemensamma delare för motsvarande element i *matris1* och *matris2*.

```
gcd([2,4;6,8],[4,8;12,16]) ENTER
                          [2 4
                           6 8]
```

Get CATALOG

Get *var*

Hämtar ett CBL™ (Calculator-Based Laboratory™) eller CBR™ (Calculator-Based Ranger™) värde från länkporten och sparar det i variabeln *var*.

Programsegment:

```
  ⋮
:Send {3,1,-1,0}
:For i,1,99
:  Get data[i]
:  PtOn i,data[i]
:EndFor
  ⋮
```

GetCalc CATALOG

GetCalc *var*

Hämtar ett värde från länkporten och sparar det i variabeln *var*. Detta gäller för länkning mellan två räknare.

Obs! Om du vill överföra en variabel från en räknare till en annan använder du 2nd [VAR-LINK] på den andra räknaren för att välja och överföra variabeln, eller använder **SendCalc** på den andra räknaren.

Programsegment:

```
  ⋮
:Disp "Press Enter when ready"
:Pause
:GetCalc L1
:Disp "List L1 received"
  ⋮
```

getConfg() CATALOG

getConfg() ⇒ *Parlista*

Returnerar en lista med information om din räknare. På varje rad står först attributets namn och därefter värdet.

TI-89:

```
getConfg() ENTER
        {"Product Name" "Advanced
            Mathematics Software"
    "Version" "2.00, 09/25/1999"
        "Product ID" "03-1-4-68"
        "ID #" "01012 34567 ABCD"
                "Cert. Rev. #" 0
              "Screen Width" 160
            "Screen Height" 100
              "Window Width" 160
              "Window Height" 67
               "RAM Size" 262132
               "Free RAM" 197178
           "Archive Size" 655360
         "Free Archive" 655340}
```

TI-92 Plus:

```
getConfg() ENTER
        {"Product Name" "Advanced
            Mathematics Software"
    "Version" "2.00, 09/25/1999"
        "Product ID" "01-1-4-80"
        "ID #" "01012 34567 ABCD"
                "Cert. Rev. #" 0
              "Screen Width" 240
            "Screen Height" 120
              "Window Width" 240
              "Window Height" 91
               "RAM Size" 262144
               "Free RAM" 192988
           "Archive Size" 720896
         "Free Archive" 720874}
```

Obs! Din skärm kan se annorlunda ut. Attributet Cert. Rev. # visas bara om du har köpt och installerat extra programvara till räknaren.

getDenom() Menyn MATH/Algebra/Extract

getDenom(*uttryck1***)** ⇒ *uttryck*

Transformerar *uttryck1* till ett uttryck med en förenklad gemensam nämnare och returnerar sedan nämnaren.

```
getDenom((x+2)/(y-3)) ENTER   y-3

getDenom(2/7) ENTER            7

getDenom(1/x+(y^2+y)/y^2) ENTER
                              x·y
```

getFold() CATALOG

getFold() ⇒ *namnsträng*

Returnerar namnet på den aktuella mappen som en sträng.

```
getFold() ENTER              "main"

getFold()→oldfoldr ENTER     "main"

oldfoldr ENTER               "main"
```

getKey() CATALOG

getKey() ⇒ *heltal*

Returnerar teckenkoden för tangenten som trycks ned. Returnerar 0 om ingen tangent trycks ned.

Prefixtangenterna (skift [↑], andrafunktion [2nd], alternativ [◆], alfa [alpha], och dra [⊙]) känns inte enbart men de ändrar teckenkoden för den tangent som följer efter dem.
Exempel: [◆] [X] ≠ [X] ≠ [2nd] [X].

I appendix B finns en fullständig lista över teckenkoder

Programlista:

```
:Disp
:Loop
:  getKey()→key
:  while key=0
:    getKey()→key
:  EndWhile
:  Disp key
:  If key = ord("a")
:  Stop
:EndLoop
```

getMode() CATALOG

getMode(*lägesnamnssträng***)** ⇒ *sträng*
getMode("ALL") ⇒ *stränglistspar*

Om argumentet är ett specifikt lägesnamn returneras en sträng som innehåller den aktuella inställningen för det läget.

Om argumentet är **"ALL"** returneras en lista med strängpar som innehåller inställningarna för alla lägen. Om du vill återställa lägesinställningarna senare måste du spara resultatet **getMode("ALL")** i en variabel och sedan använda **setMode** för att återställa lägena.

Under **setMode** finns en lista över lägesnamn och tillgängliga inställningar.

Obs! Om du vill ställa in eller läsa information lagrad i Systemenheter använder du **setUnits()** eller **getUnits()** i stället för **setMode()** eller **getMode()**.

```
getMode("angle") [ENTER]   "RADIAN"

getMode("graph") [ENTER] "FUNCTION"

getMode("all") [ENTER]

{"Graph" "FUNCTION"
"Display Digits" "FLOAT 6"
"Angle" "RADIAN"
"Exponential Format" "NORMAL"
"Complex Format" "REAL"
"Vector Format" "RECTANGULAR"
"Pretty Print" "ON"
"Split Screen" "FULL"
"Split 1 App" "Home"
"Split 2 App" "Graph"
"Number of Graphs" "1"
"Graph 2" "FUNCTION"
"Split Screen Ratio"  "1,1"
"Exact/Approx" "AUTO"
"Base" "DEC"}
```

Obs! I ditt fönster kan det visas andra lägesinställningar.

getNum() Menyn MATH/Algebra/Extract

getNum(*uttryck1***)** ⇒ *uttryck*

Transformerar *uttryck1* till ett uttryck med en förenklad gemensam nämnare och returnerar sedan täljaren.

```
getNum((x+2)/(y-3)) [ENTER]    x + 2

getNum(2/7) [ENTER]                2

getNum(1/x+1/y) [ENTER]        x + y
```

getType() CATALOG

getType(*var***)** ⇒ *sträng*

Returnerar en sträng som anger vilken datatyp som *var* är av.

Om *var* inte är definierad, returneras strängen "NONE".

```
{1,2,3}→temp [ENTER]        {1 2 3}
getType(temp) [ENTER]       "LIST"

2+3i→temp [ENTER]             2 + 3i
getType(temp) [ENTER]       "EXPR"

DelVar temp [ENTER]           Done
getType(temp) [ENTER]       "NONE"
```

Datatyp	Innehåll i variabeln
"ASM"	Assemblerprogram
"DATA"	Datatyp
"EXPR"	Uttryck (inkluderar komplexa/godtyckliga/odefinierade, ∞, ¯∞, TRUE, FALSE, pi, *e*)
"FUNC"	Funktion
"GDB"	Grafdatabas
"LIST"	Lista
"MAT"	Matris
"NONE"	Variabeln finns inte
"NUM"	Reellt tal
"OTHER"	Odefinierad datatyp för framtida användning av applikationer
"PIC"	Bild
"PRGM"	Program
"STR"	Sträng
"TEXT"	Texttyp
"VAR"	Namn på en annan variabel

getUnits() CATALOG

getUnits() ⇒ *lista*

Returnerar en lista med aktuella enheter för alla kategorier utom konstanter, temperatur, substansmängd, ljusstyrka och acceleration. *lista* har formen:

{"*system*" "*kat1*" "*enhet1*" "*kat2*" "*enhet2*" …}

Första strängen ger systeminställningen (SI, ENG/US eller CUSTOM). Följande par av strängar innehåller en kategori (som Length) följt av dess förvalda enhet (t ex _m för meter).

När du vill ställa in förvalda enheter använder du **setUnits()**.

```
getUnits()ENTER
            {"SI"  "Area"  "NONE"
              "Capacitance"  "_F"
                "Charge"  "_coul"
                              … }
```

Obs! I ditt fönster kan det visas andra förvalda enheter.

Goto CATALOG

Goto *etikettnamn*

Flyttar programkontrollen till etiketten *etikettnamn*.

etikettnamn måste definieras i samma program genom att använda en **Lbl**-instruktion.

Programsegment:

```
  ⋮
:0→temp
:1→i
:Lbl TOP
:  temp+i→temp
:  If i<10 Then
:   i+1→i
:   Goto TOP
:  EndIf
:Disp temp
  ⋮
```

Graph CATALOG

Graph *uttryck1*[,*uttryck 2*] [, *var1*] [, *var2*]

Funktionen Smart Graph plottar de begärda uttrycken/funktionerna i det aktuella grafläget.

Uttryck som har matats in med kommandot **Graph** eller **Table** är tilldelade ökande funktionsnummer med början på 1. De kan ändras eller tas bort en och en med hjälp av redigeringsfunktionerna som är tillgängliga när tabellen visas, efter att du har tryckt på F4 Header. De för tillfället markerade Y=-funktionerna ignoreras.

Om du utelämnar ett valfritt *var*-argument använder **Graph** den oberoende variabeln i det aktuella grafläget.

Obs! Alla valfria argument är inte giltiga i alla lägen, eftersom du aldrig kan ha alla fyra argument på en och samma gång.

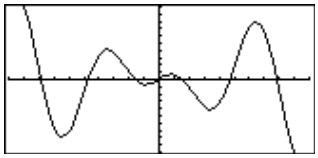
En del giltiga variationer i denna instruktion är följande:

Funktionsplottning	**Graph** *uttryck*, *x*
Parameterplottning	**Graph** *X-uttr*, *Y-uttr*, *t*
Polärplottning	**Graph** *uttryck*, θ
Talföljdsplottning	Inte tillåten.
3D-plottning	**Graph** *uttryck*, *x*, *y*
Plottning av lösningar till differential-ekvationer är	Inte tillåten.

Obs! Använd **ClrGraph** för att ta bort dessa funktioner, eller gå till Y= Editor för att återigen aktivera system-Y=-funktionerna.

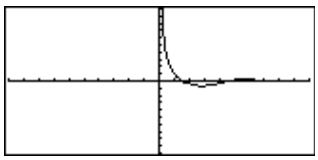
I funktionsplottningsläge och fönstret ZoomStd:

```
Graph 1.25a*cos(a),a ENTER
```



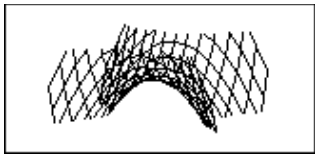
I parametergrafläge och fönstret ZoomStd:

```
Graph time,2cos(time)/time,time
ENTER
```



I 3D-grafläge:

```
Graph (v^2 - w^2)/4,v,w ENTER
```



▸Hex MATH/Base menu

heltal1 **▸Hex** ⇒ *heltal*

Konverterar *heltal1* till ett hexadecimalt värde. Binära och hexadecimala värden har alltid prefixet 0b respektive 0h.

Noll, inte bokstaven O, följt av b alternativt h.

0b *binärt tal*
0h *hexadecimalt tal*

Ett binärt tal kan bestå av upp till 32 tecken. Ett hexadecimalt tal kan bestå av upp till 8 tecken.

Värden utan prefix behandlas i decimalt läge (bas 10). Resultatet visas alltid i hexadecimalt läge oavsett inmatningsläge.

Om du skriver in ett decimaltal som är för stort för 32 bitar används en symmetrisk modulo-operation för att konvertera talet till gällande begränsningar.

```
256 ▸Hex ENTER                    0h100
0b111100001111 ▸Hex ENTER         0hF0F
```

identity() Menyn MATH/Matrix

identity(*uttryck***)** ⇒ *matris*

Returnerar enhetsmatrisen av ordning *uttryck*.

uttryck måste beräknas till ett positivt heltal.

```
identity(4) ENTER
```

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

If CATALOG

If *booleskt uttryck*
programsats

If *booleskt uttryck* **Then**
block
EndIf

Om *booleskt uttryck* beräknas till sant körs den enda programsatsen *programsats* eller blocket av programsatser *block* innan körningen fortsätter.

Om *booleskt uttryck* beräknas till falskt fortsätter körningen utan att köra programsatsen eller blocket av programsatser.

block kan vara antingen en enstaka programsats eller en serie programsatser som är avgränsade med tecknet ":".

Programsegment:

```
  :
  :
:If x<0
:Disp "x is negative"
  :
```

–eller–

```
  :
:If x<0 Then
:  Disp "x is negative"
:  abs(x)→x
:EndIf
  :
```

If *booleskt uttryck* **Then**
block1
Else
block2
EndIf

Om *booleskt uttryck* beräknas till sant körs *block1* och därefter hoppas *block2* över.

Om *booleskt uttryck* beräknas till falskt, hoppas *block1* över men *block2* körs.

block1 och *block2* kan vara en enstaka programsats.

Programsegment:

```
  :
:If x<0 Then
:  Disp "x is negative"
:    Else
:  Disp "x is positive or zero"
:EndIf
  :
```

If *booleskt uttryck1* **Then**
block1
ElseIf *booleskt uttryck2* **Then**
block2
⋮
ElseIf *booleskt uttryckN* **Then**
blockN
EndIf

Tillåter programuppdelning. Om *booleskt uttryck1* beräknas till sant körs *block1*. Om *booleskt uttryck1* beräknas till falskt beräknas *booleskt uttryck2* osv.

Programsegment:

```
  :
:If choice=1 Then
:  Goto option1
:  ElseIf choice=2 Then
:    Goto option2
:  ElseIf choice=3 Then
:    Goto option3
:  ElseIf choice=4 Then
:    Disp "Exiting Program
:    Return
:EndIf
  :
```

imag() Menyn MATH/Complex

imag(*uttryck***)** ⇒ *uttryck*

imag(*uttryck1***)** returnerar imaginärdelen av argumentet.

Obs! Alla odefinierade variabler behandlas som reella variabler. Se även **real()**.

```
imag(1+2i) ENTER        2
imag(z) ENTER           0
imag(x+iy) ENTER        y
```

imag(*lista1***)** ⇒ *lista*

Returnerar en lista med imaginärdelarna av elementen.

```
imag({-3,4-i,i}) ENTER      {0 -1 1}
```

imag(*matris1***)** ⇒ *matris*

Returnerar en matris av imaginärdelarna av elementen.

```
imag([a,b;ic,id]) ENTER      [0 0; c d]
```

Input CATALOG

Input

Gör paus i programmet, visar det aktuella graffönstret och låter dig uppdatera variablerna *xc* och *yc* (även *rc* och θ*c* för polärkoordinatläge) genom att placera grafmarkören.

När du trycker på ENTER fortsätter programmet.

Programsegment:

```
  ⋮
:◉Get 10 points from the Graph
  Screen
:For i,1,10
:  Input
:  xc→XLIST[i]
:  yc→YLIST[i]
:EndFor
  ⋮
```

Input [*frågesträng*,] *var*

Input [*frågesträng*], *var* gör paus i programmet, visar *frågesträng* i Program I/O-fönstret, väntar tills du matar in ett uttryck och sparar uttrycket i *var*.

Om du utelämnar *frågesträng*, visas "?" som meddelande.

Programsegment:

```
  ⋮
:For i,1,9,1
:  "Enter x" & string(i)→str1
:  Input str1,#(right(str1,2))
:EndFor
  ⋮
```

InputStr CATALOG

InputStr [*frågesträng*,] *var*

Gör paus i programmet, visar *frågesträng* i Program I/O-fönstret, väntar tills du matar in ett svar och sparar svaret som en sträng i variabeln *var*.

Om du utelämnar *frågesträng*, visas "?" som meddelande.

Obs! Skillnaden mellan **Input** och **InputStr** är att **InputStr** alltid sparar resultatet som en sträng så att " " inte är nödvändiga.

Programsegment:

```
  ⋮
:InputStr "Enter Your Name",str1
  ⋮
```

inString() Menyn MATH/String

inString(*källsträng***,** *sträng***[,** *start***])** ⇒ *heltal*

Returnerar teckenpositionen i strängen *källsträng* där den första förekomsten av *sträng* börjar.

Om *start* är inkluderad, anger den tecken-positionen inuti *källsträng* där sökningen ska börja. Standard = 1 (det första tecknet i *källsträng*).

Om *källsträng* inte innehåller *sträng* eller *start* är > längden på *källsträng*, returneras noll.

```
inString("Hello there","the")
ENTER                          7
"ABCEFG"→s1:If inString(s1,
"D")=0:Disp "D not found."ENTER
                     D not found.
```

int() CATALOG

int(*uttryck***)** ⇒ *heltal*
int(*lista1***)** ⇒ *lista*
int(*matris1***)** ⇒ *matris*

Returnerar det största heltalet som är mindre än eller lika stort som argumentet. Denna funktion är identisk med **floor()**.

Argumentet kan vara ett reellt eller komplext tal.

För en lista eller matris returneras det största heltalet för varje element.

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

intDiv() CATALOG

intDiv(*tal1***, *tal2***)** ⇒ *heltal*
intDiv(*lista1***, *lista2***)** ⇒ *lista*
intDiv(*matris1***, *matris2***)** ⇒ *matris*

Returnerar heltalsdelen, med tecken, av argument 1 delat med argument 2.

För listor och matriser returneras heltalsdelen med tecken, av argument 1 delat med argument 2 för varje elementpar.

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

integrate

Se ∫(), sidan 528.

iPart() Menyn MATH/Number

iPart(*tal***)** ⇒ *heltal*
iPart(*lista1***)** ⇒ *lists*
iPart(*matris1***)** ⇒ *matris*

Returnerar heltalsdelen av argumentet.

För listor och matriser returneras heltalsdelen av varje element.

Argumentet kan vara ett reellt eller komplext tal.

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

isPrime() MATH/Test menu

IsPrime(*tal***)** ⇒ *Booleskt konstant uttryck*

Returnerar sant eller falskt på frågan om *tal* är ett heltal ≥ 2 som bara kan divideras jämnt med sig självt och med 1.

Om *tal* är längre än ca 306 siffror och inte innehåller någon faktor ≤ 1021 visar **isPrime(***tal***)** ett felmeddelande.

Om du bara vill bestäma om *tal* är ett primtal använder du **isPrime()** i stället för **factor()**. Denna funktion är betydligt snabbare, speciellt då *tal* inte är ett primtal och har en näst högsta faktor med mer än fem siffror.

```
IsPrime(5) ENTER                  true
IsPrime(6) ENTER                 false
```

Funktion för att finna nästa primtal efter ett angivet värde:

```
Define nextPrim(n)=Func:Loop:
n+1→n:if isPrime(n):return n:
EndLoop:EndFunc ENTER            Done

nextPrim(7) ENTER                  11
```

Item CATALOG

Item *menyalternativsträng*
Item *menyalternativsträng*, *etikett*

Endast giltigt tillsammans med en **Custom...EndCustm**- eller **ToolBar...EndTBar**-programsats. Ställer in ett alternativ på en meny så att du kan klistra in text vid markören (**Custom**) eller hoppa till en etikett (**ToolBar**).

Obs! Hopp till en etikett är inte tillåten inom ett **Custom**-block.

Se exemplet om **Custom**.

Lbl CATALOG

Lbl *etikettnamn*

Definierar en etikett med namnet *etikettnamn* i programmet.

Du kan använda en **Goto** *etikettnamn*-instruktion för att överföra programkontroll till den instruktion som kommer direkt efter etiketten.

etikettnamn måste följa samma konventioner som ett variabelnamn.

Programsegment:

```
  ⋮
:Lbl lbl1
:InputStr "Enter password",
str1
:If str1≠password
:  Goto lbl1
:Disp "Welcome to ..."
  ⋮
```

lcm() Menyn MATH/Number

lcm(*tal1*, *tal2***)** ⇒ *uttryck*
lcm(*lista1*, *lista2***)** ⇒ *lista*
lcm(*matris1*, *matris2***)** ⇒ *matris*

Returnerar minsta gemensamma multipel till de två argumenten. Den minsta gemensamma multipeln till två bråk är minsta gemensamma multipel till deras täljare delat med största gemensamma delare till deras nämnare. Minsta gemensamma multipeln till decimaltal på bråkform är deras produkt.

För två listor eller matriser returneras den minsta gemensamma multipeln av de motsvarande elementen.

```
lcm(6,9) ENTER                    18

lcm({1/3,-14,16},{2/15,7,5})
ENTER
                    {2/3 14 80}
```

left() Menyn MATH/String

left(*källsträng*[, *antal*]**)** ⇒ *sträng*

Returnerar *antal* tecken från vänster i teckensträngen *källsträng*.

Om du utelämnar *antal*, returneras alla tecken i *källsträng*.

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

left(*lista1*[, *antal*]**)** ⇒ *lista*

Returnerar *antal* element från vänster i *lista1*.

Om du utelämnar *antal* returneras alla element i *lista1*.

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

left(*jämförelse***)** ⇒ *uttryck*

Returnerar den vänstra sidan av en ekvation eller olikhet.

```
left(x<3) ENTER                x
```

limit() Menyn MATH/Calculus

limit(*uttryck1*, *var*, *punkt*[, *riktning*]**)** ⇒ *uttryck*
limit(*lista1*, *var*, *punkt*[, *riktning*]**)** ⇒ *lista*
limit(*matris1*, *var*, *punkt*[, *riktning*]**)** ⇒ *matris*

Returnerar det begärda gränsvärdet.

riktning: negativ=från vänster, positiv=från höger, annars=båda. (Om den utelämnas är *riktning* som standard båda.)

```
limit(2x+3,x,5) ENTER                13
limit(1/x,x,0,1) ENTER               ∞
limit(sin(x)/x,x,0) ENTER            1
limit((sin(x+h)-sin(x))/h,h,0)
ENTER
                                     cos(x)
limit((1+1/n)^n,n,∞) ENTER           e
```

Gränsvärden vid positiva ∞ och vid negativa ∞ konverteras alltid till ett ensidigt gränsvärde från den begränsade sidan.

Beroende på omständigheterna returnerar **limit()** sig själv eller undef när det inte kan bestämma ett unikt gränsvärde. Detta betyder inte nödvändigtvis att ett unikt gränsvärde inte finns, undef betyder att resultatet antingen är ett okänt tal med begränsad eller obegränsad storlek eller att hela uppsättningen består av sådana tal.

limit() använder metoder som exempelvis L'Hopitals regel, så det finns unika gränsvärden som inte kan bestämmas. Om *uttryck1* innehåller odefinierade variabler förutom *var*, måste du kanske begränsa dem för att erhålla ett mer koncist resultat.

```
limit(a^x,x,∞) ENTER                 undef
limit(a^x,x,∞)|a>1 ENTER             ∞
limit(a^x,x,∞)|a>0 and a<1
ENTER                                0
```

Gränser kan vara mycket känsliga för avrundningsfel. Så långt det är möjligt bör du undvika APPROX-inställningen i Exact/Approx-läget och approximera talen när du beräknar gränvärden. Annars kommer gränser, som bör vara noll eller ha oändlig storlek, antagligen inte vara det och gränser som ska ha en bestämd storlek skild från noll kanske inte kommer att ha det.

Line CATALOG

Line *xStart*, *yStart*, *xSlut*, *ySlut*[, *ritläge*]

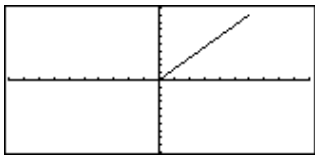
Visar graffönstret och ritar, tar bort eller inverterar ett linjesegment mellan fönster-koordinaterna (*xStart*, *yStart*) och(*xSlut*, *ySlut*), inklusive de båda ändpunkterna.

Om *ritläge* = 1 ritas linjen (standard).
Om *ritläge* = 0 stängs linjen av.
Om *ritläge* = ⁻1 inverteras bildpunkter längs linjen.

Obs! Vid omritning tas alla ritobjekt bort. Se även **PxlLine**.

I fönstret ZoomStd ritar du en linje och tar sedan bort den.

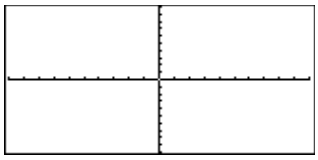
```
Line 0,0,6,9 ENTER
```



TI-89: HOME

TI-92 Plus: ♦ [HOME]

```
Line 0,0,6,9,0 ENTER
```



LineHorz CATALOG

LineHorz *y* [, *ritläge*]

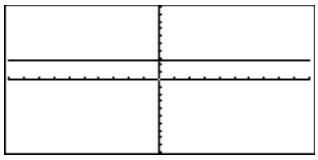
Visar graffönstret och ritar, tar bort eller inverterar en horisontell linje vid fönsterpositionen *y*.

Om *ritläge* = 1 ritas linjen (standard).
Om *ritläge* = 0 stängs linjen av.
Om *ritläge* = ⁻1 inverteras bildpunkter längs linjen.

Obs! Vid omritning tas alla ritobjekt bort. Se även **PxlHorz**.

I fönstret ZoomStd:

```
LineHorz 2.5 ENTER
```



LineTan CATALOG

LineTan *uttryck1*, *uttryck2*

Visar graffönstret och ritar en linje som är tangent till *uttryck1* vid den angivna punkten.

uttryck1 är ett uttryck eller namnet på en funktion, där x förutsätts vara den oberoende variabeln och *uttryck2* är x-värdet av punkten som är tangent.

Obs! I exemplet som visas är *uttryck1* plottat separat. **LineTan** plottar inte *uttryck1*.

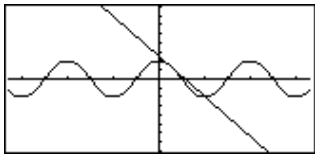
I funktionsplottningsläge och fönstret ZoomTrig:

```
Graph cos(x)
```

TI-89: HOME

TI-92 Plus: ♦ [HOME]

```
LineTan cos(x),π/4 ENTER
```



LineVert CATALOG

LineVert *x* [, *ritläge*]

Visar graffönstret ritar, tar bort eller inverterar en vertikal linje vid fönsterpositionen *x*.

If *ritläge* = 1 ritar linjen (standard).
If *ritläge* = 0 stänger av linjen.
If *ritläge* = ⁻1 inverteras bildpunkter längs linjen.

Obs! Vid omritning tas alla ritobjekt bort. Se även **PxlVert**.

I fönstret ZoomStd:

```
LineVert ⁻2.5 ENTER
```

LinReg Menyn MATH/Statistics/Regressions

LinReg *lista1*, *lista2*[, [*lista3*] [, *lista4*, *lista5*]]

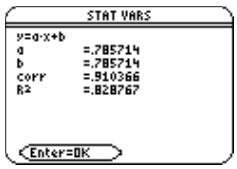
Beräknar regressionslinjen och uppdaterar alla systemstatistikvariabler.

Alla listor måste ha samma dimensioner förutom *lista5*.

lista1 motsvarar x-listan.
lista2 motsvarar y-listan.
lista3 motsvarar frekvens.
lista4 motsvarar kategorivärden.
lista5 motsvarar de kategorivärden som ska inkluderas.

I funktionsplottningsläge:

```
{0,1,2,3,4,5,6}→L1 ENTER
                          {0 1 2 ...}
{0,2,3,4,3,4,6}→L2 ENTER
                          {0 2 3 ...}
LinReg L1,L2 ENTER              Done
ShowStat ENTER
```

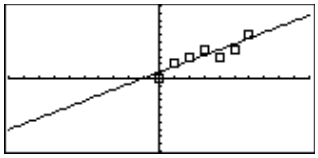


Obs! *lista1* till och med *lista4* måste vara ett variabelnamn eller c1–c99 (kolumner i den senaste datavariabeln som visas i Data/Matrix Editor). *lista5* behöver inte vara ett variabelnamn och kan inte vara någon av c1–c99.

ENTER

```
Regeq(x)→y1(x) ENTER          Done
NewPlot 1,1,L1,L2 ENTER       Done
```

◆ [GRAPH]



list▸mat() Menyn MATH/List

list▸mat(*lista* [, *elementPerRad*]**)** ⇒ *matris*

Returnerar en matris som är fylld rad för rad med elementen från *lista*.

elementPerRad, om det är inkluderat, anger antalet element per rad. Standard är antalet element i *lista* (en rad).

Om *lista* inte fyller den resulterande matrisen fylls den ut med nollor.

```
list▸mat({1,2,3}) ENTER      [1 2 3]
```

```
list▸mat({1,2,3,4,5},2) ENTER
```

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 0 \end{bmatrix}$$

Δlist() MATH/List menu

list(*list1*) ⇒ *list*

Returnerar en lista med differensen mellan efterföljande element i *lista1*. Varje element i *lista1* subtraheras från nästa element i *lista1*. Den nya listan är alltid ett element kortare än den ursprungliga *lista1*.

```
Δlist({20,30,45,70}) ENTER
                    {10,15,25}
```

ln() TI-89: 2nd LN tangenten TI-92 Plus: LN tangenten

ln(*uttryck1***)** ⇒ *uttryck*
ln(*lista1***)** ⇒ *lista*

Returnerar den naturliga logaritmen för argumentet.

För en lista returneras de naturliga logaritmerna för elementen.

```
ln(2.0) ENTER                .693...
```

Om det komplexa formatläget är REAL:

```
ln({⁻3,1.2,5}) ENTER
        Error: Non-real result
```

Om det komplexa formatläget är RECTANGULAR:

```
ln({⁻3,1.2,5}) ENTER
    {ln(3) + π·i   .182...   ln(5)}
```

---

ln(*kvadratiskMatris1***)** ⇒ *kvadratiskMatris*

Returnerar den naturliga matrislogaritmen av matrisen *kvadratiskMatris1*. Detta är *inte* samma sak som att beräkna den naturliga logaritmen för varje element. Mer information om beräkningsmetoden finns i avsnittet om **cos()**.

kvadratiskMatris1 måste vara diagonaliserbar. Resultatet anges alltid med flyttalsaritmetik.

Med vinkelinställningen Radian och komplext rektangulärt format:

```
ln([1,5,3;4,2,1;6,⁻2,1]) ENTER
```

$$\begin{bmatrix} 1.831...+1.734...\cdot i & .009...-1.490...\cdot i & ... \\ .448...-.725...\cdot i & 1.064...+.623\cdot i & ... \\ ^{-}.266...-2.083...\cdot i & 1.124...+1.790...\cdot i & ... \end{bmatrix}$$

LnReg — Menyn H/Statistics/Regressions

LnReg *lista1*, *lista2*[, [*lista3*] [, *lista4*, *lista5*]]

Beräknar den logaritmiska regressionen och uppdaterar alla systemstatistikvariabler.

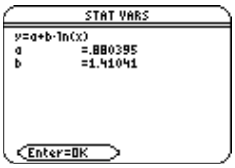
Alla listor måste ha samma dimensioner förutom *lista5*.

lista1 motsvarar x-listan.
lista2 motsvarar y-listan.
lista3 motsvarar frekvens.
lista4 motsvarar kategorivärden.
lista5 motsvarar de kategorivärden som ska inkluderas

Obs! *lista1* till och med *lista4* måste vara ett variabelnamn eller c1–c99 (kolumner i den senaste datavariabeln som visades i Data/Matrix Editor). *lista5* behöver inte vara ett variabelnamn och kan inte vara någon av c1–c99.

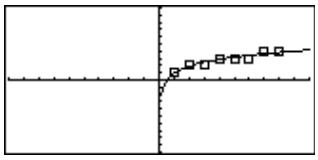
I funktionsplottningsläge:

```
{1,2,3,4,5,6,7,8}→L1 [ENTER]
                          {1 2 3 ...}
{1,2,2,3,3,3,4,4}→L2 [ENTER]
                          {1 2 2 ...}
LnReg L1,L2 [ENTER]              Done
ShowStat [ENTER]
```




```
[ENTER]
Regeq(x)→y1(x) [ENTER]           Done
NewPlot 1,1,L1,L2 [ENTER]        Done
```

[◆] [GRAPH]



Local — CATALOG

Local *var1*[, *var2*] [, *var3*] ...

Deklarerar de angivna *variablerna* som lokala variabler. Dessa variabler existerar bara under beräkningen av ett program eller en funktion och tas bort när programmet eller funktionen är klar med åtgärden.

Obs! Lokala variabler sparar minne eftersom de bara existerar tillfälligt. De stör heller inte befintliga globala variabelvärden. Lokala variabler måste användas för **For**-slingor och för att tillfälligt spara värden i en flerradsfunktion eftersom ändringar i globala variabler inte är tillåtna i en funktion.

Programlista:

```
:prgmname()
:Prgm
:Local x,y
:Input "Enter x",x
:Input "Enter y",y
:Disp x*y
:EndPrgm
```

Obs! x och y finns inte kvar efter att programmet har körts klart.

Lock — CATALOG

Lock *var1*[, *var2*] ...

Låser de angivna värdena. Detta förhindrar att du oavsiktligt tar bort eller ändrar variabeln utan att först låsa upp variabeln.

I exemplet till höger är variabeln L1 låst och kan inte tas bort eller ändras.

Obs! Variablerna kan låsas upp med kommandot **Unlock**.

```
{1,2,3,4}→L1 [ENTER]        {1,2,3,4}
Lock L1 [ENTER]                  Done
DelVar L1 [ENTER]
Error: Variable is locked or
protected
```

log() CATALOG

log(*uttryck1***)** ⇒ *uttryck*
log(*lista1***)** ⇒ *lista*

Returnerar tiologaritmen för argumentet.

```
log(2.0) ENTER                    .301…
```

Om det komplexa formatläget är REAL:

```
log({-3,1.2,5}) ENTER
                Error: Non-real result
```

För en lista returneras tiologaritmen för elementen i listan.

Om det komplexa formatläget är RECTANGULAR:

```
log({-3,1.2,5}) ENTER
```

$$\left\{\frac{\ln(3)}{\ln(10)} + \frac{\pi}{\ln(10)}\cdot i.079\ldots \quad \frac{\ln(5)}{\ln(10)}\right\}$$

log(*kvadratiskMatris1***)** ⇒ *kvadratiskMatris*

Returnerar den 10-baserade matrislogaritmen av matrisen *kvadratiskMatris1*. Detta är *inte* samma sak som att beräkna tiologaritmen för varje element. Mer information om beräknings-metoden finns i avsnittet om **cos()**.

kvadratiskMatris1 måste vara diagonaliserbar. Resultatet anges alltid med flyttalsaritmetik.

Med vinkelinställningen Radian och komplext rektangulärt format:

```
log([1,5,3;4,2,1;6,-2,1]) ENTER
```

$$\begin{bmatrix} .795\ldots+.753\ldots\cdot i & .003\ldots-.647\ldots\cdot i & \ldots \\ .194\ldots-.315\ldots\cdot i & .462\ldots+.270\cdot i & \ldots \\ -.115\ldots-.904\ldots\cdot i & .488\ldots+.777\ldots\cdot i & \ldots \end{bmatrix}$$

Logistic MATH/Statistics/Regressions menu

Logistic *lista1*, *lista2* [, [*iterationer*], [*lista3*] [, *lista4*, *lista5*]]

Beräknar den logistiska regressionen och uppdaterar systemets alla statistikvariabler.

Alla listor måste ha samma dimensioner förutom *lista5*.

lista1 representerar xlist.
lista2 representerar ylist.
lista3 representerar frekvens.
lista4 representerar kategorikoder.
lista5 representerar lista med kategoriinnehåll.

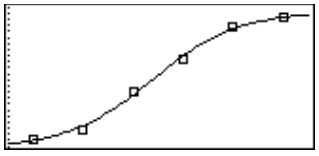
iterationer anger maximalt antal gånger en lösning prövas. Om parametern utelämnas används 64. Normalt ger högre värde exaktare svar men längre beräkningstid och vice versa.

Obs! *lista1* till och med *lista4* måste vara ett variabelnamn eller c1–c99 (kolumner i den sista datavariabeln som visas i Data/Matrix Editor). *lista5* behöver inte vara ett variabelnamn och kan inte vara c1–c99.

I funktionsplottningsläge:

```
{1,2,3,4,5,6}→L1 ENTER {1 2 3 …}
{1,1.3,2.5,3.5,4.5,4.8}→L2
ENTER
                        {1 1.3 2.5 …}
Logistic L1,L2 ENTER          Done
ShowStat ENTER
```

```
ENTER
regeq(x)→y1(x) ENTER          Done
NewPlot 1,1,L1,L2 ENTER       Done
◆ [GRAPH]
F2 9
```



Loop — CATALOG

Loop
block
EndLoop

Kör upprepningsvis programsatserna i *block*. Lägg märke till att slingan kommer att köras oändligt, om inte instruktionen **Goto** eller **Exit** inte utförs inom *block*.

block är en serie programsatser som är avgränsade med tecknet ":".

Programsegment:

```
  :
:1→i
:Loop
:  Rand(6)→die1
:  Rand(6)→die2
:  If die1=6 and die2=6
:    Goto End
:  i+1→i
:EndLoop
:Lbl End
:Disp "The number of rolls is", i
  :
```

LU — MATH/Matrix menu

LU *matris*, *lMatNamn*, *uMatNamn*, *pMatNamn*[, *tol*]

Beräknar Doolittle LU (lower-upper) sönderfallet hos en reell eller komplex *matris*. Den undre triangulära matrisen lagras i *lMatNamn*, den övre triangulära matrisen i *uMatNamn* och permuteringsmatrisen (som beskriver de radbyten som görs under beräkningen) lagras i *pMatNamn*.

lMatNamn ∗ *uMatNamn* = *pMatNamn* ∗ *matris*

Varje element i en matris betraktas som noll om dess absolutvärde är mindre än *tol*. Denna tolerans används endast om matrisen innehåller decimaltal och inte innehåller symboliska variabler som saknar värde. I andra fall ignoreras *tol*.

- Om du använder ◆ ENTER eller anger läget Exact/Approx=APPROXIMATE utförs beräkningar med decimaltal.
- Om *tol* utesluts eller inte används beräknas standardtoleransen enligt:

  5E⁻14 ∗ **max(dim(***matris***))** ∗ **rowNorm(***matris***)**

Faktoriseringsalgoritmen **LU** använder partiell flyttning med radutbyten.

```
[6,12,18;5,14,31;3,8,18]→m1 ENTER
```

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

```
LU m1,lower,upper,perm ENTER Done
```

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

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

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

[m,n;o,p]→m1 ENTER
$$\begin{bmatrix} m & n \\ o & p \end{bmatrix}$$

```
LU m1,lower,upper,perm ENTER Done
```

lower ENTER
$$\begin{bmatrix} 1 & 0 \\ \frac{m}{o} & 1 \end{bmatrix}$$

upper ENTER
$$\begin{bmatrix} o & p \\ 0 & n - \frac{m \cdot p}{o} \end{bmatrix}$$

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

mat▸list() — Menyn MATH/List

mat▸list(*matris***)** ⇒ *lista*

Returnerar en lista fylld med elementen i *matris*. Elementen kopieras från *matris* rad för rad.

```
mat▸list([1,2,3]) ENTER    {1 2 3}
[1,2,3;4,5,6]→M1 ENTER
```

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

```
mat▸list(M1) ENTER   {1 2 3 4 5 6}
```

max() Menyn MATH/List

max(*uttryck 1*, *uttryck 2***)** ⇒ *uttryck*
max(*lista1*, *lista2***)** ⇒ *lista*
max(*matris1*, *matris2***)** ⇒ *matris*

Returnerar maximum av de två argumenten. Om argumenten är två listor eller matriser, returneras en lista eller matris som innehåller maxvärdet för varje par av motsvarande element.

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

---

max(*lista***)** ⇒ *uttryck*

Returnerar det största element i *lista*.

```
max({0,1,-7,1.3,.5}) ENTER      1.3
```

---

max(*matris1***)** ⇒ *matris*

Returnerar en radvektor som innehåller det största elementet i varje kolumn i *matris1*.

Obs! Se även **fMax()** och **min()**.

```
max([1,-3,7;-4,0,.3]) ENTER
                          [1  0  7]
```

mean() Menyn MATH/Statistics

mean(*lista*[, *frekvlista*]**)** ⇒ *uttryck*

Returnerar medelvärdet av elementen i *lista*.

Varje *frekvlista* element räknar antalet efter varandra följade förekomster av motsvarande element i *lista*.

```
mean({.2,0,1,-.3,.4}) ENTER     .26
mean({1,2,3},{3,2,1}) ENTER     5/3
```

---

mean(*matris1*[, *frekvmatris*]**)** ⇒ *matris*

Returnerar en radvektor med medelvärdet hos alla kolumner i *matris1*.

Varje *frekvmatris* element räknar antalet efter varandra följade förekomster av motsvarande element i *matris1*.

Om vektorformatläget är RECTANGULAR:

```
mean([.2,0;-1,3;.4,-.5]) ENTER
                  [-.133...  .833...]
mean([1/5,0;-1,3;2/5,-1/2])
ENTER
                      [-2/15   5/6]
mean([1,2;3,4;5,6],[5,3;4,1;
6,2]) ENTER          [47/15, 11/3]
```

median() Menyn MATH/Statistics

median(*lista***)** ⇒ *uttryck*

Returnerar medianen av elementen i *lista1*.

```
median({.2,0,1,-.3,.4}) ENTER   .2
```

---

median(*matris1***)** ⇒ *matris*

Returnerar en radvektor som innehåller medianerna för kolumnerna i *matris1*.

Obs! Alla inmatningar i listan eller matrisen måste kunna förenklas till tal.

```
median([.2,0;1,-.3;.4,-.5])
ENTER
                          [.4  -.3]
```

MedMed — Menyn MATH/Statistics/Regressions

MedMed *lista1*, *lista2*[, [*lista3*] [, *lista4*, *lista5*]]

Beräknar median-median linjen och uppdaterar alla systemstatistikvariabler.

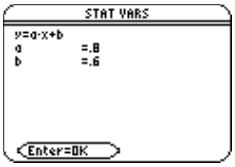
Alla listor måste ha samma dimensioner förutom *lista5*.

lista1 motsvarar x-listan.
lista2 motsvarar y-listan.
lista3 motsvarar frekvens.
lista4 motsvarar kategorivärden.
lista5 motsvarar de kategorivärden som ska inkluderas.

Obs! *lista1* till och med *lista4* måste vara ett variabelnamn eller c1–c99 (kolumner i den senaste datavariabeln som visades i Data/Matrix Editor). *lista5* behöver inte vara ett variabelnamn och kan inte vara någon av c1–c99.

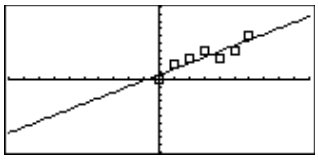
I funktionsplottningsläge:

```
{0,1,2,3,4,5,6}→L1 [ENTER] {0 1 2 ...}
{0,2,3,4,3,4,6}→L2 [ENTER] {0 2 3 ...}
MedMed L1,L2 [ENTER]                Done
ShowStat [ENTER]
```




```
[ENTER]
Regeq(x)→y1(x) [ENTER]              Done
NewPlot 1,1,L1,L2 [ENTER]           Done
```

[◆] [GRAPH]



mid() — Menyn MATH/String

mid(*källsträng*, *start*[,*antal*]**)** ⇒ *sträng*

Returnerar *antal* tecken från teckensträngen *källsträng*, med början på teckennummer *start*.

Om du utelämnar *antal* eller det är större än längden på *källsträng*, returneras alla tecken från *källsträng*, med början på teckennummer *start*.

antal måste vara ≥ 0. Om *antal* = 0, returneras en tom sträng.

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

---

mid(*källista*, *start* [, *antal*]**)** ⇒ *lista*

Returnerar *antal* element från *källistan*, med början på elementnummer *start*.

Om du utelämnar *antal* eller det är större än dimensionen för *källistan*, returneras alla element från *källistan*, med början på element-nummer *start*.

antal måste vara ≥ 0. Om *antal* = 0, returneras en tom lista.

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

---

mid(*källsträngslista*, *start*[, *antal*]**)** ⇒ *lista*

Returnerar *antal* strängar från *källsträngslista* med början på elementnummer *start*.

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

min() Menyn MATH/List

min(*uttryck1*, *uttryck2***)** ⇒ *uttryck*
min(*lista1*, *lista2***)** ⇒ *lista*
min(*matris1*, *matris2***)** ⇒ *matris*

Returnerar minimum av de två argumenten. Om argumenten är två listor eller matriser returneras en lista eller matris som innehåller minimivärdet för varje par av motsvarande element.

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

min(*lista***)** ⇒ *uttryck*

Returnerar det minsta element i *lista*.

```
min({0,1,-7,1.3,.5}) ENTER            -7
```

min(*matris1***)** ⇒ *matris*

Returnerar en radvektor som innehåller det minsta elementet i varje kolumn i *matris1*.

Obs! Se även **fMin()** och **max()**.

```
min([1,-3,7;-4,0,.3]) ENTER
                        [-4 -3 .3]
```

mod() Menyn MATH/Number

mod(*uttryck1*, *uttryck2***)** ⇒ *uttryck*
mod(*lista1*, *lista2***)** ⇒ *lista*
mod(*matris1*, *matris2***)** ⇒ *matris*

Returnerar resten av det första argumentet dividerat med det andra argumentet enligt definitionen:

mod(x,0) ≡ x
mod(x,y) ≡ x−y floor(x/y)

När det andra argumentet är något annat än noll är resultatet periodiskt i det argumentet. Resultatet är antingen noll eller har samma tecken som det andra argumentet.

Om argumentet är två listor eller två matriser returneras en lista eller matris som innehåller funktionen applicerad på varje par av motsvarande element.

Obs! Se även **remain()**.

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

MoveVar CATALOG

MoveVar *var*, *gammalMapp*, *nyMapp*

Flyttar *var* från *gammalMapp* till *nyMapp*. Om *nyMapp* inte finns skapar **MoveVar** den.

```
{1,2,3,4}→L1 ENTER        {1 2 3 4}
MoveVar L1,Main,Games ENTER  Done
```

mRow() Menyn MATH/Matrix/Row ops

mRow(*uttryck*, *matris1*, *index***)** ⇒ *matris*

Returnerar en kopia av *matris1* med varje element i rad *index* av *matris1* multiplicerat med *uttryck*.

```
mRow(-1/3,[1,2;3,4],2) ENTER
                  [ 1    2
                  -1   -4/3 ]
```

mRowAdd() Menyn MATH/Matrix/Row ops

mRowAdd(*uttryck*, *matris1*, *index1*, *index2***)** ⇒ *matris*

Returnerar en kopia av *matris1* med varje element i rad *index2* av *matris1* ersatt med:

uttryck × rad*index1* + rad*index2*

```
mRowAdd(⁻3,[1,2;3,4],1,2) ENTER
```

$$\begin{bmatrix} 1 & 2 \\ 0 & -2 \end{bmatrix}$$

```
mRowAdd(n,[a,b;c,d],1,2) ENTER
```

$$\begin{bmatrix} a & b \\ a\cdot n+c & b\cdot n+d \end{bmatrix}$$

nCr() Menyn MATH/Probability

nCr(*uttryck1*, *uttryck2***)** ⇒ *uttryck*

För heltals *uttryck1* och *uttryck2* med *uttryck1* ≥ *uttryck2* ≥ 0 är **nCr()** antalet kombinationer av *uttryck1* utvalda bland *uttryck2*.
(Detta kallas också en binomialkoefficient.)
Båda argumenten kan vara heltal eller symboliska uttryck.

```
nCr(z,3)
```

$$\frac{z\cdot(z-2)\cdot(z-1)}{6}$$

```
ans(1)|z=5
```

10

```
nCr(z,c)
```

$$\frac{z!}{c!(z-c)!}$$

```
ans(1)/nPr(z,c)
```

$$\frac{1}{c!}$$

nCr(*uttryck*, 0**)** ⇒ 1

nCr(*uttryck*, *negativtHeltal***)** ⇒ 0

nCr(*uttryck*, *positivtHeltal***)** ⇒
uttryck • (*uttryck* −1)...
(*uttryck* − *positivtHeltal* +1)/ *positivtHeltal*!

nCr(*uttryck*, *ickeHeltal***)** ⇒ *uttryck*!/
((*uttryck* − *ickeHeltal*)! • *ickeHeltal*!)

---

nCr(*lista1*, *lista2***)** ⇒ *lista*

Returnerar en lista med kombinationer baserat på motsvarande elementpar i de två listorna. Listorna måste ha samma dimension.

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

---

nCr(*matris1*, *matris2***)** ⇒ *matris*

Returnerar en matris av kombinationer baserat på motsvarande elementpar i de två matriserna. Matriserna måste ha samma dimension.

```
nCr([6,5;4,3],[2,2;2,2]) ENTER
```

$$\begin{bmatrix} 15 & 10 \\ 6 & 3 \end{bmatrix}$$

nDeriv() Menyn MATH/Calculus

nDeriv(*expression1*, *var*[, *h*]**)** ⇒ *expression*
nDeriv(*expression1*, *var*, *list***)** ⇒ *lista*
nDeriv(*lista*, *var*[, *h*]**)** ⇒ *lista*
nDeriv(*matris*, *var*[, *h*]**)** ⇒ *matris*

Returnerar den numeriska derivatan som ett uttryck. Använder den symmetriska differenskvoten.

h är stegvärdet. Om *h* utelämnas är standardvärdet 0,001.

Om deriveringen utförs på en lista eller en matris beräkna den numeriska derivatan på alla element i listan eller matrisen.

Obs! Se även **avgRC()** och ***d*()**.

```
nDeriv(cos(x),x,h) ENTER
```

$$\frac{-(\cos(x-h)-\cos(x+h))}{2\cdot h}$$

```
limit(nDeriv(cos(x),x,h),h,0)
ENTER
```

-sin(x)

```
nDeriv(x^3,x,0.01) ENTER
```

$3.\cdot(x^2+.000033)$

```
nDeriv(cos(x),x)|x=π/2 ENTER
```

-1.

```
nDeriv(x^2,x,{.01,.1}) ENTER
{2.•x  2.•x}
```

NewData CATALOG

NewData *dataVar*, *lista1*[, *lista2*] [, *lista3*]...

Skapar datavariabeln *dataVar*, där kolumnerna visas i ordningsföljd.

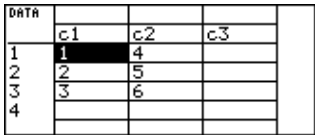
Måste ha minst en lista.

lista1, *lista2*, ..., *listan* kan vara listor som i bilden, uttryck som beräknas till listor eller listvariabelnamn.

NewData gör den nya variabeln aktuell i Data/Matrix Editor.

```
NewData mydata,{1,2,3},{4,5,6}
ENTER
                                Done
```

(Gå till Data/Matrix Editor och öppna *variabeln* `mydata` för att visa datavariabeln nedan.)



NewData *dataVar*, *matris*

Skapar datavariabeln *dataVar* baserad på *matris*.

NewData sysData, *matris*

Laddar innehållet i *matris* till systemdata-variabeln sysData.

NewFold CATALOG

NewFold *mappnamn*

```
NewFold games ENTER             Done
```

Skapar en användardefinierad mapp med namnet *mappnamn* och ställer sedan in den aktuella mappen till den nya mappen.
När du har kört denna instruktion är du i den nya mappen.

newList() CATALOG

newList(*antalElement***)** ⇒ *list*

```
newList(4) ENTER                {0 0 0 0}
```

Returnerar en lista med dimensionen *antalElement*. Varje element är noll.

newMat() CATALOG

newMat(*antalRader*, *antalKolumner***)** ⇒ *matris*

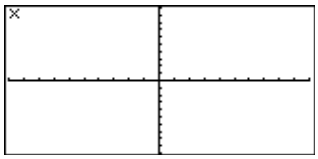
```
newMat(2,3) ENTER               [0 0 0
                                 0 0 0]
```

Returnerar en matris med nollor med dimensionen *antalRader* gånger *antalKolumner*.

NewPic CATALOG

NewPic *matris*, *bildvar* [, *maxRader*][, *maxKolumner*]

```
NewPic [1,1;2,2;3,3;4,4;5,5;
  5,1;4,2;2,4;1,5],xpic ENTER Done

RclPic xpic ENTER
```



Skapar bildvariabeln *bildvar* baserat på *matris*. *matris* måste vara en *n*×*2*-matris där varje rad motsvarar en bildpunkt. Bildpunktskoordinater börjar vid 0,0. Om *bildvar* redan finns ersätter **NewPic** den.

Standardinställningen för *bildvar* är den minsta arean som krävs för matrisvärdena. De valfria argumenten, *maxRader* och *maxKolumner*, bestämmer den maximala storleken på *bildvar*.

NewPlot CATALOG

NewPlot *n*, *typ*, *xLista* [,[*yLista*], [*frkLista*], [*katLista*], [*inklKatLista*], [*m*] [, *stapelbredd*]]

Skapar en ny plottningsdefinition för plottningsnumret *n*.

typ anger typen av grafplottningen.
1 = spridnings-/punktdiagram
2 = xy-linje-diagram
3 = lådagram
4 = histogram
5 = modifierat lådagram

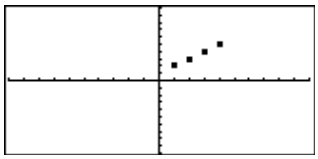
m anger visningstypen.
1 = ▫ (box)
2 = × (cross)
3 = + (plus)
4 = ▪ (square)
5 = • (dot)

stapelbredd är bredden på varje stapel i histogrammet (*typ* = 4) och varierar lite beroende på Window-variablerna xmin och xmax. *stapelbredd* måste vara > 0. Standard = 1.

Obs! *n* kan vara 1–9. Listor måste vara variabelnamn eller c1–c99 (kolumner i den senaste datavariabeln som visades i Data/Matrix Editor), förutom *inklKatLista*, som inte behöver vara ett variabelnamn och kan inte vara någon av c1–c99.

```
FnOff ENTER                          Done
PlotsOff ENTER                       Done
{1,2,3,4}→L1 ENTER          {1 2 3 4}
{2,3,4,5}→L2 ENTER          {2 3 4 5}
NewPlot 1,1,L1,L2,,,,4 ENTER         Done
```

Tryck på ◆ [GRAPH] för att visa:



NewProb CATALOG

NewProb

```
NewProb ENTER                        Done
```

Utför en rad operationer som låter dig starta en ny beräkning från ett nollställt läge utan att radera minnet

- Raderar alla variabelnamn med ett tecken (Clear a–z) i aktuell katalog om inte variabeln är låst eller sparad.
- Stänger av alla funktioner och plottningar (**FnOff** och **PlotsOff**) för aktuellt grafläge.
- Utför **ClrDraw**, **ClrErr**, **ClrGraph**, **ClrHome**, **ClrIO** och **ClrTable**.

nInt() Menyn MATH/Calculus

nInt(*uttryck1*, *var*, *undre*, *övre***)** ⇒ *uttryck*

```
nInt(e^(-x^2),x,-1,1) ENTER
                                 1.493...
```

Om integranden *uttryck1* inte innehåller några variabler förutom *var*, och om *undre* och *övre* är konstanter, positiv ∞ eller negativ ∞, returnerar **nInt()** en approxition av ∫**(***uttryck1*, *var*, *undre*, *övre***)**. Denna approxition är ett vägt medelvärde av vissa värden av integranden i intervallet *undre*<*var*<*övre*.

Målet är sex signifikanta siffror. Algoritmen anpassar sig och avgör när det verkar lämpligt att målet har uppnåtts, eller när det verkar osannolikt att ytterligare exempelvärden orsakar någon förbättring.

En varning visas ("Questionable accuracy") när det verkar som om målet inte har uppnåtts.

```
nInt(cos(x),x,⁻π,π+1E⁻12) ENTER
                          ⁻1.041...E⁻12
∫(cos(x),x,⁻π,π+10^(⁻12)) ENTER
                          ⁻sin(1/1000000000000)
ans(1)◆ ENTER             ⁻1.E⁻12
```

Nästla flera **nInt()** för att göra flera numeriska integrationer. Integrationsgränser kan bero på integrationsvariabler utanför dem.

```
nInt(nInt(e^(⁻x*y)/√(x^2-y^2),
  y,⁻x,x),x,0,1) ENTER      3.304...
```

Obs! Se även **∫()**.

norm() Menyn MATH/Matrix/Norms

norm(*matris***)** ⇒ *uttryck*

Returnerar Frobenius-normen.

```
norm([a,b;c,d]) ENTER
                          √(a²+b²+c²+d²)
norm([1,2;3,4]) ENTER     √30
```

not Menyn MATH/Test

not *booleskt uttryck1* ⇒ *booleskt uttryck*

Returnerar sant, falskt eller ett förenklat *booleskt uttryck1*.

```
not 2>=3                  true
notx <2 ENTER             x ≥ 2
not not innocent ENTER    innocent
```

---

not *heltal1* ⇒ *heltal*

Returnerar komplementvärdet av ett reellt heltal. Internt konverteras *heltal1* till 32 bitar. Värdet av varje bit växlas (0 blir 1 och vice versa). Resultatet visas i aktuellt visningsläge

Du kan skriva in heltalen i valfritt läge. Om du använder binärt eller hexadecimalt läge börjar du talet med 0b alternativt 0h. Värden utan prefix behandlas i decimalt läge (bas 10).

Om du skriver in ett decimalt tal som är för stort för 32 bitar används en symmetrisk modulo-operation för att konvertera talet till gällande begränsningar

I hexadecimalt läge:

```
not 0h7AC36 ENTER         0hFFF853C9
```

Viktigt! Noll, inte bokstaven O.

I binärt läge:

```
0b100101▸dec ENTER        37
not 0b100101 ENTER
0b11111111111111111111111111011010
ans(1)▸dec ENTER          ⁻38
```

Obs: Ett binärt tal kan bestå av upp till 32 tecken (förutom prefixet 0b).
Ett hexadecimalt tal kan bestå av upp till 8 tecken.

Obs: Om du vill skriva in konverterings-operatorn ▸ trycker du [2nd] [▸]. Du kan också välja konverteringar från menyn MATH/Base.

nPr() Menyn MATH/Probability

nPr(*uttryck1*, *uttryck2***)** ⇒ *uttryck*

```
nPr(z,3) ENTER        z·(z-2)·(z-1)
ans(1)|z=5 ENTER                  60
nPr(z,⁻3) ENTER   1/((z+1)·(z+2)·(z+3))
nPr(z,c) ENTER                z!/((z-c)!)
ans(1)*nPr(z-c,⁻c) ENTER           1
```

För heltals *uttryck1* och heltals *uttryck2* med *uttryck1* ≥ *uttryck2* ≥ 0, är **nPr()** antalet permutationer i *uttryck1* utvalda bland *uttryck2*. Båda argumenten kan vara heltal eller symboliska uttryck.

nPr(*uttryck*, 0**)** ⇒ 1

nPr(*uttryck*, *negativtHeltal***)** ⇒ 1/((*uttryck* +1) • (*uttryck*+2)... (*uttryck* – *negativtHeltal*))

nPr(*uttryck*, *positivtHeltal***)** ⇒ *uttryck* • (*uttryck* – 1) ... (*uttryck* – *positivtHeltal* +1)

nPr(*uttryck*, *ickeHeltal***)** ⇒ *uttryck*!/(*uttryck* – *ickeHeltal*)!

---

nPr(*lista1*, *lista2***)** ⇒ *lista*

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

Returnerar en lista med permutationer baserat på motsvarande elementpar i de två listorna. Listorna måste ha samma dimension.

---

nPr(*matris1*, *matris2***)** ⇒ *matris*

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

Returnerar en matris av permutationer baserat på motsvarande elementpar i de två matriserna. Matriserna måste ha samma dimension.

nSolve() Menyn MATH/Algebra

nSolve(*ekvation*, *varEllerGissning***)** ⇒ *tal eller fel_sträng*

```
nSolve(x^2+5x-25=9,x) ENTER
                      3.844...
nSolve(x^2=4,x=⁻1) ENTER   ⁻2.
nSolve(x^2=4,x=1) ENTER     2.
```

Söker iterativt efter en approximativ reell numerisk lösning till *ekvation* för dess enda *var*. Ange *varEllerGissning* till:

variabel
– eller –
variabel = *reellt värde*

Till exempel är x ett giltigt värde liksom x=3.

nSolve() är ofta mycket snabbare än **solve()** eller **zeros()**, särskilt om operatorn "|" används för att begränsa sökningen till ett relativt litet intervall som innehåller en exakt enkel lösning.

Obs! Om det finns flera lösningar kan du använda en gissning för att hitta en särskild lösning.

```
nSolve(x^2+5x-25=9,x)|x<0 ENTER
                      ⁻8.844...
nSolve(((1+r)^24-1)/r=26,r)|r>
0 and r<.25 ENTER
                        .0068...
nSolve(x^2=⁻1,x) ENTER
            "no solution found"
```

nSolve() försöker hitta antingen en punkt där resten är noll eller två relativt nära punkter där resten för de två punkterna har olika tecken och storleken på resten inte är så stor. Om det inte går att åstadkomma med ett fåtal antal testpunkter, returneras strängen "no solution found".

Om du därför ser **nSolve()** i ett program kan du använda **getType()** för att kontrollera ett numeriskt resultat innan du använder resultatet i ett algebraiskt uttryck.

Obs! Se även **cSolve()**, **cZeros()** , **solve()** och **zeros()**.

OneVar — Menyn MATH/Statistics

OneVar *lista1* [[, *lista2*] [, *lista3*] [, *lista4*]]

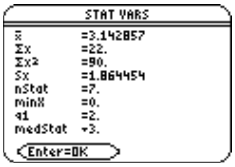
Utför en analys med en variabel med *lista1* och uppdaterar alla systemstatistikvariabler.

Alla listor måste ha samma dimensioner förutom *lista4*.

lista1 motsvarar x-listan.
lista2 motsvarar frekvens.
lista3 motsvarar kategorivärden.
lista4 motsvarar de kategorivärden som ska inkluderas.

Obs! *lista1* till och med *lista3* måste vara variabelnamn eller c1–c99 (kolumner i den senaste datavariabeln som visades i Data/Matrix Editor). *lista4* behöver inte vara ett variabelnamn och kan inte vara någon av c1–c99.

```
{0,2,3,4,3,4,6}→L1 ENTER
OneVar L1 ENTER                    Done
ShowStat ENTER
```



or — Menyn MATH/Test

booleskt uttryck1 **or** *booleskt uttryck2* ⇒ *booleskt uttryck*

Returnerar sant eller falskt eller en förenklad form av den ursprungliga inmatningen.

Returnerar sant om det ena eller båda uttrycken förenklas till sant. Returnerar falskt endast om båda uttrycken beräknas till falskt.

Obs! Se **xor**.

```
x≥3 or x≥4 ENTER                   x ≥ 3
```

Programsegment:

```
  ⋮
If x<0 or x≥5
  Goto END
  ⋮
If choice=1 or choice=2
  Disp "Wrong choice"
  ⋮
```

---

heltal1 **or** *heltal2* ⇒ *heltal*

Jämför två reella heltal bit för bit med operationen **or**. Internt konverteras båda talen till 32 bitar. Vid jämförelsen är resultatet 1 om någon bit är 1 och 0 endast om båda bitarna är 0. Resultatet visas i aktuellt visningsläge.

Du kan skriva in heltalen i valfritt läge. Om du använder binärt eller hexadecimalt läge börjar du talet med 0b alternativt 0h. Värden utan prefix behandlas i decimalt läge (bas 10).

Om du skriver in ett decimalt tal som är för stort för 32 bitar används en symmetrisk modulo-operation för att konvertera talet till gällande begränsningar.

Obs! Se **xor**.

I hexadecimalt läge:

```
0h7AC36 or 0h3D5F ENTER   0h7BD7F
```

Viktigt! Noll, inte bokstaven O.

I binärt läge:

```
0b100101 or 0b100 ENTER 0b100101
```

Obs! Ett binärt tal kan bestå av upp till 32 tecken (förutom prefixet 0b).
Ett hexadecimalt tal kan bestå av upp till 8 tecken.

ord() Menyn MATH/String

ord(*sträng***)** ⇒ *heltal*
ord(*lista1***)** ⇒ *lista*

Returnerar den numeriska koden för det första tecknet i teckensträngen *sträng*, eller en lista över de första tecknen i varje listelement.

I appendix B finns en komplett lista med teckenkoder.

```
ord("hello") ENTER                 104
char(104) ENTER                    "h"
ord(char(24)) ENTER                 24
ord({"alfa","beta"}) ENTER
                              {97  98}
```

Output CATALOG

Output *rad*, *kolumn*, *uttryckEllerSträng*

Visar *uttryckEllerSträng* (ett uttryck eller en teckensträng) i Program I/O-fönstret vid textkoordinaterna (*rad*, *kolumn*).

Ett uttryck kan innehålla konverteringar som **▸DD** och **▸Rect**. Du kan också använda operatorn ▸ om du vill utföra enhets- och lägesomvandlingar.

Om Pretty Print = ON, visas *uttryckEllerSträng* i pretty print-läge.

I Program I/O fönstret kan du trycka på F5 för att visa Home-fönstret, i ett program kan du använda **DispHome**.

Programsegment:

```
⋮
:RandSeed 1147
:ClrIO
:For i,1,90,10
:  Output i, rand(100),"Hello"
:EndFor
   ⋮
```

Resultat efter körning:

```
        Hello
   Hello
           Hello
      Hello
                Hello
    Hello
                  Hello
```

P▸Rx() Menyn MATH/Angle

P▸Rx(*rUttryck***, θ***Uttryck***)** ⇒ *uttryck*
P▸Rx(*rLista***, θ***Lista***)** ⇒ *lista*
P▸Rx(*rMatris***, θ***Matris***)** ⇒ *matris*

Returnerar motsvarande x-koordinat för paret (r, θ).

Obs! Argumentet θ tolkas som en vinkel i grader eller radianer med avseende på den aktuella vinkelenheten. Om argumentet är ett uttryck kan du använda ° eller ʳ för att tillfälligt åsidosätta inställningen för vinkelenhet.

Med vinkelenheten radianer:

```
P▸Rx(r,θ) ENTER                cos(θ)·r
P▸Rx(4,60°) ENTER                     2
P▸Rx({-3,10,1.3},{π/3,-π/4,0})
ENTER
                 {-3/2   5·√2   1.3}
```

P▸Ry() Menyn MATH/Angle

P▸Ry(*rUttryck***, θ***Uttryck***)** ⇒ *uttryck*
P▸Ry(*rLista***, θ***Lista***)** ⇒ *lista*
P▸Ry(*rMatris***, θ***Matris***)** ⇒ *matris*

Returnerar motsvarande y-koordinat för paret (r, θ).

Obs! Argumentet θ tolkas som en vinkel i grader eller radianer enligt den aktuella vinkelinställningen. Om argumentet är ett uttryck kan du använda ° eller ʳ för att tillfälligt åsidosätta inställningen för vinkelenhet.

Med vinkelenheten radianer:

```
P▸Ry(r,θ) ENTER                sin(θ)·r
P▸Ry(4,60°) ENTER              2·√3
P▸Ry({-3,10,1.3},{π/3,-π/4,0})
ENTER
                 {-3·√3/2   -5·√2   0.}
```

part() CATALOG

part(*uttryck1*[**,***ickeNegativtHeltal*]**)**

Denna avancerade programmeringsfunktion låter dig identifiera och ta ut alla sub-uttryck i det förenklade resultatet av *uttryck1*.

Om till exempel *uttryck1* förenklas till cos(π∗ x+3) gäller:

- Funktionen **cos()** har ett argument: (π∗ x+3).
- Summan av (π∗ x+3) har två operander: π∗ x och 3.
- Talet 3 har inga argument eller operander.
- Produkten π∗x har två operander: π och x.
- Variabeln x och den symboliska konstanten π har inga argument eller operander.

Om x har ett numeriskt värde och du trycker på ◆ ENTER så beräknas det numeriska värdet av π∗x, resultatet adderas till 3 och därefter beräknas cosinus. **cos()** är den **översta** operatorn eftersom den används **sist**.

---

part(*uttryck1* **)** ⇒ *tal*

Förenklar *uttryck1* och returnerar antalet övergripande argument eller operander. Returnerar 0 om *uttryck1* är ett tal, en variabel eller en symbolisk konstant som π, *e*, ***i***, or ∞.

```
part(cos(π∗ x+3)) ENTER          1
```

Obs! cos(π∗ x+3) har ett argument.

part(*uttryck1* **, 0)** ⇒ *sträng*

Förenklar *uttryck1* och returnerar en sträng med namnet på den översta funktionen eller operanden. Returnerar **string(***uttryck1***)** om *uttryck1* är ett tal, en variabel eller en symbolisk konstant som π, *e*, ***i***, or ∞.

```
part(cos(π∗ x+3),0) ENTER    "cos"
```

part(*uttryck1***,** *n***)** ⇒ *uttryck*

Förenklar *uttryck1* och returnerar det *n:te* argumentet eller operanden där *n* är > 0 och ≤ det antal övergripande argument eller operander returnerat av **part(***uttryck1***)**. I övriga fall returneras ett felmeddelande.

```
part(cos(π∗ x+3),1) ENTER    3+π•x
```

Obs! Förenklingen ändrade argumentens inbördes ordning.

Genom att kombinera de olika varianterna av **part()** kan du ta ut alla sub-uttryck ur det förenklade resultatet av *uttryck1*. I exemplet till höger visas hur du kan lagra ett argument eller en operand och sedan använda **part()** för att ta ut ytterligare sub-uttryck.

Obs! När du använder **part()**, kan du inte förlita dig på någon inbördes ordning i summor och produkter.

```
part(cos(π∗ x+3)) ENTER          1
part(cos(π∗ x+3),0) ENTER    "cos"
part(cos(π∗ x+3),1)→temp ENTER
                             3+π•x
temp ENTER                   π•x+3
part(temp,0) ENTER             "+"
part(temp) ENTER                 2
part(temp,2) ENTER               3
part(temp,1)→temp ENTER        π•x
part(temp,0) ENTER             "∗"
part(temp) ENTER                 2
part(temp,1) ENTER               π
part(temp,2) ENTER               x
```

Uttryck som (x+y+z) och (x− y− z) representeras internt som (x+y)+z och (x− y)− z. Detta påverkar de värden som returneras för det första och det andra argumentet. Det finns tekniska orsaker till varför **part(**x+y+z,1**)** returnerar y+x i stället för x+y.

```
part(x+y+z) ENTER                2
part(x+y+z,2) ENTER              z
part(x+y+z,1) ENTER            y+x
```

På samma sätt representeras x∗ y∗ z internt som (x∗ y)∗ z. Återigen finns det tekniska skäl till varför det första argumentet returneras som y·x i stället för som x·y.

```
part(x∗y∗z) ENTER                2
part(x∗y∗z,2) ENTER              z
part(x∗y∗z,1) ENTER            y·x
```

När du tar ut sub-uttryck från en matris måste du komma ihåg att matriser lagras som listor av listor som i exemplet till höger.

```
part([a,b,c;x,y,z],0) ENTER   "{"
part([a,b,c;x,y,z]) ENTER        2
part([a,b,c;x,y,z],2)→temp
ENTER
                        {x   y   z}
part(temp,0) ENTER              "{"
part(temp) ENTER                 3
part(temp,3) ENTER               z
delVar temp ENTER             Done
```

Exemplet till höger använder **getType()** och **part()** för att delvis implementera symbolisk differentiering. Genom att studera och slutföra exemplet kan du lära dig mer om manuell differentiering. Du kan till och med inkludera funktioner som din TI-89 / TI-92 Plus inte kan differentiera, som t ex Bessel-funktioner.

```
:d(y,x)
:Func
:Local f
:If getType(y)="VAR"
:  Return when(y=x,1,0,0)
:If part(y)=0
:  Return 0 ◉ y=π,∞,i,numbers
:part(y,0)→f
:If f="-" ◉ if negate
:  Return ⁻d(part(y,1),x)
:If f="−" ◉ if minus
:  Return d(part(y,1),x)
     −d(part(y,2),x)
:If f="+"
:  Return d(part(y,1),x)
     +d(part(y,2),x)
:If f="∗"
:  Return
part(y,1)∗d(part(y,2),x)
     +part(y,2)∗d(part(y,1),x)
:If f="{"
:  Return seq(d(part(y,k),x),
     k,1,part(y))
:Return undef
:EndFunc
```

PassErr CATALOG

PassErr

Se **ClrErr** programlista för exempel.

Överför ett fel till nästa nivå.

Om "errornum" är noll gör **PassErr** ingenting.

Else-satsen i programmet bör använda **ClrErr** eller **PassErr**. Om felet ska rättas till eller ignoreras använder du **ClrErr**. Om du inte vet vad du ska göra med felet använder du **PassErr** för att överföra det till nästa felhanterare. (Se även **ClrErr**.)

Pause CATALOG

Pause [*uttryck*]

Gör paus i programkörningen. Om du inkluderar *uttryck* visas *uttryck* i Program I/O-fönstret.

uttryck kan inkludera omvandlingsoperationer som **▸DD** och **▸Rect**. Du kan också använda operatorn ▸ om du vill utföra enhets- och lägesomvandlingar.

Om resultatet av *uttryck* är för stort för att rymmas på en skärm kan du använda pekaren för att bläddra på skärmen.

Programkörningen återupptas när du trycker på ENTER.

Programsegment:

```
  ⋮
:ClrIO
:DelVar temp
:1→temp[1]
:1→temp[2]
:Disp temp[2]
:◉Guess the Pattern
:For i,3,20
:  temp[i-2]+temp[i-1]→temp[i]
:  Disp temp[i]
:  Disp temp,"Can you guess the
     next","number?"
:  Pause
:EndFor
  ⋮
```

PlotsOff CATALOG

PlotsOff [**1**] [, **2**] [, **3**] **...** [, **9**]

Avmarkerar angivna plottningar. När du är i tvågrafsläge påverkas bara den aktiva grafen.

Om det inte finns några parametrar avmarkeras alla plottningar.

```
PlotsOff 1,2,5 ENTER        Done
PlotsOff ENTER              Done
```

PlotsOn CATALOG

PlotsOn [**1**] [, **2**] [, **3**] **...** [, **9**]

Markerar angivna plottningar. När du är i tvågrafsläge påverkas bara den aktiva grafen.

Om det inte finns några parametrar markeras alla plottningar.

```
PlotsOn 2,4,5 ENTER         Done
PlotsOn ENTER               Done
```

▸Polar Menyn MATH/Matrix/Vector ops

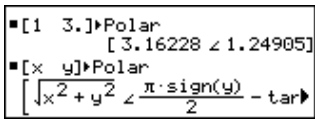
vektor **▸Polar**

Visar *vektor* i polärform [r ∠θ]. Vektor måste vara tvådimensionell och kan vara en rad- eller kolumnvektor.

Obs! ▸Polar är en instruktion för visningsformat, inte en omvandlingsfunktion. Du kan bara använda den i slutet av en inmatningsrad och den uppdaterar inte ans.

Obs! Se även **▸Rect**.

```
[1,3.]▸Polar ENTER
[x,y]▸Polar ENTER
```



komplexVektor **▸Polar**

Visar *komplexVektor* i polärform.

- Med vinkelenheten grader visas (r∠θ).
- Med vinkelenheten radianer visas $re^{i\theta}$.

komplexVektor kan ha valfri komplex form. En inmatning av $re^{i\theta}$ med vinkelenheten grader ger dock ett felmeddelande.

Obs! Du måste använda parenteser runt en (r∠θ) polär inmatning.

Med vinkelenheten radianer:

3+4*i*▸Polar ENTER $e^{i\cdot\left(\frac{\pi}{2}-\tan^{-1}(3/4)\right)}\cdot 5$

(4∠π/3)▸Polar ENTER $e^{\frac{i\cdot\pi}{3}}\cdot 4$

Med vinkelenheten grader:

3+4*i*▸Polar ENTER (5∠90−tan⁻¹(3/4))

polyEval() Menyn MATH/List

polyEval(*lista1*, *uttryck1***)** ⇒ *uttryck*
polyEval(*lista1*, *lista2***)** ⇒ *uttryck*

Tolkar det första argumentet som koefficienterna för ett fallande n:tegrads-polynom och returnerar polynomet som beräknats för värdet av det andra argumentet.

```
polyEval({a,b,c},x) ENTER
                          a·x²+b·x+c
polyEval({1,2,3,4},2) ENTER      26
polyEval({1,2,3,4},{2,⁻7})
ENTER                     {26 ⁻262}
```

PopUp CATALOG

PopUp *alternativlista*, *var*

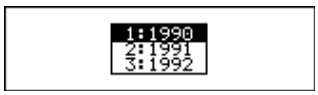
Visar en snabbmeny som innehåller teckensträngarna från *alternativlista*, väntar tills du har valt ett alternativ och sparar numret för alternativet i *var*.

Elementen i *alternativlista* måste vara teckensträngar: {*alternativ1sträng*, *alternativ2sträng*, *alternativ3sträng*, ...}

Om *var* redan finns och har ett giltigt alternativnummer, visas den posten som standardalternativ.

alternativlista måste innehålla minst ett alternativ.

```
PopUp
{"1990","1991","1992"},var1
ENTER
```



PowerReg Menyn MATH/Statistics/Regressions

PowerReg *lista1*, *lista2*[, [*lista3*] [, *lista4*, *lista5*]]

Beräknar potens-regressionen och uppdaterar alla systemstatistikvariabler.

Alla listor måste ha samma dimensioner förutom *lista5*.

lista1 motsvarar x-listan.
lista2 motsvarar y-listan.
lista3 motsvarar frekvens.
lista4 motsvarar kategorivärden.
lista5 motsvarar de kategorivärden som ska inkluderas.

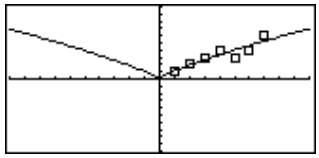
Obs! *lista1* till och med *lista4* måste vara variabelnamn eller c1–c99 (kolumner i den senaste datavariabeln som visades i Data/Matrix Editor). *lista5* behöver inte vara ett variabelnamn och kan inte vara någon av c1–c99.

I funktionsplottningsläge:

```
{1,2,3,4,5,6,7}→L1 ENTER
                          {1 2 3 ...}
{1,2,3,4,3,4,6}→L2 ENTER
                          {1 2 3 ...}
PowerReg L1,L2 ENTER          Done
ShowStat ENTER
```

```
ENTER
Regeq(x)→y1(x) ENTER          Done
NewPlot 1,1,L1,L2 ENTER       Done
```

◆ [GRAPH]



Prgm CATALOG

Prgm
⋮
EndPrgm

Obligatorisk instruktion som identifierar början av ett program. Den sista raden i ett program måste vara **EndPrgm**.

Programsegment:

```
:prgmname()
:Prgm
:
:EndPrgm
```

product() Menyn MATH/List

product(*lista***[**, *start***[**, *slut***]])** ⇒ *uttryck*

Returnerar produkten av elementen som finns i *lista*. *Start* och *slut* är valfria. De anger ett område med element.

```
product({1,2,3,4}) ENTER          24
product({2,x,y}) ENTER        2·x·y
product({4,5,8,9},2,3) ENTER    40
```

---

product(*matris1***[**, *start***[**, *slut***]])** ⇒ *matris*

Returnerar en radvektor som innehåller produkterna av elementen i de kolumner som finns i *matris1*. *Start* och *slut* är valfria. De anger ett område med rader.

```
product([1,2,3;4,5,6;7,8,9])
ENTER                  [28 80 162]
product([1,2,3;4,5,6;7,8,9],
1,2) ENTER               [4,10,18]
```

Prompt CATALOG

Prompt *var1*[,*var2*] [,*var3*] **...**

Visar en prompt i Program I/O-fönstret för varje variabel i argumentlistan. Genom att exempelvis använda prompt var1 visas var1? i I/O-fönstret och det inmatade uttrycket sparas i motsvarande variabel.

Prompt måste ha minst ett argument.

Programsegment:

```
  ⋮
Prompt A,B,C
  ⋮
EndPrgm
```

propFrac() Menyn MATH/Algebra

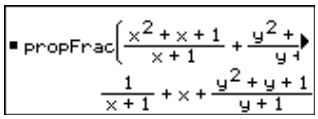
propFrac(*uttryck1***[**, *var***])** ⇒ *uttryck*

propFrac(*rationellt_tal***)** returnerar *rationellt_tal* som summan av ett heltal och ett bråktal med samma tecken och med en större nämnare än täljare.

```
propFrac(4/3) ENTER          1 + 1/3
propFrac(⁻4/3) ENTER        ⁻1−1/3
```

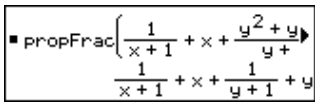
propFrac(*rationellt_uttryck,var***)** returnerar summan av ett egentligt bråk och ett polynom med avseende på *var*. Graden av *var* i nämnaren överskrider graden av *var* i täljaren i det egentliga bråket. Liknande potenser av *var* samlas. Termerna och deras faktorer sorteras med *var* som huvudvariabel.

```
propFrac((x^2+x+1)/(x+1)+
  (y^2+y+1)/(y+1),x) ENTER
```



Om *var* utelämnas görs en egentlig bråkutveckling med avseende på den mest förekommande variabeln. Koefficienterna i den polynoma delen görs sedan egentliga med avseende på deras mest förekommande variabel först osv.

```
propFrac(ans(1))
```



För rationella uttryck är **propFrac()** ett snabbare men mindre extremt alternativ till **expand()**.

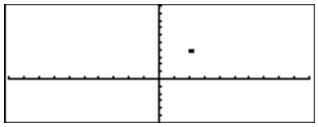
PtChg CATALOG

PtChg x, y
PtChg *xLista*, *yLista*

Visar graffönstret och inverterar bildpunkten som är närmast fönsterkoordinaterna (x, y).

Obs! Under **PtChg** till och med **PtText** visas liknande exempel.

```
PtChg 2,4 ENTER
```

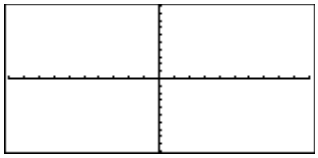


PtOff CATALOG

PtOff x, y
PtOff *xLista*, *yLista*

Visar graffönstret och stänger av bildpunkten som är närmast fönsterkoordinaterna (x, y).

```
PtOff 2,4 ENTER
```

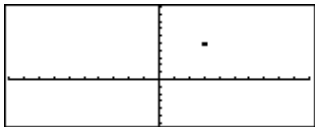


PtOn CATALOG

PtOn x, y
PtOn *xLista*, *yLista*

Visar graffönstret och slår på bildpunkten som är närmast fönsterkoordinaterna (x, y).

```
PtOn 3,5 ENTER
```



ptTest() CATALOG

ptTest (x, y) ⇒ *booleskt konstantuttryck*
ptTest (*xLista*, *yLista*) ⇒ *booleskt konstantuttryck*

Returnerar true eller false. Returnerar true endast om den bildskärmspunkt närmast fönsterkoordinaterna (x, y) är på.

```
ptTest(3,5) ENTER                    true
```

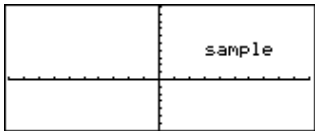
PtText CATALOG

PtText *sträng*, x, y

Visar graffönstret och placerar teckensträngen *sträng* i fönstret vid bildpunkten närmast de angivna fönsterkoordinaterna (x, y).

sträng placeras med det övre vänstra hörnet av det första tecknet vid koordinaterna.

```
PtText "sample",3,5 ENTER
```



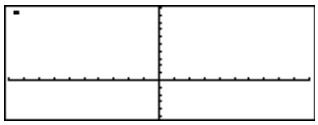
PxlChg CATALOG

PxlChg *rad*, *kolumn*
PxlChg *radlista*, *kolumnlista*

Visar graffönstret och inverterar bildpunkten med koordinaterna (*rad*, *kolumn*).

Obs! Vid omritning tas alla ritobjekt bort.

```
PxlChg 2,4 ENTER
```



PxlCrcl CATALOG

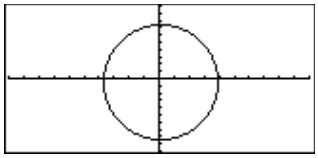
PxlCrcl *rad*, *kolumn*, *r* [, *ritläge*]

Visar graffönstret och ritar en cirkel som är centrerad vid bildpunkterna (*rad*, *kolumn*) med en radie på *r* bildpunkter.

Om *ritläge* = 1 ritar cirkeln (standard).
Om *ritläge* = 0 stängs cirkeln av.
Om *ritläge* = -1 inverteras bildpunkter längs cirkeln.

Obs! Vid omritning tas alla ritobjekt bort.
Se även **Circle**.

TI-89: `PxlCrcl 40,80,30,1` ENTER
TI-92 Plus: `PxlCrcl 50,125,40,1` ENTER



PxlHorz CATALOG

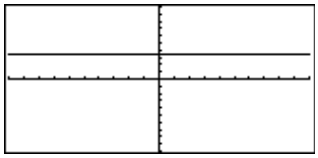
PxlHorz *rad* [, *ritläge*]

Visar graffönstret och ritar en horisontell linje vid fönsterbildpunkt *rad*.

Om *ritläge* = 1, ritas linjen (standard).
Om *ritläge* = 0, stängs linjen av.
Om *ritläge* = -1, inverteras bildpunkter längs linjen.

Obs! Vid omritning tas alla ritobjekt bort.
Se även **LineHorz**.

`PxlHorz 25,1` ENTER



PxlLine CATALOG

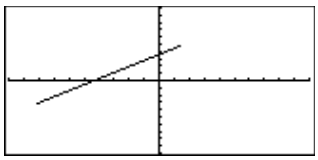
PxlLine *radstart*, *kolumnstart*, *radslut*, *kolumnslut* [, *ritläge*]

Visar graffönstret och ritar en linje mellan bildpunktskoordinater (*radstart*, *kolumnstart*) och (*radslut*, *kolumnslut*), inklusive båda ändpunkterna.

Om *ritläge* = 1, ritas linjen (standard).
Om *ritläge* = 0, stängs linjen av.
Om *ritläge* = -1, inverteras bildpunkter längs linjen.

Obs! Vid omritning tas alla ritobjekt bort.
Se även **Line**

TI-89: `PxlLine 50,15,20,90,1` ENTER
TI-92 Plus: `PxlLine 80,20,30,150,1` ENTER



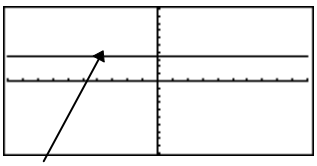
PxlOff CATALOG

PxlOff *rad*, *kolumn*
PxlOff *radlista*, *kolumnlista*

Visar graffönstret och stänger av bildpunkten med koordinaterna (*rad*, *kolumn*).

Obs! Vid omritning tas alla ritobjekt bort.

`PxlHorz 25,1` ENTER
`PxlOff 25,50` ENTER



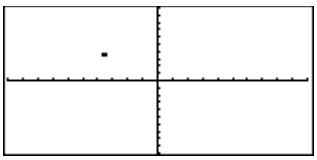
PxlOn CATALOG

PxlOn *rad*, *kolumn*
PxlOn *radlista*, *kolumnlista*

Visar graffönstret och slår på bildpunkten med koordinaterna (*rad*, *kolumn*).

Obs! Vid omritning tas alla ritobjekt bort.

`PxlOn 25,50` ENTER



pxlTest() CATALOG

pxlTest (*rad*, *kolumn*) ⇒ *booleskt uttryck*
pxlTest (*radlista*, *kolumnlista*) ⇒ *booleskt uttryck*

Returnerar true om bildpunkten vid koordinaterna (*rad*, *kolumn*) är på. Returnerar false om bildpunkten är av.

Obs! Vid omritning tas alla ritobjekt bort.

```
PxlOn 25,50 ENTER
```

TI-89: HOME
TI-92 Plus: ◆ [HOME]

```
PxlTest(25,50) ENTER          true
PxlOff 25,50 ENTER
```

TI-89: HOME
TI-92 Plus: ◆ [HOME]

```
PxlTest(25,50) ENTER         false
```

PxlText CATALOG

PxlText *sträng*, *rad*, *kolumn*

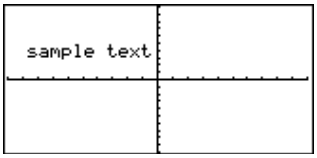
Visar graffönstret och placerar teckensträngen *sträng* i fönstret, med början vid bildpunktskoordinaterna (*rad*, *kolumn*).

sträng är placerad med det övre vänstra hörnet av det första tecknet vid koordinaterna.

Obs! Vid omritning tas alla ritobjekt bort.

TI-89: `PxlText "sample text",20,10` ENTER

TI-92 Plus: `PxlText "sample text",20,50` ENTER



PxlVert CATALOG

PxlVert *kolumn* [, *ritläge*]

Ritar en vertikal linje i fönstret vid bildpunktsposition *kolumn*.

Om *ritläge* = 1, ritas linjen (standard).
Om *ritläge* = 0, stängs linjen av.
Om *ritläge* = -1, inverteras bildpunkter längs linjen.

Obs! Vid omritning tas alla ritobjekt bort.
Se även **LineVert**.

```
PxlVert 50,1 ENTER
```

QR MATH/Matrix menu

QR *matris*, *qMatNamn*, *rMatNamn*[, *tol*]

Beräknar Householder QR faktorisering av en reell eller komplex *matris*. De resulterande Q- och R-matriserna sparas i specificerade *MatNamn*. Matrisen Q är enhetlig. R-matrisen är övre triangulär.

Varje element i en matris betraktas som noll om dess absolutvärde är mindre än *tol*. Denna tolerans används endast om matrisen innehåller decimalvärden och inte innehåller symboliska variabler som saknar värde. I andra fall ignoreras *tol*.

- Om du använder ◆ ENTER eller anger läget Exact/Approx=APPROXIMATE utförs beräkningar med flyttalsaritmetik.
- Om *tol* utesluts eller inte används beräknas standardtoleransen enligt:

  5E⁻14 ∗**max(dim(***matris***))** ∗**rowNorm(***matris***)**

Decimalvärdet (9.) i m1 gör att resultatet beräknas i decimaltalsform.

```
[1,2,3;4,5,6;7,8,9.]→m1 ENTER
```

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9. \end{bmatrix}$$

```
QR m1,qm,rm ENTER            Done
```

qm ENTER
$$\begin{bmatrix} .123\ldots & .904\ldots & .408\ldots \\ .492\ldots & .301\ldots & -.816\ldots \\ .861\ldots & -.301\ldots & .408\ldots \end{bmatrix}$$

rm ENTER
$$\begin{bmatrix} 8.124\ldots & 9.601\ldots & 11.078\ldots \\ 0. & .904\ldots & 1.809\ldots \\ 0. & 0. & 0. \end{bmatrix}$$

```
[m,n;o,p]→m1 ENTER
```

$$\begin{bmatrix} m & n \\ o & p \end{bmatrix}$$

QR-faktoriseringen beräknas numeriskt med Householder-transformeringar.
Den symboliska lösningen beräknas med Gram-Schmidt. Kolumnerna i *kMatNamn* är de ortonormerade basvektorerna som omfattar utrymmet definierat av *matris*.

QR m1,qm,rm [ENTER] Done

qm [ENTER]

$$\begin{bmatrix} \frac{m}{\sqrt{m^2+o^2}} & \frac{-\text{sign}(m\cdot p-n\cdot o)\cdot o}{\sqrt{m^2+o^2}} \\ \frac{o}{\sqrt{m^2+o^2}} & \frac{m\cdot\text{sign}(m\cdot p-n\cdot o)}{\sqrt{m^2+o^2}} \end{bmatrix}$$

rm [ENTER]

$$\begin{bmatrix} \sqrt{m^2+o^2} & \frac{m\cdot n+o\cdot p}{\sqrt{m^2+o^2}} \\ 0 & \frac{|m\cdot p-n\cdot o|}{\sqrt{m^2+o^2}} \end{bmatrix}$$

QuadReg Menyn MATH/Statistics/Regressions

QuadReg *lista1*, *lista2*[, [*lista3*] [, *lista4*, *lista5*]]

Beräknar den kvadratiska polynomregressionen och uppdaterar alla systemstatistikvariabler.

Alla listor måste ha samma dimensioner förutom *lista5*.

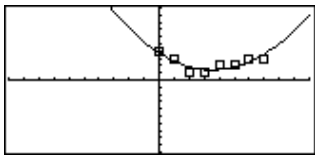
lista1 motsvarar x-listan.
lista2 motsvarar y-listan.
lista3 motsvarar frekvens.
lista4 motsvarar kategorivärden.
lista5 motsvarar de kategorivärden som ska inkluderas.

Obs! *lista1* till och med *lista4* måste vara variabelnamn eller c1–c99 (kolumner i den senaste datavariabeln som visades i Data/Matrix Editor). *lista5* behöver inte vara ett variabelnamn och kan inte vara någon av c1–c99.

I funktionsplottningsläge:

{0,1,2,3,4,5,6,7}→L1 [ENTER]
{1 2 3 ...}
{4,3,1,1,2,2,3,3}→L2 [ENTER]
{4 3 1 ...}
QuadReg L1,L2 [ENTER] Done
ShowStat [ENTER]

[ENTER]
Regeq(x)→y1(x) [ENTER] Done
NewPlot 1,1,L1,L2 [ENTER] Done
◆ [GRAPH]



QuartReg — Menyn MATH/Statistics/Regressions

QuartReg *lista1*, *lista2*[, [*lista3*] [, *lista4*, *lista5*]]

Beräknar den kvartära regressionen och uppdaterar alla systemstatistikvariabler.

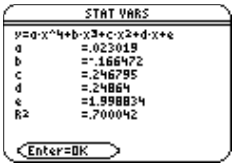
Alla listor måste ha samma dimensioner förutom *lista5*.

lista1 motsvarar x-listan.
lista2 motsvarar y-listan.
lista3 motsvarar frekvens.
lista4 motsvarar kategorivärden.
lista5 motsvarar de kategorivärden som ska inkluderas.

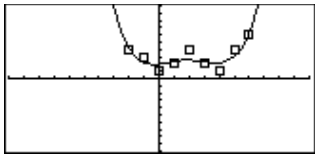
Obs! *lista1* till och med *lista4* måste vara variabelnamn eller c1–c99 (kolumner i den senaste datavariabeln som visades i Data/Matrix Editor). *lista5* behöver inte vara ett variabelnamn och kan inte vara någon av c1–c99.

I funktionsplottningsläge:

```
{-2,-1,0,1,2,3,4,5,6}→L1 ENTER
                    {-2 -1 0 ...}
{4,3,1,2,4,2,1,4,6}→L2 ENTER
                    {4 3 1 ...}
QuartReg L1,L2 ENTER        Done
ShowStat ENTER
```




```
ENTER
Regeq(x)→y1(x) ENTER        Done
NewPlot 1,1,L1,L2 ENTER     Done
◆ [GRAPH]
```



R▸Pθ() — Menyn MATH/Angle

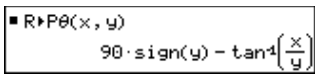
R▸Pθ (*xUttryck*, *yUttryck*) ⇒ *uttryck*
R▸Pθ (*xLista*, *yLista*) ⇒ *list*
R▸Pθ (*xMatris*, *yMatris*) ⇒ *matris*

Returnerar motsvarande θ-koordinat för pararguementen (*x*,*y*).

Obs! Resultatet returneras som en vinkel i grader eller radianer enligt den aktuella vinkelinställningen.

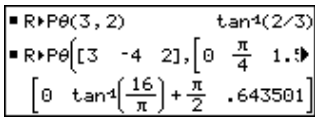
Med vinkelenheten grader:

```
R▸Pθ(x,y) ENTER
```



Med vinkelenheten radianer:

```
R▸Pθ(3,2) ENTER
R▸Pθ([3,-4,2],[0,π/4,1.5]) ENTER
```



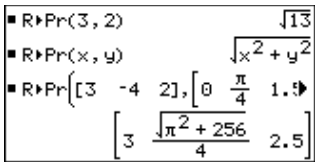
R▸Pr() — Menyn MATH/Angle

R▸Pr (*xUttryck*, *yUttryck*) ⇒ *uttryck*
R▸Pr (*xLista*, *yLista*) ⇒ *list*
R▸Pr (*xMatris*, *yMatris*) ⇒ *matris*

Returnerar motsvarande r-koordinat för pararguementen (*x*,*y*).

Med vinkelenheten radianer:

```
R▸Pr(3,2) ENTER
R▸Pr(x,y) ENTER
R▸Pr([3,-4,2],[0,π/4,1.5]) ENTER
```



rand() Menyn MATH/Probability

rand(*n***)** ⇒ *uttryck*

n är ett heltal ≠ noll.

Utan parametrar returneras nästa slumpmässigt utvalda tal mellan 0 och 1 i talföljden. När ett argument är positivt returneras ett slumpmässigt utvalt heltal i intervallet [1, *n*]. När ett argument är negativt returneras ett slumpmässigt utvalt heltal i intervallet [⁻*n*, ⁻1].

```
RandSeed 1147 ENTER                Done
```

(Ställer in startvärdet för slumptal.)

```
rand() ENTER                       .158...
rand(6) ENTER                          5
rand(⁻100) ENTER                     ⁻49
```

randMat() Menyn MATH/Probability

randMat(*antalRader***, *antalKolumner***)** ⇒ *matris*

Returnerar en matris med heltal mellan ⁻9 och 9 i den angivna dimensionen.

Båda argumenten måste kunna förenklas till heltal.

```
RandSeed 1147 ENTER                Done
randMat(3,3) ENTER
```

$$\begin{bmatrix} 8 & ^-3 & 6 \\ ^-2 & 3 & ^-6 \\ 0 & 4 & ^-6 \end{bmatrix}$$

Obs! Värdena i denna matris kommer att variera varje gång du trycker på ENTER.

randNorm() Menyn MATH/Probability

randNorm(*medelvärde***, *stdavv***)** ⇒ *uttryck*

Returnerar ett decimaltal från den angivna normalfördelningen. Det kan vara vilket reellt tal som helst men är koncentrerat i intervallet [*medelvärde*-3∗ *stdavv*, *medelvärde*+3∗ *stdavv*].

```
RandSeed 1147 ENTER                Done
randNorm(0,1) ENTER                .492...
randNorm(3,4.5) ENTER            ⁻3.543...
```

randPoly() Menyn MATH/Probability

randPoly(*var***, *ordning***)** ⇒ *uttryck*

Returnerar ett polynom i *var* i den angivna ordningen. Koefficienterna är slumpmässigt utvalda heltal i intervallet ⁻9 till 9. Den första koefficienten är inte noll.

ordning måste vara 0–99.

```
RandSeed 1147 ENTER                Done
randPoly(x,5) ENTER
        ⁻2·x⁵+3·x⁴−6·x³+4·x−6
```

RandSeed Menyn MATH/Probability

RandSeed *tal*

Om *tal* = 0 initieras slumptalsgeneratorn. Om *tal* ≠ 0 används det för att generera två "frön", som sparas i systemvariablerna seed1 och seed2.

```
RandSeed 1147 ENTER                Done
rand() ENTER                       .158...
```

RclGDB CATALOG

RclGDB *GDBvar*

Återställer alla inställningar som sparats i grafdatabasvariabeln GDBvar.

En lista över inställningar finns under **StoGDB**.

Obs! Du måste spara något i GDBvar innan du kan återställa det.

```
RclGDB GDBvar ENTER                Done
```

RclPic CATALOG

RclPic *bildvar* [, *rad*, *kolumn*]

Visar graffönstret och lägger till bilden som är sparad i *bildvar* med det övre vänstra hörnet vid koordinaterna (*rad*, *kolumn*) med hjälp av OR-logik (bilden läggs ovanpå).

bildvar måste vara en bilddatatyp.

Standardkoordinater är (0, 0).

real() Menyn MATH/Complex

real(*uttryck1***)** ⇒ *uttryck*

Returnerar den reella delen av ett argument.

Obs! Alla odefinierade variabler behandlas som reella variabler. Se även **imag()**.

```
real(2+3i) ENTER                2
real(z) ENTER                   z
real(x+iy) ENTER                x
```

real(*lista1***)** ⇒ *lista*

Returnerar de reella delarna av alla element.

```
real({a+i*b,3,i}) ENTER  {a  3  0}
```

real(*matris1***)** ⇒ *matris*

Returnerar de reella delarna av alla element.

```
real([a+i*b,3;c,i]) ENTER  [a  3
                            c  0]
```

▸Rect Menyn MATH/Matrix/Vector ops

vektor **▸Rect**

Visar *vektor* i rektangulär form [x, y, z]. *vektor* måste vara två- eller tredimensionell och kan antingen vara en rad- eller en kolumnvektor.

Obs! **▸Rect** är en instruktion för visningsformat, inte en omvandlingsfunktion. Du kan bara använda den i slutet av en inmatningsrad och den uppdaterar inte ans.

Obs! Se även **▸Polar**.

```
[3,∠π/4,∠π/6]▸Rect ENTER
```

$$\left[\frac{3\cdot\sqrt{2}}{4}\quad\frac{3\cdot\sqrt{2}}{4}\quad\frac{3\cdot\sqrt{3}}{2}\right]$$

```
[a,∠b,∠c] ENTER [a·cos(b)·sin(c)
    a·sin(b)·sin(c)  a·cos(c)]
```

komplextVärde **▸Rect**

Visar *komplextVärde* i den rektangulära formen a+b*i*. *komplextVärde* kan ha valfri komplex form. Om du matar in $re^{i\theta}$ returneras dock ett felmeddelande om du har vinkelenheten grader.

Obs! Du måste använda parenteser för inmatning av (r∠θ) polära värden.

Med vinkelenheten radianer:

```
4e^(π/3)▸Rect ENTER
```

$$4\cdot e^{\frac{\pi}{3}}$$

```
(4∠π/3)▸Rect ENTER
```

$$2+2\cdot\sqrt{3}\cdot i$$

Med vinkelenheten grader:

```
(4∠60)▸Rect ENTER
```

$$2+2\cdot\sqrt{3}\cdot i$$

Obs! Om du vill skriva in **▸Rect** via tangentbordet tryck [2nd] [▸] för operatorn ▸.
När du vill skriva in ∠ trycker du [2nd] [∠].

ref() Menyn MATH/Matrix

ref(*matris1***)** ⇒ *matris*

Returnerar radtrappstegsformen av *matris1*.

Varje element i en matris betraktas som noll om dess absolutvärde är mindre än *tol*. Denna tolerans används endast om matrisen innehåller decimaltal och inte innehåller symboliska variabler som saknar värde. I andra fall ignoreras *tol*.

- Om du använder ◆ ENTER eller anger läget Exact/Approx=APPROXIMATE utförs beräkningar med decimalvärden.
- Om *tol* utesluts eller inte används beräknas standardtoleransen enligt:

  5E⁻14 ∗ **max(dim(***matris1***))** ∗ **rowNorm(***matris1***)**

Obs! Se även **rref()**.

```
ref([-2,-2,0,-6;1,-1,9,-9;-5,2,4,-4]) ENTER
```

$$\begin{bmatrix} 1 & -2/5 & -4/5 & 4/5 \\ 0 & 1 & 4/7 & 11/7 \\ 0 & 0 & 1 & -62/71 \end{bmatrix}$$

```
[a,b,c;e,f,g]→m1 ENTER
```

$$\begin{bmatrix} a & b & c \\ e & f & g \end{bmatrix}$$

```
ref(m1) ENTER
```

$$\begin{bmatrix} 1 & \frac{f}{e} & \frac{g}{e} \\ 0 & 1 & \frac{a\cdot g-c\cdot e}{a\cdot f-b\cdot e} \end{bmatrix}$$

remain() Menyn MATH/Number

remain(*uttryck1***,** *uttryck2***)** ⇒ *uttryck*
remain(*lista1***,** *lista2***)** ⇒ *lista*
remain(*matris1***,** *matris2***)** ⇒ *matris*

Returnerar resten av det första argumentet med avseende på det andra argumentet så som det har definierats av identiteterna:

remain(x,0) ≡ x
remain(x,y) ≡ x− y∗iPart(x/y)

Som en följd av ovanstående är **remain(**−x,y**)** ≡ −**remain(**x,y**)**. Resultatet är antingen noll eller har samma tecken som det första argumentet.

Obs! Se även **mod()**.

```
remain(7,0) ENTER                    7
remain(7,3) ENTER                    1
remain(-7,3) ENTER                  -1
remain(7,-3) ENTER                   1
remain(-7,-3) ENTER                 -1
remain({12,-14,16},{9,7,-5}) ENTER
                              {3  0  1}
remain([9,-7;6,4],[4,3;4,-3]) ENTER
                              [1  -1
                               2   1]
```

Rename CATALOG

Rename *gammaltVarNamn*, *nyttVarNamn*

Ändrar namn på variabeln *gammaltVarNamn* till *nyttVarNamn*.

```
{1,2,3,4}→L1 ENTER          {1,2,3,4}
Rename L1, lista1 ENTER          Done
lista1 ENTER                {1,2,3,4}
```

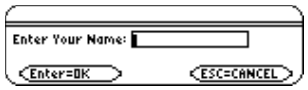
Request CATALOG

Request *frågesträng*, *var*

Om **Request** är inuti en **Dialog**...**EndDlog**-konstruktion skapas en inmatningsruta där användare kan mata in information. Om instruktionen är fristående skapas en dialogruta för denna inmatning. I båda fallen gäller att om *var* innehåller en sträng visas och markeras den i inmatningsrutan som standardalternativ. *frågesträng* måste vara ≤ 20 tecken.

Denna instruktion kan vara fristående eller en del av en dialogrutekonstruktion.

```
Request "Enter Your Name",str1 ENTER
```



Return CATALOG

Return [*uttryck*]

Returnerar *uttryck* som resultatet av funktionen. Använd inom ett **Func**...**EndFunc**-block eller **Prgm**...**EndPrgm**-block.

Obs! Använd **Return** utan ett argument för att avsluta ett program.

Obs! Skriv in texten som en lång rad i grundfönstret (utan radbrytningar).

```
Define factoral(n)=Func
  :local answer,count:1→answer
  :For count,1,n
  :answer*count→answer:EndFor
  :Return answer:EndFunc ENTER Done
factoral(3) ENTER                    6
```

right() Menyn MATH/List

right(*lista1*[, *antal*]**)** ⇒ *lista*

Returnerar element från höger i *lista1*.

Om du utelämnar *antal*, returneras alla element i *lista1*.

```
right({1,3,-2,4},3) ENTER
                          {3 -2 4}
```

right(*källsträng*[,*antal*]**)** ⇒ *sträng*

Returnerar *antal* tecken från höger i teckensträngen *källsträng*.

Om du utelämnar *antal* returneras alla tecken i *källsträng*.

```
right("Hello",2) ENTER          "lo"
```

right(*jämförelse***)** ⇒ *uttryck*

Returnerar den högra sidan av en ekvation eller olikhet.

```
right(x<3) ENTER                 3
```

rotate() MATH/Base menu

rotate(*heltal1*[,*antalSteg*]**)** ⇒ *heltal*

Roterar bitarna i ett binärt heltal. Du kan mata in *heltal1* i valfritt läge. Det konverteras automatiskt till 32-bitars binär form.
Om värdet är för stort för 32 bitar används en symmetrisk modulo-operation för att anpassa det till gällande begränsningar.

Om *antalSteg* är positivt roteras bitarna till vänster. Om *antalSteg* är negativt roteras bitarna till höger. Förvalt värde är -1 (rotera bitarna ett steg till höger).

Till exempel en rotering till höger:

Varje bit roteras till höger.

0b00000000000000111101011000011010 1

Biten längst till höger flyttas längst till vänster.

ger:

0b10000000000000111101011000011010

Resultatet visas i aktuellt visningsläge.

I binärt läge:

```
rotate(0b11110101100001101 01) ENTER
0b10000000000000111101011000011010
rotate(256,1) ENTER 0b1000000000
```

I hexadecimalt läge:

```
rotate(0h78E) ENTER          0h3C7
rotate(0h78E,-2) ENTER 0h800001E3
rotate(0h78E,2) ENTER       0h1E38
```

Viktigt: När du anger tal i binär eller hexadecimal form måste du alltid inleda med med 0b eller 0h (nolla, inte bokstaven O).

rotate(*lista1*[,*antalSteg*]**)** ⇒ *lista*

Returnerar en kopia av *lista1* roterad till höger eller vänster med *antalSteg*. Ändrar inte *lista1*.

Om *antalSteg* är positivt roteras elementen till vänster. Om *antalSteg* är negativt roteras elementen till höger. Förvalt värde är ⁻1 (rotera elementen ett steg till höger).

I decimalt läge:

```
rotate({1,2,3,4}) ENTER
                          {4 1 2 3}
rotate({1,2,3,4},⁻2) ENTER
                          {3 4 1 2}
rotate({1,2,3,4},1) ENTER
                          {2 3 4 1}
```

rotate(*sträng1*[,*antalSteg*]**)** ⇒ *sträng*

Returnerar en kopia av *sträng1* roterad till höger eller vänster med *antalSteg*. Ändrar inte *sträng1*.

Om *antalSteg* är positivt roteras tecknen till vänster. Om *antalSteg* är negativt roteras tecknen till höger. Förvalt värde är ⁻1 (rotera tecknen ett steg till höger).

```
rotate("abcd") ENTER        "dabc"
rotate("abcd",⁻2) ENTER     "cdab"
rotate("abcd",1) ENTER      "bcda"
```

round() Menyn MATH/Number

round(*uttryck1*[, *decimaler*]**)** ⇒ *uttryck*

Returnerar argumentet avrundat till angivet antal siffror efter decimalkommat.

decimaler måste vara ett heltal i intervallet 0 till 12. Om *decimaler* utelämnas returneras argumentet avrundat till 12 signifikanta siffror.

Obs! Läget Display Digits kan fortfarande påverka hur detta visas.

```
round(1.234567,3) ENTER      1.235
```

round(*lista1*[, *decimaler*]**)** ⇒ *lista*

Returnerar en lista med element avrundade till angivet antal siffror.

```
round({π,√(2),ln(2)},4) ENTER
                 {3.1416 1.4142 .6931}
```

round(*matris1*[,*decimaler*]**)** ⇒ *matris*

Returnerar en matris med element avrundad till angivet antal siffror.

```
round([ln(5),ln(3);π,e^(1)],1)
ENTER
                    [1.6  1.1
                     3.1  2.7]
```

rowAdd() Menyn MATH/Matrix/Row ops

rowAdd(*matris1*, *radindex1*, *radindex2***)** ⇒ *matris*

Returnerar en kopia av *matris1* med raden *radindex2* utbytt mot summan av raderna *radindex1* och *radindex2*.

```
rowAdd([3,4;⁻3,⁻2],1,2) ENTER
                        [ 3 4
                          0 2]
rowAdd([a,b;c,d],1,2) ENTER
                        [a     b
                         a+c   b+d]
```

rowDim() Menyn MATH/Matrix/Dimensions

rowDim(*matris***)** ⇒ *uttryck*

Returnerar antalet rader i *matris*.

Obs! Se även **colDim()**.

```
[1,2;3,4;5,6]→M1 ENTER
                        [1 2
                         3 4
                         5 6]
rowdim(M1) ENTER
                          3
```

rowNorm() Menyn MATH/Matrix/Norms

rowNorm(*matris***)** ⇒ *uttryck*

Returnerar den största radsumman av elementens absolutvärde i *matris*.

Obs! Alla matriselementen måste kunna förenklas till tal. Se även **colNorm()**.

```
rowNorm([-5,6,-7;3,4,9;9,-9,-7])
ENTER                                25
```

rowSwap() Menyn MATH/Matrix/Row ops

rowSwap(*matris1***, *radindex1***, *radindex2***)** ⇒ *matris*

Returnerar *matris1* med raderna *radindex1* och *radindex2* bytta mot varandra.

```
[1,2;3,4;5,6]→Mat ENTER
```

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

```
rowSwap(Mat,1,3) ENTER
```

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

RplcPic CATALOG

RplcPic *bildvar*[, *rad*][, *kolumn*]

Rensar graffönstret och placerar bilden *bildvar* vid bildpunktskoordinaterna (*rad*, *kolumn*). Om du inte vill rensa fönstret använder du **RclPic**.

bildvar måste vara en bilddatatypsvariabel. *rad* och *kolumn*, om de inkluderas, anger bildpunktskoordinaterna för det övre vänstra hörnet av bilden. Standardkoordinater är (0, 0).

Obs! För andra än fullfönsterbilder rensas bara den yta som bilden överlappar.

rref() Menyn MATH/Matrix

rref(*matris1***)** ⇒ *matris*

Returnerar den radreducerade formen av *matris1*.

Varje element i en matris betraktas som noll om dess absoluta värde är mindre än *tol*. Denna tolerans används endast om matrisen innehåller decimalvärden och inte innehåller symboliska variabler som saknar värde. I andra fall ignoreras *tol*.

- Om du använder ◆ ENTER eller anger läget Exact/Approx=APPROXIMATE utförs beräkningar med decimalvärden.
- Om *tol* utesluts eller inte används beräknas standardtoleransen enligt:

  5E⁻14 ∗**max(dim(***matris1***))** ∗**rowNorm(***matris1***)**

Obs! Se även **ref()**.

```
rref([-2,-2,0,-6;1,-1,9,-9;
-5,2,4,-4]) ENTER
```

$$\begin{bmatrix} 1 & 0 & 0 & 66/71 \\ 0 & 1 & 0 & \frac{147}{71} \\ 0 & 0 & 1 & -62/71 \end{bmatrix}$$

```
rref([a,b,x;c,d,y]) ENTER
```

$$\begin{bmatrix} 1 & 0 & \frac{d\cdot x-b\cdot y}{a\cdot d-b\cdot c} \\ 0 & 1 & \frac{-(c\cdot x-a\cdot y)}{a\cdot d-b\cdot c} \end{bmatrix}$$

Send CATALOG

Send *lista*

CBL™ (Calculator-Based Laboratory™) eller CBR™ (Calculator-Based Ranger™) instruktion. Skickar *lista* till länkporten.

Programsegment:

```
  ⋮
:Send {1,0}
:Send {1,2,1}
  ⋮
```

SendCalc CATALOG

SendCalc *var*

Skickar variabeln *var* till den länkade porten där en annan enhet som är länkad kan ta emot variabelvärdet. Den mottagande enheten måste ha grundfönstret aktiverat eller använda **GetCalc** från ett program.

Om du skickar data från en TI-89 eller TI-92 Plus till en TI-92 uppstår ett fel om TI-92 använder **GetCalc** från ett program. I dessa fall måste den sändande enheten i stället använda **SendChat**.

Programsegment:

```
  ⋮
:a+b→x
:SendCalc x
  ⋮
```

SendChat CATALOG

SendChat *var*

Ett allmänt alternativ till **SendCalc** som är särskilt användbart om den mottagande enheten är en TI-92 (eller för användning som ett allmänt "chat"-program där både TI-92 och TI-92 Plus kan användas). Se **SendCalc** för mer information.

SendChat skickar endast variabler som är kompatibla med en TI-92, vilket normalt är fallet för chat-program. Med **SendChat** kan du dock inte skicka variabler från användarminnet, TI-89 grafdatabaser, etc.

Programsegment:

```
  ⋮
:a+b→x
:SendChat x
  ⋮
```

seq() Menyn MATH/List

seq(*uttryck*, *var*, *undre*, *övre*[, *steg*]**)** ⇒ *lista*

Ökar *var* med *steg* från *undre* till *övre*, beräknar *uttryck* för varje värde på *var* och returnerar resultaten som en lista. Det ursprungliga innehållet i *var* finns fortfarande kvar efter att **seq()** har slutförts.

var kan inte vara en systemvariabel.

Standardvärdet för *steg* = 1.

```
seq(n^2,n,1,6) ENTER {1 4 9 16 25
seq(1/n,n,1,10,2) ENTER
                {1 1/3 1/5 1/7 1/9}
sum(seq(1/n^2,n,1,10,1)) ENTER
                              196...
                              127...
```

eller tryck på ◆ ENTER för att få: `1.549...`

setFold() CATALOG

setFold(*nyttMappnamn***)** ⇒ *gammalMappsträng*

Returnerar namnet på den aktuella mappen som en sträng och ställer in *nyttMappnamn* som den aktuella mappen.

Mappen *nyttMappnamn* måste finnas.

```
newFold chris ENTER          Done
setFold(main) ENTER        "chris"
setFold(chris)→oldfoldr ENTER
                           "main"
1→a ENTER                      1
setFold(#oldfoldr) ENTER   "chris"
a ENTER                        a
chris\a ENTER                  1
```

setGraph() CATALOG

setGraph(*lägesnamnSträng***,** *inställningSträng***)** ⇒ *sträng*

Ställer in grafläget *lägesnamnSträng* till den nya inställningen *inställningSträng* och returnerar den aktuella inställningen av läget. Om du sparar den aktuella inställningen kan du återställa den senare.

lägesnamnSträng är en teckensträng som anger vilket läge du vill ställa in. Det måste vara ett av namnen från tabellen nedan.

inställningSträng är en teckensträng som anger den nya inställningen för läget. Den måste vara en av inställningarna nedan för det specifika läge du ställer in.

```
setGraph("Graph Order","Seq")
ENTER                          "SEQ"
setGraph("Coordinates","Off")
ENTER                          "RECT"
```

Obs! Versaler och mellanslag är valfria när du matar in namn på lägen.

Namn på läge	Inställningar	
"Coordinates"	"Rect", "Polar", "Off"	
"Graph Order"	"Seq", "Simul" [1]	
"Grid"	"Off", "On" [2]	
"Axes"	"Off", "On" "Box", "Axes", "Off"	(inte 3D-grafläge) (3D-grafläge)
"Leading Cursor"	"Off", "On" [2]	
"Labels"	"Off", "On"	
"Style"	"Wire Frame", "Hidden Surface","Contour Levels", "Wire and Contour", "Implicit Plot" [3]	
"Seq Axes"	"Time", "Web", "U1-vs-U2" [4]	
"DE Axes"	"Time", "t-vs-y' ", "y-vs-y' ", "y1-vs-y2", "y1-vs-y2' ", "y1'-vs-y2' " [5] **Tips:** Om du vill skriva en primtalssymbol (') tryck 2nd [''].	
"Solution Method"	"RK", "Euler" [5]	
"Fields"	"SlpFld", "DirFld", "FldOff" [5]	

[1]Inte tillgänglig i Sequence-, 3D- eller Diff Equations-grafläge.
[2]Inte tillgänglig i 3D-grafläge.
[3]Endast tillgänglig i 3D-grafläge.
[4]Endast tillgänglig i Sequence-grafläge.
[5]Endast tillgänglig i Diff Equations grafläge.

setMode() CATALOG

setMode(*lägesnamnSträng***, *inställningSträng*)** ⇒ *sträng*
setMode(*lista***)** ⇒ *stränglista*

Ställer in läget *lägesnamnSträng* till den nya inställningen *inställningSträng* och returnerar den aktuella inställningen för det läget.

lägesnamnSträng är en teckensträng som anger vilket läge du vill ställa in. Det måste vara ett av namnen från tabellen nedan.

inställningSträng är en teckensträng som anger den nya inställningen för läget. Den måste vara en av inställningarna nedan för det specifika läget du ställer in.

lista innehåller par av lägesnamnssträngar och ställer in alla på en gång. Detta rekommenderas för ändringar av flera lägen. Exemplet som visas kanske inte fungerar om varje par matas in med en separat **setMode()**-inställning i ordningsföljden som visas.

Använd **setMode(***var***)** för att återställa inställningar som har sparats med **getMode("ALL")→** *var*.

Obs! Om du vill ställa in eller läsa information lagrad i Systemenheter använder du **setUnits()** eller **getUnits()** i stället för **setMode()** eller **getMode()**.

```
setMode("Angle","Degree")
ENTER                          "RADIAN"
```

```
sin(45) ENTER                  √2/2
```

```
setMode("Angle","Radian")
ENTER                          "DEGREE"
```

```
sin(π/4) ENTER                 √2/2
```

```
setMode("Display Digits",
  "Fix 2") ENTER               "FLOAT"
π ◆ ENTER                        3.14
```

```
setMode ("Display Digits",
  "Float") ENTER               "FIX 2"
π ◆ ENTER                      3.141...
```

```
setMode ({"Split Screen",
  "Left-Right","Split 1 App",
  "Graph","Split 2
  App","Table"})
  ENTER
          {"Split 2 App" "Graph"
            "Split 1 App" "Home"
          "Split Screen" "FULL"}
```

Obs! Versaler och mellanslag är valfria när du matar in namn på lägen. Resultaten i dessa exempel kan skilja sig från de du får på din räknare.

Namn på läge	Inställningar
"Graph"	"Function", "Parametric", "Polar", "Sequence", "3D", "Diff Equations"
"Display Digits"	"Fix 0", "Fix 1", ..., "Fix 12", "Float", "Float 1", ..., "Float 12"
"Angle"	"Radian", "Degree"
"Exponential Format"	"Normal", "Scientific", "Engineering"
"Complex Format"	"Real", "Rectangular", "Polar"
"Vector Format"	"Rectangular", "Cylindrical", "Spherical"
"Pretty Print"	"Off", "On"
"Split Screen"	"Full", "Top-Bottom", "Left-Right"
"Split 1 App"	"Home", "Y= Editor", "Window Editor", "Graph", "Table", "Data/Matrix Editor", "Program Editor", "Text Editor", "Numeric Solver", "*FlashProg*"
"Split 2 App"	"Home", "Y= Editor", "Window Editor", "Graph", "Table", "Data/Matrix Editor", "Program Editor", "Text Editor", "Numeric Solver", "*FlashProg*"
"Number of Graphs"	"1", "2"
"Graph2"	"Function", "Parametric", "Polar", "Sequence", "3D", "Diff Equations"
"Split Screen Ratio"	"1:1", "1:2", "2:1" (TI-92 Plus bara)
"Exact/Approx"	"Auto", "Exact", "Approximate"
"Base"	"Dec", "Hex", "Bin"
"Språk"	"Engelska", "*alternativt språk*"

setTable() CATALOG

setTable(*lägesNamnSträng*, *inställningSträng***)** ⇒ *sträng*

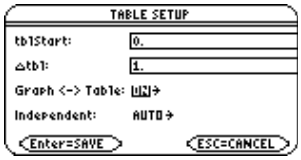
Ställer in tabellparametern *lägesnamnSträng* till den nya inställningen *inställningSträng* och returnerar den aktuella inställningen av parametern. Om du sparar den aktuella inställningen kan du återställa den senare.

lägesnamnSträng är en teckensträng som anger vilken parameter du vill ställa in. Det måste vara en av parametrarna i tabellen nedan.

inställningSträng är en teckensträng som anger den nya inställningen för parametern. Det måste vara en av inställningarna som visas nedan för den specifika parametern du ställer in.

```
setTable("Graph <->
Table","ON")
ENTER                          "OFF"

setTable("Independent","AUTO")
ENTER                          "ASK"
```

◆ [TblSet]



Obs! Versaler och mellanslag är valfria när du matar in namn på parametrar.

Namn på parameter	Inställningar
"Graph <-> Table"	"Off", "On"
"Independent"	"Auto", "Ask"

setUnits() CATALOG

setUnits(*lista1***)** ⇒ *lista*

Ställer in förvalda enheter till de värden som anges i *lista1* och returnerar en lista över tidigare förval.

- För att ange inbyggda enheter enlig systemen SI (metrisk) eller ENG/US använder *lista1* formen:

  {"SI"} eller {"ENG/US"}

- För att ange en anpassad uppsättning enheter använder *lista1* formen:

  {"CUSTOM", "*kat1*", "*enhet1*"[, "*kat2*", "*enhet2*", …]}

  där varje par med *kat* och *enhet* anger en kategori och dess förvalda enhet. (Du kan bara ange inbyggda enheter, inte användardefinierade.) Kategorier som inte anges behåller sin tidigare inställning.

- För att återgå till föregående anpassade förval använder *lista1* formen:

  {"CUSTOM"}

Om du vill använda olika förvalda enheter beroende på uppgiften kan du skapa olika listor med inställningar och spara dem under olika namn. När du vill använda inställningarna anger du bara listans namn med **setUnits()**.

Du kan använda **setUnits()** för att återställa inställningar som du tidigare sparat med **setUnits()** → *var* eller med **getUnits()** → *var*.

Alla enhetsnamn måste börja med ett understrykningstecken _.

TI-89: ◆ [_]

TI-92 Plus: 2nd [_]

Du kan också välja enheter från en meny genom att trycka på:

TI-89: 2nd [UNITS]

TI-92 Plus: ◆ [UNITS]

```
setUnits({"SI"}) ENTER
          {"SI"  "Area"  "NONE"
      "Capacitance"  "_F"  ...}

setUnits({"CUSTOM","Length",
"_cm","Mass","_gm"}) ENTER
          {"SI"  "Length"  "_m"
            "Mass"  "_kg"  ...}
```

Obs! Din skärm kan visa andra enheter.

Shade CATALOG

Shade *uttryck1*, *uttryck2*, [*xUndre*], [*xÖvre*], [*mönster*], [*mönsterUppl*]

Visar graffönstret, plottar *uttryck1* och *uttryck2* och skuggor ytor där *uttryck1* är mindre än *uttryck2*. (*uttryck1* och *uttryck2* måste vara uttryck som använder x som den oberoende variabeln.)

xUndre och *xÖvre*, om de tas med, anger den vänstra och högra gränsen för skuggningen. Giltiga inmatningar är mellan xmin och xmax. Standard är xmin och xmax.

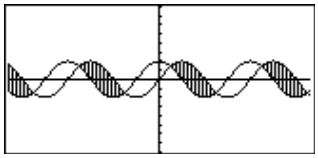
mönster anger ett av fyra skuggmönster:
1 = vertikal (standard)
2 = horisontell
3 = negativ lutning 45°
4 = positiv lutning 45°

mönsterUppl anger upplösningen för skuggmönstren:
1= fyllt mönster
2= 1 bildpunkts avstånd (standard)
3= 2 bildpunkters avstånd
⋮
10= 9 bildpunkters avstånd

Obs! Interaktiv skuggning är tillgänglig i graffönstret med instruktionen **Shade**. Automatisk skuggning av en viss funktion är tillgänglig med instruktionen **Style**. **Shade** är inte tillgänglig i 3D-grafläge.

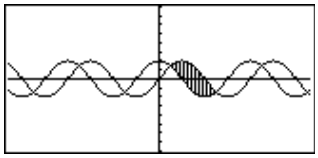
I visningsfönstret ZoomTrig:

```
Shade cos(x),sin(x) ENTER
```



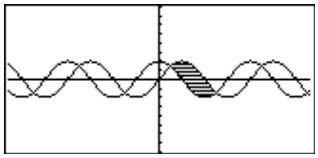
TI-89: HOME
TI-92 Plus: ♦ [HOME]

```
ClrDraw ENTER                    Done
Shade cos(x),sin(x),0,5 ENTER
```



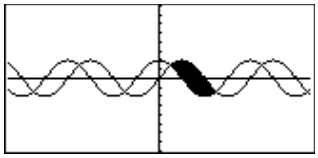
TI-89: HOME
TI-92 Plus: ♦ [HOME]

```
ClrDraw ENTER                    Done
Shade cos(x),sin(x),0,5,2 ENTER
```



TI-89: HOME
TI-92 Plus: ♦ [HOME]

```
ClrDraw ENTER                    Done
Shade cos(x),sin(x),0,5,2,1
ENTER
```



shift(*heltal1*[,*antalSteg*]**)** ⇒ *heltal*

Skiftar bitarna i ett binärt heltal. Du kan mata in *heltal1* i valfritt läge. Det konverteras automatiskt till 32-bitars binär form.
Om värdet är för stort för 32 bitar används en symmetrisk modulo-operation för att anpassa det till gällande begränsningar.

Om *antalSteg* är positivt skiftas bitarna till vänster. Om *antalSteg* är negativt skiftas bitarna till höger. Förvalt värde är ⁻1 (skifta bitarna ett steg till höger).

I en skiftning till höger försvinner biten längst till höger och 0 eller 1 sätts in för att matcha biten längst till vänster. I en skiftning till vänster försvinner biten längst till vänster och 0 sätts in längst till höger.

Vi visar här ett exempel på en skiftning till höger:

Varje bit skiftas till höger.

0b00000000000000111101011000011010 1

Sätter in 0 om vänstra biten är 0, eller 1 om vänstra biten är 1.

Släpps

ger:

0b00000000000000111101011000011010

Resultatet visas i aktuellt visningsläge. Inledande nollor visas inte.

I binärt läge:

```
shift(0b1111010110000110101)
ENTER
                0b111101011000011010
shift(256,1) ENTER   0b1000000000
```

I hexadecimalt läge:

```
shift(0h78E) ENTER          0h3C7
shift(0h78E,⁻2) ENTER       0h1E3
shift(0h78E,2) ENTER       0h1E38
```

Viktigt: När du anger tal i binär eller hexadecimal form måste du alltid inleda med med 0b eller 0h (nolla, inte bokstaven O).

---

shift(*lista1* [,*antalSteg*]**)** ⇒ *lista*

Returnerar en kopia av *lista1* skiftad till höger eller vänster med *antalSteg*. Ändrar inte *lista1*.

Om *antalSteg* är positivt skiftas elementen till vänster. Om *antalSteg* är negativt skiftas elementen till höger. Förvalt värde är ⁻1 (skifta elementen ett steg till höger).

Element som tillkommer i början eller slutet av *lista* genom skiftningen anges med symbolen "undef".

I decimalt läge:

```
shift({1,2,3,4}) ENTER
                    {undef 1 2 3}
shift({1,2,3,4},⁻2) ENTER
                 {undef undef 1 2}
shift({1,2,3,4},1) ENTER
                    {2 3 4 undef}
```

---

shift(*sträng1*[,*antalSteg*]**)** ⇒ *sträng*

Returnerar en kopia av *sträng1* skiftad till höger eller vänster med *antalSteg*. Ändrar inte *sträng1*.

Om *antalSteg* är positivt skiftas tecknen till vänster. Om *antalSteg* är negativt skiftas tecknen till höger. Förvalt värde är ⁻1 (skifta tecknen ett steg till höger).

Tecken som tillkommer i början eller slutet av *sträng* genom skiftningen sätts till mellanslag.

```
shift("abcd") ENTER        " abc"
shift("abcd",⁻2) ENTER     "  ab"
shift("abcd",1) ENTER      "bcd "
```

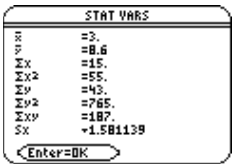
ShowStat CATALOG

ShowStat

Visar en dialogruta som innehåller de senast beräknade statistikresultaten om de fortfarande är giltiga. Statistiska resultat rensas automatiskt om informationen som ligger till grund för beräkningen ändras.

Använd denna instruktion efter en statistikberäkning, exempelvis **LinReg**.

```
{1,2,3,4,5}→L1 [ENTER]  {1 2 3 4 5}
{0,2,6,10,25}→L2 [ENTER]
                        {0 2 6 10 25}
TwoVar L1,L2 [ENTER]
ShowStat [ENTER]
```



sign() Menyn MATH/Number

sign(*uttryck1***)** ⇒ *uttryck*
sign(*lista1***)** ⇒ *lista*
sign(*matris1***)** ⇒ *matris*

För reella och komplexa *uttryck1* returneras *uttryck1*/**abs(***uttryck1***)** när *uttryck1*≠ 0.

Returnerar 1 om *uttryck1* är positivt. Returnerar ⁻1 om *uttryck1* är negativt. **sign(0)** returnerar ±1 om läget för komplext format är REAL. I andra lägen returneras det egna värdet. **sign(0)** motsvarar enhetscirkeln i den komplexa domänen.

För en lista eller matris returneras signum för alla element.

```
sign(⁻3.2) [ENTER]                    ⁻1.
sign({2,3,4,⁻5}) [ENTER]
                          {1  1  1  ⁻1}
sign(1+abs(x)) [ENTER]                 1
```

Om läget för komplext format är REAL:

```
sign([⁻3,0,3]) [ENTER]     [⁻1 ±1 1]
```

simult() MATH/Matrix menu

simult(*koeffMatris*, *konstVektor*[, *tol*]**)** ⇒ *matris*

Returnerar en kolumnvektor som innehåller lösningen på ett linjärt ekvationssystem.

koeffMatris måste vara en kvadratisk matris och bestå av koefficienterna i ekvationen.

konstVektor måste ha samma antal rader (samma dimension) som *koeffMatris* och innehålla konstanterna.

Varje element i en matris betraktas som noll om dess absolutvärde är mindre än *tol*. Denna tolerans används endast om matrisen innehåller decimalvärden och inte innehåller symboliska variabler som saknar värde. I andra fall ignoreras *tol*.

- Om du använder [◆] [ENTER] eller anger läget Exact/Approx=APPROXIMATE utförs beräkningar med decimaltal.
- Om *tol* utesluts eller inte används beräknas standardtoleransen enligt:

  5E⁻14 ∗ **max(dim(***matrisuttryck***))** ∗ **rowNorm(***matrisuttryck***)**

Lös för x och y: $x + 2y = 1$
$3x + 4y = {}^{-}1$

```
simult([1,2;3,4],[1;⁻1]) [ENTER]
                              [⁻3 ]
                              [ 2 ]
```

Lösningen är x=⁻3 och y=2.

Lös för x och y: $ax + by = 1$
$cx + dy = 2$

```
[a,b;c,d]→matx1 [ENTER]      [a b]
                             [c d]
simult(matx1,[1;2]) [ENTER]
```

$$\begin{bmatrix} \frac{-(2\cdot b-d)}{a\cdot d-b\cdot c} \\ \frac{2\cdot a-c}{a\cdot d-b\cdot c} \end{bmatrix}$$

simult(*koeffMatris*, *konstMatris*[, *tol*]**)** ⇒ *matris*

Löser flera system med linjära ekvationer där varje system har samma ekvationskoefficienter men olika konstanter.

Varje kolumn i *konstMatris* måste innehålla konstanten för ett system av ekvationer. Varje kolumn i den resulterande matrisen innehåller lösningen för motsvarande system.

Lös för x och y: $x + 2y = 1$ $\quad x + 2y = 2$
$3x + 4y = {}^{-}1$ $\quad 3x + 4y = {}^{-}3$

simult([1,2;3,4],[1,2;⁻1,⁻3]) ENTER

$$\begin{bmatrix} -3 & -7 \\ 2 & 9/2 \end{bmatrix}$$

För det första systemet, x=⁻3 och y=2.
För det andra systemet, x=⁻7 och y=9/2.

sin()

TI-89: 2nd [SIN] **tangenten** **TI-92 Plus:** [SIN] **tangenten**

sin(*uttryck1***)** ⇒ *uttryck*
sin(*lista1***)** ⇒ *lista*

sin(*uttryck1***)** returnerar sinus av argumentet som ett uttryck.

sin(*lista1***)** returnerar en lista med sinus av alla argument i *lista1*.

Obs! Argumentet tolkas som en vinkel i grader eller radianer, enligt den aktuella inställningen för vinkelenheter. Du kan använda ° eller ʳ för att tillfälligt åsidosätta inställningen för vinkelenhet.

Med vinkelenheten grader:

sin((π/4)ʳ) ENTER $\quad \frac{\sqrt{2}}{2}$

sin(45) ENTER $\quad \frac{\sqrt{2}}{2}$

sin({0,60,90}) ENTER $\quad \{0 \quad \frac{\sqrt{3}}{2} \quad 1\}$

En el modo Angle en radianes:

sin(π/4) ENTER $\quad \frac{\sqrt{2}}{2}$

sin(45°) ENTER $\quad \frac{\sqrt{2}}{2}$

sin(*kvadratiskMatris1***)** ⇒ *kvadratiskMatris*

Returnerar sinus av matrisen *kvadratiskMatris1*. Detta är *inte* samma sak som att beräkna sinus för varje element. Mer information om beräkningsmetoden finns i avsnittet om **cos()**.

kvadratiskMatris1 måste vara diagonaliserbar. Resultatet anges alltid med flyttalsaritmetik.

Med vinkelenheten radianer:

sin([1,5,3;4,2,1;6,⁻2,1]) ENTER

$$\begin{bmatrix} .942\ldots & -.045\ldots & -.031\ldots \\ -.045\ldots & .949\ldots & -.020\ldots \\ -.048\ldots & -.005\ldots & .961\ldots \end{bmatrix}$$

$\sin^{-1}()$

TI-89: ◆ [SIN⁻¹] **tangenten** **TI-92 Plus:** 2nd [SIN⁻¹] **tangenten**

sin⁻¹(*uttryck 1***)** ⇒ *uttryck*
sin⁻¹(*lista1***)** ⇒ *lista*

sin⁻¹(*uttryck1***)** returnerar vinkeln vars sinusvärde är *uttryck1* som ett uttryck.

sin⁻¹(*lista1***)** returnerar en lista med de vinklar som har sinusvärdena som element i *lista1*.

Obs! Resultatet returneras som ett vinkelmått i antingen grader eller radianer, enligt den aktuella inställningen för vinkelenheter.

Med vinkelenheten grader:

sin⁻¹(1) ENTER $\quad 90$

Med vinkelenheten radianer:

sin⁻¹({0,.2,.5}) ENTER

$\{0 \quad .201\ldots \quad .523\ldots\}$

sin⁻¹(*kvadratiskMatris1***)** ⇒ *kvadratiskMatris*

Returnerar arcussinus av matrisen *kvadratiskMatris1*. Detta är *inte* samma sak som att beräkna arcussinus för varje element. Mer information om beräkningsmetoden finns i avsnittet om **cos()**.

kvadratiskMatris1 måste vara diagonaliserbar. Resultatet anges alltid med flyttalsaritmetik.

Med vinkelinställningen Radian och komplext rektangulärt format:

sin⁻¹([1,5,3;4,2,1;6,⁻2,1]) ENTER

$$\begin{bmatrix} -.164\ldots-.064\ldots\cdot i & 1.490\ldots-2.105\ldots\cdot i & \ldots \\ .725\ldots-1.515\ldots\cdot i & .947\ldots-.778\ldots\cdot i & \ldots \\ 2.083\ldots-2.632\ldots\cdot i & -1.790\ldots+1.271\ldots\cdot i & \ldots \end{bmatrix}$$

sinh() Menyn MATH/Hyperbolic

sinh(*uttryck 1***)** ⇒ *uttryck*
sinh(*lista1***)** ⇒ *lista*

sinh (*uttryck1***)** returnerar sinus hyperbolicus av argumentet som ett uttryck.

sinh (*lista***)** returnerar en lista med sinus hyperbolicus av alla elementen i *lista1*.

```
sinh(1.2) ENTER                    1.509…

sinh({0,1.2,3.}) ENTER
                 {0   1.509…   10.017…}
```

---

sinh(*kvadratiskMatris1***)** ⇒ *kvadratiskMatris*

Returnerar sinus hyperbolicus av matrisen *kvadratiskMatris1*. Detta är *inte* samma sak som att beräkna sinus hyperbolicus för varje element. Mer information om beräknings-metoden finns i avsnittet om **cos()**.

kvadratiskMatris1 måste vara diagonaliserbar. Resultatet anges alltid med flyttalsaritmetik.

Med vinkelinställningen Radian:

```
sinh([1,5,3;4,2,1;6,⁻2,1])
ENTER
```

$$\begin{bmatrix} 360.954 & 305.708 & 239.604 \\ 352.912 & 233.495 & 193.564 \\ 298.632 & 154.599 & 140.251 \end{bmatrix}$$

$\sinh^{-1}()$ Menyn MATH/Hyperbolic

$\sinh^{-1}$(*uttryck1***)** ⇒ *uttryck*
$\sinh^{-1}$(*lista1***)** ⇒ *lista*

$\sinh^{-1}$(*uttryck1***)** returnerar arcussinus hyperbolicus av argumentet som ett uttryck.

$\sinh^{-1}$(*lista1***)** returnerar en lista med arcussinus hyperbolicus av alla element i *lista1*.

```
sinh⁻¹(0) ENTER                    0

sinh⁻¹({0,2.1,3}) ENTER
                 {0   1.487…   sinh⁻¹(3)}
```

$\sinh^{-1}$(*kvadratiskMatris1***)** ⇒ *kvadratiskMatris*

Returnerar arcussinus hyperbolicus av matrisen *kvadratiskMatris1*. Detta är *inte* samma sak som att beräkna arcussinus hyperbolicus för varje element. Mer information om beräknings-metoden finns i avsnittet om **cos()**.

kvadratiskMatris1 måste vara diagonaliserbar. Resultatet anges alltid med flyttalsaritmetik.

Med vinkelinställningen Radian:

```
sinh⁻¹([1,5,3;4,2,1;6,⁻2,1])
ENTER
```

$$\begin{bmatrix} .041\ldots & 2.155\ldots & 1.158\ldots \\ 1.463\ldots & .926\ldots & .112\ldots \\ 2.750\ldots & -1.528\ldots & .572\ldots \end{bmatrix}$$

SinReg MATH/Statistics/Regressions menu

SinReg *lista1*, *lista2* [, [*iterationer*], [*period*] [, *lista3*, *lista4*]]

Beräknar den sinusoidala regressionen och uppdaterar systemets alla statistikvariabler.

Alla listor måste ha samma dimensioner förutom *lista4*.

lista1 representerar xlist.
lista2 representerar ylist.
lista3 representerar kategorikoder.
lista4 representerar lista med kategoriinnehåll.

iterationer anger maximalt antal gånger en lösning prövas (1 till 16). Om parametern utelämnas används 8. Normalt ger högre värde exaktare svar men längre beräkningstid och vice versa.

I funktionsplottningsläge:

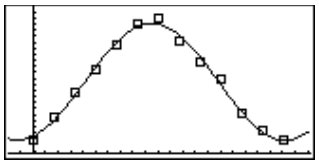
```
seq(x,x,1,361,30)→L1 ENTER
                    {1 31 61 …}
{5.5,8,11,13.5,16.5,19,19.5,17,
14.5,12.5,8.5,6.5,5.5}→L2 ENTER
                    {5.5 8 11 …}
SinReg L1,L2 ENTER          Done
ShowStat ENTER
```

period anger en uppskattad period. Om parametern utelämnas måste differensen mellan värdena i *lista1* vara lika och i sekventiell ordning. Om du anger *period* kan differensen mellan x-värden vara olika.

Obs! *lista1* till och med *lista3* måste vara ett variabelnamn eller c1–c99 (kolumner i den sista datavariabeln som visas i Data/Matrix Editor). *lista4* behöver inte vara ett variabelnamn och kan inte vara c1–c99.

Resultatet från **SinReg** visas alltid i radianer oavsett valda vinkelenheter.

```
ENTER
regeq(x)→y1(x) ENTER          Done
NewPlot 1,1,L1,L2 ENTER       Done
◆ [GRAPH]
F2 9
```



solve() Menyn MATH/Algebra

solve(*ekvation*, *var***)** ⇒ *booleskt uttryck*
solve(*olikhet*, *var***)** ⇒ *booleskt uttryck*

Returnerar reella lösningar till en ekvation eller olikhet för *var*. Målet är att returnera alla lösningar. Det kan dock finnas ekvationer eller olikheter för vilka antalet lösningar är oändligt.

```
solve(a*x^2+b*x+c=0,x) ENTER
```

$$x = \frac{\sqrt{-(4\cdot a\cdot c-b^2)}-b}{2\cdot a}$$

$$\text{or } x = \frac{-(\sqrt{-(4\cdot a\cdot c-b^2)}+b)}{2\cdot a}$$

Lösningar kan inte vara reella begränsade lösningar för vissa kombinationer av värden för odefinierade variabler.

```
ans(1)| a=1 and b=1 and c=1
ENTER
          Error: Non-real result
```

För AUTO-inställningen i Exact/Approx-läget är målsättningen att producera exakta lösningar när de är koncisa, och utökade med upprepande sökningar med approximativ aritmetik när exakta lösningar är opraktiska.

```
solve((x-a)e^(x)=⁻x*(x-a),x)
ENTER
                x = a or x = ⁻.567...
```

På grund av standardmässig eliminering av största gemensamma delaren från täljaren och nämnaren i bråk, kan lösningar vara ensidiga lösningar från en eller två sidor.

```
(x+1)(x-1)/(x-1)+x-3 ENTER 2·x-2
solve(entry(1)=0,x) ENTER     x = 1
entry(2)|ans(1) ENTER       undef
limit(entry(3),x,1) ENTER       0
```

För olikheter av typen ≥, ≤, < eller > är explicita lösningar osannolika, såvida inte olikheten är linjär och endast innehåller *var*.

```
solve(5x-2 ≥ 2x,x) ENTER     x ≥ 2/3
```

För EXACT-inställningen i Exact/Approx-läget returneras delar som inte kan lösas som en implicit ekvation eller olikhet.

```
exact(solve((x-a)e^(x)=⁻x*
  (x-a),x)) ENTER
                e^x + x = 0 or x = a
```

Använd operatorn "|" för att begränsa lösningsintervallet och/eller andra variabler som uppstår i ekvationen eller olikheten. När du hittar en lösning i ett intervall kan du använda operatorerna för olikhet för att exkludera det intervallet från fortsatta sökningar.

Med vinkelenheten radianer:

```
solve(tan(x)=1/x,x)|x>0 and x<1
ENTER                       x =.860...
```

false returneras när inga reella lösningar kan hittas. true returneras om **solve()** kan avgöra att ett begränsat reellt värde på *var* tillgodoser ekvationen eller olikheten.

```
solve(x=x+1,x) ENTER          false

solve(x=x,x) ENTER             true
```

Eftersom **solve()** alltid returnerar ett booleskt resultat kan du använda "and", "or" och "not" för att kombinera resultat från **solve()** med varandra eller med andra booleska uttryck.

```
2x-1≤1 and solve(x^2≠9,x) ENTER
                    x ≤ 1 and x ≠ ⁻3
```

Lösningar kan innehålla unika nya odefinierade variabler på formen @n*j* där *j* är ett heltal i intervallet 1–255. Sådana variabler anger ett godtyckligt heltal.

I reellt läge beräknas endast den reella grenen av bråkpotenser annars, med mångdelade uttryck som exempelvis bråkpotenser, logaritmer och inversa trigonometriska funktioner beräknar endast principalgrenen. Följaktligen så producerar **solve()** endast lösningar som motsvarar den enda reella grenen eller principalgrenen.

Obs! Se även **cSolve()**, **cZeros()**, **nSolve()** och **zeros()**.

Med vinkelinställningen Radian:

```
solve(sin(x)=0,x) ENTER   x = @n1·π
solve(x^(1/3)=-1,x) ENTER   x = -1
solve(√(x)=-2,x) ENTER   false
solve(-√(x)=-2,x) ENTER   x = 4
```

---

solve(*ekvation1* **and** *ekvation2* [**and** ...], **{***varEllerGissning1*, *varEllerGissning2* [, ...]**})**
⇒ *Booleskt uttryck*

Returnerar möjliga reella lösningar till ekvationssystem där varje *varEllerGissning* motsvarar en variabel som du vill lösa ekvationssystemet för.

Alternativt kan du ange en gissning för en variabel. Varje *varEllerGissning* måste vara på formen:

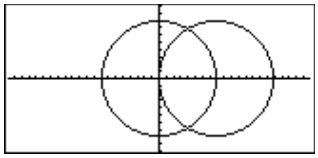
variabel
– eller –
variabel = reellt eller icke-reellt värde

Till exempel är x ett giltigt värde liksom x=3.

Om alla ekvationerna är polynoma och du INTE anger några gissningar använder **solve()** Gröbner/Buchberger elimineringsmetod för att försöka bestämma **alla** reella lösningar.

```
solve(y=x^2-2 and
x+2y=-1,{x,y}) ENTER
                    x=1 and y=-1
              or x=-3/2 and y=1/4
```

Om du t ex har en cirkel med radien r vid mittpunkten och en annan cirkel med radien r med centrum där den första cirkeln skär den positiva delen av x-axeln kan du använda **solve()** för att finna skärningspunkterna.



Som illustreras av r i exemplet till höger kan ekvationssystem ha extra variabler utan värde som representerar givna numeriska värden vilka senare kan bytas ut.

```
solve(x^2+y^2=r^2 and
(x-r)^2+y^2=r^2,{x,y}) ENTER
```

$$x=\frac{r}{2} \text{ and } y=\frac{\sqrt{3}\cdot r}{2}$$
$$\text{or } x=\frac{r}{2} \text{ and } y=\frac{-\sqrt{3}\cdot r}{2}$$

Du kan också eller i stället ange lösningsvariabler som inte ingår i ekvationen. Du kan t ex inkludera z som en lösningsvariabel för att utöka det föregående exemplet till två parallella skärande cylindrar med radien r.

```
solve(x^2+y^2=r^2 and
(x-r)^2+y^2=r^2,{x,y,z}) ENTER
```

$$x=\frac{r}{2} \text{ and } y=\frac{\sqrt{3}\cdot r}{2} \text{ and } z=@1$$

$$\text{or } x=\frac{r}{2} \text{ and } y=\frac{-\sqrt{3}\cdot r}{2} \text{ and } z=@1$$

Dessa lösningar visar hur grupper av lösningar kan innehålla godtyckliga konstanter av formen @*k*, där *k* är ett heltalssuffix mellan 1 och 255. Suffixet återställs till 1 när du använder **ClrHome** eller F1 8:Clear Home

Beräkningstid och minnesbehov för polynoma system kan i stor utsträckning bero på i vilken ordning du listar dina lösningsvariabler. Om ditt första val överskrider tillgängligt minne, eller ditt tålamod, kan du ändra ordningen på variablerna i ekvationen och/eller i listan med variablerna *varEllerGissning*.

Om du inte anger några gissningar eller om någon ekvation är icke-polynom, men alla ekvationer är linjära med alla lösningsvariabler använder **solve()** Gauss elimineringsmetod för att försöka bestämma alla reella lösningar.

```
solve(x+e^(z)*y=1 and
x-y=sin(z),{x,y}) ENTER
```

$$x=\frac{e^z\cdot\sin(z)+1}{e^z+1} \text{ and } y=\frac{-(\sin(z)-1)}{e^z+1}$$

Om ett system varken är polynomt i alla variabler eller linjärt i alla lösningsvariabler bestämmer **solve()** högst en lösning genom en ungefärlig iterativ metod. För att kunna göra detta måste antalet lösningsvariabler motsvara antalet ekvationer och alla andra variabler måste förenklas till numeriska värden.

```
solve(e^(z)*y=1 and
-y=sin(z),{y,z}) ENTER
                y=.041… and z=3.183…
```

Varje lösningsvariabel startar med gissat värde om det finns något, annars används 0.0.

Använd gissningar för att söka flera lösningar. Konvergenskraven gör att gissningen måste vara ganska nära lösningen.

```
solve(e^(z)*y=1 and
-y=sin(z),{y,z=2π}) ENTER
                y=.001… and z=6.281…
```

SortA

Menyn MATH/List

SortA *listVar1*[, *listVar2*] [, *listVar3*] **...**
SortA *vektorVar1*[,*vektorVar2*] [,*vektorVar3*] **...**

Sorterar elementen i det första argumentet i stigande ordning.

Om du inkluderar ytterligare argument sorteras elementen i varje argument så att deras nya positioner matchar de nya positionerna för elementen i det första argumentet.

Alla argument måste vara namn på listor eller vektorer. Alla argument måste ha samma dimensioner.

```
{2,1,4,3}→list1 ENTER      {2,1,4,3}
SortA list1 ENTER                Done

list1 ENTER                {1 2 3 4}
{4,3,2,1}→list2 ENTER      {4 3 2 1}
SortA list2,list1 ENTER          Done

list2 ENTER                {1 2 3 4}
list1 ENTER                {4 3 2 1}
```

SortD

Menyn MATH/List

SortD *listVar1*[, *listVar2*] [, *listVar3*] **...**
SortD *vektorVar1*[,*vektorVar2*] [,*vektorVar3*] **...**

Identisk med **SortA**, förutom att **SortD** sorterar elementen i fallande ordning.

```
{2,1,4,3}→list1 ENTER      {2 1 4 3}
{1,2,3,4}→list2 ENTER      {1 2 3 4}
SortD list1,list2 ENTER          Done
list1 ENTER                {4 3 2 1}
list2 ENTER                {3 4 1 2}
```

▸Sphere Menyn MATH/Matrix/Vector ops

vektor ▸**Sphere**

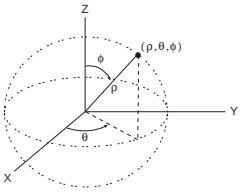
Visar rad- eller kolumnvektorn i sfärisk form [ρ ∠θ ∠ϕ].

vektor måste vara tredimensionell och kan antingen vara en rad- eller kolumnvektor.

Obs! ▸**Sphere** är en instruktion för visningsformat, inte en konverteringsfunktion. Du kan bara använda den i slutet av en inmatningsrad och den uppdaterar inte ans.

```
[1,2,3]▸Sphere
◆ ENTER [3.741... ∠1.107... ∠.640...]

[2,∠π/4,3]▸Sphere
◆ ENTER  [3.605... ∠.785... ∠.588...]
```

ENTER $\left[\sqrt{13}\ \angle\frac{\pi}{4}\ \angle\cos^{-1}\left(\frac{3\cdot\sqrt{13}}{13}\right)\right]$



stdDev() Menyn MATH/Statistics

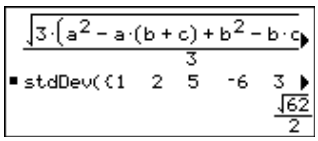
stdDev(*lista***[**, *frekvlista***])** ⇒ *uttryck*

Returnerar standardavvikelsen för elementen i *lista*.

Varje *frekvlista* element räknar antalet efter varandra följade förekomster av motsvarande element i *lista*.

Obs! *lista* måste ha minst två element.

```
stdDev({a,b,c}) ENTER
stdDev({1,2,5,⁻6,3,⁻2}) ENTER
```




```
stdDev({1.3,2.5,-6.4},{3,2,5})
ENTER                          4.33345
```

stdDev(*matris1***[**, *frekvmatris***])** ⇒ *matris*

Returnerar en radvektor med standardavvikelsen för elementen i *matris1*.

Varje *frekvlista* element räknar antalet efter varandra följade förekomster av motsvarande element i *matris1*.

Obs! *matris1* måste ha minst två element.

```
stdDev([1,2,5;⁻3,0,1;.5,.7,3])
ENTER
                [2.179...   1.014...    2]

stdDev([-1.2,5.3;2.5,7.3;6,-4],
[4,2;3,3;1,7]) ENTER
                    [2.7005,5.44695]
```

StoGDB CATALOG

StoGDB *GDBvar*

Skapar en variabel för grafdatabasen (GDB) som innehåller aktuella:

* Graflägen
* Y=-funktioner
* Window-variabler
* Grafformatinställningar
  Ett- eller tvågrafsläge (inställningar för Split Screen och Split Screen Ratio om tvågrafsläget är valt)
  Vinkelenhet
  Reellt eller komplext läge
* Initialvärden om läget Sequence eller Diff Equations är valt
* Tabellflaggor
* tblStart, Δtbl, tblInput

Du kan använda **RclGDB** *GDBvar* för att återställa grafmiljön.

***Obs!** Dessa poster sparas för båda graferna i tvågrafsläge.

Stop CATALOG

Stop

Används som en programinstruktion för att stoppa programkörning.

Programsegment:

```
  ⋮
For i,1,10,1
  If i=5
  Stop
EndFor
  ⋮
```

StoPic CATALOG

StoPic *bildvar* [, *bildpktrad*, *bildpktkolumn*] [, *bredd*, *höjd*]

Visar graffönstret och kopierar en rektangulär yta av fönstret till variabeln *bildvar*.

bildpktrad och *bildpktkolumn*, om de inkluderas, anger det övre vänstra hörnet av ytan som ska kopieras (standard är 0, 0).

bredd och *höjd*, om de inkluderas, anger dimensionerna (i bildpunkter) av ytan. Standardinställningen är bredden och höjden (i bildpunkter) på det aktuella graffönstret.

Store

Se →(spara) på sidan 534.

string() Menyn MATH/String

string(*uttryck***)** ⇒ *sträng*

Förenklar *uttryck* och returnerar resultatet som en teckensträng.

```
string(1.2345) ENTER        "1.2345"
string(1+2) ENTER                "3"
string(cos(x)+√(3)) ENTER
                      "cos(x)+√(3)"
```

Style CATALOG

Style *ekvationsnummer*, *egenskapssträng*

Ställer in graffunktionen *ekvationsnummer* i det aktuella grafläget för att använda grafegenskapen *egenskapssträng*.

ekvationsnummer måste vara ett heltal mellan 1 och 99 och funktionen måste redan finnas.

egenskapssträng måste vara någon av: "Line", "Dot", "Square", "Thick", "Animate", "Path", "Above" eller "Below".

Observera att det vid parameterplottning endast är x-komponenten av paret som innehåller formatinformationen.

Giltiga formatnamn kontra grafläge:

Function:	all styles
Parametric/Polar:	line, dot, square, thick, animate, path
Sequence:	line, dot, square, thick
3D:	none
Diff Equations:	line, dot, square, thick, animate, path

Obs! Versaler och mellanslag är tillåtna när du matar innamn på *egenskapssträng*.

```
Style 1,"thick" ENTER          Done
Style 10,"path" ENTER          Done
```

Obs! I funktionsplottningsläge ställer dessa exempel in formatet för y1(x) till "Thick" och y10(x) till "Path".

subMat() CATALOG

subMat(*matris1***[**, *startrad***]** [, *startkolumn***]** [, *slutrad***]** [, *slutkolumn***])** ⇒ *matris*

Returnerar den angivna submatrisen för *matris1*.

Standard: *startrad*=1, *startkolumn*=1, *slutrad*=sista rad, *slutkolumn*=sista kolumn.

```
[1,2,3;4,5,6;7,8,9]→m1 ENTER
```

$$\begin{bmatrix}1 & 2 & 3\\4 & 5 & 6\\7 & 8 & 9\end{bmatrix}$$

```
subMat(m1,2,1,3,2) ENTER
```

$$\begin{bmatrix}4 & 5\\7 & 8\end{bmatrix}$$

```
subMat(m1,2,2) ENTER
```

$$\begin{bmatrix}5 & 6\\8 & 9\end{bmatrix}$$

sum() Menyn MATH/List

sum(*lista***[**, *start***[**, *slut***]])** ⇒ *uttryck*

Returnerar summan av elementen i *lista*.

Start och *slut* är valfria. De anger ett område med element.

```
sum({1,2,3,4,5}) ENTER          15
sum({a,2a,3a}) ENTER            6·a
sum(seq(n,n,1,10)) ENTER        55
sum({1,3,5,7,9},3) ENTER        21
```

---

sum(*matris1***[**, *start***[**, *slut***]])** ⇒ *matris*

Returnerar en radvektor med summan av elementen i kolumnerna i *matris1*.

Start och *slut* är valfria. De anger ett område med rader.

```
sum([1,2,3;4,5,6]) ENTER [5  7  9]
sum([1,2,3;4,5,6;7,8,9]) ENTER
                         [12 15 18]
sum([1,2,3;4,5,6;7,8,9],2,3)
ENTER
                         [11,13,15]
```

switch() CATALOG

switch([*heltal1*]) ⇒ *heltal*

Returnerar numret för det aktiva fönstret. Kan också ställa in det aktiva fönstret.

Obs! Window 1 är till vänster eller överst; Window 2 är till höger eller längst ned.

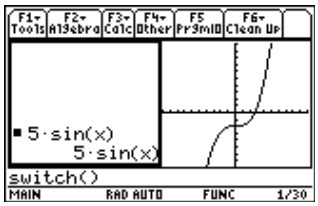
Om *heltal1* = 0, returneras numret för det aktiva fönstret.

Om *heltal1* = 1, aktiveras fönster 1 och returneras numret för det fönster som förut var aktivt.

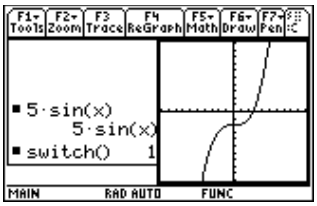
Om *heltal1* = 2, aktiveras fönster 2 och returneras numret för det fönster som förut var aktivt.

Om *heltal1* utelämnas växlas fönstren och returneras numret för det fönster som förut var aktivt.

heltal1 ignoreras om inte ett delat fönster visas.



switch ENTER



T (transponera) Menyn MATH/Matrix

*matris1*T ⇒ *matris*

Returnerar den konjugerade komplexa, transponerade matrisen till *matris1*.

[1,2,3;4,5,6;7,8,9]→mat1 ENTER

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$$

mat1T ENTER

$$\begin{bmatrix} 1 & 4 & 7 \\ 2 & 5 & 8 \\ 3 & 6 & 9 \end{bmatrix}$$

[a,b;c,d]→mat2 ENTER $\begin{bmatrix} a & b \\ c & d \end{bmatrix}$

mat2T ENTER $\begin{bmatrix} a & c \\ b & d \end{bmatrix}$

[1+*i*,2+*i*;3+*i*,4+*i*]→mat3 ENTER

$$\begin{bmatrix} 1+i & 2+i \\ 3+i & 4+i \end{bmatrix}$$

mat3T ENTER $\begin{bmatrix} 1-i & 3-i \\ 2-i & 4-i \end{bmatrix}$

Table CATALOG

Table *uttryck1*[, *uttryck2*] [, *var1*]

Genererar en tabell av de angivna uttrycken och funktionerna.

Uttrycken i tabellen kan också plottas. Uttryck som matas in med kommandona **Table** eller **Graph** tilldelas ökande funktionsnummer med början på 1. Uttrycken kan ändras eller tas bort ett och ett med hjälp av de redigeringsfunktioner som är tillgängliga när tabellen visas, genom att trycka på F4 Header. De för tillfället markerade funktionerna i Y= Editor ignoreras tillfälligt.

Du tar bort de funktioner som skapades med **Table** eller **Graph** genom att köra kommandot **ClrGraph** eller genom att visa Y= Editor.

Om parametern *var* utelämnas antas den oberoende variabeln för grafläget. Några av de giltiga variationerna för denna instruktion är följande:

Funktionsplottning: **Table** *uttryck*, *x*

Parameterplottning: **Table** *xUttryck*, *yUttryck*, *t*

Polärplottning: **Table** *uttryck*, θ

Obs! Kommandot **Table** är inte giltigt för 3D- och talföljdsplottning, eller vid plottning av lösningar till differentialekvationer. Som ett alternativ kan du använda **BldData**.

I funktionsplottningsläge.

Table 1.25x∗cos(x) ENTER

x	1		
0.	0.		
1.	.67538		
2.	-1.04		
3.	-3.712		
4.	-3.268		

Table cos(time),time ENTER

x	1	2	3
0.	0.	1.	
1.	.67538	.5403	
2.	-1.04	-.4161	
3.	-3.712	-.99	
4.	-3.268	-.6536	

tan() TI-89: 2nd TAN tangenten TI-92 Plus: TAN tangenten

tan(*uttryck1***)** ⇒ *uttryck*
tan(*lista1***)** ⇒ *lista*

tan(*uttryck1***)** returnerar tangens av argumentet som ett uttryck.

tan(*lista1***)** returnerar en lista med tangens av alla element i *lista1*.

Obs! Argumentet tolkas som en vinkel i grader eller radianer, enligt den aktuella inställningen för vinkelenheter. Du kan använda ° eller ʳ för att tillfälligt åsidosätta inställningen för vinkelenhet.

Med vinkelinställningen Degree:

tan((π/4)ʳ) ENTER 1

tan(45) ENTER 1

tan({0,60,90}) ENTER
{0 √3 undef}

Med vinkelinställningen Radian:

tan(π/4) ENTER 1

tan(45°) ENTER 1

tan({π,π/3,-π,π/4}) ENTER
{0 √3 0 1}

---

tan(*kvadratiskMatris1***)** ⇒ *kvadratiskMatris*

Returnerar tangens av matrisen *kvadratiskMatris1*. Detta är *inte* samma sak som att beräkna tangens för varje element. Mer information om beräkningsmetoden finns i avsnittet om **cos()**.

kvadratiskMatris1 måste vara diagonaliserbar. Resultatet anges alltid med flyttalsaritmetik.

Med vinkelinställningen Radian:

tan([1,5,3;4,2,1;6,-2,1]) ENTER

$$\begin{bmatrix} -28.291\ldots & 26.088\ldots & 11.114\ldots \\ 12.117\ldots & -7.835\ldots & -5.481\ldots \\ 36.818\ldots & -32.806\ldots & -10.459\ldots \end{bmatrix}$$

tan⁻¹() TI-89: ◆ [TAN⁻¹] tangenten TI-92 Plus: 2nd [TAN⁻¹] tangenten

tan⁻¹(*uttryck1***)** ⇒ *uttryck*
tan⁻¹(*lista1***)** ⇒ *lista*

tan⁻¹(*uttryck1***)** returnerar vinkeln vars tangensvärde är *uttryck1* som ett uttryck.

tan⁻¹(*lista1***)** returnerar en lista med de vinklar som har tangensvärdena som element i *lista1*.

Obs! Resultatet returneras som ett vinkelmått i antingen grader eller radianer, enligt den aktuella inställningen för vinkelenhet.

Med vinkelinställningen Degree:

tan⁻¹(1) ENTER 45

Med vinkelinställningen Radian:

tan⁻¹({0,.2,.5}) ENTER
{0 .197... .463...}

---

tan⁻¹(*kvadratiskMatris1***)** ⇒ *kvadratiskMatris*

Returnerar arcustangens av matrisen *kvadratiskMatris1*. Detta är *inte* samma sak som att beräkna arcustangens för varje element. Mer information om beräkningsmetoden finns i avsnittet om **cos()**.

kvadratiskMatris1 måste vara diagonaliserbar. Resultatet anges alltid med flyttalsaritmetik.

Med vinkelinställningen Radian:

tan⁻¹([1,5,3;4,2,1;6,⁻2,1]) ENTER

$$\begin{bmatrix} -.083... & 1.266... & .622... \\ .748... & .630... & -.070... \\ 1.686... & -1.182... & .455... \end{bmatrix}$$

tanh() Menyn MATH/Hyperbolic

tanh(*uttryck1***)** ⇒ *uttryck*
tanh(*lista1***)** ⇒ *lista*

tanh(*uttryck1***)** returnerar tangens hyperbolicus av argumentet som ett uttryck.

tanh(*lista***)** returnerar en lista med tangens hyperbolicus av alla elementen i *lista1*.

tanh(1.2) ENTER .833...

tanh({0,1}) ENTER {0 tanh(1)}

---

tanh(*kvadratiskMatris1***)** ⇒ *kvadratiskMatris*

Returnerar tangens hyperbolicus av matrisen *kvadratiskMatris1*. Detta är *inte* samma sak som att beräkna tangens hyperbolicus för varje element. Mer information om beräknings-metoden finns i avsnittet om **cos()**.

kvadratiskMatris1 måste vara diagonaliserbar. Resultatet anges alltid med flyttalsaritmetik.

Med vinkelinställningen Radian:

tanh([1,5,3;4,2,1;6,⁻2,1]) ENTER

$$\begin{bmatrix} -.097... & .933... & .425... \\ .488... & .538... & -.129... \\ 1.282... & -1.034... & .428... \end{bmatrix}$$

tanh⁻¹() Menyn MATH/Hyperbolic

tanh⁻¹(*uttryck 1***)** ⇒ *uttryck*
tanh⁻¹(*lista1***)** ⇒ *lista*

tanh⁻¹(*uttryck1***)** returnerar arcustangens hyperbolicus av argumentet som ett uttryck.

tanh⁻¹(*lista1***)** returnerar en lista med arcus-tangens hyperbolicus av alla element i *lista1*.

Med det komplexa formatläget rectangular:

tanh⁻¹(0) ENTER 0

tanh⁻¹({1,2.1,3}) ENTER

$$\{\infty \quad .518... - 1.570...\cdot i \quad \frac{\ln(2)}{2} - \frac{\pi}{2}\cdot i\}$$

---

tanh⁻¹(*kvadratiskMatris1***)** ⇒ *kvadratiskMatris*

Returnerar arcustangens hyperbolicus av matrisen *kvadratiskMatris1*. Detta är *inte* samma sak som att beräkna arcustangens hyperbolicus för varje element. Mer information om beräkningsmetoden finns i avsnittet om **cos()**.

kvadratiskMatris1 måste vara diagonaliserbar. Resultatet anges alltid med flyttalsaritmetik.

Med vinkelinställningen Radian och komplext rektangulärt format:

tanh⁻¹([1,5,3;4,2,1;6,⁻2,1]) ENTER

$$\begin{bmatrix} -.099...+.164...\cdot i & .267...-1.490...\cdot i & ... \\ -.087...-.725...\cdot i & .479...-.947...\cdot i & ... \\ .511...-2.083...\cdot i & -.878...+1.790...\cdot i & ... \end{bmatrix}$$

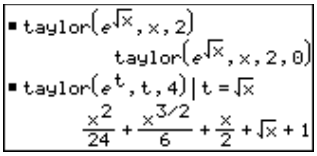
taylor() Menyn MATH/Calculus

taylor(*uttryck1*, *var*, *ordning*[, *punkt*]**)** ⇒ *uttryck*

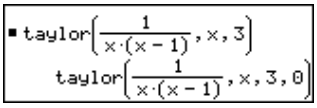
Returnerar det begärda Taylor-polynomet. Polynomet inkluderar termer som är skilda från noll och vars gradtal är ett heltal från noll till *ordning* i (*var* minus *punkt*). **taylor()** returnerar sig själv om det inte finns några trunkerade serier av denna ordning eller om det skulle krävas negativa exponenter eller bråktals-exponenter. Använd substitution och/eller temporär multiplikation av en potens av (*var* minus *punkt*) för att bestämma generellare potensserier.

punkt är som standard noll och det är kring denna punkt som utvecklingen sker.

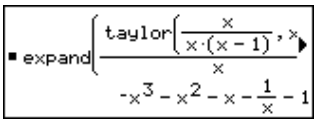
```
taylor(e^(√(x)),x,2) ENTER
taylor(e^(t),t,4)|t=√(x) ENTER
```




```
taylor(1/(x*(x-1)),x,3) ENTER
```




```
expand(taylor(x/(x*(x-1)),
x,4)/x,x) ENTER
```



tCollect() Menyn MATH\Algebra\Trig

tCollect(*uttryck1***)** ⇒ *uttryck*

Returnerar ett uttryck där produkter och heltalspotenser av sinus och cosinus konverteras till en linjär kombination av sinus och cosinus med multipla vinklar, vinkelsummor och vinkeldifferenser. Transformationen konverterar trigonometriska polynom till en linjär kombination av deras harmoniska komponenter.

Ibland kan **tCollect()** åstadkomma det du vill när vanliga trigonometriska förenklingar inte gör det. **tCollect()** har en tendens att omvända transformationer som är gjorda av **tExpand()**. Ibland kan du förenkla ett uttryck genom att tillämpa **tExpand()** på ett resultat från **tCollect()**, eller tvärtom, i två enskilda steg.

```
tCollect((cos(α))^2) ENTER
```

$$\frac{\cos(2\cdot\alpha)+1}{2}$$

```
tCollect(sin(α)cos(β)) ENTER
```

$$\frac{\sin(\alpha-\beta)+\sin(\alpha+\beta)}{2}$$

tExpand() Menyn MATH\Algebra\Trig

tExpand(*uttryck1***)** ⇒ *uttryck*

Returnerar ett uttryck där sinus och cosinus av heltalsmultipla vinklar, vinkelsummor och vinkeldifferenser är utvecklade. På grund av den trigonometriska identiteten $(\sin(x))^2+(\cos(x))^2=1$, finns det många möjliga ekvivalenta resultat. Därför kan ett resultat i räknaren kanske skilja sig från ett resultat som du sett någon annanstans.

Ibland kan **tExpand()** åstadkomma det du vill när vanliga trigonometriska förenklingar inte gör det. **tExpand()** har en tendens att omvända transformationer som är gjorda av **tCollect()**. Ibland kan du förenkla ett uttryck genom att tillämpa **tCollect()** på ett resultat från **tExpand()**, eller tvärtom, i två enskilda steg.

Obs! Skalning med π/180 i vinkelläget Degree stör funktionen som känner igen expanderbara former i **tExpand()** Bästa resultat med **tExpand()** får du om du använder vinkelinställningen Radian.

```
tExpand(sin(3ϕ)) ENTER
      4·sin(ϕ)·(cos(ϕ))²−sin(ϕ)

tExpand(cos(α−β)) ENTER
    cos(α)·cos(β)+sin(α)·sin(β)
```

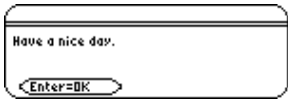
Text CATALOG

Text *textsträng*

Visar teckensträngen *textsträng* i en dialogruta.

Om den används som en del av ett **Dialog...EndDlog block**, visas *textsträng* inuti den dialogrutan. Om den används som en fristående instruktion skapar **Text** en dialogruta som visar strängen.

```
Text "Have a nice day." ENTER
                          Done
```



Then

Se **If** på sidan 455.

Title CATALOG

Title *titelsträng*, [*etikett*]

Skapar rubriken på en snabbmeny eller dialogruta när den används inuti en **Toolbar**- eller **Custom**-konstruktion eller ett **Dialog...EndDlog**-block.

Obs! *etikett* är bara giltigt i en **Toolbar**-konstruktion. När den inkluderas kan menyn förgrena sig till en angiven etikett inuti programmet.

Programsegment:

```
  ⋮
:Dialog
:Title    "This is a dialog
box"
:Request  "Your name",Str1
:Dropdown "Month you were
born",
  seq(string(i),i,1,12),Var1
:EndDlog
  ⋮
```

tmpCnv() CATALOG

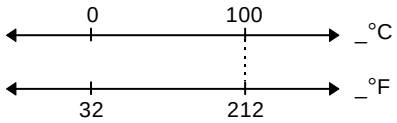
tmpCnv(*uttryck1_°tempEnhet1***, _***°tempEnhet2***)**
⇒ *uttryck _°tempEnhet2*

Omvandlar ett temperaturvärde definierat av *uttryck1* från en enhet till en annan. Giltiga enheter för temperatur är:

_°C	Celsius
_°F	Fahrenheit
_°K	Kelvin
_°R	Rankine

└─ För °, tryck [2nd] [°].
TI-89: För _, tryck [◆] [_].
TI-92 Plus: För _, tryck [2nd] [_].

Till exempel omvandlas 100_°C till 212_°F:



Om du vill konvertera ett temperaturområde använder du **ΔtmpCnv()** i stället.

```
tmpCnv(100_°c,_°f) ENTER 212.•_°F

tmpCnv(32_°f,_°c) ENTER    0.•_°C

tmpCnv(0_°c,_°k) ENTER 273.15•_°K

tmpCnv(0_°f,_°r) ENTER 459.67•_°R
```

Obs! Du kan välja temperaturenhet från en meny genom att trycka:

TI-89: [2nd] [UNITS]
TI-92 Plus: [◆] [UNITS]

ΔtmpCnv() CATALOG

ΔtmpCnv(*uttryck1_°tempEnhet1***, _***°tempEnhet2***)**
⇒ *uttryck _°tempEnhet2*

Omvandlar ett temperaturområde (differensen mellan två temperaturer) angivna med *uttryck1* från en enhet till en annan. Giltiga enheter för temperatur är:

_°C	Celsius
_°F	Fahrenheit
_°K	Kelvin
_°R	Rankine

└─ För °, tryck [2nd] [°].
TI-89: För _, tryck [◆] [_].
TI-92 Plus: För _, tryck [2nd] [_].

1_°C och 1_°K har samma magnitud liksom 1_°F och 1_°R. Trots detta är 1_°C 9/5 så stor som 1_°F.

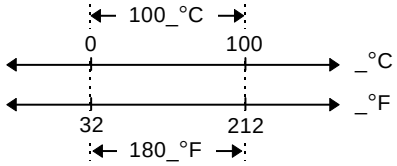
För Δ, tryck [◆] [(] [↑] [D] (eller [2nd] [CHAR] 1 5).

```
ΔtmpCnv(100_°c,_°f) ENTER
                    180.•_°F

ΔtmpCnv(180_°f,_°c) ENTER
                    100.•_°C

ΔtmpCnv(100_°c,_°k) ENTER
                    100.•_°K

ΔtmpCnv(100_°f,_°r) ENTER
                    100.•_°R

ΔtmpCnv(1_°c,_°f) ENTER
                     1.8•_°F
```

Obs! Du kan välja temperaturenheter från en meny genom att trycka:

TI-89: [2nd] [UNITS]
TI-92 Plus: [◆] [UNITS]

Till exempel är området 100_°C (från 0_°C till 100_°C) lika stort som området 180_°F:



Om du vill omvandla ett specifikt temperaturvärde använder du **tmpCnv()**.

Toolbar CATALOG

Toolbar
block
EndTBar

Skapar en meny.

block kan vara antingen en enstaka programsats eller en följd av satser som är avgränsade med tecknet ":". Programsatsen kan vara Title eller Item.

Alternativ (Item) på menyn måste ha etiketter. En titel (Title) måste också ha en etikett om den inte har ett alternativ under sig.

Programsegment:

```
   ⋮
:Toolbar
:  Title "Examples"
:  Item "Trig", t
:  Item "Calc", c
:  Item "Stop", Pexit
:EndTbar
   ⋮
```

Obs! När detta segment körs i ett program skapas en meny med tre alternativ som förgrenas till tre ställen i programmet.

Trace CATALOG

Trace

Ritar en Smart Graph och placerar spårningsmarkören på den först definierade Y=-funktionen vid den föregående definierade markörpositionen, eller vid den återställda positionen om omplottning behövdes.

Tillåter användning av markören och de flesta tangenter vid redigering av koordinatvärden. Flera tangenter, exempelvis funktionstangenterna, [APPS] och [MODE] aktiveras inte under spårning.

Obs! Tryck på [ENTER] för att fortsätta.

Try CATALOG

Try
block1
Else
block2
EndTry

Kör *block1* om inte ett fel uppstår. Programkörning överförs till *block2* om ett fel uppstår i *block1*. Variabeln errornum innehåller felnumret som gör det möjligt för programmet att agera på felet.

block1 och *block2* kan vara antingen en enstaka programsats eller flera satser som är avgränsade med tecknet ":".

Programsegment:

```
   ⋮
:Try
:  NewFold(temp)
:  Else
:  ◉ Already exists
:  ClrErr
:EndTry
   ⋮
```

Obs! Se **ClrErr** och **PassErr**.

TwoVar — Menyn MATH/Statistics

TwoVar *lista1*, *lista2*[, [*lista3*] [, *lista4*, *lista5*]]

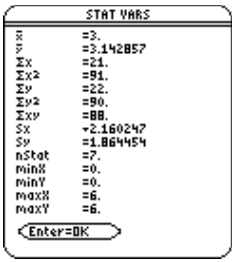
Beräknar **TwoVar**-statistiken och uppdaterar alla systemstatistikvariabler.

Alla listor måste ha samma dimensioner förutom *lista5*.

lista1 motsvarar x-listan.
lista2 motsvarar y-listan.
lista3 motsvarar frekvens.
lista4 motsvarar kategorivärden.
lista5 motsvarar de kategorivärden som ska inkluderas.

Obs! *lista1* till och med *lista4* måste vara variabelnamn eller c1–c99 (kolumner i den senaste datavariabeln som visades i Data/Matrix Editor). *list5* behöver inte vara ett variabelnamn och kan inte vara någon av c1–c99.

```
{0,1,2,3,4,5,6}→L1 [ENTER]
                          {0 1 2 ...}
{0,2,3,4,3,4,6}→L2 [ENTER]
                          {0 2 3 ...}
TwoVar L1,L2 [ENTER]            Done
ShowStat [ENTER]
```



Unarchiv — CATALOG

Unarchiv *var1* [, *var2*] [, *var3*] …

Flyttar angivna variabler från användarminnet till RAM.

Du kan anropa en lagrad variabel på samma sätt som om den låg i RAM. Du kan däremot inte ta bort, namnändra eller spara till en lagrad variabel eftersom den låses automatiskt.

Om du vill lagra variabler använder du **Archive**.

```
10→arctest [ENTER]                10
Archive arctest [ENTER]         Done
5*arctest [ENTER]                 50
15→arctest [ENTER]
```

```
[ESC]
Unarchiv arctest [ENTER]        Done
15→arctest [ENTER]                15
```

unitV() — Menyn MATH/Matrix/Vector ops

unitV(*vektor1***)** ⇒ *vektor*

Returnerar antingen en rad- eller en kolumn-enhetsvektor, beroende på formen på *vektor1*.

vektor1 måste vara antingen en enstaka radmatris eller enstaka kolumnmatris.

unitV([a,b,c]) [ENTER]

$$\left[\frac{a}{\sqrt{a^2+b^2+c^2}}\ \frac{b}{\sqrt{a^2+b^2+c^2}}\ \frac{c}{\sqrt{a^2+b^2+c^2}}\right]$$

unitV([1,2,1]) [ENTER]

$$\left[\frac{\sqrt{6}}{6}\ \frac{\sqrt{6}}{3}\ \frac{\sqrt{6}}{6}\right]$$

unitV([1;2;3]) [ENTER]

$$\begin{bmatrix} \frac{\sqrt{14}}{14} \\ \frac{\sqrt{14}}{7} \\ \frac{3\cdot\sqrt{14}}{14} \end{bmatrix}$$

Unlock — CATALOG

Unlock *var1*[, *var2*][, *var3*]...

Låser upp angivna variabler.

Obs! Variablerna kan låsas med kommandot **Lock**.

variance() Menyn MATH/Statistics

variance(*lista*[, *frekvlista*]**)** ⇒ *uttryck*

Retutnerar variansen hos *lista*.

Varje *frekvlista* element räknar antalet efter varandra följande förekomster av element i *lista*.

Obs! *lista* måste innehålla minst två element.

```
variance({a,b,c}) ENTER
```

$$\frac{a^2-a\cdot(b+c)+b^2-b\cdot c+c^2}{3}$$

```
variance({1,2,5,⁻6,3,⁻2}) ENTER
                                31/2
variance({1,3,5},{4,6,2}) ENTER
                                68/33
```

variance(*matris1*[, *frekvmatris*]**)** ⇒ *matris*

Returnerar en radvektor med variansen för varje kolumn i *matris1*.

Varje *frekvmatris* element räknar antalet efter varandra följande förekomster av element i *matris1*.

Obs! *matris1* måste innehålla minst två rader.

```
variance([1,2,5;⁻3,0,1;
.5,.7,3]) ENTER   [4.75  1.03  4]
variance([-1.1,2.2;3.4,5.1;
-2.3,4.3],[6,3;2,4;5,1]) ENTER
                [3.91731,2.08411]
```

when() CATALOG

when(*villkor*, *santResultat*[, *falsktResultat*]
[, *okäntResultat*]**)** ⇒ *uttryck*

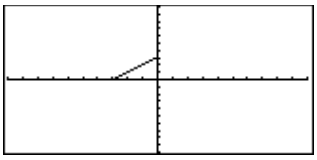
Returnerar *santResultat*, *falsktResultat* eller *okäntResultat*, beroende på om *villkor* är sant, falskt eller okänt. Returnerar inmatningen om det är för få argument för att ange det lämpliga resultatet.

Utelämna både *falsktResultat* och *okäntResultat* för att endast definiera ett uttryck i det område där *villkor* är sant.

```
when(x<0,x+3)|x=5 ENTER
                  when(x<0,3+x)
```

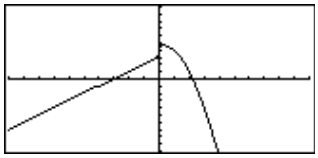
Använd en undef *falsktResultat* för att definiera ett uttryck som endast plottar inom ett intervall.

```
ClrGraph ENTER
Graph when(x≥⁻π and
x<0,x+3,undef) ENTER
```



Utelämna endast *okäntResultat* för att definiera ett uttryck i två delar.

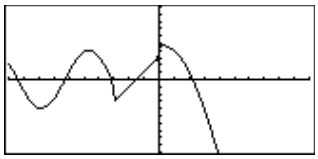
```
Graph when(x<0,x+3,5-x^2) ENTER
```



Nästla **when()** för att definiera uttryck som har mer än två delar.

TI-89: [HOME]
TI-92 Plus: [◆] [HOME]

```
ClrGraph [ENTER]                    Done
Graph when(x<0,when(x<⁻π,
4*sin(x),2x+3),5-x^2) [ENTER]
```



when() är bra för att definiera rekursiva funktioner.

```
when(n>0,n*factoral(n-1),1)
→factoral(n) [ENTER]                Done
factoral(3) [ENTER]                    6
3! [ENTER]                             6
```

While — CATALOG

While *villkor*
  block
EndWhile

Kör programsatserna i *block* så länge som *villkor* är sant.

block kan antingen vara en enstaka programsats eller flera satser som avgränsas med tecknet ":".

Programsegment:

```
  ⋮
:1→i
:0→temp
:While i<=20
:  temp+1/i→temp
:  i+1→i
:EndWhile
:Disp " Disp "sum of reciprocals
  up to 20",temp
  ⋮
```

"With"

Se | på sidan 533.

xor — Menyn MATH/Test

booleskt uttryck1 **xor** *booleskt uttryck2* ⇒ *booleskt uttryck*

Returnerar true om *booleskt uttryck1* är **true** och *booleskt uttryck2* är falskt, eller tvärtom. Returnerar falskt om *booleskt uttryck1* och *booleskt uttryck2* båda är sanna eller båda är falska. Returnerar ett förenklat booleskt uttryck om något av de ursprungliga booleska uttrycken inte kan beräknas till sant eller falskt.

Obs! Se **or**.

```
true xor true [ENTER]              false
(5>3) xor (3>5) [ENTER]             true
```

heltal1 **xor** *heltal2* ⇒ *heltal*

Jämför två reella heltal bit för bit med operationen **xor**. Internt konverteras båda talen till 32 bitar. Vid jämförelsen är resultatet 1 om någon bit är 1 (men inte båda) och 0 om båda bitarna är 0 eller 1. Resultatet visas i aktuellt visningsläge.

Du kan skriva in heltalen i valfritt läge. Om du använder binärt eller hexadecimalt läge börjar du talet med 0b alternativt 0h. Värden utan prefix behandlas i decimalt läge (bas 10).

Om du skriver in ett decimalt tal som är för stort för 32 bitar används en symmetrisk modulo-operation för att konvertera talet till gällande begränsningar.

Obs! Se **or**.

I hexadecimalt läge:

```
0h7AC36 xor 0h3D5F ENTER 0h79169
```

Viktigt! Noll, inte bokstaven O.

I binärt läge:

```
0b100101 xor 0b100 ENTER 0b100001
```

Obs! Ett binärt tal kan bestå av upp till 32 tecken (förutom prefixet 0b). Ett hexadecimalt tal kan bestå av upp till 8 tecken.

XorPic CATALOG

XorPic *bildvar*[, *rad*] [, *kolumn*]

Visar bilden som är sparad i *bildvar* i det aktuella graffönstret.

Använder **xor**-logik för varje bildpunkt. Endast de bildpunkter som är olika i fönstret och på bilden sätts på. Denna instruktion stänger av bildpunkter som är påslagna i båda bilderna.

bildvar måste innehålla datatypen pic.

rad och *kolumn*, om de inkluderas, anger bildpunktskoordinaterna för det övre vänstra hörnet i bilden. Standard är (0, 0).

zeros() Menyn MATH/Algebra

zeros(*uttryck*, *var***)** ⇒ *lista*

Returnerar en lista med reella lösningar för *var* som gör *uttryck*=0. **zeros()** gör detta genom att beräkna **exp▸list(solve(***uttryck*=0,*var***))**.

```
zeros(a*x^2+b*x+c,x) ENTER
```

$$\left\{\frac{-\left(\sqrt{b^2-4\cdot a\cdot c}+b\right)}{2\cdot a}\ \frac{\sqrt{b^2-4\cdot a\cdot c}-b}{2\cdot a}\right\}$$

```
a*x^2+b*x+c | x=ans(1)[2] ENTER   0
```

Ibland kan resultatformen från **zeros()** vara lättare att använda än den som **solve()** ger. Resultatformen från **zeros()** kan emellertid inte uttrycka implicita lösningar, lösningar med olikheter eller lösningar som inte använder *var*.

```
exact(zeros(a*(e^(x)+x)
  (sign (x)-1),x)) ENTER        {}
```

```
exact(solve(a*(e^(x)+x)
  (sign (x)-1)=0,x)) ENTER
```

$e^x + x = 0$ or $x>0$ or $a = 0$

Obs! Se även **cSolve()**, **cZeros()** och **solve()**.

zeros({*uttryck1*, *uttryck2* [, ...]**}, {***varEllerGissning1*, *varEllerGissning2* [, ...]**})** ⇒ *matris*

Returnerar möjliga reella nollställen hos de algebraiska uttrycken. Varje *varEllerGissning* motsvarar en okänd variabel vars värde du söker.

Alternativt kan du ange en gissning för en variabel. Varje *varEllerGissning* måste vara på formen:

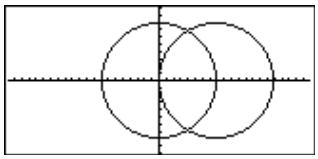
variabel

– eller –

variabel = reellt eller icke-reellt värde

Till exempel är x ett giltigt värde liksom x=3.

Om alla uttryck är polynom och du INTE anger några gissningar använder **zeros()** Gröbner/Buchberger elimineringsmetod för att försöka bestämma **alla** reella nollställen.



Om du t ex har en cirkel med radien r vid mittpunkten och en annan cirkel med radien r med centrum där den första cirkeln skär den positiva delen av x-axeln kan du använda **zeros()** för att finna skärningspunkterna.

Som illustreras av r i exemplet till höger kan ekvationssystem ha extra variabler utan värde som representerar givna numeriska värden vilka senare kan bytas ut.

```
zeros({x^2+y^2-r^2,
(x-r)^2+y^2-r^2},{x,y}) ENTER
```

$$\begin{bmatrix} \frac{r}{2} & \frac{\sqrt{3}\cdot r}{2} \\ \frac{r}{2} & \frac{-\sqrt{3}\cdot r}{2} \end{bmatrix}$$

Varje rad i den resulterande matrisen representerar ett alternativt nollställe med komponenterna ordnade i samma ordning som i listan *varEllerGissning*. Om du vill ta ut en rad anger du index i matrisen med [*row*].

Ta ut rad 2:

```
ans(1)[2] ENTER
```

$$\begin{bmatrix} \frac{r}{2} & \frac{-\sqrt{3}\cdot r}{2} \end{bmatrix}$$

Du kan också eller i stället ange lösnings-variabler som inte ingår i ekvationen. Du kans t ex inkludera z som en lösningsvariabel för att utöka det föregående exemplet till två parallella skärande cylindrar med radien r. Dessa nollställen visar hur grupper av nollställen kan innehålla godtyckliga konstanter av formen @*k*, där *k* är ett heltalssuffix mellan 1 och 255. Suffixet återställs till 1 när du använder **ClrHome** eller F1 8:Clear Home

```
zeros({x^2+y^2-r^2,
(x-r)^2+y^2-r^2},{x,y,z})ENTER
```

$$\begin{bmatrix} \frac{r}{2} & \frac{\sqrt{3}\cdot r}{2} & @1 \\ \frac{r}{2} & \frac{-\sqrt{3}\cdot r}{2} & @1 \end{bmatrix}$$

Beräkningstid och minnesbehov för polynoma system kan i stor utsträckning bero på i vilken ordning du listar dina lösningsvariabler. Om ditt första val överskrider tillgängligt minne, eller ditt tålamod, kan du ändra ordningen på variablerna i uttrycket och/eller i listan med variablerna *varEllerGissning*.

Om du inte anger några gissningar eller om något uttryck är icke-polynomt, men alla uttryck är linjära i någon variabel använder **zeros()** Gauss elimineringsmetod för att försöka bestämma alla reella nollställen.

```
zeros({x+e^(z)*y-1,x-y-sin(z)}
,{x,y}) ENTER
```

$$\begin{bmatrix} \frac{e^z\cdot \sin(z)+1}{e^z+1} & \frac{-(\sin(z)-1)}{e^z+1} \end{bmatrix}$$

Om ett system varken är polynomt i alla variabler eller linjärt i alla lösningsvariabler bestämmer **zeros()** högst en lösning genom en ungefärlig iterativ metod. För att kunna göra detta måste antalet lösningsvariabler motsvara antalet ekvationer och alla andra variabler måste förenklas till numeriska värden.

```
zeros({e^(z)*y-1,-y-sin(z)},
{y,z}) ENTER
```

[.041… 3.183…]

Varje lösningsvariabel startar med gissat värde om det finns något, annars används 0.

Använd gissningar för att söka flera nollställen.

```
zeros({e^(z)*y-1,-y-sin(z)},
```

Konvergenskraven gör att gissningen måste vara ganska nära nollstället.

```
{y,z=2π}) ENTER
                    [.001…  6.281…]
```

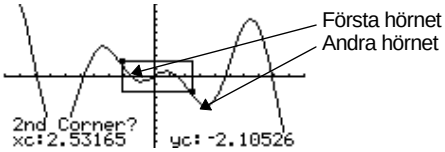
ZoomBox CATALOG

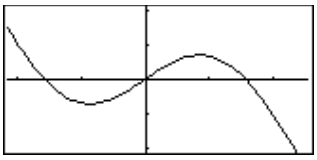
ZoomBox

Visar graffönstret, låter dig rita en ruta som definierar ett nytt visningsfönster och uppdaterar fönstret.

I funktionsplottningsläge:

```
1.25x*cos(x)→y1(x) ENTER      Done
ZoomStd:ZoomBox ENTER
```





Fönstret efter att du definierat ZoomBox genom att trycka på ENTER en andra gång.

ZoomData CATALOG

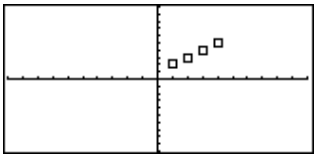
ZoomData

Justerar fönsterinställningarna utifrån de för tillfället definierade plottningarna (och data) så att alla statistiska datapunkter får plats i fönstret och visar sedan graffönstret.

Obs! Justerar inte ymin och ymax för histogram.

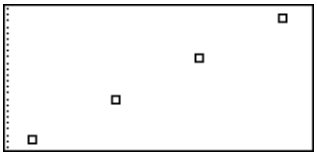
I funktionsplottningsläge:

```
{1,2,3,4}→L1 ENTER        {1 2 3 4}
{2,3,4,5}→L2 ENTER        {2 3 4 5}
newPlot 1,1,L1,L2 ENTER       Done
ZoomStd ENTER
```



TI-89: HOME
TI-92 Plus: ◆ [HOME]

```
ZoomData ENTER
```



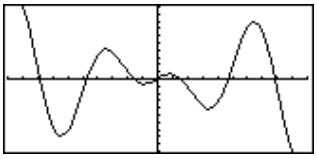
ZoomDec CATALOG

ZoomDec

Justerar visningsfönstret så att Δx och Δy = 0.1 och visar graffönstret med origo i mitten av fönstret.

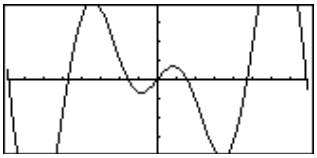
I funktionsplottningsläge:

```
1.25x*cos(x)→y1(x) ENTER        Done
ZoomStd ENTER
```



TI-89: HOME
TI-92 Plus: ♦ [HOME]

```
ZoomDec ENTER
```



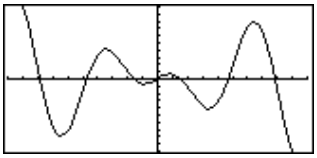
ZoomFit CATALOG

ZoomFit

Visar graffönstret och beräknar de nödvändiga fönsterdimensionerna för de beroende variablerna för att se hela bilden med avseende på den oberoende variabeln.

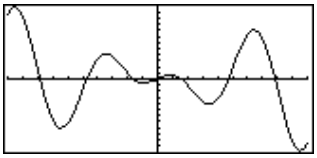
I funktionsplottningsläge:

```
1.25x*cos(x)→y1(x) ENTER        Done
ZoomStd ENTER
```



TI-89: HOME
TI-92 Plus: ♦ [HOME]

```
ZoomFit ENTER
```



ZoomIn CATALOG

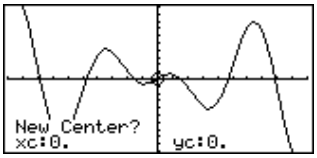
ZoomIn

Visar graffönstret, låter dig ställa in en centreringspunkt runt vilken zoomningen kommer att ske och uppdaterar sedan visningsfönstret.

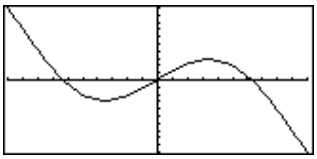
Graden av zoomning beror på zoomfaktorerna xFact och yFact. I läget 3D Graph, beror graden på xFact, yFact och zFact.

I funktionsplottningsläge:

```
1.25x*cos(x)→y1(x) ENTER        Done
ZoomStd:ZoomIn ENTER
```



ENTER



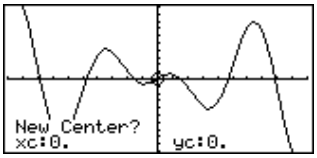
ZoomInt CATALOG

ZoomInt

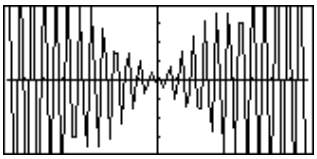
Visar graffönstret, låter dig ställa in centreringspunkt för zoomningen och justerar fönsterinställningarna så att varje bildpunkt är ett heltal i alla riktningar.

I funktionsplottningsläge:

```
1.25x*cos(x)→y1(x) ENTER        Done
ZoomStd:ZoomInt ENTER
```



ENTER



ZoomOut CATALOG

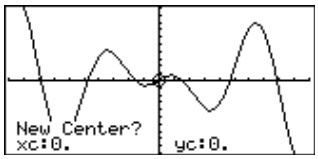
ZoomOut

Visar graffönstret, låter dig ställa in en centreringspunkt för utzoomningen och uppdaterar sedan visningsfönstret.

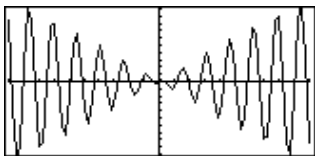
Graden av zoomning beror på zoomfaktorerna xFact och yFact. I läget 3D Graph beror graden på xFact, yFact och zFact.

I funktionsplottningsläge:

```
1.25x*cos(x)→y1(x) ENTER        Done
ZoomStd:ZoomOut ENTER
```



ENTER



ZoomPrev CATALOG

ZoomPrev

Visar graffönstret och uppdaterar visningsfönstret med de inställningar som användes innan den senaste zoomningen.

ZoomRcl CATALOG

ZoomRcl

Visar graffönstret och uppdaterar visningsfönstret med de inställningar som är sparade med instruktionen **ZoomSto**.

ZoomSqr CATALOG

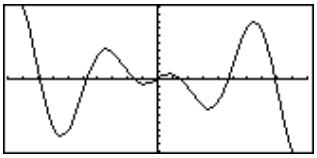
ZoomSqr

Visar graffönstret, justerar x- eller y-fönster-inställningarna så att varje bildpunkt motsvarar en lika stor bredd och höjd i koordinatsystemet och uppdaterar visningsfönstret.

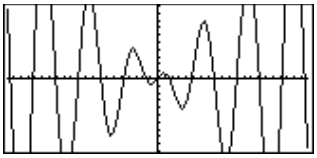
I 3D-grafläge förlänger **ZoomSqr** de två kortaste axlarna så att de är lika långa som den längsta axeln.

I funktionsplottningsläge:

```
1.25x*cos(x)→y1(x) ENTER      Done
ZoomStd ENTER
```



```
HOME
ZoomSqr ENTER
```



ZoomStd CATALOG

ZoomStd

Ställer in Window-variablerna till följande standardvärden och uppdaterar sedan visningsfönstret.

Funktionsplottning:
x: [⁻10, 10, 1], y: [⁻10, 10, 1] och xres=2

Parameterplottning:
t: [0, 2π, π/24], x:[⁻10,10,1], y:[⁻10,10,1]

Polärplottning:
θ: [0, 2π, π/24], x:[⁻10,10,1], y:[⁻10,10,1]

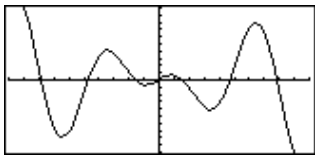
Talföljdsplottning:
nmin=1, nmax=10, plotStrt=1, plotStep=1,
x: [⁻10, 10, 1], y: [⁻10, 10, 1]

3D-plottning:
eyeθ°=20, eyeϕ°=70, eyeψ°=0
x: [⁻10, 10, 14], y: [⁻10, 10, 14],
z: [⁻10, 10], ncontour=5

Differentialekvations-plottning:
t: [0, 10, .1, 0], x: [⁻1, 10, 1], y: [⁻10, 10, 1],
ncurves=0, Estep=1, diftol=.001, fldres=20, dtime=0

I funktionsplottningsläge:

```
1.25x*cos(x)→y1(x) ENTER      Done
ZoomStd ENTER
```



ZoomSto CATALOG

ZoomSto

Sparar de aktuella fönsterinställningarna i Zoom-minnet. Du kan använda **ZoomRcl** för att återställa inställningarna.

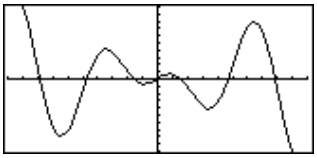
ZoomTrig CATALOG

ZoomTrig

Visar graffönstret, ställer in Δx till π/24 och xscl till π/2, centrerar origo, ställer in y-inställningarna till [⁻4, 4, .5] och uppdaterar visningsfönstret.

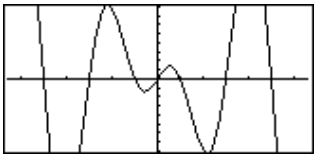
I funktionsplottningsläge:

```
1.25x*cos(x)→y1(x) ENTER        Done
ZoomStd ENTER
```



TI-89: HOME
TI-92 Plus: ◆ [HOME]

```
ZoomTrig ENTER
```



+ (addera) tangenten +

uttryck1 + *uttryck2* ⇒ *uttryck*

Returnerar summan av *uttryck1* och *uttryck2*.

```
56 ENTER                56
ans(1)+4 ENTER          60
ans(1)+4 ENTER          64
ans(1)+4 ENTER          68
ans(1)+4 ENTER          72
```

---

lista1 + *lista2* ⇒ *lista*
matris1 + *matris2* ⇒ *matris*

Returnerar en lista (eller matris) som innehåller summan av motsvarande element i *lista1* och *lista2* (eller *matris1* och *matris2*).

Dimensioner för argument måste vara lika.

```
{22,π,π/2}→L1 ENTER     {22 π π/2}
{10,5,π/2}→L2 ENTER     {10 5 π/2}
L1+L2 ENTER             {32 π+5 π}
ans(1)+{π,⁻5,⁻π} ENTER
                        {π+32 π 0}
[a,b;c,d]+[1,0;0,1] ENTER
                        [a+1 b; c d+1]
```

---

uttryck + *lista1* ⇒ *lista*
lista1 + *uttryck* ⇒ *lista*

Returnerar en lista som innehåller summan av *uttryck* och varje element i *lista1*.

```
15+{10,15,20} ENTER     {25 30 35}
{10,15,20}+15 ENTER     {25 30 35}
```

---

uttryck + *matris1* ⇒ *matris*
matris1 + *uttryck* ⇒ *matris*

Returnerar en matris med *uttryck* adderat till varje element på diagonalen i *matris1*. *matris1* måste vara kvadratisk.

Obs! Använd **.+** (elementvis addition) för att lägga till ett uttryck till varje element.

```
20+[1,2;3,4] ENTER
                        [21 2; 3 24]
```

− (subtrahera) tangenten −

uttryck1 − *uttryck2* ⇒ *uttryck*

Returnerar *uttryck1* minus *uttryck2*.

```
6−2 ENTER               4
π−π/6 ENTER             5·π/6
```

lista1 - *lista2* ⇒ *lista*
matris1 - *matris2* ⇒ *matris*

Subtraherar varje element i *lista2* (eller *matris2*) från motsvarande element i *lista1* (eller *matris1*) och returnerar resultatet.

Dimensionerna för argumenten måste vara lika.

```
{22,π,π/2}-{10,5,π/2} ENTER
                      {12 π-5 0}
[3,4]-[1,2] ENTER     [2 2]
```

uttryck - *lista1* ⇒ *lista*
lista1 - *uttryck* ⇒ *lista*

Subtraherar varje element i *lista1* från *uttryck* eller subtraherar *uttryck* från varje element i *lista1* och returnerar en lista med resultaten.

```
15-{10,15,20} ENTER   {5 0 -5}
{10,15,20}-15 ENTER   {-5 0 5}
```

uttryck - *matris1* ⇒ *matris*
matris1 - *uttryck* ⇒ *matris*

uttryck - *matris1* returnerar en matris med *uttryck* gånger enhetsmatrisen minus *matris1*. *matris1* måste vara kvadratisk.

matris1 - *uttryck* returnerar en matris med *uttryck* gånger enhetsmatrisen subtraherad från *matris1*. *matris1* måste vara kvadratisk.

Obs! Använd **.-** (elementvis subtraktion) för att subtrahera ett uttryck från varje element.

```
20-[1,2;3,4] ENTER
                      19  -2
                     [-3  16]
```

* (multiplicera) tangenten [×]

uttryck1 * *uttryck2* ⇒ *uttryck*

Returnerar produkten av *uttryck1* och *uttryck2*.

```
2*3.45 ENTER          6.9
x*y*x ENTER           x²·y
```

lista1 * *lista2* ⇒ *lista*

Returnerar en lista som innehåller produkterna av motsvarande element i *lista1* och *lista2*.

Dimensionerna på listorna måste vara lika.

```
{1.0,2,3}*{4,5,6} ENTER   {4. 10 18}
{2/a,3/2}*{a²,b/3} ENTER  {2·a  b/2}
```

matris1 * *matris2* ⇒ *matris*

Returnerar produkten av *matris1* och *matris2*.

Antalet rader i *matris1* måste vara samma som antalet kolumner i *matris2*.

```
[1,2,3;4,5,6]*[a,d;b,e;c,f]
ENTER
■[1 2 3]·[a d]
 [4 5 6] [b e]
         [c f]
 [a+2·b+3·c   d+2·e+3
 [4·a+5·b+6·c 4·d+5·e
```

uttryck * *lista1* ⇒ *lista*
lista1 * *uttryck* ⇒ *lista*

Returnerar en lista som innehåller produkterna av *uttryck* och varje element i *lista1*.

```
π*{4,5,6} ENTER       {4·π 5·π 6·π}
```

uttryck * *matris1* ⇒ *matris*
matrix1 * *uttryck* ⇒ *matris*

Returnerar en matris som innehåller produkten av *uttryck* och varje element i *matris1*.

Obs! Använd **.*** (elementvis multiplikation) för att multiplicera ett *uttryck* med varje element.

```
[1,2;3,4]*.01 ENTER   [.01 .02
                      [.03 .04]
λ*identity(3) ENTER   [λ 0 0
                      [0 λ 0
                      [0 0 λ]
```

/ (dividera) tangenten [÷]

uttryck1 / *uttryck2* ⇒ *uttryck*

Returnerar kvoten av *uttryck1* dividerat med *uttryck2*.

```
2/3.45 ENTER          .57971
x^3/x ENTER           x²
```

lista1 / lisat2 ⇒ *lista*

Returnerar en lista som innehåller kvoterna av *lista1* dividerat med *lista2*.

Dimensionerna för listorna måste vara lika.

```
{1.0,2,3}/{4,5,6} ENTER
                    {.25 2/5 1/2}
```

uttryck / lista1 ⇒ *lista*
lista1 / uttryck ⇒ *lista*

Returnerar en lista som innehåller kvoterna av *uttryck* dividerat med *lista1* eller *lista1* dividerat med *uttryck*.

```
a/{3,a,√(a)} ENTER
```

$$\{\frac{a}{3}\ 1\ \sqrt{a}\}$$

```
{a,b,c}/(a*b*c) ENTER
```

$$\{\frac{1}{b\cdot c}\ \frac{1}{a\cdot c}\ \frac{1}{a\cdot b}\}$$

matris1 / uttryck ⇒ *matris*

Returnerar en matris som innehåller kvoterna av *matris1/uttryck*.

Obs! Använd **./** (elementvis division) för att dividera ett *uttryck* med varje element.

```
[a,b,c]/(a*b*c) ENTER
```

$$[\frac{1}{b\cdot c}\ \frac{1}{a\cdot c}\ \frac{1}{a\cdot b}]$$

⁻ (negation) **tangenten** (-)

⁻*uttryck1* ⇒ *uttryck*
⁻*lista1* ⇒ *lista*
⁻*matris1* ⇒ *matris*

Returnerar negationen av argumentet.

För en lista eller matris returneras alla element negerade.

Om *uttryck1* är ett binärt eller hexadecimalt heltal returnerar negationen komplementet.

```
⁻2.43 ENTER                    ⁻2.43
⁻{⁻1,0.4,1.2E19} ENTER
                    {1 ⁻.4 ⁻1.2E19}
⁻a*⁻b ENTER                    a·b
```

I binärt läge:

```
0b100101▸dec ENTER              37
```

Viktigt: nolla, inte bokstaven O.

```
⁻0b100101 ENTER
0b11111111111111111111111111011011
ans(1)▸dec ENTER               ⁻37
```

Obs! När du vill skriva ▸ tryck 2nd [▸].

% (procent) **Menyn CHAR/Punctuation**

uttryck1 % ⇒ *uttryck*
lista1 % ⇒ *lista*
matris1 % ⇒ *matris*

Returnerar $\frac{argument}{100}$.

För en lista eller matris returneras en lista eller matris med varje element dividerat med 100.

```
13% ◆ ENTER                    .13
{1, 10, 100}% ◆ ENTER
                    {.01 .1 1.}
```

= (lika med) tangenten [=]

uttryck1 = *uttryck2* ⇒ *booleskt uttryck*
lista1 = *lista2* ⇒ *boolesk lista*
matris1 = *matris2* ⇒ *boolesk matris*

Returnerar sant om *uttryck1* är lika med *uttryck2*.

Returnerar falskt om *uttryck1* inte är lika med *uttryck2*.

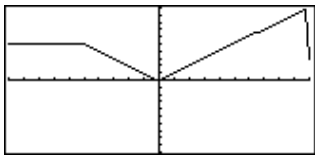
Allt annat returnerar en förenklad form av ekvationen.

För listor och matriser returneras jämförelser element för element.

Exempel på programlista som använder matematiska jämförelsesymboler:
=, ≠, <, ≤, >, ≥

```
:g(x)
:Func
:If x≤⁻5 Then
:  Return 5
:  ElseIf x>⁻5 and x<0 Then
:  Return ⁻x
:  ElseIf x≥0 and x≠10 Then
:  Return x
:  ElseIf x=10 Then
:  Return 3
:EndIf
:EndFunc
```

Graph g(x) [ENTER]



≠ tangenten [◆][=]

uttryck1 ≠ *uttryck2* ⇒ *booleskt uttryck*
lista1 ≠ *lista2* ⇒ *boolesk lista*
matris1 ≠ *matris2* ⇒ *boolesk matris*

Returnerar true om *uttryck1* inte är lika med *uttryck2*.

Returnerar false om *uttryck1* är lika med *uttryck2*.

Allt annat returnerar en förenklad form av ekvationen.

För listor och matriser returneras jämförelser element för element.

Se "="-examplet.

< tangenten [2nd][<]

uttryck1 < *uttryck2* ⇒ *booleskt uttryck*
lista1 < *lista2* ⇒ *boolesk lista*
matris1 < *matris2* ⇒ *boolesk matris*

Returnerar sant om *uttryck1* är mindre än *uttryck2*.

Returnerar falskt om *uttryck1* är större än eller lika med *uttryck2*.

Allt annat returnerar en förenklad form av ekvationen.

För listor och matriser returneras jämförelser element för element.

Se "="-exemplet.

≤ tangenten [♦] [0]

uttryck1 ≤ *uttryck2* ⇒ *booleskt uttryck*
lista1 ≤ *lista2* ⇒ *boolesk lista*
matris1 ≤ *matris2* ⇒ *boolesk matris*

Returnerar sant om *uttryck1* är mindre än eller lika med *uttryck2*.

Returnerar falskt om *uttryck1* är större än *uttryck2*.

Allt annat returnerar en förenklad form av ekvationen.

För listor och matriser returneras jämförelser element för element.

Se "="-exemplet.

> tangenten [2nd] [>]

uttryck1 > *uttryck2* ⇒ *booleskt uttryck*
lista1 > *lista2* ⇒ *boolesk lista*
matris1 > *matris2* ⇒ *boolesk matris*

Returnerar sant om *uttryck1* är större än *uttryck2*.

Returnerar falskt om *uttryck1* är mindre än eller lika med *uttryck2*.

Allt annat returnerar en förenklad form av ekvationen.

För listor och matriser returneras jämförelser element för element.

Se "="-exemplet.

≥ tangenten [♦] [.]

uttryck1 ≥ *uttryck2* ⇒ *booleskt uttryck*
lista1 ≥ *lista2* ⇒ *boolesk lista*
matris1 ≥ *matris2* ⇒ *boolesk matris*

Returnerar true om *uttryck1* är större än eller lika med *uttryck2*.

Returnerar false om *uttryck1* är mindre än *uttryck2*.

Allt annat returnerar en förenklad form av ekvationen.

För listor och matriser returneras jämförelser element för element.

Se "="-exemplet.

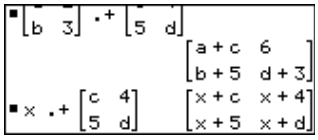
.+ (elem. add.) tangenterna [.] [+]

matris1 **.+** *matris2* ⇒ *matris*
uttryck **.+** *matris1* ⇒ *matris*

matris1 **.+** *matris2* returnerar en matris som är summan av varje par av motsvarande element i *matris1* och *matris2*.

uttryck **.+** *matris1* returnerar en matris som är summan av *uttryck* och varje element i *matris1*.

```
[a,2;b,3].+[c,4;5,d] ENTER
x.+[c,4;5,d] ENTER
```



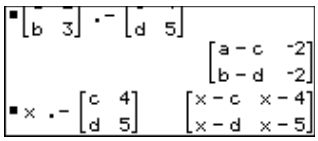
.− (elem. sub.) tangenterna [.] [−]

matris1 .− *matris2* ⇒ *matris*
uttryck .− *matris1* ⇒ *matris*

matris1 .− *matris2* returnerar en matris som är skillnaden mellan varje par av motsvarande element i *matris1* och *matris2*.

uttryck .− *matris1* returnerar en matris som är skillnaden mellan *uttryck* och varje element i *matris1*.

```
[a,2;b,3].-[c,4;d,5] ENTER
x.-[c,4;d,5] ENTER
```



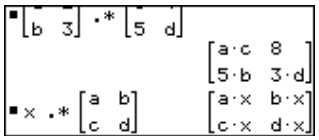
.* (elem. multi.) tangenterna [.] [×]

matris1 .* *matris2* ⇒ *matris*
uttryck .* *matris1* ⇒ *matris*

matris1 .* *matris2* returnerar en matris som är produkten av varje par av motsvarande element i *matris1* och *matris2*.

uttryck .* *matris1* returnerar en matris som innehåller produkten av *uttryck* och varje element i *matris1*.

```
[a,2;b,3].*[c,4;5,d] ENTER
x.*[a,b;c,d] ENTER
```



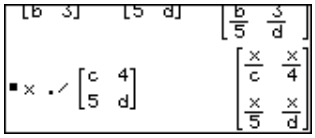
./ (elem. div.) tangenterna [.] [÷]

matris1 ./ *matris2* ⇒ *matris*
uttryck ./ *matris1* ⇒ *matris*

matris1 ./ *matris2* returnerar en matris som är kvoten av varje par av motsvarande element i *matris1* och *matris2*.

uttryck ./ *matris1* returnerar en matris som innehåller kvoten av *uttryck* och varje element i *matris1*.

```
[a,2;b,3]./[c,4;5,d] ENTER
x./[c,4;5,d] ENTER
```



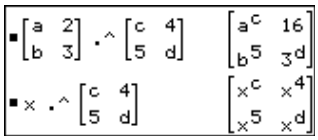
.^ (elem. potens) tangenterna [.] [^]

matris1 .^ *matris2* ⇒ *matris*
uttryck .^ *matris1* ⇒ *matris*

matris1 .^ *matris2* returnerar en matris där varje element i *matris2* är exponenten av motsvarande element i *matris1*.

uttryck .^ *matris1* returnerar en matris där varje element i *matris1* är exponenten för *uttryck*.

```
[a,2;b,3].^[c,4;5,d] ENTER
x.^[c,4;5,d] ENTER
```



! (fakultet) TI-89: tangenten [◆] [÷] TI-92 Plus: tangenten [2nd] W

uttryck1! ⇒ *uttryck*
lista1! ⇒ *lista*
matris1! ⇒ *matris*

Returnerar fakulteten av argumentet.

För en lista eller matris returneras en lista eller matris av fakulteten av elementen.

TI-89 beräknar ett numeriskt värde endast för icke-negativa heltalsvärden.

```
5! ENTER                    120
{5,4,3}! ENTER      {120  24  6}
[1,2;3,4]! ENTER        [1  2; 6  24]
```

& (sätt ihop) TI-89: tangenten [◆] [×] TI-92 Plus: tangenten [2nd] H

sträng1 & *sträng2* ⇒ *sträng*

Returnerar en textsträng som är *sträng2* tillagd *sträng1*.

```
" Hello " & "Nick" ENTER
                "Hello Nick "
```

∫() (integrera) **tangenten** [2nd] [∫]

∫(*uttryck1*, *var*[, *undre*] [,*övre*]) ⇒ *uttryck*
∫(*lista1*, *var*[, *ordning*]) ⇒ *lista*
∫(*matris1*, *var*[, *ordning*]) ⇒ *matris*

Returnerar integralen för *uttryck1* med avseende på *var* över intervallet från *undre* till *övre*.

∫(x^2,x,a,b) [ENTER]

$$\frac{-a^3}{3}+\frac{b^3}{3}$$

Returnerar den primitiva funktionen om *undre* och *övre* utelämnas. Den symboliska konstanten, exempelvis C, har utelämnats.

∫(x^2,x) [ENTER]

$$\frac{x^3}{3}$$

undre läggs däremot till som en konstant om *övre* utelämnas.

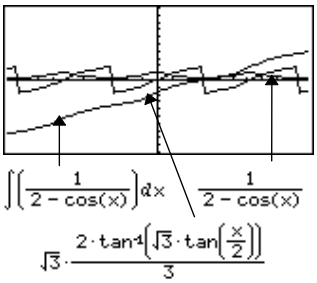
∫(a∗x^2,x,c) [ENTER]

$$\frac{a\cdot x^3}{3}+c$$

Primitiva funktioner som är lika mycket värda kan skiljas av en numerisk konstant. En sådan konstant kan vara gömd – särskilt när en primitiv funktion innehåller logaritmer eller inversa trigonometriska funktioner. Dessutom läggs uttryck som består av flera konstanter ibland till för att en primitiv funktion ska bli giltig över ett större intervall än den vanliga formeln.

∫(1/(2−cos(x)),x)→tmp(x) [ENTER]

ClrGraph:Graph tmp(x):Graph 1/(2−cos(x)):Graph √(3)(2tan⁻¹(√(3)(tan(x/2)))/3) [ENTER]



∫() returnerar sig själv för delar av *uttryck1* som den inte kan bestämma som en explicit begränsad kombination av de inbyggda funktionerna och operatorerna.

∫(b∗e^(⁻x^2)+a/(x^2+a^2),x) [ENTER]

$$\int\left(b\cdot e^{-x^2}+\frac{a}{x^2+a^2}\right)dx$$
$$b\cdot\int\left(e^{-x^2}\right)dx+\tan^{-1}\left(\frac{x}{a}\right)$$

När både *undre* och *övre* är inkluderade görs ett försök att hitta diskontinuiteter eller diskontinuerliga derivator i intervallet *undre* < *var* < *övre* och för att dela upp intervallet vid dessa ställen.

För AUTO-inställningen i Exact/Approx-läget används numerisk integration när det är lämpligt och när en primitiv funktion eller en gränsvärde inte kan bestämmas.

För APPROX-inställningen prövas först numerisk integration, om det är lämpligt. Primitiva funktioner söks endast där numerisk integration inte är tillämplig eller misslyckas.

∫(e^(⁻x^2),x,⁻1,1)◆ [ENTER] 1.493...

∫() kan nästlas till att göra multipla integraler. Integrationsgränser kan bero på integrationsvariabler utanför dem.

∫(∫(ln(x+y),y,0,x),x,0,a) [ENTER]

$$\int_0^a\int_0^x \ln(x+y)\,dy\,dx$$
$$\frac{a^2\cdot\ln(a)}{2}+a^2\cdot(\ln(2)-3/4)$$

Obs! Se även **nInt()**.

√() (roten ur) **tangenten** [2nd] [√]

√ (*uttryck1*) ⇒ *uttryck*
√ (*lista1*) ⇒ *lista*

Returnerar roten ur argumentet.

För en lista returneras roten ur av alla element i *lista1*.

√(4) [ENTER] 2

√({9,a,4}) [ENTER] {3 √a 2}

Π() (produkt) **Menyn MATH/Calculus**

Π(*uttryck1*, *var*, *undre*, *övre*) ⇒ *uttryck*

Beräknar *uttryck1* för varje värde av *var* från *undre* till *övre* och returnerar produkten av resultaten.

Π(1/n,n,1,5) [ENTER] $\frac{1}{120}$

Π(k^2,k,1,n) [ENTER] $(n!)^2$

Π({1/n,n,2},n,1,5) [ENTER] $\{\frac{1}{120}\ 120\ 32\}$

Π(*uttryck1*, *var*, *undre*, *undre*− 1) ⇒ 1

Π(k,k,4,3) [ENTER] 1

Π(*uttryck1*, *var*, *undre*, *övre*) ⇒ 1/Π(*uttryck1*, *var*, *övre*+1, *undre*− 1) om *övre* < *undre*− 1

Π(1/k,k,4,1) [ENTER] 6

Π(1/k,k,4,1)∗**Π**(1/k,k,2,4) [ENTER] 1/4

Σ() (summa) **Menyn MATH/Calculus**

Σ(*uttryck1*, *var*, *undre*, *övre*) ⇒ *uttryck*

Beräknar *uttryck1* för varje värde av *var* från *undre* till *övre* och returnerar summan av resultaten.

Σ(1/n,n,1,5) [ENTER] $\frac{137}{60}$

Σ(k^2,k,1,n) [ENTER] $\frac{n\cdot(n+1)\cdot(2\cdot n+1)}{6}$

Σ(1/n^2,n,1,∞) [ENTER] $\frac{\pi^2}{6}$

Σ(*uttryck1*, *var*, *undre*, *undre*− 1) ⇒ 0

Σ(k,k,4,3) [ENTER] 0

Σ(*uttryck1*, *var*, *undre*, *övre*) ⇒ −Σ(*uttryck1*, *var*, *övre*+1, *undre*− 1) om *övre* < *undre*− 1

Σ(k,k,4,1) [ENTER] −5

Σ(k,k,4,1)+Σ(k,k,2,4) [ENTER] 4

^ (potens) **tangenten** [^]

uttryck1 ^ *uttryck2* ⇒ *uttryck*
lista1 ^ *lista2* ⇒ *lista*

Returnerar det första argumentet upphöjt till det andra argumentet.

För en lista returneras elementen i *lista1* upphöjda till motsvarande element i *lista2*.

I den reella domänen använder bråkpotenser som har reducerade exponenter med udda nämnare den reella grenen kontra principalgrenen för det komplexa läget.

4^2 [ENTER] 16

{a,2,c}^{1,b,3} [ENTER] $\{a\ 2^b\ c^3\}$

uttryck ^ *lista1* ⇒ *lista*

Returnerar *uttryck* upphöjt till elementen i *lista1*.

```
p^{a,2,-3} ENTER
```

$\{p^a \quad p^2 \quad \frac{1}{p^3}\}$

lista1 ^ *uttryck* ⇒ *lista*

Returnerar elementen i *lista1* upphöjt till *uttryck*.

```
{1,2,3,4}^-2 ENTER
```

{1 1/4 1/9 1/16}

kvadratiskMatris1 ^ *heltal* ⇒ *matris*

Returnerar *kvadratiskMatris1* upphöjt till *heltal*.

kvadratiskMatris1 måste vara en kvadratisk matris.

Om *heltal* = ⁻1, beräknas den inversa matrisen. Om *heltal* < ⁻1, beräknas den inversa matrisen med en passande positivt exponent.

```
[1,2;3,4]^2 ENTER
[1,2;3,4]^-1 ENTER
[1,2;3,4]^-2 ENTER
```

$$\begin{bmatrix}1 & 2\\3 & 4\end{bmatrix}^2 \qquad \begin{bmatrix}7 & 10\\15 & 22\end{bmatrix}$$

$$\begin{bmatrix}1 & 2\\3 & 4\end{bmatrix}^{-1} \qquad \begin{bmatrix}-2 & 1\\3/2 & -1/2\end{bmatrix}$$

$$\begin{bmatrix}1 & 2\\3 & 4\end{bmatrix}^{-2} \qquad \begin{bmatrix}11/2 & -5/2\\-15/4 & 7/4\end{bmatrix}$$

(omdirigerare) CATALOG

varNamnsträng

Refererar till variabeln vars namn är *varNamnsträng*. Detta gör att du kan skapa och ändra variabler från ett program med hjälp av strängar.

Programsegment:

```
  ⋮
:Request "Enter Your Name",str1
:NewFold #str1
  ⋮

  ⋮
:For i,1,5,1
:  ClrGraph
:  Graph i*x
:  StoPic #("pic" & string(i))
:EndFor
  ⋮
```

ʳ (radianer) Menyn MATH/Angle

*uttryck1*ʳ ⇒ *uttryck*
*lista1*ʳ ⇒ *lista*
*matris1*ʳ ⇒ *matris*

Med vinkelenheten grader multipliceras *uttryck1* med 180/π. Med vinkelenheten radianer returneras *uttryck1* oförändrat.

Med den här funktionen kan du använda en vinkel angiven i radianer när du använder vinkelenheten grader. (Med vinkelenheten grader förväntar sig **sin()**, **cos()**, **tan()** och polär-till-rektangulär omvandling att vinkel-argumentet ska vara i grader.)

Tips! Använd ʳ om du vill använda radianer i en funktions- eller programdefinition oavsett vilket läge som är aktivt när funktionen eller programmet används.

Med vinkelenheten grader eller radianer:

```
cos((π/4)ʳ) ENTER
```

$\frac{\sqrt{2}}{2}$

```
cos({0ʳ,(π/12)ʳ,-πʳ}) ENTER
```

$\{1 \quad \frac{(\sqrt{3}+1)\cdot\sqrt{2}}{4} \quad -1\}$

° (grader) — tangenten 2nd [°]

uttryck° ⇒ *värde*
lista1° ⇒ *lista*
matris1° ⇒ *matris*

Med vinkelenheten radianer multipliceras *uttryck* med π/180. Med vinkelenheten grader returneras *uttryck* oförändrat.

Med den här funktionen kan du använda en vinkel angiven i grader när du använder vinkelenheten radianer. (Med vinkelenheten radianer förväntar sig **sin()**, **cos()**, **tan()** och polär-till-rektangulär omvandling att vinkelargumentet ska vara i radianer.)

Med vinkelenheten radianer:

cos(45°) ENTER $\frac{\sqrt{2}}{2}$

cos({0,π/4,90°,30.12°}) ◆ ENTER
{1 .707... 0 .864...}

∠ (vinkel) — tangenten 2nd [∠]

[*radie*,∠θ_*vinkel*] ⇒ *vektor* (polär inmatning)
[*radie*,∠θ_*vinkel*,*Z_koordinat*] ⇒ *vektor* (cylindrisk inmatning)
[*radie*,∠θ_*vinkel*,∠ϕ_*vinkel*] ⇒ *vektor* (sfärisk inmatning)

Returnerar koordinaterna som en vektor beroende på inställningen för läget Vector Format: rectangular, cylindrical eller spherical.

[5,∠60°,∠45°] ENTER

Med vinkelenheten radianer och vektorformatet inställt till:

■[5 ∠60° ∠45°]
$\left[\frac{5\cdot\sqrt{2}}{4}\ \ \frac{5\cdot\sqrt{6}}{4}\ \ \frac{5\cdot\sqrt{2}}{2}\right]$ rektangulärt
■[5 ∠60° ∠45°]
$\left[\frac{5\cdot\sqrt{2}}{2}\angle\frac{\pi}{3}\ \ \frac{5\cdot\sqrt{2}}{2}\right]$ cylindriskt
■[5 ∠60° ∠45°]
$\left[5\angle\frac{\pi}{3}\angle\frac{\pi}{4}\right]$ sfäriskt

---

(*magnitud* ∠ *vinkel*) ⇒ *komplextVärde* (polär inmatning)

Anger ett komplext värde i (r∠θ) polär form. *vinkel* beräknas utifrån aktuell inställning för vinklar.

Med vinkelenheten radianer och komplext rektangulärt läge:

5+3*i*−(10∠π/4) ENTER
$5-5\cdot\sqrt{2}+(3-5\cdot\sqrt{2})\cdot i$
◆ ENTER $-2.071...-4.071...\cdot i$

°, ', " — tangenten 2nd [°] (°), tangenten 2nd [′] (′), tangenten 2nd ["] (")

dd°*mm*'*ss.ss*" ⇒ *uttryck*

dd	Ett positivt eller negativt tal
mm	Ett icke-negativt tal
ss.ss	Ett icke-negativt tal

Returnerar *dd*+(*mm*/60)+(*ss.ss*/3600).

Detta format med basen 60 låter dig:

- Mata in en vinkel i grader/minuter utan att behöva ta hänsyn till den aktuella vinkelenheten.
- Ange tid i timmar/minuter/sekunder.

Med vinkelenheten grader:

25°13'17.5" ENTER 25.221...

25°30' ENTER 51/2

' (prim) — tangenten 2nd [′]

variabel'
variabel"

Anger ett primtecken i en differentialekvation. Ett enkelt tecken anger en första ordningens ekvation, två tecken anger en andra ordningens ekvation osv.

deSolve(y''=y^(⁻1/2) and y(0)=0 and y'(0)=0,t,y) ENTER
$\frac{2\cdot y^{3/4}}{3}=t$

_ (understryk.) **TI-89: tangenten** [◆] [_] **TI-92 Plus: tangenten** [2nd] [_]

uttryck_enhet

Bestämmer enhet för *uttryck*. Alla enhetsnamn måste börja med ett understrykningstecken.

Du kan använda fördefinierade enheter eller skapa egna. En lista över alla fördefinierade enheter finns i kapitlet om konstanter och mätenheter i denna handbok. Du kan trycka på:
TI-89: [2nd] [UNITS]
TI-92 Plus: [◆] [UNITS]
för att välja enheter från en meny, eller så kan du skriva in enhetsnamnen direkt.

```
3_m▸_ft ENTER          9.842…·_ft
```

Obs! När du vill skriva ▸ tryck [2nd] [▸].

---

*variabel_*

När *variabel* saknar värde antas den representera ett komplext värde. Genom förval hanteras variabeln som reell om _ saknas i variabelnamnet.

Om *variabel* har tilldelats ett värde ignoreras _ och *variabel* behåller sin ursprungliga datatyp.

Obs! Du kan lagra komplexa värden i en variabel utan att använda _. För korrekta resultat i beräkningar med **cSolve()** och **cZeros()** bör du dock använda tecknet.

Förutsatt att z är odefinierat:

```
real(z) ENTER                  z
real(z_) ENTER          real(z_)

imag(z) ENTER                  0
imag(z_) ENTER          imag(z_)
```

▸ (omvandling) **tangenten** [2nd] [▸]

uttryck_enhet1 ▸ *_enhet2* ⇒ *uttryck_enhet2*

Omvandlar ett uttryck från en enhet till en annan. Enheterna måste vara av samma kategori.

Understrykningstecknet _ bestämmer enhet. En lista över alla fördefinierade enheter finns i kapitlet om konstanter och mätenheter i denna handbok. Du kan trycka på:
TI-89: [2nd] [UNITS]
TI-92 Plus: [◆] [UNITS] för att välja enheter från en meny, eller så kan du skriva in enhetsnamnen direkt.

När du vill skriva in ett _ understrykningstecken trycker du på:
TI-89: [◆] [_]
TI-92 Plus: [2nd] [_]

Obs! Omvandling med operatorn ▸ kan inte användas för temperaturomvandlingar. Använd istället **tmpCnv()** och **ΔtmpCnv()**.

```
3_m▸_ft ENTER          9.842… _ft
```

10^() CATALOG

10^ (*uttryck1***)** ⇒ *uttryck*
10^ (*lista1***)** ⇒ *lista*

Returnerar 10 upphöjt till argumentet.

För en lista returneras 10 upphöjt till elementen i *lista1*.

```
10^1.5 ENTER                    31.622...
10^{0,-2,2,a} ENTER
```

$\{1 \quad \frac{1}{100} \quad 100 \quad 10^a\}$

10^(*kvadratiskMatris1***)** ⇒ *kvadratiskMatris*

Returnerar 10 upphöjt till *kvadratiskMatris1*. Detta är inte samma sak som att beräkna 10 upphöjt till varje enskilt element. Mer information om beräkningsmetoden finns under **cos()**.

kvadratiskMatris1 måste vara diagonaliserbar. Resultatet innehåller alltid tal i flyttalsform.

```
10^([1,2,3;4,5,6;7,8,9]) ENTER
```

$$\begin{bmatrix} 1.143\ldots E7 & 8.171\ldots E6 & 6.675\ldots E6 \\ 9.956\ldots E6 & 7.115\ldots E6 & 5.813\ldots E6 \\ 7.652\ldots E6 & 5.469\ldots E6 & 4.468\ldots E6 \end{bmatrix}$$

x^{-1} CATALOG (^-1)

uttryck1 **x**$^{-1}$ ⇒ *uttryck*
lista1 **x**$^{-1}$ ⇒ *lista*

Returnerar det inverterade värdet av argumentet.

För en lista returneras det inverterade värdet av elementen i *lista1*.

```
3.1^-1 ENTER                    .322581
{a,4,-.1,x-2}^-1 ENTER
```

$\{\frac{1}{a} \quad \frac{1}{4} \quad -10 \quad \frac{1}{x-2}\}$

---

kvadratiskMatris1 **x**$^{-1}$ ⇒ *kvadratiskMatris*

Returnerar inversen av *kvadratiskMatris1*.

kvadratiskMatris1 måste vara en icke-singulär kvadratisk matris.

```
[1,2;3,4]^-1 ENTER
[1,2;a,4]^-1 ENTER
```

$$\blacksquare \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}^{-1} \qquad \begin{bmatrix} -2 & 1 \\ 3/2 & -1/2 \end{bmatrix}$$

$$\blacksquare \begin{bmatrix} 1 & 2 \\ a & 4 \end{bmatrix}^{-1} \qquad \begin{bmatrix} \frac{-2}{a-2} & \frac{1}{a-2} \\ \frac{a}{2\cdot(a-2)} & \frac{-1}{2\cdot(a-2)} \end{bmatrix}$$

| ("with") TI-89: tangenten [|] TI-92 Plus: tangenten [2nd] [|]

uttryck **|** *booleskt uttryck1* [*and booleskt uttryck2*]...[*and booleskt uttryckN*]

Symbolen "with" (|) fungerar som en binär operator. Operanden till vänster om | är ett uttryck. Operanden till höger om | anger en eller fler relationer som är avsedda att påverka förenklingen av uttrycket. Flera relationer efter | måste förenas med ett logiskt "och" (and).

Operatorn "with" tillhandahåller tre grundläggande typer av funktioner: substitutioner, intervallbegränsningar och undantag.

```
x+1| x=3 ENTER                  4
x+y| x=sin(y) ENTER        sin(y)+y
x+y| sin(y)=x ENTER            x+y
```

Substitutioner är på formen av en likhet, som exempelvis x=3 eller y=sin(x). För att den ska vara så effektiv som möjligt bör den vänstra sidan vara en enkel variabel. *uttryck* | *variabel* = *värde* substituerar *värde* för varje förekomst av *variabel* i *uttryck*.

```
x^3-2x+7→f(x) ENTER                    Done
f(x)| x=√(3) ENTER          3^(3/2) - 2·√3 + 7
(sin(x))^2+2sin(x)-6| sin(x)=d
ENTER
                                       d^2 +2d-6
```

Intervallbegränsningar är på formen en eller flera olikheter sammansatta av logiska "och"-(and) operatorer. Intervallbegränsningar tillåter också förenkling som annars kanske skulle vara ogiltig eller inte beräkningsbar.

```
solve(x^2-1=0,x)|x>0 and x<2
ENTER
                                       x = 1
√(x)*√(1/x)|x>0 ENTER                  1
√(x)*√(1/x) ENTER                      √(I/x)·√x
```

Undantag använder relationsoperatorn "inte lika med" (/= eller ≠) för att undanta ett specifikt värde från att beräknas. Undantagen används främst för att utesluta en exakt lösning när du använder **cSolve()**, **cZeros()**, **fMax()**, **fMin()**, **solve()**, **zeros()** osv.

```
solve(x^2-1=0,x)| x≠1 ENTER            x = -1
```

→ (spara) **tangenten** STO▸

uttryck → *var*
lista → *var*
matris → *var*
uttryck → *funktionsnamn(parameter1,...)*
lista → *funktionsnamn(parameter1,...)*
matris → *funktionsnamn(parameter1,...)*

```
π/4→myvar ENTER                        π/4
2cos(x)→Y1(x) ENTER                    Done
{1,2,3,4}→Lst5 ENTER          {1 2 3 4}
[1,2,3;4,5,6]→MatG ENTER      [1 2 3; 4 5 6]
"Hello"→str1 ENTER                     "Hello"
```

Om variabeln *var* inte finns skapas *var* och den initialiseras till *uttryck*, *lista* eller *matris*.

Om *var* redan finns och den inte är låst eller skyddad, ersätts dess innehåll med *uttryck*, *lista* eller *matris*.

Tips! Om du planerar att göra symboliska beräkningar med odefinierade variabler bör du undvika att spara någonting i variabler som du ofta använder och vars namn består av en bokstav, exempelvis a, b, c, x, y, z osv.

◉ (kommentar) **Program Editor/Control-meny** eller

TI-89: [◆] [)] tangent

TI-92 Plus: [2nd] X tangent

◉ [*text*]

◉ behandlar *text* som en kommentarrad, vilken kan användas för att kommentera programinstruktioner.

◉ kan vara i början eller någon annanstans på raden. Allt som är till höger om ◉ och som sträcker sig till slutet av raden är kommentaren.

Programsegment:

```
  ⋮
:◉ Get 10 points from the Graph
  screen
:For i,1,10 ◉ This loops 10
times
  ⋮
```

0b, 0h

TI-89: [0] [alpha] [B] tang **TI-92 Plus: [0] B tang**

TI-89: [0] [alpha] [H] tang **TI-92 Plus: [0] H tang**

0b *binärtTal*
0h *hexadecimaltTal*

Anger ett binärt eller hexadecimalt värde. Binära och hexadecimala tal måste alltid matas in med prefixen 0b alternativt 0h oavsett beräkningsläge. Utan prefix hanteras talet som ett decimalt tal (bas 10).

Resultat visas enligt inställt beräkningsläge.

I decimalt läge:

```
0b10+0hF+10 [ENTER]                27
```

I binärt läge:

```
0b10+0hF+10 [ENTER]          0b11011
```

I hexadecimalt läge:

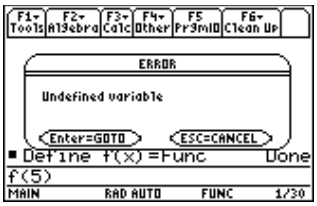
```
0b10+0hF+10 [ENTER]             0h1B
```

Referensinformation

B

TI-89 / TI-92 Plus felmeddelanden 538
Lägen 546
TI-89 / TI-92 Plus teckenkoder 551
TI-89 Tangentkoder 552
TI-92 Plus tangentkoder 555
Skriva Komplexa tal 559
Information om noggrannhet 562
Systemvariabler och reserverade namn 563
Hierarkin EOS (Equation Operating System) 564
Regressionsformler 566
Konturnivåer och algoritmer för implicit plottning 568
Runge-Kuttas metod 569

Detta appendix innehåller information om felmeddelanden och teckenkoder för TI-89 / TI-92 Plus. Det innehåller också information om hur vissa beräkningar i TI-89 / TI-92 Plus utförs.



Ytterligare information finns i appendix C. Om du till exempel har problem med att använda din TI-89 / TI-92 Plus innehåller appendix C avsnittet “Vid problem” som ger dig förslag på hur du ska åtgärda problemet.

TI-89 / TI-92 Plus felmeddelanden

Detta avsnitt listar de felmeddelanden som kan visas på grund av inmatningsfel eller interna fel. Numret till vänster om varje meddelande representerar en intern felkod som inte visas tillsammans med meddelandet. Om felet uppstår inom ett Try...EndTry block sparas felkoden i systemvariabeln *errornum*. De flesta av dessa felmeddelanden är lätta att förstå och behöver ingen extra beskrivning. För vissa felmeddelanden finns det dock en del extra information.

Felkod	Beskrivning
10	**A function did not return a value** *(Funktionen returnerade inget värde)*
20	**A test did not resolve to TRUE or FALSE** *(Ett test bestämdes inte till TRUE eller FALSE)* Generellt sett kan odefinierade variabler inte jämföras. Exempel: If a<b ger detta felmeddelande om antingen a eller b är odefinerad när If-satsen körs.
30	**Argument cannot be a folder name** *(Argument kan inte vara mappnamn)*
40	**Argument error** *(Argumentfel)*
50	**Argument mismatch** *(Fel typ av argument)* Två eller flera argument måste vara av samma datatyp. Exempel: **PtOn** *argument1*,*argument2* och **PtOn** *lista1*,*lista2* är båda giltiga medan **PtOn** *argument1*,*lista2* är ogiltigt.
60	**Argument must be a Boolean expression or integer** *(Argument måste vara ett booleskt uttryck eller ett heltal)*
70	**Argument must be a decimal number** *(Argument måste vara ett decimalt tal)*
80	**Argument must be a label name** *(Argument måste bestå av ett etikettnamn)*
90	**Argument must be a list** *(Argument måste vara en lista)*
100	**Argument must be a matrix** *(Argument måste vara en matris)*
110	**Argument must be a Pic** *(Argument måste vara en bild)*
120	**Argument must be a Pic or string** *(Argument måste vara en bild eller en sträng)*
130	**Argument must be a string** *(Argument måste vara en sträng)*
140	**Argument must be a variable name** *(Argument måste vara ett variabelnamn)* Exempel: DelVar 12 är ogiltigt eftersom ett variabelnamn inte kan vara ett tal.
150	**Argument must be an empty folder name** *(Argument måste bestå av namnet på en tom mapp)*

Felkod	Beskrivning
160	**Argument must be an expression** *(Argument måste vara ett uttryck)* Exempel: zeros(2x+3=0,x) är ogiltigt eftersom det första argumentet är en ekvation.
161	**ASAP or Exec string too long** *(ASAP eller Exec för lång)*
163	**Attribute (8-digit number) of object (8-digit number) not found** *(Attribut (8 siffror) hos objekt (8 siffror) kan inte hittas)*
165	**Batteries too low for sending/receiving product code** *(Batterierna har för låg spänning för att sända/ta emot programkod)* Sätt i nya batterier innan du tar emot/sänder programkod (baskod).
170	**Bound** *(Gränsvärdesfel)* För interaktiva grafiska matematikfunktioner som 2:Zero måste det lägre gränsvärdet vara mindre än det övre gränsvärdet för att definiera ett sökintervall.
180	**Break** *(Bryt)* Tangenten [ON] trycktes ned under en lång beräkning eller under körning av ett program.
185	**Checksum error** *(Checksumfel)*
190	**Circular definition** *(Cirkulär definition)* Detta meddelande visas för att undvika att minnet tar slut då variabler ersätts i oändlighet vid en förenkling. Exempel: a+1→a där a är en odefinierad variabel ger detta felmeddelande.
200	**Constraint expression invalid** *(Ogiltigt uttryck med operator)* Exempel: solve(3x^2−4=0, x) | x<0 or x>5 ger detta felmeddelande då de två delarna är separerade med “or” och inte “and.”
210	**Data type** *(Datatypsfel)* Ett argument är av fel datatyp.
220	**Dependent limit** *(Integrationsfel)* En integrationsgräns innehåller integrationsvariabeln. Exempel: ∫(x^2,x,1,x) är inte giltigt.
225	**Diff Eq setup** *(Felaktig inställning för differentialekvation)*
230	**Dimension** *(Dimensionsfel)* En lista eller matrisindex är ogiltig. Exempel: om listan{1,2,3,4} lagras i L1 är L1[5] ogiltigt eftersom listan L1 bara innehåller fyra element.
240	**Dimension mismatch** *(Olika dimensioner används)* Två eller flera argument måste ha samma dimension. Exempel: [1,2]+[1,2,3] är ogiltigt eftersom matriserna innehåller olika antal element.
250	**Divide by zero** *(Division med noll)*

Felkod	Beskrivning
260	**Domain error** *(Domänfel)* Ett argument måste vara inom ett visst intervall. Exempel: ans(100) är inte giltigt eftersom argument till **ans()** måste vara inom intervallet 1–99.
270	**Duplicate variable name** *(Dubblerade variabelnamn)*
280	**Else and ElseIf invalid outside of If..EndIf block** *(Else och ElseIf kan inte användas utanför If...EndIf)*
290	**EndTry is missing the matching Else statement** *(EndTry utan Else-uttryck)*
295	**Excessive iteration** *(För många iterationer)*
300	**Expected 2 or 3-element list or matrix** *(Förväntade en lista med 2 eller 3 element, eller en matris)*
307	**Flash application extension (function or program) not found** *(Flash-applikation (funktion eller program) kan inte hittas)*
308	**Falsh application not found** *(Flash-applikation kan inte hittas)*
310	**First argument of nSolve must be a univariate equation** *(Första argumentet till nSolve måste vara en ekvation med högst en obestämd variabel)* Första argumentet måste vara en ekvation och ekvationen får inte innehålla någon variabel utan värde förutom den variabel som ska beräknas. Exempel: nSolve(3x^2−4=0, x) är en giltig ekvation medan nSolve(3x^2−4, x) inte är giltig liksom nSolve(3x^2−y=0,x) inte är giltig eftersom y inte har tilldelats något värde i detta exempel.
320	**First argument of solve or cSolve must be an equation or inequality** *(Första argumentet till solve eller cSolve måste vara en ekvation)* Exempel: solve(3x^2−4, x) är inte giltigt eftersom det första argumentet inte är en ekvation.
330	**Folder** *(Mapp saknas)* Ett försök gjordes att via menyn VAR-LINK spara en variabel i en katalog som inte finns.
335	**Graph functions y1(x)...y99(x) not available in Diff Equations mode** *(Graffunktionerna y1(x)...y99(x) är inte tillgängliga i läget Diff Equations)*
345	**Inconsistent units** *(Enheterna är olika)*
350	**Index out of range** *(Index utanför giltiga värden)*
360	**Indirection string is not a valid variable name** *(Strängen består inte av ett giltigt variabelnamn)*
380	**Invalid ans()** *(Ogiltigt ans())*
390	**Invalid assignment** *(Ogiltig tilldelning)*
400	**Invalid assignment value** *(Ogiltigt tilldelningsvärde)*

Felkod	Beskrivning
405	**Invalid axes** *(Ogiltiga axlar)*
410	**Invalid command** *(Ogiltigt kommando)*
420	**Invalid folder name** *(Ogiltigt mappnamn)*
430	**Invalid for the current mode settings** *(Ogiltigt för aktuell lägesinställning)*
440	**Invalid implied multiply** *(Ogiltig implicit multiplikation)* Exempel: x(x+1) är ogiltigt medan x∗(x+1) är en korrekt syntaxe. Detta är för att undvika sammanblandning av implicit multiplikation och funktionsanrop.
450	**Invalid in a function or current expression** *(Ogiltigt i en funktion eller i det aktuella uttrycket)* Endast vissa kommandon kan användas i användardefinierade funktioner. Inmatningar som görs via Window Editor, Table Editor, Data/Matrix Editor och Solver liksom systemanrop som Lower Bound kan inte innehålla kommandon eller kolon (:). Se även “Skapa och utvärdera användardefinierade funktioner” i kapitel 5.
460	**Invalid in Custom..EndCustm block** *(Ogiltigt inom Custom...EndCustm)*
470	**Invalid in Dialog..EndDlog block** *(Ogiltigt inom Dialog...EndDlog)*
480	**Invalid in Toolbar..EndTBar block** *(Ogiltigt inom Toolbar...EndTBar)*
490	**Invalid in Try..EndTry block** *(Ogiltigt inom Try...EndTry)*
500	**Invalid label** *(Ogiltig etikett)* Namn på etiketter måste följa samma regler som för variabelnamn.
510	**Invalid list or matrix** *(Ogiltig lista eller matris)* Exempel: en lista inuti en annan lista som {2,{3,4}} är inte giltigt.
520	**Invalid outside Custom..EndCustm or ToolBar..EndTbar blocks** *(Ogiltigt utanför Custom...EndCustm eller Toolbar...EndTbar)* Exempel: kommandot **Item** anropas utanför en struktur för **Custom** eller **ToolBar**.
530	**Invalid outside Dialog..EndDlog, Custom..EndCustm, or ToolBar..EndTBar blocks** *(Ogiltigt utanför Dialog...EndDlog, Custom..EndCustm eller Toolbar...EndTbar)* Exempel: kommandot Title anropas utanför en struktur för **Dialog**, **Custom** eller **ToolBar**.
540	**Invalid outside Dialog..EndDlog block** *(Ogiltigt utanför Dialog...EndDlog)* Exempel: kommandot **DropDown** anropas utanför en struktur för **Dialog**.
550	**Invalid outside function or program** *(Ogiltigt utanför en funktion eller ett program)* Ett antal kommandon är inte giltiga utanför ett program eller en funktion. Exempel: Local kan inte användas utanför ett program eller funktion.

Felkod	Beskrivning
560	**Invalid outside Loop..EndLoop, For..EndFor, or While..EndWhile blocks** *(Ogiltigt utanför Loop...EndLoop, For...EndFor eller While...EndWhile)* Exempel: kommandot **Exit** kan bara användas innanför dessa slingor.
570	**Invalid pathname** *(Ogiltig sökväg)* Exempel: `\\var` är ogiltigt.
575	**Invalid polar complex** *(Ogiltigt polärt tal)*
580	**Invalid program reference** *(Ogiltig programreferens)* Program kan inte refereras inuti funktioner eller uttryck som `1+p(x)` där `p` är ett program.
590	**Invalid syntax block** *(Ogiltig syntax)* Ett **Dialog..EndDlog** block är tomt eller har mer än en titel. Ett **Custom..EndCustm** block kan inte innehålla PIC variabler och enheter måste föregås av en titel. Ett **Toolbar..EndTBar** block måste ha ett andra argument om inga enheter följer eller så måste enheter ha ett andra argument och föregås av en titel.
600	**Invalid table** *(Ogiltig tabell)*
605	**Invalid use of units** *(Ogiltig användning av måttenheter)*
610	**Invalid variable name in a Local statement** *(Ogiltigt variabelnamn i ett lokalt uttryck)*
620	**Invalid variable or function name** *(Ogiltigt variabel- eller funktionsnamn)*
630	**Invalid variable reference** *(Ogiltig variabelreferens)*
640	**Invalid vector syntax** *(Ogiltig syntax för vektor)*
650	**Link transmission** *(Överföringen slutfördes inte)* Överföringen mellan två enheter slutfördes inte. Kontrollera att kabeln mellan enheterna är ordentligt isatt.
665	**Matrix not diagonalizable** *(Matrisen är inte diagonaliserbar)*
670 673	**Memory** *(Minnesfel)* Beräkningen krävde mer minne än vad som fanns tillgängligt. Om du får detta felmeddelande då du kör ett stort program kan du vara tvungen att dela upp programmet i flera mindre program eller funktioner (där ett program eller funktion anropar ett annat).
680	**Missing (** *(Saknar vänsterparentes)*

Felkod	Beskrivning
690	**Missing)** *(Saknar högerparentes)*
700	**Missing "** *(Saknar ")*
710	**Missing]** *(Saknar])*
720	**Missing }** *(Saknar })*
730	**Missing start or end of block syntax** *(Saknar start eller slut i blockets syntax)*
740	**Missing Then in the If..EndIf block** *(Saknar Then i If...EndIf-satsen)*
750	**Name is not a function or program** *(Namnet är inte en funktion eller ett program)*
765	**No functions selected** *(Inga funktioner är valda)*
780	**No solution found** *(Ingen lösning kan hittas)* Om du använder de interaktiva matematikfunktionerna (F5:Math) i applikationen Graph kan detta fel uppstå. Om du t ex försöker hitta en inflexionspunkt, t.ex. på parabeln $y1(x)=x^2$, som inte finns, så uppstår detta fel.
790	**Non-algebraic variable in expression** *(Uttrycket innehåller en variabel som inte är av algebraisk typ)* Om a är namnet på en variabel av typ PIC, GDB, MAC, FIG etc. så är a+1 inte giltigt. Använd ett annat variabelnamn i uttrycket eller ta bort variabeln.
800	**Non-real result** *(Resultatet är inte ett reellt tal)* Exempel: om enhet är angiven till inställningen REAL för Complex Format så är ln(⁻2) inte giltigt.
810	**Not enough memory to save current variable. Please delete unneeded variables on the Var-Link screen and re-open editor as current OR re-open editor and use F1 8 to clear editor.** *(Minnet räcker inte för att spara den aktuella variabeln. Ta bort oanvända varibler i fönstret Var-Link och öppna editorn igen som aktuell ELLER öppna editorn igen och använd F1-8 för att radera editorn.)* Detta felmeddelande genereras vid lite ledigt minne i Data/Matrix Editor.
830	**Overflow** *(Värdefel)*
840	**Plot setup** *(Inställningar för plottning)*

Felkod	Beskrivning
850	**Program not found** *(Programmet kan inte hittas)* En referens till ett program från ett annat program kunde inte hittas i angiven sökväg vid körning.
860	**Recursion is limited to 255 calls deep** *(Rekursiva anrop är begränsat till 255 anrop)*
870	**Reserved name or system variable** *(Reserverat namn eller systemvariabel)*
875	**ROM-resident routine not available** *(ROM-baserad rutin kan inte hittas)*
880	**Sequence setup** *(Sekvensinställning)*
885	**Signature error** *(Signaturfel)*
890	**Singular matrix** *(Singulär matris)*
895	**Slope fields need one selected function and are used for 1st-order equations only** *(Lutningsfält kräver en markerad funktion och kan endast användas för första ordningens ekvationer)*
900	**Stat** *(Stat)*
910	**Syntax** *(Syntax)* Strukturen hos inmatningen är inkorrekt. Exempel: x+−y (x plus minus y) är ogiltig medan x+ ⁻y (x plus negative y) är korrekt.
930	**Too few arguments** *(För få argument)* Uttrycket eller ekvationen saknar ett eller flera argument. Exempel: *d*(f(x)) är ogiltigt medan *d*(f(x),x) är den korrekta syntaxen.
940	**Too many arguments** *(För många argument)* Uttrycket eller ekvationen har för många argument och kan inte utvärderas.
950	**Too many subscripts** *(För många underskript)*
955	**Too many undefined variables** *(För många odefinierade variabler)*
960	**Undefined variable** *(Odefinierad variabel)*
965	**Unlicensed product software or Flash application** *(Olicensierad programkod eller Flash-applikation)*
970	**Variable in use so references or changes are not allowed** *(Variabeln används redan så referenser eller ändringar kan inte ske)*
980	**Variable is locked, protected, or archived** *(Variabeln är låst, skyddad eller lagrad i användarminnet)*
990	**Variable name is limited to 8 characters** *(Variabelnamn kan bestå av högst 8 tecken)*
1000	**Window variables domain** *(Domänen Window-variabler)*
1010	**Zoom** *(Zooma)*

Felkod	Beskrivning
	Warning: ∞^0 or undef^0 replaced by 1 *(Varning: ∞^0 eller undef^0 ersatt av 1)*
	Warning: 0^0 replaced by 1 *(Varning: 0^0 ersatt av 1)*
	Warning: 1^∞ or 1^undef replaced by 1 *(Varning: 1^∞ eller 1^undef ersatt av 1)*
	Warning: cSolve may specify more zeros *(Varning: cSolve kan ange fler nollställen)*
	Warning: May produce false equation *(Varning: Kan orsaka felaktiga ekvationer)*
	Warning: Expected finite real integrand *(Varning: Förväntade finit reell integral)*
	Warning: May not be fully simplified *(Varning: Är eventuellt inte helt förenklat)*
	Warning: More solutions may exist *(Varning: Fler lösningar kan finnas)*
	Warning: May introduce false solutions *(Varning: Kan orsaka felaktiga lösningar)*
	Warning: Operation may lose solutions *(Varning: Operationen kan förlora lösningar)*
	Warning: Requires & returns 32 bit value *(Varning: Kräver och returnerar 32-bitars värde)*
	Warning: Overflow replaced by ∞ or -∞ *(Varning: Värdefel ersatt av ∞ eller -∞)*
	Warning: Questionable accuracy *(Varning: Otillräcklig noggranhet)*
	Warning: Questionable solution *(Varning: Tveksam lösning)*
	Warning: Solve may specify more zeros *(Varning: Solve kan ange fler nollställen)*
	Warning: Trig argument too big to reduce *(Varning: Trigonometriskt argument för stort för att reducera)*

I det här avsnittet beskrivs de lägen som finns i TI-89 / TI-92 Plus och här listas tillgängliga inställningar för varje läge. Dessa lägesinställningar visas när du trycker på MODE.

Graph

Anger vilka typer av grafer du kan plotta.

1:FUNCTION	y(x)-funktioner (kapitel 6)
2:PARAMETRIC	x(t) och y(t) -parameterekvationer (kapitel 7)
3:POLAR	r(θ)-polära ekvationer (kapitel 8)
4:SEQUENCE	u(n) -talföljder (kapitel 9)
5:3D	z(x,y) -3D-ekvationer (kapitel 10)
6:DIFF EQUATIONS	y'(t) differentialekvationer (kapitel 11)

Obs! Om du använder ett delat fönster med Number of Graphs = 2 är Graph för den översta eller vänstra delen av fönstret och Graph 2 är för den nedersta eller högra delen.

Current Folder

Anger den aktuella mappen. Du kan använda flera mappar med unika konfigurationer av variabler, grafdatabaser, program osv.

***Obs!** Detaljerad information om hur du använder mappar finns i kapitel 5.*

1:main	Standardmapp som ingår i TI-89 / TI-92 Plus.
2: — (egna mappar)	Övriga mappar är endast tillgängliga om de har skapats av en användare.

Display Digits

Väljer antalet siffror som visas. Dessa decimalinställningar påverkar bara hur resultaten visas; du kan mata in ett tal i vilket format som helst.

Internt behålls decimaltal med 14 signifikanta siffror. För visningsändamål rundas sådana tal av till maximalt 12 signifikanta siffror.

1:FIX 0 2:FIX 1 … D:FIX 12	Result visas alltid med angivna antalet decimaler.
E:FLOAT	Antalet decimaler varierar, beroende på resultatet.
F:FLOAT 1 G:FLOAT 2 … Q:FLOAT 12	Om heltalsdelen har mer än det angivna antalet siffror rundas resultatet av och visas i grundpotensform. I exempelvis FLOAT 4 visas 12345. som 1.235E4

Angle

Anger i vilken vinkelenhet som vinkelvärden tolkas och visas i trigonometriska funktioner och vid polära/rektangulära omvandlingar.

1:RADIAN
2:DEGREE

Exponential Format

Anger vilket notationsformat som ska användas. Dessa format påverkar bara hur ett resultat visas; du kan mata in tal i vilket format som helst. Numeriska resultat kan visas med upp till 12 siffror och med en 3-siffrig exponent.

1:NORMAL	Uttrycker tal i standardformat. Exempel: 12345.67
2:SCIENTIFIC	Grundpotensform. Uttrycker tal i två delar: • De signifikanta siffrorna visas med en siffra till vänster om decimalkommat. • Potenser av 10 visas till höger om E. Exempel: 1.234567E4 betyder 1.234567×10^4
3:ENGINEERING	Liknar grundpotensform, men har följande undantag: • Talet kan ha en, två eller tre siffror före decimalkommat. • Exponenten till potensen av 10 är i multipler om tre. Exempel: 12.34567E3 betyder 12.34567×10^3

Obs! Om du väljer NORMAL, men resultatet inte kan visas i det antal siffror som anges av Display Digits, visas resultatet i grundpotensform. Om Display Digits = FLOAT kommer grundpotensform att användas för exponenter som är större än eller lika med 12 och exponenter som är mindre än eller lika med −4.

Complex Format

Anger om komplexa resultat ska visas och i så fall i vilket format.

1:REAL	Visar inte komplexa format. (Om ett resultat är ett komplext tal och inmatningen inte innehåller den imaginära enheten ***i***, visas ett felmeddelande.)
2:RECTANGULAR	Visar komplexa tal på formatet: a+b***i***
3:POLAR	Visar komplexa tal på formatet: $re^{i\theta}$

Vector Format

Anger hur vektorer med 2 eller 3 element ska visas. Du kan mata in vektorer i vilket som helst av koordinatsystemen.

1:RECTANGULAR	Koordinater är i termer av x, y och z. [3,5,2] motsvarar exempelvis x = 3, y = 5 och z = 2.
2:CYLINDRICAL	Koordinater är i termer av r, θ, and z. [3,∠45,2] motsvarar exempelvis r = 3, θ = 45, and z = 2.
3:SPHERICAL	Koordinater är i termer av r, θ, and ϕ. [3, ∠45, ∠90] motsvarar exempelvis r = 3, θ = 45, and ϕ = 90.

Pretty Print

Anger hur resultaten visas i grundfönstret.

1:OFF	Resultat visas i en linjär, endimensionell form. Exempelvis π^2, π/2 eller √((x-3)/x)
2:ON	Resultat visas i konventionellt matematiskt format. Exempelvis π^2, $\frac{\pi}{2}$, eller $\sqrt{\frac{x-3}{x}}$

Obs! En fullständig beskrivning av dessa inställningar finns i "Format på visade resultat" i kapitel 2.

Split Screen

Låter dig dela fönstret i två delar. Du kan exempelvis visa en graf och Y= Editor på samma gång (kapitel 14).

1:FULL	Fönstret är inte delat.
2:TOP-BOTTOM	Applikationerna visas i två fönster som är ovanför varandra.
3:LEFT-RIGHT	Applikationerna visas i två fönster som är bredvid varandra.

För att bestämma vilken information som ska visas i ett delat fönster och hur den ska visas använder du detta läge tillsammans med andra lägesinställningar, som exempelvis Split 1 App, Split 2 App, och Number of Graphs, och Split Screen Ratio. (Split Screen Ratio finns bara på TI-92 Plus.)

Split 1 App och Split 2 App

Anger vilket program som visas i fönstret.

- Vid ett fönster i fullständig storlek är endast Split 1 App aktivt.
- Vid ett delat fönster är Split 1 App längst upp eller längst till vänster i fönstret och Split 2 App är längst ned eller till höger.

De tillgängliga alternativen är de som visas när du trycker på ⊙ från fönstret Page 2 eller när du trycker på [APPS]. Du måste ha olika applikationer i varje fönster, såvida du inte är i tvågrafsläge.

Number of Graphs

Anger huruvida båda delarna i ett delat fönster kan visa grafer på samma gång.

1	Endast en del kan visa grafer.
2	Båda delarna kan visa ett fristående graffönster (inställningen Graph eller Graph 2) med oberoende inställningar.

Graph 2

Anger vilken typ av grafer du kan plotta för den andra grafen i ett delat fönster i tvågrafsläge. Detta är endast aktivt när Number of Graphs = 2. I denna tvågrafsinställning ställer Graph in typen av graf för den översta eller vänstra delen av det delade fönstret, och Graph 2 ställer in den understa eller högra delen. De tillgängliga alternativen är samma som för Graph.

Split Screen Ratio (endast TI-92 Plus)

Anger storleksförhållandet mellan de två delarna av en delad skärm.

1:1	Skärmen delas i två lika delar.
1:2	Den undre eller högra delen är ungefär dubbelt så stor som den övre eller vänstra delen.
2:1	Den övre eller vänstra delen är ungefär dubbelt så stor som den undre eller högra delen.

Exact/Approx

Anger hur bråkuttryck och symboliska uttryck beräknas och visas. Genom att behålla rationella och symboliska former i inställningen EXACT, ökas precisionen genom att eliminera de flesta avrundningsfel.

1:AUTO	Använder inställningen EXACT i de flesta fall. Inställningen APPROXIMATE används dock om inmatningen innehåller ett decimalkomma.
2:EXACT	Visar icke-heltalsresultat i sin rationella eller symboliska form.
3:APPROXIMATE	Visar numeriska resultat i decimal form.

Obs! En fullständig beskrivning av dessa inställningar finns i "Format på visade resultat" i kapitel 2.

Base

Låter dig genomföra beräkningar med inmatning av värden i decimal, binär eller hexadecimal form.

1:DEC	Decimala tal består av tecknen 0 - 9 med basen 10.
2:HEX	Hexadecimala tal består av tecknen 0 - 9 och A - F med basen 16.
3:BIN	Binära tal består av tecknen 0 och 1 med basen 2.

Unit System

Bestämmer enhetsbeteckning för värden i ett uttryck, t ex 6_m * 4_m eller 23_m/_s * 10_s. Du kan även konvertera värden från en enhet till en annan och skapa egna enheter.

1:SI	Välj SI om du vill använda det metriska systemet för mätvärden.
2:ENG/US	Välj ENG/US om du inte vill använda det metriska systemet för mätvärden.
3:CUSTOM	Låter dig välja egna förvalda enheter.

Custom Units

Låter dig välja egna standardvärden. Inställningen är gråmarkerad tills du väljer Unit System, 3:CUSTOM.

Språk

Med denna funktion kan du anpassa din TI-89 / TI-92 Plus till ett av flera språk, beroende på vilka språkanpassningsprogram du har installerat.

1:English	Förvalt språk som är inkluderat i baskoden i TI-89 / TI-92 Plus.
2: — (language Flash applicatios)	Alternativa språk finns bara tillgängliga om de installerats.

TI-89 / TI-92 Plus teckenkoder

Med funktionen **char()** kan du mata in valfritt tecken genom att ange tecknets numeriska värde. Exempel: om du vill visa tecknet ◆ i fönstret Program I/O anger du `Disp char(127)`. Du kan använda **ord()** när du vill hitta det numeriska värdet för ett tecken. Exempel: `ord("A")` returnerar 65.

1.	SOH	42.	*	83.	S	124.	|	165.	¥	206.	Î	247.	÷
2.	STX	43.	+	84.	T	125.	}	166.	¦	207.	Ï	248.	ø
3.	ETX	44.	,	85.	U	126.	~	167.	§	208.	Ð	249.	ù
4.	EOT	45.	−	86.	V	127.	◆	168.	√	209.	Ñ	250.	ú
5.	ENQ	46.	.	87.	W	128.	α	169.	◐	210.	Ò	251.	û
6.	ACK	47.	/	88.	X	129.	β	170.	ª	211.	Ó	252.	ü
7.	BELL	48.	0	89.	Y	130.	Γ	171.	«	212.	Ô	253.	ý
8.	BS	49.	1	90.	Z	131.	γ	172.	¬	213.	Õ	254.	þ
9.	TAB	50.	2	91.	[	132.	Δ	173.	-	214.	Ö	255.	ÿ
10.	LF	51.	3	92.	\	133.	δ	174.	®	215.	×		
11.	⇗	52.	4	93.	]	134.	ε	175.	¯	216.	Ø		
12.	FF	53.	5	94.	^	135.	ζ	176.	°	217.	Ù		
13.	CR	54.	6	95.	_	136.	θ	177.	±	218.	Ú		
14.	🔒	55.	7	96.	`	137.	λ	178.	²	219.	Û		
15.	✓	56.	8	97.	a	138.	ξ	179.	³	220.	Ü		
16.	▪	57.	9	98.	b	139.	Π	180.	⁻¹	221.	Ý		
17.	◀	58.	:	99.	c	140.	π	181.	µ	222.	Þ		
18.	▶	59.	;	100.	d	141.	ρ	182.	¶	223.	ß		
19.	▲	60.	<	101.	e	142.	Σ	183.	•	224.	à		
20.	▼	61.	=	102.	f	143.	σ	184.	⁺	225.	á		
21.	←	62.	>	103.	g	144.	τ	185.	¹	226.	â		
22.	→	63.	?	104.	h	145.	φ	186.	º	227.	ã		
23.	↑	64.	@	105.	i	146.	ψ	187.	»	228.	ä		
24.	↓	65.	A	106.	j	147.	Ω	188.	*d*	229.	å		
25.	◖	66.	B	107.	k	148.	ω	189.	∫	230.	æ		
26.	◗	67.	C	108.	l	149.	E	190.	∞	231.	ç		
27.	⬆	68.	D	109.	m	150.	*e*	191.	¿	232.	è		
28.	∪	69.	E	110.	n	151.	***i***	192.	À	233.	é		
29.	∩	70.	F	111.	o	152.	ʳ	193.	Á	234.	ê		
30.	⊂	71.	G	112.	p	153.	T	194.	Â	235.	ë		
31.	∈	72.	H	113.	q	154.	x̄	195.	Ã	236.	ì		
32.	SPACE	73.	I	114.	r	155.	ȳ	196.	Ä	237.	í		
33.	!	74.	J	115.	s	156.	≤	197.	Å	238.	î		
34.	"	75.	K	116.	t	157.	≠	198.	Æ	239.	ï		
35.	#	76.	L	117.	u	158.	≥	199.	Ç	240.	ð		
36.	$	77.	M	118.	v	159.	∠	200.	È	241.	ñ		
37.	%	78.	N	119.	w	160.	…	201.	É	242.	ò		
38.	&	79.	O	120.	x	161.	¡	202.	Ê	243.	ó		
39.	'	80.	P	121.	y	162.	¢	203.	Ë	244.	ô		
40.	(	81.	Q	122.	z	163.	£	204.	Ì	245.	õ		
41.	)	82.	R	123.	{	164.	¤	205.	Í	246.	ö		

TI-89 Tangentkoder

Funktionen **getKey()** returnerar ett tal som värde som motsvarar den sista tangenten som trycktes ned, enligt de tabeller som visas i detta avsnitt. Om ditt program exempelvis innehåller en **getKey()**-funktion, returneras värdet 273 om du trycker på [2nd] [F6].

Tabell 1: Tangentkoder för primärtangenter

Tangent	Ändringstangent									
	Ingen		[↑]		[2nd]		[◆]		[alpha]	
	Assoc.	Värde	Assoc.	Värde	Assoc.	Värde	Assoc.	Värde	Assoc.	Värde
[F1]	F1	268	F1	268	F6	273	Y=	8460	F1	268
[F2]	F2	269	F2	269	F7	274	WINDOW	8461	F2	269
[F3]	F3	270	F3	270	F8	275	GRAPH	8462	F3	270
[F4]	F4	271	F4	271	F4	271	TblSet	8463	F4	271
[F5]	F5	272	F5	272	F5	272	TABLE	8464	F5	272
[◆]			COPY	24576	CUT	12288				
[alpha]					a-lock					
[ESC]	ESC	264	ESC	264	QUIT	4360	PASTE	8456	ESC	264
[APPS]	APPS	265	APPS	265	SWITCH	4361		8457	APPS	265
[HOME]	HOME	277	HOME	277	CUST	4373	HOME	277	HOME	277
[MODE]	MODE	266	MODE	266	▸	18	_	95	MODE	266
[CATALOG]	CATLG	278	CATLG	278	*i*	151	∞	190	CATLG	278
[←]	BS	257	BS	257	INS	4353	DEL	8449	BS	257
[CLEAR]	CLEAR	263	CLEAR	263	CLEAR	263		8455	CLEAR	263
[X]	x	120	X	88	LN	4184	e^x	8280	x	120
[Y]	y	121	Y	89	SIN	4185	SIN^{-1}	8281	y	121
[Z]	z	122	Z	90	COS	4186	COS^{-1}	8282	z	122
[T]	t	116	T	84	TAN	4180	TAN^{-1}	8276	t	116
[^]	^	94	^	94	π	140	θ	136	^	94
[|]	|	124	F	70	°	176	Format d/b	8316	f	102
[(]	(	40	B	66	{	123			b	98
[)]	)	41	C	67	}	125	◉	169	c	99
[,]	,	44	D	68	[	91		8236	d	100
[÷]	/	47	E	69	]	93	!	33	e	101
[×]	*	42	J	74	√	4138	&	38	j	106
[−]	-	45	O	79	VAR-LNK	4141	Contr. -		o	111
[+]	+	43	U	85	CHAR	4139	Contr. +		u	117

Tabell 1: Tangentkoder för primärtangenter (fortsättning)

Tangent	Ändringstangent									
	Ingen		↑		2nd		◆		alpha	
	Assoc.	Värde	Assoc.	Värde	Assoc.	Värde	Assoc.	Värde	Assoc.	Värde
ENTER	CR	13	CR	13	ENTRY	4109	APPROX	8205	CR	13
STO▸	STO▸	258	P	80	RCL	4354	@	64	p	112
=	=	61	A	65	'	39	≠	157	a	97
EE	EE	149	K	75	∠	159	SYMB	8341	k	107
(-)	-	173	SPACE	32	ANS	4372		8365	SPACE	32
.	.	46	W	87	>	62	≥	158	w	119
0	0	48	V	86	<	60	≤	156	v	118
1	1	49	Q	81	"	34		8241	q	113
2	2	50	R	82	\	92		8242	r	114
3	3	51	S	83	UNITS	4147		8243	s	115
4	4	52	L	76	:	58		8244	l	108
5	5	53	M	77	MATH	4149		8245	m	109
6	6	54	N	78	MEM	4150		8246	n	110
7	7	55	G	71	∫	4151		8247	g	103
8	8	56	H	72	d	4152		8248	h	104
9	9	57	I	73	;	59		8249	i	105

Tabell 2: Piltangenter (inklusive diagonala rörelser)

Piltangent	Normal	↑	2nd	◆	alpha
⊖	338	16722	4434	8530	33106
⊕	340	16724	4436	8532	33108
⊖	344	16728	4440	8536	33112
⊕	337	16721	4433	8529	33105
⊖ och ⊕	339	16723	4435	8531	33107
⊖ och ⊕	342	16726	4438	8534	33110
⊖ och ⊕	345	16729	4441	8537	33113
⊖ och ⊕	348	16732	4444	8540	33116

Tabell 3: Grekiska tecken (föregås av [◆] [(])

Tangent	Andra modifierare			
	[alpha]		[↑]	
	Assoc.	**Värde**	**Assoc.**	**Värde**
[=] [A]	α	128		
[(] [B]	β	129		
[,] [D]	δ	133	Δ	132
[÷] [E]	ε	134		
[|] [F]	ϕ	145		
[7] [G]	γ	131	Γ	130
[4] [L]	λ	137		
[5] [M]	μ	181		
[STO▸] [P]	π	140	Π	139
[2] [R]	ρ	141		
[3] [S]	σ	143	Σ	142
[T] [T]	τ	144		
[.] [W]	ω	148	Ω	147
[X]	ξ	138		
[Y]	ψ	146		
[Z]	ζ	135		

TI-92 Plus tangentkoder

Funktionen **getKey()** returnerar ett tal som värde som motsvarar den sista tangenten som trycktes ned, enligt de tabeller som visas i detta avsnitt. Om ditt program exempelvis innehåller en **getKey()**-funktion, returneras värdet 268 om du trycker på [2nd] [F1].

Tabell 1: Tangentkoder för primärtangenter

Tangent	Ändringstangent							
	Ingen		[↑]		[2nd]		[♦]	
	Assoc.	Värde	Assoc.	Värde	Assoc.	Värde	Assoc.	Värde
[F1]	F1	268	F1	268	F1	268		8460
[F2]	F2	269	F2	269	F2	269		8461
[F3]	F3	270	F3	270	F3	270		8462
[F4]	F4	271	F4	271	F4	271		8463
[F5]	F5	272	F5	272	F5	272		8464
[F6]	F6	273	F6	273	F6	273		8465
[F7]	F7	274	F7	274	F7	274		8466
[F8]	F8	275	F8	275	F8	275		8467
[MODE]	MODE	266	MODE	266	MODE	266		8458
[CLEAR]	CLEAR	263	CLEAR	263	CLEAR	263		8455
[LN]	LN	262	LN	262	e^x	4358		8454
[ESC]	ESC	264	ESC	264	QUIT	4360		8456
[APPS]	APPS	265	APPS	265	SWITCH	4361		8457
[ENTER]	CR	13	CR	13	ENTRY	4109	APPROX	8205
[SIN]	SIN	259	SIN	259	SIN^{-1}	4355		8451
[COS]	COS	260	COS	260	COS^{-1}	4356		8452
[TAN]	TAN	261	TAN	261	TAN^{-1}	4357		8453
[^]	^	94	^	94	π	140		8286
[(]	(	40	(	40	{	123		8232
[)]	)	41	)	41	}	125		8233
[,]	,	44	,	44	[	91		8236
[÷]	/	47	/	47	]	93		8239
[×]	*	42	*	42	Ö	4138		8234
[−]	-	45	-	45	VAR-LNK	4141	Contrast −	
[+]	+	43	+	43	CHAR	4139	Contrast +	
[STO▸]	STO▸	258	STO▸	258	RCL	4354		8450
SPACE		32		32		32		8224
[=]	=	61	=	61	\	92		8253
[←]	BS	257	BS	257	INS	4353	DEL	8449
[θ]	θ	136	θ	136	:	58		8328
[(-)]	-	173	-	173	ANS	4372		8365
[.]	.	46	.	46	>	62		8238

Tabell 1: Tangentkoder för primärtangenter (fortsättning)

Tangent	Ändringstangent							
	Ingen		↑		2nd		◆	
	Assoc.	Värde	Assoc.	Värde	Assoc.	Värde	Assoc.	Värde
0	0	48	0	48	<	60		8240
1	1	49	1	49	E	149		8241
2	2	50	2	50	CATLG	4146		8242
3	3	51	3	51	CUST	4147		8243
4	4	52	4	52	Σ	4148		8244
5	5	53	5	53	MATH	4149		8245
6	6	54	6	54	MEM	4150		8246
7	7	55	7	55	VAR-LNK	4151		8247
8	8	56	8	56	∫	4152		8248
9	9	57	9	57	δ	4153		8249
A	a	97	A	65	Table 3			8257
B	b	98	B	66	‘	39		8258
C	c	99	C	67	Table 4		COPY	8259
D	d	100	D	68	°	176		8260
E	e	101	E	69	Table 5		WINDOW	8261
F	f	102	F	70	∠	159	FORMAT	8262
G	g	103	G	71	Table 6			8263
H	h	104	H	72	&	38		8264
I	i	105	I	73	¡	151		8265
J		106	J	74	¥	190		8266
K	k	107	K	75	|	124	KEY	8267
L	l	108	L	76	“	34		8268
M	m	109	M	77	;	59		8269
N	n	110	N	78	Table 7		NEW	8270
O	o	111	O	79	Table 8		OPEN	8271
P	p	112	P	80	_	95	UNITS	8272
Q	q	113	Q	81	?	63	HOME	8273
R	r	114	R	82	@	64	GRAPH	8274
S	s	115	S	83	β	223	SAVE	8275
T	t	116	T	84	#	35	TblSet	8276
U	u	117	U	85	Table 9			8277
V	v	118	V	86	≠	157	PASTE	8278
W	w	119	W	87	!	33	Y=	8279
X	x	110	X	88	◉	169	CUT	8280
Y	y	111	Y	89	▸	18	TABLE	8281
Z	z	112	Z	90	Caps Lock			8282

Tabell 2: Piltangenter

Piltangent	Normal	⇧	2nd	◆	⊙
⊙	338	16722	4434	8530	33106
⊙	342	16726	4438	8534	33110
⊙	340	16724	4436	8532	33108
⊙	348	16732	4444	8540	33116
⊙	344	16728	4440	8536	33112
⊙	345	16729	4441	8537	33113
⊙	337	16721	4433	8529	33105
⊙	339	16723	4435	8531	33107

Obs! Ändringstangenten ⊙ påverkar bara piltangenterna.

Tabell 3: Tecken med grav accent (föregås av 2nd A)

Tangent	Assoc.	Normal	⇧
A	à	224	192
E	è	232	200
I	ì	236	204
O	ò	242	210
U	ù	249	217

Tabell 4: Cedilla-tecken (föregås av 2nd C)

Tangent	Assoc.	Normal	⇧
C	ç	231	199

Tabell 5: Akut accent tecken (föregås av 2nd E)

Tangent	Assoc.	Normal	⇧
A	á	225	193
E	é	233	201
I	í	237	205
O	ó	243	211
U	ú	250	218
Y	ý	253	221

Tabell 6: Grekiska tecken (föregås av 2nd G)

Tangent	Assoc.	Normal	↑
A	α	128	
B	β	129	
D	δ	133	132
E	ε	134	
F	ϕ	145	
G	γ	131	130
L	λ	137	
M	μ	181	
P	π	140	139
R	ρ	141	
S	σ	143	142
T	τ	144	
W	ω	148	147
X	ξ	138	
Y	ψ	146	
Z	ζ	135	

Tabell 7: Tilde tecken (föregås av 2nd N)

Tangent	Assoc.	Normal	↑
N	ñ	241	209
O	õ	245	

Tabell 8: Cirkumflex tecken (föregås av 2nd O)

Tangent	Assoc.	Normal	↑
A	â	226	194
E	ê	234	202
I	î	238	206
O	ô	244	212
U	û	251	219

Tabell 9: Omljud tecken (föregås av 2nd U)

Tangent	Assoc.	Normal	↑
A	ä	228	196
E	ë	235	203
I	ï	239	207
O	ö	246	214
U	ü	252	220
Y	ÿ	255	

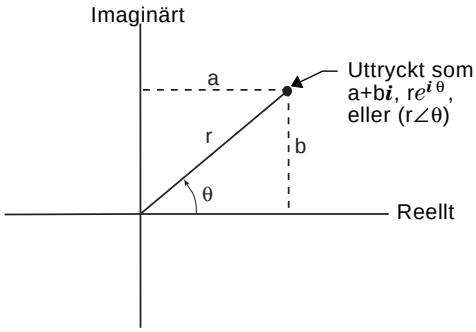
Du kan skriva in komplexa tal i polär form (r∠θ), där r motsvarar magnituden och θ vinkeln, eller i den polära formen $re^{i\theta}$.
Du kan även skriva in komplexa tal i rektangulär form a+b*i*.

Översikt över komplexa tal

Ett komplext tal har reella och imaginära komponenter som identifierar en punkt i det komplexa planet. Dessa komponenter mäts längs de reella och imaginära axlarna, som liknar x- och y-axlarna i det reella planet.

Punkten kan beskrivas i rektangulär form eller i någon av de två polära formerna.

Symbolen ***i*** identifierar det imaginära talet $\sqrt{-1}$.

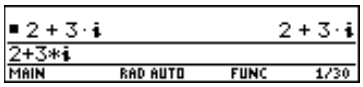


Vilken form du kan använda beror på den aktuella inställningen för Angle. Se nedan.

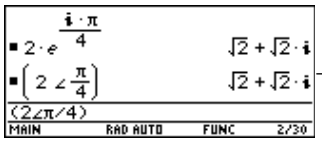
Du kan använda formen:	När vinkelinställningen är:
a+b*i*	Radian eller Degree
$re^{i\theta}$	endast Radian (Med vinkelinställningen Degree orsakar detta ett fel av typen Domain error.)
(r∠θ)	Radian eller Degree

Använd följande metoder då du vill mata in komplexa tal.

Om du vill mata in den:	Använder du tangentföljden:
Rektangulära formen a+b*i*	Substituera värden eller variabelnamn mot a och b i uttrycket nedan. a [+] b [2nd] [i] Exempel:



Obs! *Skriv in symbolen* ***i*** *med* [2nd] [i], *inte genom att skriva in ett alfabetiskt i.*

Om du vill mata in den:	Använder du tangentföljden:
Polära formen $re^{i\theta}$ – eller – (r∠θ) Parenteser krävs för formen (r∠θ).	Ersätt värdena eller variabelnamnen för r och θ, där θ beräknas enligt den aktuella vinkelinställningen. **TI-89:** [alpha] [R] [◆] [e^x] [2nd] [i] [◆] [θ] [)] – eller – [(] [alpha] [R] [2nd] [∠] [◆] [θ] [)] **TI-92 Plus:** R [2nd] [e^x] [2nd] [i] [θ] [)] – eller – [(] R [2nd] [∠] [θ] [)] Exempel:  Resultaten visas i rektangulär form, men du kan om du vill välja polär form.

Viktigt: *Använd inte den polära formen $re^{i\theta}$ med vinkelinställningen* Degree*. Detta orsakar ett fel av typen* Domain error*.*

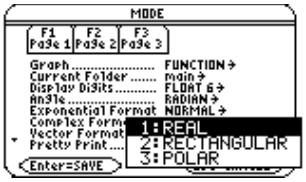
Obs! *Skriv in symbolen e genom att trycka:*
TI-89: [◆] [e^x].
TI-92 Plus: [2nd] [e^x]
Du kan inte skriva in ett alfabetiskt e.

Tip! *Om du vill mata in symbolen ∠, press* [2nd] [∠].

Tip! *Om du vill ange θ i grader för (r∠θ), kan du mata in en °-symbol (som 45°). Mata in °-symbolen genom att trycka på* [2nd] [°]*. Du kan inte använda grader för $re^{i\theta}$.*

Visa resultat i komplext format

Använda [MODE] för att ställa in läget Complex Format till en av tre inställningar.



Du kan när som helst mata in ett komplext tal, oavsett inställningen för läget Complex Format. Lägesinställningen anger dock hur resultaten visas.

Om inställningen för läget Complex Format är:	Händer följande:
REAL	Visar inte komplexa resultat om du inte: • Anger ett komplext tal. – eller – • Använder en komplex funktion som **cFactor()**, **cSolve()**, eller **cZeros()**. Om komplexa resultat visas är det i formen a+bi eller re^i.
RECTANGULAR	Komplexa resultat visas som a+bi.
POLAR	Komplexa resultat visas som: • r$e^{i\theta}$ om Angle mode = Radian – eller – • (r∠θ) om Angle mode = Degree

Obs! *Du kan mata in komplexa tal i valfri form (eller en blandning av former) beroende på aktuell* vinkelinställning*.*

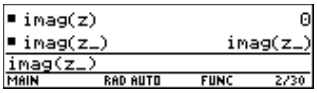
Använda komplexa variabler i symboliska beräkningar

Oavsett inställningen för läget Complex Format behandlas alla odefinierade variabler som reella tal. Om du vill utföra komplexa symboliska analyser måste du definiera en komplex variabel. Exempel:

Metod 1: Använd ett understrykningstecken _ (**TI-89:** ◆ [_] **TI-92 Plus:** 2nd [_]) som det sista tecknet i variabelnamnet för att ange att det är ett komplext tal. T ex:

Obs! *För bästa resultat med beräkningar som* **cSolve()** *och* **cZeros()** *bör du använda metod 1.*

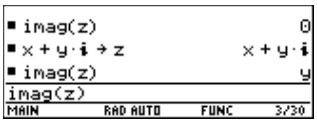
z_ hanteras som en komplex variabel (om z redan existerar bibehåller den sin tidigare datatyp).



Metod 2: Definiera en komplex variabel. T ex:

x+y*i*→z

I detta fall hanteras z som en komplex variabel.



Komplexa tal och vinkelinställningar

Vinkelinställningen Radian rekommenderas vid beräkningar med komplexa tal. Internt konverterar TI-89 / TI-92 Plus alla inmatade trigonometriska värden till radianer, men den konverterar inte värden för exponentiella, logaritmiska eller hyperboliska funktioner.

Obs! *Om du använder vinkelinställningen* Degree *måste du mata in polära värden i formen (r∠θ). Med vinkelinställningen* Degree *ger en inmatning av typen* $re^{i\theta}$ *ett felmeddelande.*

Med vinkelinställningen Degree är komplexa identiteter som e^(iθ) = cos(θ) + i sin(θ) inte generellt sanna eftersom värdena för cos och sin konverteras till radianer, medan de för e^( ) inte konverteras. Exempel, e^(i45) = cos(45) + i sin(45) hanteras internt som e^(i45) = cos(π/4) + i sin(π/4). Komplexa identiteter är alltid sanna med vinkelinställningen Radian.

Information om noggrannhet

> För att maximera noggrannheten används fler siffror internt än vad som visas.

Noggrannhet vid beräkningar

Decimaltalsvärden i minnet sparas med upp till 14 siffrors noggrannhet och med en 3-siffrig exponent.

- För min- och max-Window-variablerna (xmin, xmax, ymin, ymax osv) kan du spara värden med upp till 12 siffror. Andra Window-variabler använder 14 siffror.
- När ett decimaltalsvärde visas rundas det av enligt vad som är angivet för gällande lägesinställningar (Display Digits, Exponential Format osv) med maximalt 12 siffror och med en 3-siffrig exponent.
- RegEQ visar upp till 14-siffriga koefficienter.

Heltalsvärden i minnet sparas med upp till 614 siffror.

Noggrannhet vid plottning

Window-variabeln xmin är mittpunkten av den bildpunkt som är längst till vänster, och xmax är mittpunkten av den bildpunkt som är längst till höger. Δx är avståndet mellan mittpunkterna hos två horisontellt intilliggande bildpunkter.

__Obs!__ I "Aktivera och avaktivera läget för delat fönster" i kapitel 14 finns en tabell som visar antalet bildpunkter i ett fönster i fullständig storlek eller i ett delat fönster.

- Δx beräknas som (xmax − xmin) / (antal bildpunkter − 1).
- Om Δx matas in från grundfönstret eller ett program beräknas xmax som xmin + Δx ∗ (antal bildpunkter − 1).

Window-variabeln ymin är mittpunkten av den understa bildpunkten som används och ymax är mittpunkten av den översta bildpunkten som används. Δy är avståndet mellan mittpunkterna hos två vertikalt intilliggande bildpunkter.

- Δy beräknas som (ymax − ymin) / (antal bildpunkter − 1).
- Om Δy matas in från grundfönstret eller ett program beräknas ymax som ymin + Δy ∗ (antal bildpunkter − 1).

Markörkoordinater visas med åtta tecken (som kan inkludera ett negativt tecken, decimalkomma och exponent). Koordinatvärdena (xc, yc, zc osv) uppdateras med en noggrannhet på maximalt 12 siffror.

Systemvariabler och reserverade namn

I det här avsnittet listas namnen på systemvariabler och reserverade funktionsnamn som används av TI-89 / TI-92 Plus. Du kan endast ta bort de systemvariabler och reserverade funktionsnamn som markeras med en asterisk (*) med hjälp av DelVar *var* på inmatningsraden.

Graf	y1(x)–y99(x)*	y1'(t)–y99'(t)*	yi1–yi99*	r1(θ)–r99(θ)*
	xt1(t)–xt99(t)*	yt1(t)– yt99(t)*	z1(x,y)–z99(x,y)*	u1(n)–u99(n)*
	ui1–ui99*	xc	yc	zc
	tc	rc	θc	nc
	xfact	yfact	zfact	xmin
	xmax	xscl	xgrid	ymin
	ymax	yscl	ygrid	xres
	Δx	Δy	zmin	zmax
	zscl	eyeθ	eyeϕ	eyeψ
	ncontour	θmin	θmax	θstep
	tmin	tmax	tstep	t0
	tplot	ncurves	diftol	dtime
	Estep	fldpic	fldres	nmin
	nmax	plotStrt	plotStep	sysMath
Grafzoom	zxmin	zxmax	zxscl	zxgrid
	zymin	zymax	zyscl	zygrid
	zxres	zθmin	zθmax	zθstep
	ztmin	ztmax	ztstep	zt0de
	ztmaxde	ztstepde	ztplotde	zzmin
	zzmax	zzscl	zeyeθ	zeyeϕ
	zeyeψ	znmin	znmax	zpltstrt
	zpltstep			
Statistik	$\bar{x}$	$\bar{y}$	Σx	σx
	Σx^2	Σxy	Σy	σy
	Σy^2	corr	maxX	maxY
	medStat	medx1	medx2	medx3
	medy1	medy2	medy3	minX
	minY	nStat	q1	q3
	regCoef*	regEq(x)*	seed1	seed2
	Sx	Sy	R^2	
Tabell	tblStart	Δtbl	tblInput	
Data/Matris	c1–c99	sysData*		
Diverse	main	ok	errornum	
Ekvationslösaren	eqn*	exp*		

Hierarkin EOS (Equation Operating System)

I det här avsnittet beskrivs EOS™ (Equation Operating System), som används av TI-89 / TI-92 Plus. Tal, variabler och funktioner matas in i en enkel ordningsföljd. EOS beräknar uttryck och ekvationer med hjälp av parentesgruppering och enligt den prioritetsordning som beskrivs här nedan.

Ordningsföljd för beräkning

Nivå	Operator
1	Parenteser ( ), hakparenteser [ ], klammerparenteser { }
2	Omdirigerare (#)
3	Funktionsanrop
4	Operatorer efter argumentet: grader-minuter-sekunder (°,',"), fakultet (!),procent (%), radianer (ʳ),indexnummer ([ ]), transponering (ᵀ)
5	Potenser, "upphöjt till"-operatorn (^)
6	Negation (⁻)
7	Strängsammansättning (&)
8	Multiplikation (∗), division (/)
9	Addition (+),subtraktion (−)
10	Likhetsrelationer: lika med (=), inte lika med (≠ eller /=), mindre än (<), mindre än eller lika med (≤ eller <=), större än (>), större än eller lika med (≥ eller >=)
11	Logisk **not**
12	Logisk **and**
13	Logisk **or**, logisk "exklusive eller" **xor**
14	Begränsning "with"-operatorn (|)
15	Spara (→)

Parenteser, hakparenteser och klammerparenteser

All beräkning inom parenteser, hakparenteser eller klammerparenteser görs först. I exempelvis uttrycket 4(1+2), beräknar EOS först den del av uttrycket som är inom parentes, 1+2, och därefter multipliceras resultatet, 3, med 4.

Antalet vänster- och högerparenteser, hakparenteser och klammerparenteser måste vara lika i ett uttryck eller en ekvation. Om de inte är det visas ett felmeddelande som anger det saknade elementet. Exempelvis skulle (1+2)/(3+4 visa felmeddelandet “Missing)”.

Obs! Eftersom du kan definiera dina egna funktioner tolkas ett variabelnamn följt av ett uttryck inom parentes som ett "funktionsanrop" i stället för en implicit multiplikation. Anta att a(b+c) är funktionen a beräknad med b+c. För att multiplicera uttrycket b+c med variabeln a använder du explicit multiplikation: a∗(b+c).

Omdirigerare

Omdirigeringsoperator (#) konverterar en sträng till ett variabel- eller ett funktionsnamn. Exempelvis skapar #("x"&"y"&"z") variabelnamnet xyz. Med omdirigering kan du också skapa och ändra variabler inifrån ett program. Om exempelvis 10→r och "r"→s1 är #s1=10.

Operatorer efter argument

Operatorer efter argumentet är exempelvis 5!, 25% eller 60° 15' 45". Argument som följs av en operator beräknas efter fjärde prioritetsnivån. I exempelvis uttrycket 4^3! beräknas först 3!. Resultatet, 6, blir sedan exponenten för 4 med resultatet 4096.

Potenser

Potenser (^) och elementvis exponentiering (.^) beräknas från höger till vänster. Uttrycket 2^3^2 beräknas exempelvis på samma sätt som 2^(3^2) och resulterar i 512. Detta till skillnad från (2^3)^2, som är 64.

Negation

Om du vill mata in ett negativt tal trycker du på [(-)] följt av talet. Operatorer efter argument och potenser utförs före negation. Resultatet av $^-x^2$ är exempelvis ett negativt tal och $^-9^2 = ^-81$. Använd parentes för att kvadrera ett negativt tal som exempelvis $(^-9)^2$ med resultatet 81. Lägg också märke till att 5 negativt ($^-5$) inte är det samma som minus 5 (–5) och $^-3!$ beräknas som $^-(3!)$.

Begränsning (|)

Argumentet som följer operatorn "with" (|) tillhandahåller en uppsättning begränsningar som påverkar beräkningen av argumentet som föregår operatorn "with".

Regressionsformler

Detta avsnitt beskriver hur statistiska regressioner beräknas.

Algoritmen minsta kvadrat

De flesta regressioner använder icke-linjära, rekursiva, metoder för att finna den minsta kvadraten för att optimera följande kostnadsfunktion, vilken är summan av alla kvadrater på resterande fel:

$$J = \sum_{i=1}^{N} \left[residualExpression \right]^2$$

där: *residualExpression* är i termer av x_i och y_i
x_i är den oberoende variabellistan
y_i är den beroende variabellistan
N är listornas dimensioner

Denna teknik försöker att rekursivt uppskatta konstanterna i modelluttrycket för att göra J så litet som möjligt.

Exempel: $y=a\ sin(bx+c)+d$ är modellekvationen för **SinReg**. Dess resterande uttryck är därför:

$a\ sin(bx_i+c)+d-y_i$

För **SinReg** hittar algoritmen minsta kvadrat därför konstanterna a, b, c och d som minimerar funktionen:

$$J = \sum_{i=1}^{N} \left[a\ sin(bx_i + c) + d - y_i \right]^2$$

Regressioner

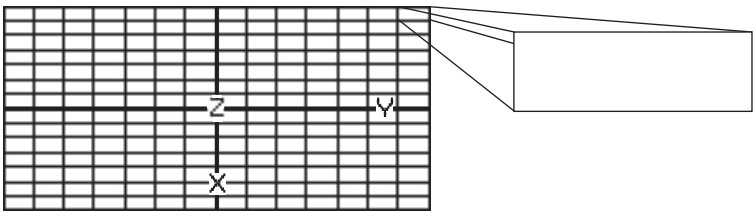
Regression	Beskrivning
CubicReg	Använder algoritmen minsta kvadrat för att hitta tredje gradens polynom: $y=ax^3+bx^2+cx+d$ Med fyra datapunkter passar den exakt i ett polynom. Med fem eller flera punkter är det en polynom regression. Minst fyra datapunkter krävs.
ExpReg	Använder algoritmen minsta kvadrat och transformerade värden x och $\ln(y)$ för att passa i modellekvationen: $y=ab^x$
LinReg	Använder algoritmen minsta kvadrat för att passa i modellekvationen: $y=ax+b$ där a är lutningen och b är skärningen med y-axeln.

Regression	Beskrivning
LnReg	Använder algoritmen minsta kvadrat och transformerade värden ln(x) och y för att passa i modellekvationen: $y=a+b\ \ln(x)$
Logistic	Använder algoritmen minsta kvadrat för att passa i modellekvationen: $y=a/(1+b*e^{\wedge}(c*x))+d$
MedMed	Använder linjen median-median för att beräkna summapunkterna x1, y1, x2, y2, x3, and y3, och passa i modellekvationen: $y=ax+b$ där a är lutningen och b är skärningen med y-axeln.
PowerReg	Använder algoritmen minsta kvadrat och transformerade värden av ln(x) och ln(y) för att passa i modellekvationen: $y=ax^b$
QuadReg	Använder algoritmen minsta kvadrat för att hitta andra gradens polynom: $y=ax^2+bx+c$ Med tre datapunkter passar den exakt i ett polynom. Med fyra eller flera punkter är det en polynom regression. Minst tre datapunkter krävs.
QuartReg	Använder algoritmen minsta kvadrat för att hitta fjärde gradens polynom: $y=ax^4+bx^3+cx^2+dx+e$ Med fem datapunkter passar den exakt i ett polynom. Med sex eller flera punkter är det en polynom regression. Minst fem datapunkter krävs.
SinReg	Använder algoritmen minsta kvadrat för att passa i modellekvationen: $y=a\ sin(bx+c)+d$

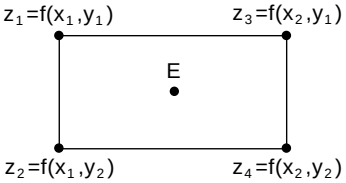
Konturer beräknas och plottas enligt följande metod. En implicit plottning är detsamma som en kontur förutom att en implicit plottning endast gäller för konturen z=0.

Algoritm

Utifrån dina Windowsvariabler för x och y delas avståndet mellan xmin och xmax samt mellan ymin och ymax upp i ett antal linjer specificerade av xgrid och ygrid. Dessa linjer korsas så att en serie rektanglar skapas.



För varje rektangel beräknas ekvationen för varje hörn av rektangeln (även kallade rasterpunkter) och ett medelvärde (E) beräknas:

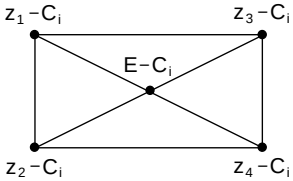


$$E = \frac{z_1 + z_2 + z_3 + z_4}{4}$$

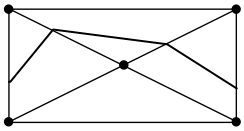
Värdet hos E är ekvationens värde vid rektangelns mitt.

För varje angivet konturvärde (C_i):

- För var och en av de fem punkterna som visas till höger beräknas differensen mellan punktens z-värde och konturvärdet.



- En teckenändring mellan två närliggande punkter anger att konturen korsar linjen som sammanbinder de två punkterna. Linjär interpolation används för att uppskatta var nollstället skär linjen.
- Inom rektangeln ansluts nollställen med raka linjer.
- Denna process upprepas för varje konturvärde.



Varje rektangel i rastret behandlas på samma sätt.

Runge-Kuttas metod

Runge-Kutta integrationer av vanliga differentialekvationer beräknas med TI-89 / TI-92 Plus enligt formeln Bogacki-Shampine 3(2) som den presenterades i *Applied Math Letters*, 2 (1989), pp. 1–9.

Bogacki-Shampine 3(2) formel

Formeln Bogacki-Shampine 3(2) ger ett resultat med tredje gradens exakthet och en feluppskattning baserad på en inbäddad andra gradens formel. För ett problem av formen:

$$y' = f(x, y)$$

och en angiven stegning h, kan Bogacki-Shampine formeln skrivas som:

$$F_1 = f(x_n, y_n)$$

$$F_2 = f\left(x_n + h\frac{1}{2}, y_n + h\frac{1}{2}F_1\right)$$

$$F_3 = f\left(x_n + h\frac{3}{4}, y_n + h\frac{3}{4}F_2\right)$$

$$y_{n+1} = y_n + h\left(\frac{2}{9}F_1 + \frac{1}{3}F_2 + \frac{4}{9}F_3\right)$$

$$x_{n+1} = x_n + h$$

$$F_4 = f(x_{n+1}, y_{n+1})$$

$$errest = h\left(\frac{5}{72}F_1 - \frac{1}{12}F_2 - \frac{1}{9}F_3 + \frac{1}{8}F_4\right)$$

Feluppskattningen *errest* används för att kontrollera steglängden automatiskt. En utförlig beskrivning finns i *Numerical Solution of Ordinary Differential Equations* av L. F. Shampine (New York: Chapman & Hall, 1994).

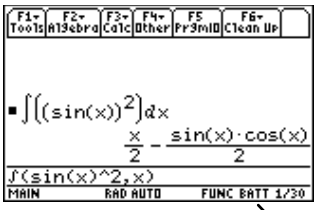
Programvaran i TI-89 / TI-92 Plus justerar inte steglängden för att hamna på särskilda punkter. Istället tar räknaren så stora steg som möjligt (utifrån felmarginalen som angetts i diftol) och erhåller resultaten för $x_n \le x \le x_{n+1}$ genom de kubiskt interpolerade polynomen som passerar genom punkten (x_n, y_n) med lutningen F_1 och genom (x_{n+1}, y_{n+1}) med lutningen F_4. Interpoleringen är effektiv och ger resultat genom steget som är lika exakta som resultaten vid stegets slut.

Information om service och garanti

C

Batterierna 572
Om du får problem 575
TI produktservice och garantiinformation 576

I den här appendix finns ytterligare information som kan vara till hjälp när du använder TI-89 / TI-92 Plus. Här finns procedurer som du kan följa för att rätta till problem med räknaren och information om den service och garanti som Texas Instruments ger.



Indikatorn BATT

När indikatorn BATT visas i statusfältet är det dags att byta batterier.

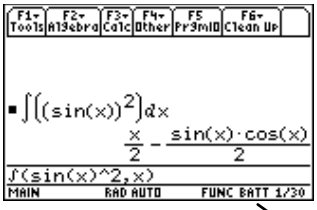
I TI-89 / TI-92 Plus används två typer av batterier: alkaliska batterier och ett litiumbatteri som används för att bevara minnet under tiden du byter dealkaliska batterierna.

När ska batterierna bytas ut

__Obs!__ TI-89 använder fyra AAA alkaliska batterier.

TI-92 Plus använder fyra AA alkaliska batterier.

När alkaliska batterierna börjar ta slut blir teckenfönstret suddigt (särskilt under beräkningar). För att kompensera för detta måste du justera kontrasten. Om du märker att du ofta behöver justera kontrasten är det dags att byta ut alkaliska batterierna. Som en hänvisning visas indikatorn BATT (BATT) i statusfältet när batterierna nästan är helt slut och du bör byta dem snart. När indikatorn BATT visas i vitt mot svart bakgrund (BATT) måste du byta alkaliska batterierna omedelbart.
Du bör byta litiumbatteriet en gång var tredje år.



Indikatorn BATT

__Obs!__ Undvik förlust av information som är lagrad i din TI-89 / TI-92 Plus genom att se till att räknaren är avslagen. Ta inte ur alkaliska batterierna eller litiumbatteriet samtidigt.

Litium-batteriet för strömbackup sitter i batterifacket, ovanför alkaliska batterierna. Undvik förlust av data genom att aldrig ta ur litium-batteriet utom när det sitter fyra nya alkaliska batterier i räknaren. Byt ut litium-batteriet vart tredje eller fjärde år.

Vad händer när du byter batterier

Om du inte tar ur båda typerna av batteri samtidigt eller låter dem ta slut helt och hållet, kan du byta vilken typ av batterier som helst, utan att förlora informationen i minnet.

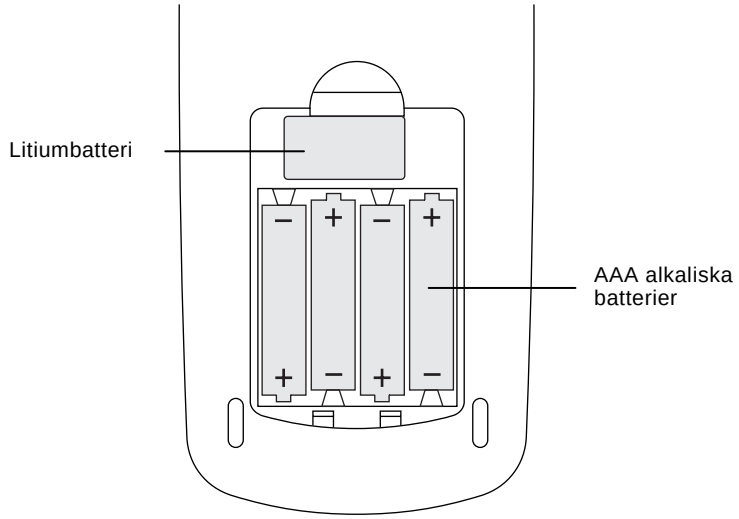
Batteriskötsel

Följ noga nedanstående instruktioner vid byte av batterier:

- Lämna aldrig batterier inom räckhåll för barn.
- Blanda aldrig nya och gamla batterier. Blanda aldrig batterier av olika fabrikat (eller batterityper av ett märke).
- Blanda inte laddningsbara och vanliga batterier.
- Kontrollera att du sätter i batterierna åt rätt håll enligt bilden (+ och -).
- Försök aldrig att ladda upp vanliga batterier.
- Kassera omedelbart använda batterier enligt gällande miljöregler.
- Bränn eller öppna aldrig batterier.

Byta alkaliska batterier i en TI–89

1. Om räknaren är påslagen stänger du av den (tryck på [2nd] [OFF]) för att undvika förlust av data i minnet.
2. Skjut skyddslocket över tangentbordet.
3. Håll räknaren upprätt, tryck ned luckans låsning och ta bort batteriluckan.
4. Ta ur de fyra AAA batterierna.
5. Sätt i fyra nya AAA alkaliska batterier åt rätt håll (+ och -) enligt illustrationen i batterifacket.



6. Sätt tillbaks batteriluckan genom att passa in de två pinnarna i öppningarna vid batterifacket och skjut sedan locket på plats.

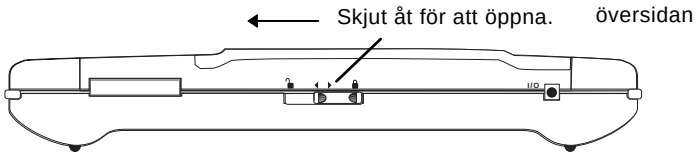
Byta ut Litiumbatteri i en TI–89

När du vill byta ut litiumbatteriet öppnar du batterifacket och lossar den lilla skruven som håller locket för BACK UP BATTERY på plats.

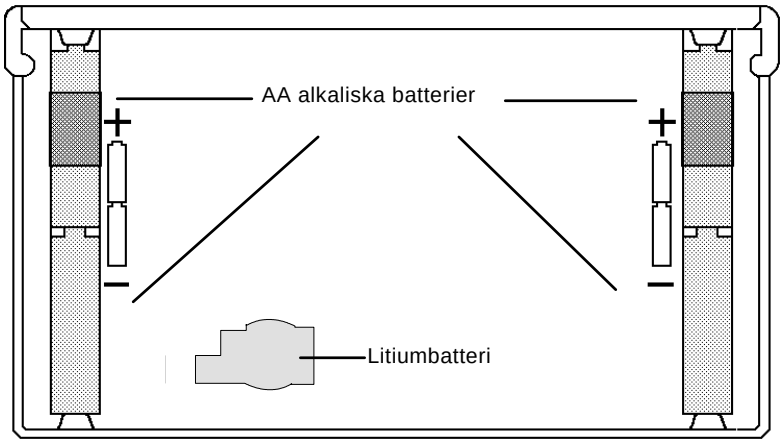
Ta ur det gamla batteriet och sätt i ett nytt CR1616 eller CR1620 med plus-sidan (+) uppåt. Sätt tillbaks locket och skruven.

Byta alkaliska batterier i en TI–92 Plus

1. Stäng av räknaren och placera den upp och ned på en ren yta för att undvika att slå på räknaren av misstag.
2. Håll räknaren upprätt med teckenfönstret mot dig, skjut spärren ovanpå räknaren till den olåsta positionen med fingret och ta bort det bakre locket från räknaren.



3. Ta ur de fyra AA batterierna.
4. Ta ur de gamla AA-batterierna och sätt i nya, enligt polaritetsdiagrammet inuti batterifacket.



5. Sätt tillbaka det bakre locket och skjut tillbaka spärren till den låsta positionen.
6. Slå på räknaren och justera kontrasten om det behövs.

Byta ut Litiumbatteri i en TI–92 Plus

När du vill byta ut litiumbatteriet öppnar du batterifacket och lossar den lilla skruven som håller locket för litiumbatteri på plats.

Ta ur det gamla batteriet och sätt i ett nytt CR2032 med plus-sidan (+) uppåt. Sätt tillbaks locket och skruven.

Om du får problem

Om du får problem när du arbetar med TI-89 / TI-92 Plus kanske följande förslag kan hjälpa dig att lösa dem.

Förslag

Problem:	Åtgärd:
Du kan inte se någonting i teckenfönstret.	Tryck på [◆] [+] för att göra kontrasten mörkare eller [◆] [−] ljusare.
Indikatorn BATT visas.	Byt batterier. Om BATT visas i vitt mot svart bakgrund (BATT) bör du byta ut batterierna omedelbart.
Indikatorn BUSY visas.	En beräkning pågår. Om du vill avbryta beräkningen trycker du på [ON].
Indikatorn PAUSE visas.	En graf eller ett program har stoppats tillfälligt och räknaren väntar på inmatning. Tryck på [ENTER].
Ett felmeddelande visas.	I Appendix B finns en lista över felmeddelanden. Tryck på [ESC] för att ta bort meddelandet.
Räknaren verkar inte fungera ordentligt.	Tryck på [ESC] flera gånger för att stänga en meny eller dialogruta och flytta tillbaka markören till inmatningsraden. — eller — Kontrollera att batterierna är korrekt installerade och att de är uppladdade.
TI-89 verkar ha "låst sig" och svarar inte på inmatningar från tangentbordet.	1. Ta bort ett av de fyra AAA-batterierna. 2. Tryck på och håll ned [(-)] och [)] när du sätter tillbaka batteriet. 3. Fortsätt att hålla ned [(-)] och [)] i fem sekunder innan du släpper upp tangenterna.
TI-92 Plus verkar ha "låst sig" och svarar inte på inmatningar från tangentbordet.	Tryck och håll ned [2nd] och [⏻]. Tryck och släpp upp [ON]. — eller — Om [2nd] [⏻] och [ON] inte löser problemet: 1. Ta bort ett av de fyra AA-batterierna. 2. Tryck på och håll ned [(-)] och [)] när du sätter tillbaka batteriet. 3. Fortsätt att hålla ned [(-)] och [)] i fem sekunder innan du släpper upp tangenterna.

Obs! *När du rättar till en "låsning" nollställs TI-89 / TI-92 Plus och minnet töms.*

TI produktservice och garantiinformation

Mer information om TI:s produkter, service och garantivillkor hittar du nedan.

TI produkt och serviceinformation

Mer information om TI:s produkter och tjänster kan du få via e-post eller från vår webb-server.

e-postadress: **ti-cares@ti.com**

webbadress: **http://www.ti.com/calc**

Service och garantiinformation

Mer information om garantitidens längd och omfattning eller om våra tjänster hittar du på garantisedeln som medföljde räknaren. Du kan även kontakta din återförsäljare eller serviceleverantör.

Programmeringshandbok

D

sättLäge( ) och hämtLäge( ) 578
sättGraf( ) 581
sättTabl( ) 583

Parametrar och lägesalternativ för funktionerna sättLäge( ), hämtLäge( ), sättGraf( ) och sättTabl( ) översätts inte till andra språk när de används i program. Om du t ex skriver ett program på svenska och sedan växlar till norska kommer programmet att ge ett felmeddelande. Du kan undvika detta fel genom att ersätta de alfabetiska tecknen med siffror. Siffervärdena fungerar med alla språk. I denna appendix hittar du de numeriska värdena för samtliga dessa inställningar.

I exemplet nedan visas hur du ersätter parametrarna för sättLäge( ) med siffror.

Exempel 1: Ett program som använder alfabetiska parametrar/lägesinställningar:

```
sättLäge("Graf"," Talföljd")
```

Exempel 2: Samma program, men med numeriska värden:

```
sättLäge("1","4")
```

Parameter/Lägesinställning	Värde
ALLA	0
Graf	1
FUNKTION	1
PARAMETERFORM	2
POLAR	3
TALFÖLJD	4
3D	5
DIFF EKVATION	6
Visa siffror	2
FIX 0	1
FIX 1	2
FIX 2	3
FIX 3	4
FIX 4	5
FIX 5	6
FIX 6	7
FIX 7	8
FIX 8	9
FIX 9	10
FIX 10	11
FIX 11	12
FIX 12	13
FLOAT	14
FLOAT 1	15
FLOAT 2	16
FLOAT 3	17
FLOAT 4	18
FLOAT 5	19
FLOAT 6	20
FLOAT 7	21
FLOAT 8	22
FLOAT 9	23

Parameter/Lägesinställning	Värde
FLOAT 10	24
FLOAT 11	25
FLYT 12	26
Vinkel	3
RADIAN	1
GRADER	2
Exponentformat	4
NORMAL	1
GRUNDPOTENS	2
ENGINEERING	3
Komplext format	5
REEL	1
REKTANGULÄR	2
POLAR	3
Vektorformat	6
REKTANGULÄR	1
CYLINDRISK	2
SFÄRISK	3
Pretty Print	7
AV	1
PÅ	2
Delad skärm	8
HEL	1
TOPP-BOTTEN	2
VÄNSTER-HÖGER	3
Dela Prog 1	9
(program är inte numrerade)	
Dela Prog 2	10
(program är inte numrerade)	
Antal grafer	11
1	1
2	2

Parameter/Lägesinställning	Värde
Graf 2	12
FUNKTION	1
PARAMETERFORM	2
POLAR	3
TALFÖLJD	4
3D	5
DIFF EKVATION	6
Delningsförh skärm	13
1:1	1
1:2	2
2:1	3
Exakt/Approx	14
AUTO	1
EXAKT	2
APPROXIMATIV	3
Bas	15
DEC	1
HEX	2
BIN	3

Parameter/Lägesinställning	Värde
Koordinater	1
REKT	1
POLAR	2
AV	3
Grafordning	2
TALF	1
SIMUL	2
Rutnät	3
AV	1
PÅ	2
Vinkel	4
I 3D-läge:	
AV	1
VINKEL	2
BOX	3
Inte i 3D-läge:	
AV	1
PÅ	2
Ledmarkör	5
AV	1
PÅ	2
Etiketter	6
AV	1
PÅ	1
Seq Vinkel	7
TID	1
VÄV	2
EGET	3
Lösningsmetod	8
RK	1
EULER	2

Parameter/Lägesinställning	Värde
Fält	9
LUTNFLT	1
RIKTFLT	2
FLTAV	3
DE Vinkel	10
TID	1
Y1-VS-Y2	2
T-VS-Y'	3
Y-VS-Y'	4
Y1-VS-Y2'	5
Y1'-VS-Y2'	6
XR Stil	11
TRÅD	1
GÖMD YTA	2
KONTURNIVÅER	3
TRÅD OCH KONTUR	4
IMPLICIT PLOT	5

Parameter/Lägesinställning	Värde
Graf <-> Tabell	1
AV	1
PÅ	2
Oberoende	2
AUTO	1
FRÅGA	2
Vinkel	4

Innehållsförteckning

Kommandon och funktioner är sorterade efter sina korrekta namn. Matematiska symboler och operatorer är också sorterade efter namn.

Symboler

!, fakultet, *8*, *534*. *omslagets främre insida*, *omslagets bakre insida*
", andranotation, *538*
∫**( )**, integrera, *10*, *61*, *62*, *63*, *66*, *75*, *76*, *535*
√**( )**, kvadratrot, *536*
Π( ), produkt, *75*, *536*
Σ( ), summa, *75*, *536*
≠, **/=**, inte lika med, *294*, *532*, *1*
≤, **<=**, mindre än eller lika med, *294*, *533*. *omslagets främre insida*, *omslagets bakre insida*
≥, **>=**, större än eller lika med, *294*, *533*, *omslagets främre insida*, *omslagets bakre insida*
/, division, *530*
E, exponent, *441*
°, gradnotation, *400*, *538*
#, inriktning, *293*, *537*, *565*. *omslagets främre insida*, *omslagets bakre insida*
◉, kommentar, *282*, *542*. *omslagets främre insida*, *omslagets bakre insida*
→, lagra, *289*, *541*
∗, multiplicera, *530*
-, negation, *25*
-, negera, *531*
∞, oändlighet, *80*
▸, omvandla, *85*, *539*
ʳ, radian, *537*
−, subtrahera, *529*
ᵀ, transponera, *512*
∠, vinkel, *538*
▸**Bin**, visa som binär, *345*, *417*
▸**Cylind**, visa som cylindrisk vektor, *429*
▸**DD**, visa som decimal vinkel, *432*
▸**Dec**, visa som decimalt heltal, *345*, *432*
▸**DMS**, visa som timmar/minuter/sekunder, *438*
∫f(x)dx (graf math verktyg), *122*, *124*
▸**Hex**, visa som hexadecimal, *345*, *456*
Δlist( ), lista skillnader, *463*
θmax window-variabel, *137*
θmin window-variabel, *137*
▸**Polär**, visa som polär vektor, *481*
▸**Rect**, visa som rektangulär vektor, *491*
▸**Sphere**, visa som sfärisk vektor, *509*
θstep window-variabel, *137*
Δtbl, tabell ökning, *224*
ΔtmpCnv( ), temperaturområde omvandling, *86*, *517*
Δx window-variabel, *119*, *562*
Δy window-variabel, *119*, *562*
%, procent, *531*
&, lägg till, *293*, *534*. *omslagets främre insida*, *omslagets bakre insida*
', minut-notation, *538*
', prim, *538*
./, dot division, *534*
.∗, dot multiplikation, *534*
.−, dot subtraktion, *534*
.^, dot potens, *534*
.+, dot addition, *533*
@, godtyckligt heltal, *80*. *omslagets främre insida*, *omslagets bakre insida*
^, potens, *536*, *565*
_, understrykningstecken, *539*
|, with, *10*, *58*, *60*, *67*, *540*, *565*
+, addera, *529*
<, mindre än, *294*, *532*
<<...>>, otillräckligt minne, *103*
=, lika med, *294*, *532*
>, större än, *294*, *533*
0b, binär indikator, *542*
0h, hexadecimal indikator, *542*
10^( ), 10-potens, *540*
3D-plottning, *153* – *173*. *omslagets främre insida*, *omslagets bakre insida*
  animering, *154*, *164*. *omslagets främre insida*, *omslagets bakre insida*
  CONTOUR LEVELS, *155*, *166*
  HIDDEN SURFACE, *155*, *166*
  WIRE AND CONTOUR, *155*, *166*
  WIRE FRAME, *155*, *166*

A

abs( ), absolut värde, *402*, *414*
addera, **+**, *529*
Algebra meny, *70*, *72*
algebra operationer, *410*
and, Booleskt and, *67*, *294*, *347*, *414*
AndPic, and picture, *306*, *415*
andranotation, **"**, *538*
ange verktygsfält, **Toolbar**, *302*, *518*
Angle lägesinställning, *41*, *108*, *547*
angle( ), vinkel, *415*
anpassat verktygsfält. *Se* verktygsfält
ans( ), senaste resultat, *50*, *416*

A (fort.)

användardefinierade enheter, *88*
användardefinierade funktioner, *46*, *77*, *78*, *97 – 99*, *157*, *205*, *207*, *285*, *286*, *432*
APD (Automatic Power Down), *14*
APPLICATIONS meny, *34*, *38*
approx( ), approximativ, *70*, *416*
Approximate lägesinställning, *29*, *41*, *54*, *62*, *549*
approximativt svar. *omslagets främre insida*, *omslagets bakre insida*
Arc (graf math verktyg), *122*, *125*, *138*
Archive, variabler i användarminnet, *289*, *361*, *416*
arcLen( ), båglängd, *75*, *416*
arctangent, **tan^{-1}( )**, *514*
arcuscosinus, **cos^{-1}( )**, *424*
arcussinus, **sin^{-1}( )**, *503*
assemblerspråk, *313*, *314*, *443*
augment( ), sammanfoga/konkatenera, *388*, *417*
Auto lägesinställning, *29*, *41*, *54*, *63*, *549*
automatisk förenkling, *64*
automatisk inklistring, *52*, *95*
automatiska tabeller, *226*
avbryta en beräkning, *28*
avgRC( ), genomsnittling förändringstakt, *417*
avrunda, **round( )**, *494*
avsluta, **Exit**, *444*
Axes grafformat, *114*, *181*, *190*, *191*
Axes inställningar, *162*, *165*
axlar (talföljd), CUSTOM, *146*

B

båglängd, **arcLen( )**, *75*, *416*
bas för naturlig logaritm, *e*, *80*
Base lägesinställning, *42*, *550*
baskod, *373*, *374*, *375*, *376*
BATT meddelande, *54*, *572*, *575*
batterier, *2*, *3*, *14*, *15*, *54*, *572*, *573*, *574*, *575*
bilder, *217*, *218*
  cykel, **CyclePic**, *306*, *429*
  ersätt, **RplcPic**, *306*, *495*
  exklusivt or, **XorPic**, *306*, *522*
  hämta, **RclPic**, *306*, *490*
  ny, **NewPic**, *289*, *306*, *472*
  och, **AndPic**, *306*, *415*
  spara, **StoPic**, *306*, *510*
  ta bort, *218*
binär
  indikator, **0b**, *542*
  rotera, **rotate( )**, *348*
  skift, **shift( )**, *348*
  visning, **▸Bin**, *345*, *417*
bläddra. *omslagets främre insida*, *omslagets bakre insida*
BldData, bygg data, *193*, *289*, *418*
Bogacki-Shampine formel, *569*
Booleskt
  and, **and**, *67*, *294*, *347*, *414*
  exklusivt or, **xor**, *294*, *347*, *521*
  not, **not**, *294*, *474*
  or, **or**, *294*, *347*, *476*
Box Plot, *267*
bråkdelar, *70*, *74*, *394*, *483*
Build Web, skapa väv, *146*, *147*
BUSY-indikator, *54*, *115*, *278*
bygg
  data, **BldData**, *193*, *289*, *418*
  tabell, **Table**, *305*, *513*
byt
  bild, **CyclePic**, *219*, *306*, *429*
  namn, **Rename**, *289*, *492*
  ut bild, **RplcPic**, *306*, *495*

C

Calc meny, *75*
Calculator-Based Laboratory. *Se* CBL
Calculator-Based Ranger. *Se* CBR
CATALOG meny, *44*
CBL
  hämta/returnera, **Get**, *451*
  program, *309*, *399*
  skicka listvariabel, **Send**, *495*
  statistisk data, *273*, *274*
  övning, *399*
CBR
  hämta/returnera, **Get**, *451*
  program, *309*, *399*
  skicka listvariabel, **Send**, *495*
  statistisk data, *273*, *274*
ceiling( ), tak, *389*, *418*
certifikat, *369*, *373*, *374*, *375*, *376*, *377*, *378*
cFactor( ), komplex faktor, *71*, *406*, *419*, *560*
CHAR (tecken) meny, *34*
char( ), tecken sträng, *293*, *419*, *551*
Circle, rita cirkel, *308*, *419*
Circulär definition felstatus, *289*
cirkel, plottning, *106*
cirklar, rita, *214*
Clean Up meny, *43*
ClrDraw, radera ritning, *213*, *307*, *420*
ClrErr, radera felstatus, *310*, *420*
ClrGraf, radera graf, *205*, *305*, *340*, *420*
ClrHome, radera home, *421*
ClrIO, radera I/O, *279*, *302*, *421*
colDim( ), matris kolumndimension, *421*
colNorm( ), matris kolumn norm, *421*
comDenom( ), gemensam nämnare, *70*, *71*, *74*, *421*
Complex Format lägesinställning, *41*, *547*
Complex meny, *71*
conj( ), komplex konjugera, *422*
Coordinates grafformat, *114*, *137*
CopyVar, kopiera variabel, *289*, *358*, *422*
cos^{-1}( ), arcuscosinus, *424*

C (fort.)

cos( ), cosinus, *423*
cosh^{-1}( ), hyperbolisk arcuscosinus, *424*
cosh( ), hyperbolisk cosimus, *424*
cosinus, **cos( )**, *423*
crossP( ), korsprodukt, *425*
cSolve( ), komplex lösning, *61*, *425*, *560*
CubicReg, tredjeordningens regression, *262*, *427*, *566*
cumSum( ), kumulativ summa, *250*, *428*
Current folder lägesinställning, *41*, *546*
CustmOff, anpassat verktygsfält av, *37*, *428*
CustmOn, anpassat verktygsfält på, *37*, *428*
CUSTOM axlar (talföljd), *146*
CUSTOM egen plottning, *142*, *190*, *191*
Custom Enheter lägesinställning, *42*, *550*
CUSTOM meny, *34*, *37*
Custom, definiera verktygsfält, *302*, *428*
Cycle, cycle, *429*
CyclePic, byt bild, *219*, *306*, *429*
cylindrisk vektordisplay, **▸Cylind**, *429*
cZeros( ), komplexa nollställen, *61*, *71*, *429*, *560*

D

d **( )**, förstaderivata, *10*, *66*, *75*, *76*, *431*
Data/Matrix Editor, *203*, *237* – ***2***52. *Se även*
  matriser
  Auto-calculate, *249*
  cellbredd, *245*
  datavariabel, *240*, *241*, *242*
  fylla, *244*
  kolumnrubrik, *248*, *249*, *250*
  kopiera, *252*
  låsa, *248*
  listvariabel, *239*, *241*, *242*
  matrisvariabel, *239*, *240*, *241*, *242*
  nytt, **NewData**, *240*, *249*, *289*
  rullning, *244*
  sätta in, *246*, *247*
  skapa, *241*, *242*
  skifta, **shift( )**, *250*, *501*
  sortera kolumner, *251*
  statistikplottning, *265*
  ta bort, *246*, *247*
  värden, *243*
  variabler, *240*, *241*, *242*
datafiltrering, *396*
dataplottning, *254*
decimal
  heltal, **▸Dec**, *345*, *432*
  vinkel, **▸DD**, *432*
Define, definiera, *77*, *97*, *110*, *130*, *142*, *157*, *179*, *196*, *204*, *207*, *287*, *289*, *305*, *384*, *432*
delad skärm, *209*, *211*, *231* – ***2***36, *329*
  växla, **switch( )**, *300*, *512*
delat fönsterläge, *341*
  avsluta, *234*
  inmatningsrad, *235*, *236*
  pixelkoordinater, *234*
  ställa in, *233*
  växla, *235*
DelFold, ta bort mapp, *102*, *289*, *433*
DelVar, ta bort variabel, *60*, *77*, *102*, *289*, *291*, *433*
derivata, *10*
  förstaderivata, *d* **( )**, *10*, *66*, *75*, *76*, *431*
  numerisk derivata, **nDeriv( )**, *75*, *471*
Derivatives (graf math verktyg), *122*, *124*, *132*, *138*
deSolve( ), lösning, *75*, *196*, *433*
det( ), matris determinant, *435*
diag( ), matris diagonal, *435*
diagram och plottning
  koordinater. *omslagets främre insida*, *omslagets bakre insida*
Dialog, definiera dialogruta, *302*, *436*
dialogrutor, *35*
differentialekvationer
  andraordningens, *187*, *196*
  DIRFLD, riktningsfält, *180*, *185*, *198*
  felsökning, *197*
  FLDOFF, fält av, *180*, *185*, *199*
  förstaordningens, *186*, *196*
  lösningsmetoder, *180*, *193*, *569*
  plottning, *175* – ***1***99
  SLPFLD, lutningsfält, *180*, *185*, *197*
  startvärden, *184*
  tredjeordningens, *189*
diftol window-variabel, *182*
dim( ), dimension, *293*, *437*
DIRFLD, riktningsfält, *180*, *185*, *198*
Disp, visa I/O-fönster, *277*, *283*, *302*, *310*, *437*, *551*
DispG, visa graf, *302*, *305*, *437*
DispHome, visa grundfönster, *302*, *438*
display
  graf, **DispG**, *302*, *305*, *437*
  grundfönster, **DispHome**, *302*, *438*
  I/O-fönster, **Disp**, *277*, *283*, *302*, *310*, *437*, *551*
  tabell, **DispTbl**, *302*, *305*, *438*
Display Digits lägesinställning, *31*, *41*, *546*
DispTbl, display tabell, *302*, *305*, *438*
Distance (graf math verktyg), *122*, *125*, *132*, *138*
division, **/**, *530*
domänrestriktioner, *69*
dotP( ), punktprodukt, *438*
DrawFunc, ritfunktion, *212*, *308*, *439*
DrawInv, rita invers, *212*, *308*, *439*

D (fort.)

DrawParm, rita paramteriskt, *212*, *308*, *439*
DrawPol, rita polär, *212*, *308*, *439*
DrawSlp, rita slope, *215*, *308*, *440*
DropDown, rullgardinsmeny, *302*, *440*
DrwCtour, rita kontur, *168*, *308*, *440*
dtime window-variabel, *182*

E

e upphöjt till, ***e*^( )**, *441*
e, bas för naturlig logaritm, *80*
***e*^( )**, *e* upphöjt till, *441*
egen plottning, CUSTOM, *142*, *190*, *191*
egentligt bråk, **propFrac**, *70*, *74*, *394*, *483*
eigVc( ), egenvektor, *441*
eigVl( ), egenvärde, *442*
ekvationer, lösa, *333 – 341*
  delat fönster, *340*
  ekvationer, *335*, *336*
  plottning, *340*
  variabler, *336*
Else, else, *296*, *457*
ElseIf, else if, *207*, *296*, *442*
end
  custom, **EndCustm**, *302*, *428*
  dialog, **EndDlog**, *302*, *436*
  for, **EndFor**, *283*, *297*, *450*
  funktion, **EndFunc**, *207*, *286*, *451*
  if, **EndIf**, *283*, *295*, *457*
  loop, **EndLoop**, *299*, *466*
  program, **EndPrgm**, *276*, *287*, *482*
  toolbar, **EndTBar**, *302*, *518*
  try, **EndTry**, *310*, *518*
  while, **EndWhile**, *298*, *521*
EndCustm, end custom, *302*, *428*
EndDlog, end dialog, *302*, *436*
EndFor, end for, *283*, *297*, *450*
EndFunc, end function, *207*, *286*, *451*
EndIf, end if, *283*, *295*, *457*
EndLoop, end loop, *299*, *466*
EndPrgm, end program, *276*, *287*, *482*
EndTBar, end toolbar, *302*, *518*
EndTry, end try, *310*, *518*
EndWhile, end while, *298*, *521*
enheter, *83*
  användardefinierade, *88*
  förval, *87*, *89*
  hämta/returnera, **getUnits( )**, *455*
  inställningar, **setUnits( )**, *300*, *499*
  lägesinställningar, *42*, *82*, *550*
  mått, *81 – 91*
  omvandla, *85*
  visa, *87*
entry( ), entry, *50*, *443*
en-variabel-statistik, **OneVar**, *262*, *476*
EOS (Equation Operating System), *564*
Equation Operating System (EOS), *564*
Estep window-variabel, *182*
etikett, **Lbl**, *287*, *296*, *299*, *460*
Euler metod, *180*, *193*
exact( ), exact, *443*
Exact/Approx lägesinställning, *29*, *41*, *54*, *61*, *62*, *63*, *549*
Exec, kör assemblerspråk, *314*, *443*
exempel, förhandsvisning, övningar
  3D-plottning, *154*, *390*
  andraordningens differentialekvation, *187*, *196*
  baseball, *400*
  bollkurva, *128*
  CBL-program, *399*
  cos(x)=sin(x), *389*
  data/matris editor, *238*
  datafiltrering, *396*
  dela upp en rationell funktion, *394*
  delad skärm, *232*, *400*
  derivata, *10*
  differentialekvationer, *176*
  divergerande vävdiagram, *148*
  ekvationslösaren, *334*
  expandera uttryck, *9*
  faktorer, *8*
  faktorisera polynom, *9*, *72*
  Fibonacci-talföljd, *151*
  förkorta uttryck, *9*
  funktionsplottning, *106*
  graffunktioner, *11*
  hörnstolpsproblemet, *384*
  implicit plottning, *173*
  integraler, *10*
  komplex faktor, *406*
  komplex modulus-yta, *170*
  komplexa nollställen, *402*
  komplexa tal, *8*
  konstanter, *82*
  konvergerande vävdiagram, *148*
  kvadratisk formel, *386*
  lösa linjära ekvationer, *9*, *10*, *73*
  måttenheter, *82*
  minneshantering, *350*, *351*, *352*
  modellen jägare-byte, *150*, *191*
  oscillerande vävdiagram, *149*
  parameterplottning, *128*, *400*
  polär ros, *134*
  population, *254 – 258*
  primfaktorer, *8*
  programmering, *276*, *277*, *311*, *312*
  Pythagoras sats, *384*
  rationella faktorer, *406*
  reella faktorer, *406*
  samla in mätvärden, *407*
  standardannuitet, *404*
  statistik, *254 – 258*

E (fort.)

stegvisa funktioner, *202*
symbolisk manipulation, *58*
tabeller, *222*
talbaser, *344*
talföljdsplottning, *140*
textoperationer, *316*
träd och skogar, *140*
tredjegradspolynom, *402*
tredjeordningens differentialekvation, *189*
värdeändring över tid, *405*
variabelhantering, *350*, *351*, *352*
övningsskript med texteditor, *392*
Exit, avsluta, *444*
exklusivt or (Booleskt), **xor**, *294*, *347*, *521*
exklusivt or picture, **XorPic**, *306*, *522*
exp▸list( ), uttryck till lista, *444*
expand( ), expandera, *9*, *70*, *72*, *386*, *402*, *445*
exponent, E, *441*
Exponential Format lägesinställning, *31*, *41*, *547*
expr( ), sträng till uttryck, *292*, *293*, *301*, *381*, *446*
ExpReg, exponentiell regression, *262*, *446*, *566*
Extract meny, *71*
eyeψ rotation window-variabel, *158*, *162*, *163*
eyeθ x-axel window-variabel, *158*, *162*
eyeϕ z-axel window-variabel, *158*, *162*, *163*

F

factor( ), faktor, *8*, *9*, *61*, *70*, *72*, *387*, *406*, *447*
faktorisering, *9*, *72*
övning, *406*
fakultet, **!**, *8*, *534*. *omslagets främre insida*, *omslagets bakre insida*
false meddelande, *80*
fält av, FLDOFF, *180*, *185*, *199*
fel och felsökning
program, *310*
överföring, *369*, *377*
felsökning. *Se* felstatus och felsökning
felstatus och felsökning, *575*, *576*
Circulär definition felstatus, *289*
meddelanden, *538* – ***5***45
Memory felstatus, *364*
Out-of-memory, *79*
radera felstatus, **ClrErr**, *310*, *420*
skicka felstatus vidare, **PassErr**, *310*, *480*
varningar, *545*
Fibonacci-följd, *151*
Field graf format, *180*
field picture, fldpic, *183*
Fill, matris fyll, *448*
Flash, uppgradera produktkod, *373*, *374*
Flash-program, *4*, *38*, *45*, *79*, *353*, *356*. *omslagets främre insida*, *omslagets bakre insida*
ta bort, *369*
FLDOFF, fält av, *180*, *185*, *199*
fldpic, field picture, *183*
fldres window-variabel, *182*
flersatsfunktioner, *207*
floor( ), golv, *389*, *448*
flytta variabel, **MoveVar**, *289*, *470*
fMax( ), funktion maximum, *61*, *75*, *448*
fMin( ), funktion minimum, *61*, *75*, *449*
FnOff, funktion av, *111*, *305*, *449*
FnOn, funktion på, *111*, *305*, *449*
For, for, *283*, *297*, *450*
förenkling
fördröjd, *66*
regler, *64*
stoppa, *65*
förfrågan, **Request**, *301*, *302*, *492*
förhandsvisning. *Se* exempel, förhandsvisning, övningar
format( ), format sträng, *293*, *302*, *450*
FORMATS dialogruta, *114*, *155*, *165*, *166*, *167*, *171*, *176*, *245*, *325*. *omslagets främre insida*, *omslagets bakre insida*
fpart( ), funktion part, *451*
fritt rörlig markör, *116*, *132*, *138*, *145*, *159*, *183*
Frobenius norm, **norm( )**, *474*
Func, program funktion, *207*, *286*, *451*
funktioner, *26*, *409* – *542*
användardefinierade, *46*, *77*, *78*, *97* – *99*, *157*, *205*, *207*, *285*, *286*, *432*
av, **FnOff**, *111*, *305*, *449*
del, **fpart( )**, *451*
flera satser, *207*
fördröjd förenkling, *66*
maximum, **fMax( )**, *61*, *75*, *448*
minimum, **fMin( )**, *61*, *75*, *449*
på, **FnOn**, *111*, *305*, *449*
plottning, *105* – ***1***26
program funktion, **Func**, *207*, *286*, *451*

G

Garbage collection meddelande, *362*, *363*
gcd( ), största gemensamma nämnare, *451*
gemensam nämnare, **comDenom( )**, *70*, *71*, *74*, *421*
genomsnittling förändringstakt, **avgRC( )**, *417*
Get, hämta/returnera CBL/CBR-värde, *273*, *309*, *451*
GetCalc, hämta/returnera räknare, *309*, *371*, *452*
getConfg( ), hämta/returnera konfiguration, *300*, *452*
getDenom( ), hämta/returnera nämnare, *71*, *453*
getFold( ), hämta/returnera mapp, *289*, *300*
getKey( ), hämta/returnera tangent, *301*, *453*, *552*, *555*
getMode( ), hämta/returnera lägesinställning, *300*, *453*
getNum( ), hämta/returnera tal, *71*, *454*
getType( ), hämta/returnera typ, *59*, *454*

G (fort.)

getUnits( ), hämta/returnera enheter, *300*, *455*
globala variabler, *291*
godtyckligt heltal, @, *80*. *omslagets främre insida*, *omslagets bakre insida*
golv, **floor( )**, *448*
gömd yta, *155*, *161*, *166*
Goto, go to, *287*, *296*, *299*, *455*
gradnotation, ° , *400*, *538*
Graf 2 lägesinställning, *41*, *549*
Graf lägesinställning, *41*, *54*, *108*, *130*, *142*, *157*, *179*, *546*
Graf Order graf format, *114*, *180*
Graf<->Tabell, tabell-graf, *224*
grafer och plottning
  ∫f(x)dx, *122*, *124*
  3D, *153* – ***1***73
  anpassade axlar, *146*
  anpassade grafer, *142*, *190*, *191*
  Arc, *122*, *125*, *138*
  bilder, *217*, *218*
  delad skärm, *209*, *211*
  delat fönster, *233*
  Derivatives, *122*, *124*, *132*, *138*
  differentialekvationer, *175* – ***1***99
  Distance, *122*, *125*, *132*, *138*
  format, *114*, *137*, *144*, *180*
  funktion av, **FnOff**, *305*, *449*
  funktion på, **FnOn**, *305*, *449*
  funktioner, *105* – ***1***26
  graf, **Graf**, *205*, *305*, *456*
  grafdatabaser, *220*
  Grundfönster, *204*, *205*
  hämta grafdatabas, **RclGDB**, *306*, *490*
  implicit plottning, *171*, *172*, *173*
  Inflection, *122*, *124*
  inställningar, **setGraf( )**, *300*, *305*, *497*
  Intersection, *122*, *123*
  inversa funktioner, *212*
  konturplottning, *167*, *168*, *169*
  koordinater, *11*, *116*
  kurvfamiljer, *208*, *209*
  lägesinställning, *136*
  lägesinställningar, *41*, *54*, *108*, *130*, *142*, *157*, *179*, *546*
  linjestilar, *112*, *131*, *136*, *143*, *157*, *179*
  matematiska funktioner, *122*
  matrisdata, *203*
  Maximum, *122*, *123*
  Minimum, *11*, *122*, *123*
  nästlade funktioner, *206*
  oberoende variabel, *204*
  operationer, *410*
  panorering, *118*
  parametrisk, *127* – ***1***32
  paus, *115*
  pixlar, *562*
  polär, *133* – ***1***38
  program, *305*
  QuickCenter, *118*
  radera, **ClrGraf**, *205*, *305*, *340*, *420*
  rita, *213*
  ritning, *307*
  sekvens, *139* – ***1***51
  Shade, *122*, *126*
  simultana grafer, *208*
  skuggning, **Shade**, *308*, *500*
  spåra, *117*, *118*, *132*, *138*, *145*, *159*, *183*
  spara grafdatabas, **StoGDB**, *306*, *510*
  spåra, **Trace**, *117*, *305*, *390*, *398*, *399*, *402*, *518*
  spårning, *11*
  stegvisa funktioner, *206*
  stil, **Style**, *305*, *511*
  Tangent, *122*, *125*, *132*, *138*
  text, *216*
  tidsgrafer, *142*, *146*, *190*, *191*
  två-grafsläge, *209*, *210*, *233*
  välja funktioner, *111*, *131*, *143*, *179*
  Value, *122*, *123*, *132*, *138*, *145*, *159*, *183*
  vävdiagram, *142*, *146*, *147*
  visningsfönster, *113*, *131*, *137*, *143*, *144*, *158*
  window-variabler, *113*, *131*, *137*, *143*, *144*, *158*
  Y= editor, *106*, *109*, *130*, *136*, *142*, *157*, *179*, *204*
  Zero, *122*, *123*
  zoom, *119*, *132*, *138*, *145*, *159*, *305*
  zoom Memory, *119*, *121*
  zoomfaktorer, *119*, *121*
  översikt, *107*, *129*, *135*, *141*, *156*, *178*
grafiskt användargränssnitt, GUI, *302*
grafs and plottning
  animering, *219*
gräns, **limit( )**, *461*
Graph lägesinställning, *136*
Graph, graf, *110*, *202*, *205*, *208*, *305*, *456*
grekiska tecken, *325*, *326*, *327*. *omslagets främre insida*, *omslagets bakre insida*
Grid graf-format, *114*
Grundfönster, *6*, *23*
GUI, grafiskt användargränssnitt, *302*

H

hämta
  bild, **RclPic**, *306*, *490*
  grafdatabas, **RclGDB**, *220*, *306*, *490*
hämta variabler från användarminnet, **Unarchiv**, *289*, *361*, *519*
hämta/returnera
  CBL/CBR-värde, **Get**, *273*, *309*, *451*
  enheter, **getUnits( )**, *300*, *455*
  konfiguration, **getConfg( )**, *300*, *452*
  lägesinställning, **getMode( )**, *300*, *453*
  mapp, **getFold( )**, *289*, *300*, *453*
  nämnare, **getDenom( )**, *71*, *453*

H (fort.)

räknare, **GetCalc**, *309*, *371*, *452*
tal, **getNum( )**, *71*, *454*
tangent, **getKey( )**, *301*, *453*, *552*, *555*
typ, **getType( )**, *59*, *454*
heltal, **int( )**, *459*
heltalsdel, **iPart( )**, *140*, *459*
heltalsdivision, **intDiv( )**, *346*, *459*
hexadecimal
indikator, **0h**, *542*
visning, **▸Hex**, *345*, *456*
Histogram, *268*
historik, *6*
historikområde, *23*, *329*
höger, **right( )**, *71*, *293*, *492*
hyperbolisk
arcuscosinus, **cosh^{-1}( )**, *424*
arcussinus, **sinh^{-1}( )**, *504*
arcustangent, **tanh^{-1}( )**, *514*
cosinus, **cosh( )**, *424*
sinus, **sinh( )**, *504*
tangent, **tanh( )**, *514*

I

i sträng, **inString( )**, *293*, *458*
ID listor, *379*
identity( ), identity matris, *457*
ID-listor, *378*
ID-nummer, *55*, *373*, *374*, *378*, *379*
If, if, *207*, *283*, *295*, *296*, *457*
imag( ), imaginär del, *457*
implicit multiplikation, *26*, *130*
implicit plottning, *171*, *172*, *173*, *568*
Independent AUTO/ASK, oberoende auto/fråga, *224*, *226*, *229*
Inflection (graf math verktyg), *122*, *124*
Input, inmatning, *301*, *305*, *458*
InputSt, inmatningssträng, *292*, *301*, *371*, *458*
inriktning, **#**, *293*, *537*, *565*. *omslagets främre insida*, *omslagets bakre insida*
inString( ), i sträng, *293*, *458*
instruktioner, *26*
int( ), heltal, *459*
intDiv( ), heltalsdivision, *346*, *459*
inte lika med, ≠, **/=**, *294*, *532*, *1*
integrera, ∫**( )**, *10*, *61*, *62*, *63*, *66*, *75*, *76*, *535*
internationella tecken, *21*, *324*, *325*, *326*. *omslagets bakre insida*
Intersection (graf math verktyg), *122*, *123*
iPart( ), heltalsdel, *140*, *459*
isPrime( ), primtalstest, *459*
Item, menyalternativ, *302*, *303*, *460*

K

klipp ut, *95*, *321*. *omslagets bakre insida*
klistra in, *95*, *96*, *321*. *omslagets bakre insida*
kombinationer, **nCr( )**, *471*
kommandomärkning, *328*
kommandon, *409 – 542*
kommandoskript, *94*, *328*, *329*
övning, *392*
kommentar, ◉, *282*, *542*. *omslagets främre insida*, *omslagets bakre insida*
komplex
faktor, **cFactor( )**, *71*, *406*, *419*, *560*
konjugera, **conj( )**, *422*
lägesinställning, Complex Format, *41*, *547*
lösning, **cSolve( )** , *61*, *425*, *560*
modulus yta, *170*
nollställen, **cZeros( )**, *61*, *71*, *429*, *560*
komplexa
tabeller, *227*
tal, *8*, *559 – 561*
konstanter, *81 – 91*, *83*
fördefinierade, *89*, *90*, *91*
kontrast, justera, *4*, *15*. *omslagets främre insida*, *omslagets bakre insida*
konturnivåplottning, *155*, *166*, *568*
konturplottning, *167*, *168*, *169*
DrwCtour, rita kontur, *168*
kopiera, *95*, *96*, *321*. *omslagets bakre insida*
kopiera variabel, **CopyVar**, *289*, *358*, *422*
kör assemblerspråk, **Exec**, *314*, *443*
kör program, **Prgm**, *276*, *287*, *482*
kumulativ summa, **cumSum( )**, *250*, *428*
kurvfamilj, *208*, *209*
kvadratisk regression, **QuadReg**, *263*, *488*, *567*
kvadratrot, √**( )**, *536*
kvartär regression, **QuartReg**, *263*, *488*, *567*

L

labbrapporter, *330*, *331*
Labels grafformat, *114*
lägen
hämta/returnera, **getMode( )**, *300*, *453*
inställningar, **setMode( )**, *300*, *305*, *498*
lägesinställningar, *40*, *546 – 550*
Angle, *41*, *108*, *547*
Approximate, *29*, *41*, *54*, *62*, *549*
Auto, *29*, *41*, *54*, *63*, *549*
Base, *42*, *550*
Complex Format, *41*, *547*
Current mapp, *41*, *546*
Custom Enheter, *42*, *550*
Display Digits, *31*, *41*, *546*
Exact/Approx, *29*, *41*, *54*, *61*, *62*, *63*, *549*
Exponential Format, *31*, *41*, *547*

L (fort.)

Graf, *41*, *54*, *108*, *130*, *136*, *142*, *157*, *179*, *546*
Graf 2, *41*, *549*
inställningar i program, *300*
Language, *42*, *550*
Number of Grafs, *41*, *549*
Pretty Print, *29*, *41*, *548*
Split App, *41*, *549*
Split Screen, *41*, *548*
Unit System, *42*, *82*, *550*
Vektor Format, *41*, *548*
lägg till, **&**, *293*, *534*. *omslagets främre insida*, *omslagets bakre insida*
lagra
bild, **StoPic**, *306*
symbol, **→**, *289*, *541*
Language lägesinställning, *42*, *550*
länka och överföra, *495*, 496
avbryta, *368*
fel, *369*, *376*, *377*
skicka chat, **SendChat**, 309, 371
skicka till räknare, **SendCalc**, *309*, *371*
länkning och överföring, *365* – *381*
Flash-program, *367*, *370*
hämta/returnera CBL/CBR-värde, **Get**, *273*, *309*, *451*
inkompatibilitet, *380*, *381*
kompatibilitet, *380*, *381*
mappar, *367*, *368*, *369*
program, *309*, *371*
räknare till räknare, *309*, *366*, *367*, *371*, *372*
skicka listvariabel, **Send**, *309*, *495*
variabler, *367*, *368*
lås upp, **Unlock**, *289*, *519*
lås variabel, **Lock**, *289*, *465*
Lbl, etikett, *287*, *296*, *299*, *460*
lcm, minsta gemensamma multipel, *460*
Leading Cursor grafformat, *114*
left( ), vänster, *71*, *293*, *460*
lika med, **=**, *294*, *532*
limit( ), gräns, *66*, *75*, *76*, *461*
Line, rita linje, *308*, *462*
LineHorz, rita horisontell linje, *308*, *462*
LineTan, rita tangent, *308*, *462*
LineVert, rita vertikal linje, *308*, *463*
LinReg, linjär regression, *262*, *463*, *566*
list▸mat( ), lista till matris, *249*, *463*
lista skillnader, **Δlist( )**, *463*
listor. *Se även* data/matrix editor
Auto-calculate, *249*
differens, **Δlist( )**, *463*
dimension, **dim( )**, *437*
dot produkt, **dotP( )**, *438*
kolumnrubrik, *248*, *249*, *250*
kopiera, *252*
korsprodukt, **crossP( )**, *425*
kumulativ summa, **cumSum( )**, *250*, *428*
låsa, *248*
lista till matris, **list▸mat( )**, *249*, *463*
matris till lista, **mat▸list( )**, *467*
maximum, **max( )**, *467*
minimum, **min( )**, *470*
mittsträng, **mid( )**, *469*
ny data, **NewData**, *240*, *249*, *289*, *472*
ny, **newList( )**, *472*
operationer, *410*
produkt, **product( )**, *483*
sammanfoga/konkatenera, **augment( )**, *417*
sätta in, *246*, *247*
skapa, *241*, *242*
sortera kolumner, *251*
sortera stigande, **SortA**, *508*
summa, **sum( )**, *493*, *511*
ta bort, *246*, *247*
tabellvariabler, *230*
uttryck till lista, **exp▸list( )**, *444*
variabler, *239*, *241*, *242*
ljusare/mörkare, *4*, *15*. *omslagets främre insida*, *omslagets bakre insida*
ln( ), naturlig logaritm, *464*
LnReg, logaritmisk regression, *262*, *464*, *567*
Local, lokal variabel, *286*, *288*, *289*, *290*, *465*
Lock, lås variabel, *289*, *465*
log( ), logaritm, *465*
logaritmer, *464*, *465*
Logistic, logistisk regression, *262*, *466*, *567*
Loop, slinga, *299*, *466*
lös, **solve( )**, *9*
lösa linjära ekvationer, *9*, *10*, *73*
lösning, **deSolve( )**, *75*, *196*, *433*
LU, matris undre-övre uppdelning, *467*
lutningsfält, SLPFLD, *180*, *185*, *197*

M

mappar, *41*, *100*, *546*
byta namn, *358*
hämta/returnera, **getFold( )**, *453*
inställningar, **setFold( )**, *101*, *300*, *496*
klistra in namn, *359*
låsa/låsa upp, *358*
ny, **NewFold**, *101*, *289*, *472*
ta bort, *357*
ta bort, **DelFold**, *102*, *289*, *433*
VAR-LINK, *102*, *356*, *357*, *358*
överföra, *367*, *368*
markera text, *32*, *320*. *omslagets främre insida*, *omslagets bakre insida*
markör
3D-graf, *160*
bredvid kurva, *161*
flytta, *16*, *17*, *32*. *omslagets främre insida*, *omslagets bakre insida*

M (fort.)

fritt rörlig, *116*, *132*, *138*, *145*, *159*, *183*
gömd yta, *161*
spåra, *117*
mat▸list( ), matris till lista, *467*
MATH meny, *34*, *122*
math operationer, *411*
matris till lista, **mat▸list( )**, *467*
matriser. *Se även* data/matrix editor
Auto-calculate, *249*
data från en graf, *203*
determinant, **det( )**, *435*
diagonal, **diag( )**, *435*
dimension, **dim( )**, *437*
dot addition, **.+**, *533*
dot division, **./**, *534*
dot multiplikation, **.∗**, *534*
dot potens, **.^**, *534*
dot subtraktion, **.−**, *534*
egenvärde, **eigVl( )**, *442*
egenvektor, **eigVc( )**, *441*
fylla, **Fill**, *448*
identitet, **identity( )**, *457*
kolumndimension, **colDim( )**, *421*
kolumnnorm, **colNorm( )**, *421*
kolumnrubrik, *248*, *249*, *250*
kopiera, *252*
kumulativ summa, **cumSum( )**, *250*, *428*
låsa, *248*
lista till matris, **list▸mat( )**, *463*
matris till lista, **mat▸list( )**, *467*
maximum, **max( )**, *467*
minimum, **min( )**, *470*
ny data, **NewData**, *289*, *472*
ny, **newMat( )**, *472*
operationer, *411*
pretty print, *240*
produkt, **product( )**, *483*
QR faktorisering, **QR**, *487*
radaddition, **rowAdd( )**, *494*
radbyte, **rowSwap( )**, *494*
raddimension, **rowDim( )**, *494*
radnorm, **rowNorm( )**, *494*
radoperation, **mRow( )**, *470*
radtrappsstegsform, **ref( )**, *491*
reducerad radtrappsstegsform, **rref( )** , *73*, *388*, *495*
row multiplikation och addition, **mRowAdd( )**, *471*
sammanfoga/konkatenera, **augment( )**, *388*, *417*
sätta in, *246*, *247*
skapa, *241*, *242*
slumptal, **randMat( )**, *388*, *489*
sortera kolumner, *251*
submatris, **subMat( )**, *511*
summa, **sum( )**, *493*, *511*
ta bort, *246*, *247*
transponera, ᵀ, *512*
undre-övre uppdelning, **LU**, *467*
variabler, *239*, *240*, *241*, *242*
max( ), maximum, *467*
Maximum (graf math verktyg), *122*, *123*
mean( ), medel, *468*
meddelanden. *Se även* felstatus och felsökning
BATT, *54*, *572*, *575*
false, *80*
Garbage collection, *362*, *363*
otillräckligt minne, <<...>>, *103*
true, *80*
undef (odefinierat), *80*
median( ), median, *468*
MedMed, medium-medium linje regression, *263*, *469*, *567*
Memory (zoom), *119*, *121*
Memory felstatus, *364*
meny item, **Item**, *302*, *303*, *460*
menyer, *34*
Algebra, *70*, *72*
anpassade, *303*, *304*
använda, *34*
APPLICATIONS, *34*, *38*
Calc, *75*
CATALOG, *44*
CHAR (character), *34*
Clean Up, *43*
Complex, *71*
CUSTOM, *34*, *37*
Extract, *71*
MATH, *34*, *122*
Trig, *71*
verktygsfält, *34*, *37*
mid( ), mittsträng, *293*, *469*
min( ), minimum, *470*
mindre än eller lika med, ≤, **<=**, *294*, *533*. *omslagets främre insida*, *omslagets bakre insida*
mindre än, **<**, *294*, *532*
Minimum (graf math verktyg), *11*, *122*, *123*
minne, *349* – *364*
arkivera, **Archive**, *289*, *361*, *416*
hämta från användarmninne, **Unarchiv**, *289*, *361*, *519*
kontrollera, *353*, *354*
otillräckligt minne, <<...>>, *103*
VAR-LINK skärm, *355*, *356*, *357*, *358*, *361*
återställning, *353*, *354*
minsta gemensamma multipel, **lcm**, *460*
minut-notation, ', *538*
mittsträng, **mid( )**, *293*, *469*
mod( ), modulo, *470*
mörkare/ljusare, *4*, *15*. *omslagets främre insida*, *omslagets bakre insida*
MoveVar, flytta variabel, *289*, *470*
mRow( ), matrisradoperation, *470*

M (fort.)

mRowAdd( ), matrisrad multiplikation och addition, *471*
multiplicera, *, *530*

N

nämnare, *421*
nätplottning. *Se* vävplottning
naturlig logaritm, **ln( )**, *464*
ncontour window-variabel, *158*
nCr( ), kombinationer, *471*
ncurves window-variabel, *182*
nDeriv( ), numerisk derivata, *75*, *471*
negation, -, *25*
negera, -, *531*
NewData, ny data, *240*, *249*, *274*, *289*, *472*
NewFold, ny mapp, *101*, *289*, *472*
newList( ), ny lista, *472*
newMat( ), ny matris, *472*
NewPic, ny bild, *289*, *306*, *472*
NewPlot, ny plot, *267*, *305*, *473*
NewProb, nytt problem, *43*, *473*
nInt( ), numerisk integral, *75*, *473*
nmax window-variabel, *143*, *144*
nmin window-variabel, *143*, *144*
noggranhet, *562*
nollställen, övning, *402*
nollställen, **zeroes( )**, *61*, *70*, *74*, *384*, *522*
norm( ), Frobenius norm, *474*
not, Booleskt not, *294*, *474*
nPr( ), permutationer, *475*
nSolve( ), numerisk lösning, *70*, *475*
Number of Grafs lägesinställning, *41*, *549*
numerisk
  derivata, **nDeriv( )**, *75*, *471*
  integral, **nInt( )**, *75*, *473*
  lösning, **nSolve( )**, *70*, *475*
ny
  bild, **NewPic**, *289*, *306*, *472*
  data, **NewData**, *240*, *249*, *274*, *289*, *472*
  lista, **newList( )**, *472*
  mapp, **NewFold**, *101*, *289*, *472*
  matris, **newMat( )**, *472*
  plot, **NewPlot**, *267*, *305*, *473*
  problem, **NewProb**, *43*, *473*

O

oändlighet, ∞, *80*
oberoende auto/fråga, Independent AUTO/ASK, *224*, *226*, *229*
omvandla, ▸, *85*, *539*
OneVar, en-variabel-statistik, *262*, *476*
operationer, *409 – 542*
operatorer, *26*
or, Booleskt or, *294*, *347*, *476*
ord( ), numerisk teckenkod, *293*, *477*, *551*
otillräckligt minne, <<...>>, *103*
Out-of-memory felstatus, *79*
Output, utmatning, *302*, *477*

P

P▸Rx( ), rektangulär x-koordinat, *477*
P▸Ry( ), rektangulär y-koordinat, *477*
på/av, *4*, *7*, *14*. *omslagets främre insida*, *omslagets bakre insida*
panorering, *118*
parallellpiped övning, *390*
parameterplottning, *127 – **1**32*
parenteser, hakparenteser, och vinklar, *27*, *564*
part( ), del, *478*
PassErr, skicka fel, *310*, *480*
Pause, paus, *302*, *310*, *481*
PAUSE-indikator, *54*
Permanent minne, *14*
permutationer, **nPr( )**, *475*
pixel
  ändra, **PxlChg**, *307*, *484*
  av, **PxlOff**, *307*, *485*
  cirkel, **PxlCrcl**, *308*, *484*
  horisontell linje, **PxlHorz**, *308*, *485*
  linje, **PxlLine**, *216*, *308*, *485*
  på, **PxlOn**, *216*, *307*, *485*
  test, **pxlTest( )**, *307*, *486*
  text, **PxlText**, *307*, *486*
  vertikal linje, **PxlVert**, *308*, *486*
PlotsOff, plottning av, *111*, *305*, *481*
PlotsOn, plottning på, *111*, *305*, *481*
plotStep window-variabel, *143*, *144*
plotStrt window-variabel, *143*, *144*
plottning
  av, **PlotsOff**, *111*, *305*, *481*
  data, *254 – **2**58*
  markera, *266*, *269*
  ny, **NewPlot**, *267*, *305*, *473*
  på, **PlotsOn**, *111*, *305*, *481*
  rensa, *266*
  spåra, *270*
  visningsfönster, *270*
  Y= Editor, *269*, *270*
polär
  koordinat, **R▸Pθ( )**, *489*
  koordinat, **R▸Pr( )**, *489*
  plottning, *133 – **1**38*
  vektordisplay, **▸Polar**, *481*
polyEval( ), utvärdera polynom, *482*
polynom, *9*, *72*, *76*
  slumptal, **randPoly( )**, *490*
  utvärdera, **polyEval( )**, *482*
  övning, *402*
PopUp, popupmeny, *301*, *482*
potens, **^**, *536*, *565*
PowerReg, potensregression, *263*, *482*, *567*
pretty print, *6*, *11*, *23*, *29*
Pretty Print lägesinställning, *29*, *41*, *548*

P (fort.)

Prgm, kör program, *276*, *287*, *482*
prim, ', *538*
primtal, *8*
primtalstest, **isPrime( )**, *459*
problem (nytt), **NewProb**, *43*, *473*
problem vid användning. *Se* felstatus och felsökning
procent, **%**, *531*
product( ), produkt, *483*
produkt ID, *55*
produkt, **Π( )**, *75*, *536*
produktkod, uppgradering, *373*, *374*
program och programmering, *275 – 314*
  aktivera anpassat verktygsfält, **CustmOn**, *37*, *302*, *428*
  anropa andra program, *287*
  argument, *284*
  assemblerspråk, *313*, *314*
  CBL, *309*, *399*
  CBR, *309*, *399*
  definiera dialogruta **Dialog**, *302*, *436*
  definiera verktygsfält, **Custom**, *302*, *428*
  definiera verktygsfält, **Toolbar**, *302*, *518*
  definiera, **Define**, *287*, *305*, *384*, *432*
  else if, **ElseIf**, *207*, *296*, *442*
  else, **Else**, *296*, *457*
  end custom, **EndCustm**, *302*, *428*
  end dialog, **EndDlog**, *302*, *436*
  end for, **EndFor**, *283*, *297*, *450*
  end function, **EndFunc**, *207*, *286*, *451*
  end if, **EndIf**, *283*, *295*, *296*, *457*
  end loop, **EndLoop**, *299*, *466*
  end program, **EndPrgm**, *276*, *287*, *482*
  end toolbar, **EndTBar**, *302*, *518*
  end try, **EndTry**, *310*, *518*
  end while, **EndWhile**, *298*, *521*
  etikett, **Lbl**, *287*, *296*, *299*, *460*
  exekvera assemblerspråk, **Exec**, *314*, *443*
  exekvera program, **Prgm**, *276*, *287*, *482*
  exit, **Exit**, *444*
  felsökning, *310*
  for, **For**, *283*, *297*, *450*
  förfrågan, **Request**, *301*, *302*, *492*
  formatsträng, **format( )**, *302*, *450*
  funktion, **Func**, *207*, *286*, *451*
  funktioner, *280*, *285*, *286*
  go to, **Goto**, *287*, *296*, *299*, *455*
  grafer, *305*
  grafiskt användargränssnitt, GUI, *302*
  hämta/returnera enheter, **getUnits( )**, *300*, *455*
  hämta/returnera från räknare, **GetCalc**, *309*, *371*, *452*
  hämta/returnera konfiguration, **getConfg( )**, *300*, *452*
  hämta/returnera mapp, **getFold( )**, *300*, *453*
  hämta/returnera tangent, **getKey( )**, *301*, *453*, *552*, *555*
  if, **If**, *207*, *283*, *295*, *296*, *457*
  inmatning, *279*, *283*, *301*
  inmatning, **Input**, *301*, *305*, *458*
  kommentarer, ◉, *282*, *542*
  konditionella test, *294*
  konditioner, *283*, *295*, *296*
  kopiera, *281*
  köra, *278*. *omslagets främre insida*, *omslagets bakre insida*
  läget hämta/returnera, **getMode( )**, *300*, *453*
  listmeny, **DropDown**, *302*, *440*
  lokal, **Local**, *286*, *288*, *289*, *290*, *465*
  loop, **Loop**, *299*, *466*
  mata in, *280*, *281*, *282*, *283*
  meny item, **Item**, *302*, *303*, *460*
  menyer, *303*, *304*
  operationer, *412*
  pass error, **PassErr**, *310*
  paus, **Pause**, *302*, *310*, *481*
  popupmeny, **PopUp**, *301*, *482*
  prompt, **Prompt( )**, *301*, *483*
  rader med flera kommandon, *282*
  radera felstatus, **ClrErr**, *310*, *420*
  radera graf, **ClrGraf**, *205*, *305*, *420*
  radera grundfönstret, **ClrHome**, *421*
  radera I/O, **ClrIO**, *279*, *302*, *421*
  radera tabell, **ClrTable**, *421*
  returnera, **Return**, *286*, *287*, *492*
  skicka fel, **PassErr**, *480*
  skicka värden, *284*
  slingor, *283*, *297*, *298*
  stäng av anpassat verktygsfält, **CustmOff**, *37*, *302*, *428*
  stopp, **Stop**, *282*, *510*
  stoppa, *278*
  subrutiner, *287*
  ta bort, *281*
  tabeller, *305*
  text, **Text**, *302*, *516*
  Then, **Then**, *295*, *296*, *457*
  title, **Title**, *302*, *516*
  try, **Try**, *310*, *518*
  utmatning, *279*, *283*, *301*, *302*
  utmatning, **Output**, *302*, *477*
  variabler, *288*
  visa graf, **DispG**, *302*, *305*, *437*
  visa grundfönster, **DispHome**, *302*, *438*
  visa I/O-skärm, **Disp**, *277*, *283*, *302*, *310*, *437*, *551*
  visa tabell, **DispTbl**, *302*, *305*, *438*
  while, **While**, *298*, *521*
programversion, *55*
Prompt( ), prompt, *301*, *483*
propFrac, egentligt bråk, *9*, *70*, *74*, *394*, *483*
PtChg, punkt ändra, *307*, *483*
PtOff, punkt av, *307*, *484*

P (fort.)

PtOn, punkt på, *307*, *484*
ptTest( ), punkt test, *307*, *484*
PtText, punkt text, *307*, *484*
punkt
  addition, **.+**, *533*
  av, **PtOff**, *307*, *484*
  division, **./**, *534*
  multiplikation, **.∗**, *534*
  på, **PtOn**, *307*, *484*
  potens, **.^**, *534*
  produkt, **dotP( )**, *438*
  subtraktion, **.−**, *534*
  test, **ptTest( )**, *307*, *484*
  text, **PtText**, *307*, *484*
  ändra, **PtChg**, *307*, *483*
PxlChg, pixel ändra, *307*, *484*
PxlCrcl, pixel cirkel, *308*, *484*
PxlHorz, pixel horisontell linje, *308*, *485*
PxlLine, pixel linje, *216*, *308*, *485*
PxlOff, pixel av, *307*, *485*
PxlOn, pixel på, *216*, *307*, *485*
pxlTest( ), pixel test, *307*, *486*
PxlText, pixel text, *307*, *486*
PxlVert, pixel vertikal linje, *308*, *486*

Q

QR, QR faktorisering, *487*
QuadReg, kvadratisk regression, *263*, *488*, *567*
QuartReg, kvartär regression, *263*, *488*, *567*
QuickCenter, *118*
Quick-Find sökning, *410*

R

R▸Pθ( ), polär koordinat, *489*
R▸Pr( ), polär koordinat, *489*
radera
  felstatus, **ClrErr**, *310*, *420*
  graf, **ClrGraf**, *205*, *305*, *340*, *420*
  home, **ClrHome**, *421*
  I/O, **ClrIO**, *279*, *302*, *421*
  ritning, **ClrDraw**, *213*, *307*, *420*
radian, ʳ, *537*
radtrappsstegsform, **ref( )**, *491*
räkneoperationer, *410*
rand( ), slumptal, *489*
randMat( ), slumpmatris, *388*, *489*
randNorm( ), slumpnorm, *490*
randPoly( ), slumppolynom, *490*
RandSeed, slumptalsvärde, *388*, *490*
rationella funktioner övning, *394*
RclGDB, hämta grafdatabas, *220*, *306*, *490*
RclPic, hämta bild, *306*, *490*
real( ), real, *490*
reciproka, x^{-1}, *540*
redigera, *32*
reducerad radtrappsstegsform, **rref( )**, *73*, *388*, *495*
ref( ), radtrappsstegsform, *491*
regression, *463*
  andraordningens, **QuadReg**, *263*, *488*, *567*
  exponentiell, **ExpReg**, *262*, *446*, *566*
  fjärdeordningens, **QuartReg**, *263*, *488*, *567*
  formler, *566*, *567*
  kvadratisk formel övning, *386*
  linjär regression, **LinReg**, *262*, *463*, *566*
  logaritmisk, **LnReg**, *262*, *464*, *567*
  logistisk, **Logistisk**, *262*, *466*, *567*
  medium-medium linje, **MedMed**, *263*, *469*, *567*
  sinusoidal, **SinReg**, *263*, *505*, *567*
  tredjeordningens, **CubicReg**, *262*, *427*, *566*
  upphöjt till regression, **PowerReg**, *263*, *482*, *567*
  välja, *262*
rektangulär x-koordinat, **P▸Rx( )**, *477*
rektangulär y-koordinat, **P▸Ry( )**, *477*
rektangulär-vektordisplay, **▸Rect**, *491*
remain( ), rest, *492*
Rename, byt namn, *289*, *492*
Request, förfrågan, *301*, *302*, *492*
reserverade namn, *563*, *564*
resultat (senaste), **ans( )**, *50*, *416*
Return, returnera, *207*, *286*, *287*, *492*
returnera. *Se* hämta/returnera
right( ), höger, *71*, *293*, *492*
riktningsfält, DIRFLD, *180*, *185*, *198*
ritningar och ritning
  cirkel, **Circle**, *308*, *419*
  cirklar, *214*
  frihand, *213*
  funktion, **DrawFunc**, *212*, *308*, *439*
  horisontell linje, **LineHorz**, *308*, *462*
  invers, **DrawInv**, *212*, *308*, *439*
  kontur, **DrwCtour**, *308*, *440*
  linje, **Line**, *308*, *462*
  linjer, *214*, *215*
  lutning, **DrawSlp**, *215*, *308*, *440*
  på en graf, *307*
  parametrisk, **DrawParm**, *212*, *308*, *439*
  Pencil, *213*
  polär, **DrawPol**, *212*, *308*, *439*
  radera, **ClrDraw**, *214*, *307*, *420*
  tangent, **LineTan**, *308*, *462*
  vertikal linje, **LineVert**, *308*, *463*
rotate( ), rotera, *293*, *348*, *493*
round( ), avrunda, *494*
rowAdd( ), matris radaddition, *494*
rowDim( ), matris raddimension, *494*
rowNorm( ), matris radnorm, *494*

R (fort.)

rowSwap( ), matris radbyte, *494*
RplcPic, byt ut bild, *306*, *495*
rref( ), reducerad radtrappsstegsform, *73*, *388*, *495*
rullgardinsmeny, **DropDown**, *302*, *440*
rullning, *7*, *103*, *227*
Runge-Kutta metod, *180*, *191*, *193*, *569*

S

Scatter plottning, *267*
scientific notation, *25*
sekvens, **seq( )**, *496*
sekvensplottning, *139 – **1**51*
senaste inmatning, 20, 49, 50
senaste resultat, 20, 28, 49, 51
Send, skicka listvariabel, *309*, *495*
SendCalc, skicka till räknare, *309*, *371*, *495*
SendChat, skicka chat, *309*, *371*, *496*
seq( ), sekvens, *496*
serienummer, *55*
serviceinformation, *576*
Set factors (zoom), *119*, *121*
setFold( ), ställ in mapp, *101*, *300*, *496*
setGraf( ), ställ in graf, *300*, *305*, *497*
setMode( ), ställ in lägesinställning, *300*, *305*, *498*
setTable( ), ställ in tabell, *225*, *300*, *305*, *499*
setUnits( ), ställ in enheter, *300*, *499*
sfärisk vektordisplay, **▸Sphere**, *509*
Shade (graf math verktyg), *122*, *126*
Shade, skugga, *308*, *500*
shift( ), skift, *250*, *293*, *348*, *501*
ShowStat, visa statistiska resultat, *263*, *502*
sign( ), tecken, *502*
simult( ), simultana ekvationer, *73*, *502*
sin^{-1}( ), arcussinus, *503*
sin( ), sinus, *503*
sinh^{-1}( ), hyperbolisk arcussinus, *504*
sinh( ), hyperbolisk sinus, *504*
SinReg, sinusoidal regression, *263*, *505*, *567*
skapa väv, Build Web, *146*, *147*
skicka chat, **SendChat**, *309*, *371*, *496*
skicka listvariabel, **Send**, *309*, *495*
skicka till räknare, **SendCalc**, *309*, *371*, *495*
skift, **shift( )**, *250*, *293*, *348*, *501*
skript, *94*, *328*, *329*
  handledning, *392*
  övning, *392*
skugga, **Shade**, *308*, *500*
slinga, **Loop**, *299*, *466*
SLPFLD, lutningsfält, *180*, *185*, *197*
slump
  matris, **randMat( )**, *388*, *489*
  norm, **randNorm( )**, *490*
  polynom, **randPoly( )**, *490*
  startvärde, **RandSeed**, *388*, *490*
  tal, **rand( )**, *489*
Smart Graph, *115*
Solution Method grafformat, *180*
solve( ), lös, *9*, *58*, *61*, *62*, *63*, *66*, *68*, *70*, *73*, *196*, *505*
SortA, sortera stigande, *508*
SortD, sortera fallande, *508*
spåra, *117*, *118*, *132*, *138*, *145*, *159*, *183*
spåra, **Trace**, *305*, *390*, *398*, *399*, *402*, *518*
spårning, *11*
specialtecken, *21*, *324*, *325*
Split App lägesinställning, *41*, *549*
Split Screen lägesinställning, *41*, *548*
språk, *4*
språkanpassning, *4*
ställ in
  enheter, **setUnits( )**, *300*, *499*
  graf, **setGraf( )**, *300*, *305*, *497*
  lägesinställning, **setMode( )**, *300*, *305*, *498*
  mapp, **setFold( )**, *101*, *300*, *496*
  tabell, **setTable( )**, *225*, *300*, *305*, *499*
standardannuitet övning, *404*
standardavvikelse, **stdDev( )**, *509*
startvärden, *184*
statistik, *253 – **2**74*. *Se även* regression
  Box Plot, *267*
  Calculation Type, *260*, *262*
  Category, *260*, *261*
  en-variabel statistik, **OneVar**, *262*, *476*
  fakultet, **!** , *8*, *534*
  frekvens, *271*, *272*
  Freq, *260*, *261*
  Histogram plottning, *268*
  kategorier, *271*, *272*
  kombinationer, **nCr( )**, *471*
  medel, **mean( )**, *468*
  median, **median( )**, *468*
  ny plottning, **NewPlot**, *267*, *473*
  operationer, *412*
  permutationer, **nPr( )**, *475*
  plottning, *265*, *266*, *267*, *268*, *269*, *270*
  plottning av, **PlotsOff**, *111*, *305*, *481*
  plottning på, **PlotsOn**, *111*, *305*, *481*
  Scatter-plottning, *267*
  slumpnorm, **randNorm( )**, *490*
  slumptal, **rand( )**, *489*
  slumptalsstart, **RandSeed**, *388*, *490*
  standardavvikelse, **stdDev( )**, *509*
  två-variabel resultat, **TwoVar**, *262*, *519*
  variabler, *261*, *264*
  varians, **variance( )**, *520*

S (fort.)

visa resultat, **ShowStat**, *263*, *502*
xyline plottning, *267*
översikt, *259*
statusrad, *53*, *54*, *108*
stdDev( ), standardavvikelse, *509*
stegvisa funktioner, *202*, *206*
stil, **Style**, *305*, *511*
StoGDB, spara grafdatabas, *220*, *306*, *510*
Stop, stopp, *282*, *510*
StoPic, lagra bild, *306*, *510*
storing
graph database, **StoGDB**, *220*, *306*, *510*
picture, **StoPic**, *510*
större än eller lika med, ≥, **>=**, *294*, *533*. *omslagets främre insida*, *omslagets bakre insida*
större än, **>**, *294*, *533*
största gemensamma nämnare, **gcd( )**, *451*
sträng
del av, **InString**, *293*, *458*
dimension, **dim( )**, *293*, *437*
format, **format( )**, *450*
höger, **right( )**, *293*, *492*
inriktning, **#**, *293*, *537*, *565*
lägg till, **&**, *293*, *534*
mitt, **mid( )**, *293*, *469*
operationer, *413*
rotera, **rotate( )**, *293*, *493*
skift, **shift( )**, *293*, *501*
skriva in, **InputSt**, *292*, *301*, *371*
sträng till uttryck, **expr( )**, *292*, *293*, *301*, *381*, *446*
teckenkod, **ord( )**, *293*, *477*, *551*
uttryck till sträng, **string( )**, *293*, *510*
vänster, **left( )**, *293*, *460*
strängar
format, **format( )**, *293*, *302*
operationer, *292*, *293*
teckensträng, **char( )**, *293*, *419*, *551*
string( ), uttryck till sträng, *293*, *510*
Style, stil, *112*, *305*, *511*
subMat( ), submatris, *511*
substituera, *67*, *68*, *69*
subtrahera, **–**, *529*
sum( ), summa, *493*, *511*
summa, **Σ( )**, *75*, *536*
switch( ), växla, *300*, *512*
symbolisk manipulation, *57 – 80*
sysdata, systemdata, *203*
systemdata, sysdata, *203*
systemvariabler, *563*, *564*

T

t0 window-variabel, *181*
ta bort
mapp, **DelFold**, *102*, *289*, *433*
variabel, **DelVar**, *60*, *77*, *102*, *289*, *291*, *433*
tabeller, *221 – 230*
Δtbl, *224*
automatiska, *226*
cellbredd, *227*, *230*
differentialekvationer, *199*
funktioner, *228*
Independent AUTO/ASK, *224*, *226*, *229*
inställningar, *225*
inställningar, **setTable( )**, *225*, *300*, *305*, *499*
inställningar, TABLE SETUP, *224*
komplexa tal, *227*
manuellt, *229*
plottning, Graf<->Tabell, *224*
program, *305*
radera, **ClrTable**, *421*
skapa med talföljder, *151*
skapa, **Table**, *305*, *513*
starta, tblStart, *224*
tblStart, *224*
visa, **DispTbl**, *302*, *305*, *438*
ökning, Δtbl, *224*
översikt, *223*
tabell-graf, Graf<->Tabell, *224*
TABLE SETUP, tabellinställningar, *224*
Table, skapa tabell, *305*, *513*
tak, **ceiling( )**, *418*
tal
irrationella, *61*, *62*
komplexa, *559 – 561*
negativa, *25*
rationella, *61*, *62*, *63*
talbaser, *343 – 348*
Booleska operationer, *347*
matematiska operationer, *346*
omvandlingar, *345*
tan^{-1}( ), arctangent, *514*
tan( ), tangent, *513*
Tangent (graf math verktyg), *122*, *125*, *132*, *138*
tangentbord, *16*, *17*
alpha (alfa), *18*
2nd (andrafunktion), *18*
◆ (diamant), *18*
(hand), *18*
↑ (skift), *18*
genvägar, *325*. *omslagets främre insida*, *omslagets bakre insida*
karta, *324*, *325*. *omslagets främre insida*, *omslagets bakre insida*
tangentkoder, *301*, *552*
tanh^{-1}( ), hyperbolisk arctangent, *514*
tanh( ), hyperbolisk tangent, *514*
taylor( ), Taylor-polynom, *75*, *76*, *515*

T (fort.)

tblStart, tabellstart, *224*
tCollect( ), trigonometrisk samling, *71*, *515*
tecken
  Grekiska, *325*, *326*, *327*. *omslagets främre insida*, *omslagets bakre insida*
  internationella, *21*, *324*, *325*, *326*. *omslagets bakre insida*
  koder, *551*
  meny, *34*
  numeriskt värde, **ord( )**, *293*, *477*, *551*
  special, *21*, *324*, *325*
  sträng, **char( )**, *293*, *419*, *551*
  symboler, *21*, *325*
  versaler/gemener, *21*, *319*. *omslagets främre insida*
tecken, **sign( )**, *502*
temperatur omvandling, **tmpCnv( )**, *86*, *517*
temperaturområde omvandling, **ΔtmpCnv( )**, *86*, *517*
testvärden övning, *407*
tExpand( ), trigonometrisk expansion, *71*, *516*
Text, text, *302*, *516*
textredigering, *315* – *331*
  dator, *322*
  klipp ut, kopiera, klistra in, *95*, *96*, *321*
  markera, *320*. *omslagets främre insida*, *omslagets bakre insida*
  söka, *321*
Then, Then, *295*, *296*, *457*
tidsplottning, TIME, *142*, *146*, *190*, *191*
TI-GRAPH LINK, *314*, *322*, *323*, *374*, *378*
TIME, tidsplottning, *142*, *146*, *190*, *191*
timmar/minuter/sekunder, **▸DMS**, *438*
Title, titel, *516*
tmax window-variabel, *131*, *181*
tmin window-variabel, *131*
tmpCnv( ), temperatur omvandling, *86*, *517*
Toolbar, verktygsfält, *302*, *518*
tplot window-variabel, *181*
Trace, spåra, *117*, *305*, *390*, *398*, *399*, *402*, *518*
tråd- och konturplottning, *155*, *166*
trådplottning, *155*, *166*
transponera, ᵀ, *512*
tredjeordningens regression, **CubicReg**, *262*, *427*, *566*
Trig meny, *71*
trigonometrisk expansion, **tExpand( )**, *71*, *516*
trigonometrisk samling, **tCollect( )**, *71*, *515*
true meddelande, *80*
Try, try, *310*, *518*
tstep window-variabel, *131*, *181*
TwoVar, två-variabel resultat, *262*, *519*

U

Unarchiv, hämta variabler från användarminnet, *289*, *361*, *519*
undef (odefinierat) meddelande, *80*
undermenyer, *35*
understrykningstecken, _, *539*
Unit System lägesinställning, *42*, *82*, *550*
unitV( ), unit vektor, *519*
Unlock, lås upp, *289*, *519*
uppgradera programkod, *373*, *374*
Urklipp, *95*, *96*, *321*
utmatning, **Output**, *302*, *477*
uttryck, *26*, *27*, *32*
  expandera, *9*
  förkorta, *9*
  sträng till uttryck, **expr( )**, *292*, *293*, *301*, *381*, *446*
  uttryck till lista, **exp▸list( )**, *444*
utvärdera polynom, **polyEval( )**, *482*

V

Value (graf math verktyg), *122*, *123*, *132*, *138*, *159*, *183*
vänster, **left( )**, *71*, *293*, *460*
värdeändring över tid övning, *405*
variabler, *47*, *48*
  byta namn, *358*
  data, *240*, *241*, *242*
  definierade, *337*
  definition, *59*
  flytta, **MoveVar**, *289*
  fördröjd förenkling, *66*
  gå förbi, *60*
  hämta från användarminne, **Unarchiv**, *289*, *361*, *519*
  i användarminnet, **Archive**, *289*, *361*, *416*
  i program, *359*
  klistra in namn, *359*
  kopiera, *358*
  kopiera, **CopyVar**, *289*, *358*, *422*
  låsa upp, **Unlock**, *289*
  låsa, **Lock**, *289*
  låsa/låsa upp, *54*, *358*
  lista, *239*, *241*, *242*
  lokala, **Local**, *286*, *288*, *289*, *290*, *465*
  matris, *239*, *240*, *241*, *242*
  odefinierade, *59*, *337*
  okända, lösa, *337*, *339*
  rensa, *341*
  reserverade namn, *563*, *564*
  spara, *100*
  spara i användarminne, **Archive**, *289*, *361*, *416*
  spara och hämta från användarminne, *360*
  statistik, *261*, *264*
  system, *563*, *564*

V (fort.)

ta bort, *369*
ta bort, **DelVar**, *60*, *77*, *102*, *289*, *291*, *433*
text, *94*
VAR-LINK, *102*, *355*, *356*, *357*, *358*, *361*
överföra, *366*, *368*
variance( ), varians, *520*
vävdiagram WEB, *142*, *146*, *147*
vävplottning
divergens, *148*
konvergens, *148*
oscillation, *149*
Vector Format lägesinställning, *41*, *548*
vektorer
cylindrisk vektordisplay, **▸Cylind**, *429*
dot produkt, **dotP( )**, *438*
enhet, **unitV( )**, *519*
korsprodukt, **crossP( )**, *425*
Vector Format lägesinställning, *41*, *548*
verkligt bråktal, **propFrac**, *9*
verktygsfält
av, **CustmOff**, *37*, *428*
definiera, **Custom**, *302*, *428*
på, **CustmOn**, *37*, *428*
vinkel, ∠, *538*
vinkel, **angle( )**, *415*
visa som
binär, **▸Bin**, *345*, *417*
cylindrisk vektor, **▸Cylind**, *429*
decimal vinkel, **▸DD**, *432*
decimalt heltal, **▸Dec**, *345*, *432*
hexadecimal, **▸Hex**, *345*, *456*
polär vektor, **▸Polär**, *481*
rektangulär vektor, **▸Rect**, *491*
sfärisk vektor, **▸Sphere**, *509*
timmar/minuter/sekunder, **▸DMS**, *438*
visa statistiska resultat, **ShowStat**, *263*, *502*
visningsorbit, *164*
visningsvinkel, *162*

W

WEB, vävdiagram, *142*, *146*, *147*
when( ), when, *202*, *206*, *520*
While, while, *298*, *521*
window-variabel
θmax, *137*
θmin, *137*
θstep, *137*
Δx, *562*
Δy, *562*
diftol, *182*
dtime, *182*
Estep, *182*
eyeψ (rotation), *158*, *162*, *163*
eyeθ (x axis), *158*, *162*
eyeϕ (z axis), *158*, *162*, *163*
fldres, *182*
ncontour, *158*
ncurves, *182*
nmax, *143*, *144*
nmin, *143*, *144*
plotStep, *143*, *144*
plotStrt, *143*, *144*
t0, *181*
tmax, *131*, *181*
tmin, *131*
tplot, *181*
tstep, *131*, *181*
xgrid, *158*
xmax, *113*, *131*, *137*, *143*, *144*, *158*, *182*, *562*
xmin, *113*, *131*, *137*, *143*, *144*, *158*, *182*, *562*
xres, *113*, *131*, *158*
xscl, *113*, *131*, *137*, *143*, *144*, *158*, *182*
ygrid, *158*
ymax, *113*, *131*, *137*, *143*, *144*, *158*, *182*, *562*
ymin, *113*, *131*, *137*, *143*, *144*, *158*, *182*, *562*
yscl, *113*, *131*, *137*, *143*, *144*, *158*, *182*
zmax, *158*
zmin, *158*
with, |, *10*, *58*, *60*, *67*, *540*, *565*

X

x^{-1}, reciproka, *540*
xgrid window-variabel, *158*
xmax window-variabel, *113*, *131*, *137*, *143*, *144*, *158*, *182*, *562*
xmin window-variabel, *113*, *131*, *137*, *143*, *144*, *158*, *182*, *562*
xor, Booleskt exklusivt or, *294*, *347*, *521*
XorPic, exklusivt or picture, *306*, *522*
xres window-variabel, *113*
xscl window-variabel, *113*, *131*, *137*, *143*, *144*, *182*, *562*
xyline plottning, *267*

Y

Y= editor, *106*, *109*, *130*, *136*, *142*, *157*, *179*, *204*
ygrid window-variabel, *158*
ymax window-variabel, *113*, *131*, *137*, *143*, *144*, *158*, *182*, *562*
ymin window-variabel, *113*, *131*, *137*, *143*, *144*, *158*, *182*, *562*
yscl window-variabel, *113*, *131*, *137*, *143*, *144*, *182*, *562*

Z

Zero (graf math verktyg), *122*, *123*
zeroes( ), nollställen, *61*, *70*, *74*, *384*, *522*
zmax window-variabel, *158*
zmin window-variabel, *158*
zoom
data, **ZoomData**, *119*, *525*
decimal, **ZoomDec**, *119*, *525*
faktorer, *119*, *121*

Z (fort.)

föregående, **ZoomPrev**, *121*, *527*
heltal, **ZoomInt**, *119*, *526*
in, **ZoomIn**, *119*, *120*, *526*
kvadrat, **ZoomSqr**, *119*, *527*
Memory, *119*, *121*
passa, **ZoomFit**, *119*, *526*
ruta, **ZoomBox**, *119*, *120*, *524*
spara, **ZoomSto**, *121*, *528*
standard, **ZoomStd**, *119*, *528*
tidigare, **ZoomRcl**, *121*, *527*
trig, **ZoomTrig**, *119*, *528*
ut, **ZoomOut**, *119*, *120*, *527*
Zoom meny, *119*
ZoomBox, zoom ruta, *119*, *120*, *524*
ZoomData, zoom data, *119*, *525*
ZoomDec, zoom decimal, *119*, *525*
ZoomFit, zoom passa, *119*, *526*
ZoomIn, zoom in, *119*, *120*, *526*
ZoomInt, zoom heltal, *119*, *526*
ZoomOut, zoom ut, *119*, *120*, *527*
ZoomPrev, zoom föregående, *121*, *527*
ZoomRcl, zoom tidigare, *121*, *527*
ZoomSqr, zoom kvadrat, *119*, *527*
ZoomStd, zoom standard, *119*, *528*
ZoomSto, zoom spara, *121*, *528*
ZoomTrig, zoom trig, *119*, *528*

Ö

överföring. *Se* länkning och överföring
övningar. *Se* exempel, förhandsvisning, övningar

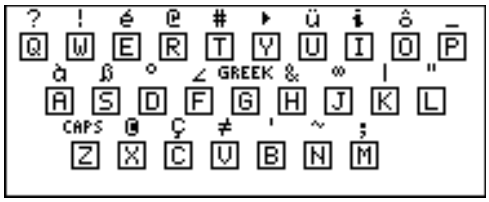
TI-92 Plus Snabbtangenter

Allmänna

◆ APPS	Lista med Flash-applikationer
2nd [⇥]	Växla mellan de två sista programmen eller delade skärmarna
◆ D	Kopiera grafkoordinater till sysdata
◆ F	Öppna dialogrutan FORMATS
◆ H	Kopiera grafkoordinater till grundfönstrets historiklista
◆ N	Skapa ny variabel
◆ O	Öppna variabel
◆ S	Spara kopia som
◆ −, ◆ +	Öka/minska kontrasten
◆ ENTER	Beräkna approximativt svar
◆ ON	Stänger av enheten så att den återgår till aktuellt program nästa gång den startas
◆ 1 – ◆ 9	Kör program kbdprgm1() till kbdprgm9()

Tangentbord på skärmen (◆ [KEY])

Tryck på ESC när du vill stänga det.



Se tabellen nedan för genvägar som inte är utmärkta på tagentbordet TI-92 Plus. Se nästa kolumn för accenter och grekiska tecken.

2nd Q	?	
2nd W	!	(faktorisera)
2nd R	@	
2nd T	#	(indirection)
2nd H	&	(lägg till)
2nd X	◉	(kommentar)
◆ =	≠	
◆ 0 (noll)	≤	
◆ .	≥	

Redigering

◆ ⌃	Flytta markören längst upp
◆ ⌄	Flytta markören längst ned
2nd ◂	Flytta markören längst till vänster
2nd ▸	Flytta markören längst till höger
⌃, ⌄	Bläddra stora objekt i historiklistan
2nd ⌃, 2nd ⌄	Sida upp och sida ned
◆ X	Klipp ut
◆ C	Kopiera
◆ V	Klistra in

3D-plottning

⌃, ⌄, ◂, ▸	Animera graf
+, −	Ändra animeringshastighet
X, Y, Z	Visa längs axlar
0 (zero)	Återgå till orginalvy
F	Ändra stil för graf
×	Expanderad/normal vy

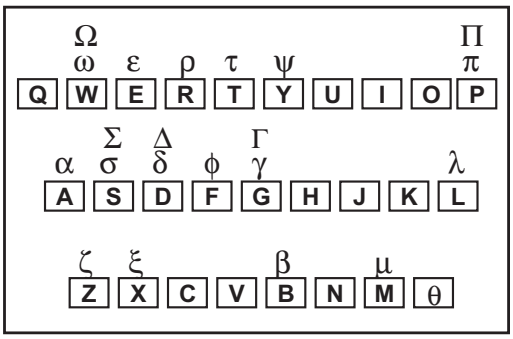
Accenttecken

2nd A + tecken	à, è, ì, ò, ù, À, È, Ì, Ò, Ù
2nd C + tecken	ç, Ç
2nd E + tecken	á, é, í, ó, ú, ý, Á, É, Í, Ó, Ú, Ý
2nd N + tecken	ã, ñ, õ, Ã, Ñ, Õ
2nd O + tecken	â, ê, î, ô, û, Â, Ê, Î, Ô, Û
2nd U + tecken	ä, ë, ï, ö, ü, ÿ, Ä, Ë, Ï, Ö, Ü

Grekiska tecken

2nd G	Använd grekiska tecken.
2nd G + tecken	Använd gemena grekiska tecken. Exempel: 2nd G W ger ω
2nd G ⇧ + tecken	Använd versala grekiska tecken. Exempel: 2nd G ⇧ W ger Ω

Om du trycker ned en kombination av tecken som inte motsvarar något grekiskt tecken används det normala tecknet för den tangenten.



Texas Instruments U.S.A.
7800 Banner Dr.
Dallas, TX 75251

Texas Instruments Holland B.V.
Rutherfordweg 102
3542 CG Utrecht - The Netherlands

Printed by:

ti-cares@ti.com
www.ti.com/calc

© 1999 Texas Instruments

S
XX/OM/1E6/A