

# TI-Nspire™ CX CAS Sovelluksen käsikirja

## **Tärkeitä tietoja**

Ellei muuten ilmoiteta ohjelman mukaan liitetyssä käyttöluvassa, Texas Instruments ei anna minkäänlaista suoraa tai välillistä takuuta mukaan lukien, mutta ei näihin rajoittuen, kaikki välilliset takuut, jotka koskevat kaikkien ohjelmien ja kirjojen myyntikelpoisuutta tai erityiseen tarkoitukseen sopivuutta, ja tarjoaa kyseisiä materiaaleja ainoastaan "sellaisina kuin ne ovat" -pohjalla. Texas Instruments ei ole missään tapauksessa vastuussa kenellekään mistään erityisistä, rinnakkaisista, tahattomista tai seuraussellista vaurioista näiden materiaalien hankinnan tai käytön aiheuttamana, ja Texas Instruments:n yksinomaisen ja eksklusiivinen vastuu toimintamuodosta riippumatta ei ylitä määrää, joka on asetettu käyttöluvassa ohjelmaa varten. Texas Instruments ei myöskään vastaa mistään vaateista, joita toinen osapuoli voi esittää aiheutuen näiden materiaalien käytöstä.

© 2023 Texas Instruments Incorporated

Todelliset tuotteet saattavat erota hieman mukana tulevista kuvista.

# Sisällys

<b>Lausekemallit</b> .....	<b>1</b>
<b>Luettelo aakkosjärjestyksessä</b> .....	<b>8</b>
A .....	8
B .....	17
C .....	21
D .....	48
E .....	61
F .....	72
G .....	83
I .....	93
L .....	102
M .....	119
N .....	128
O .....	138
P .....	141
Q .....	150
R .....	154
S .....	169
T .....	197
U .....	213
V .....	214
W .....	215
X .....	217
Z .....	219
<b>Symbolit</b> .....	<b>228</b>
<b>TI-Nspire™ CX II – Piirtokomennot</b> .....	<b>256</b>
Kuvaajien ohjelmointi .....	256
Kuvaajanäyttö .....	256
Oletusnäkyvä ja asetukset .....	257
Kuvaajanäytön virheviestit .....	258
Virheelliset komennot kuvaajatilassa .....	258
C .....	260
D .....	261
F .....	264
G .....	266
P .....	267
J .....	269
I .....	271

<b>Tyhjät elementit</b> .....	<b>272</b>
<b>Matemaattisten lausekkeiden syöttäminen pikavalintojen avulla</b> .....	<b>274</b>
<b>EOS-järjestelmän (yhtälökäyttäjärjestelmä) hierarkia</b> .....	<b>276</b>
<b>TI-Nspire CX II – TI-Basic-ohjelmointiominaisuudet</b> .....	<b>278</b>
Automaattinen sisennys Ohjelmointieditorissa .....	278
TI-Basicin virheviestien parannukset .....	278
<b>Vakiot ja arvot</b> .....	<b>281</b>
<b>Virhekoodit ja viestit</b> .....	<b>282</b>
<b>Varoituskoodit ja -viestit</b> .....	<b>289</b>
<b>Yleistä</b> .....	<b>291</b>
<b>Indeksi</b> .....	<b>292</b>

# Lausekemallit

Lausekemallien avulla voit syöttää matemaattisia lausekkeita normaalissa matemaattisessa muodossa. Lisätessäsi mallin se näkyy syöterivillä siten, että elementtien syöttökohdissa on pienet ruudut. Kohdistin on syötettävän elementin kohdalla.

Voit siirtää kohdistimen kunkin elementin kohdalle nuolipainikkeilla tai painikkeella **tab**, jonka jälkeen voit kirjoittaa elementin arvon tai lausekkeen. Lauseke sievennetään painamalla painikkeita **enter** tai **ctrl enter**.

## Murtolukumalli

**ctrl** **÷** painikkeet



**Huomaa:** Katso myös / (jakolasku), sivu 230.

Esimerkki:

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

## Eksponenttimalli

**^** painike



**Huomaa:** Syötä ensimmäinen arvo, paina **^** ja syötä sen jälkeen eksponentti. Voit palauttaa kohdistimen perusviivalle painamalla oikealle osoittavaa nuolta (►).

**Huomaa:** Katso myös ^ (potenssi), sivu 231.

Esimerkki:

$$2^3 = 8$$

## Neliöjuurimalli

**ctrl** **x<sup>2</sup>** painikkeet



**Huomaa:** Katso myös  $\sqrt{\quad}$  (neliöjuuri), sivu 242.

Esimerkki:

$$\sqrt{4} = 2$$
$$\sqrt{\{9, a, 4\}} = \{3, \sqrt{a}, 2\}$$

## N:s juuri -malli

**ctrl** **^** **painikkeet**



**Huomaa:** Katso myös **root()**, sivu 166.

Esimerkki:

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

---

$$\sqrt[3]{\{8, 27, b\}} \quad \left\{ 2, 3, b^{\frac{1}{3}} \right\}$$

## e eksponenttimalli

**e<sup>x</sup>** **painikkeet**



e-kantainen eksponenttifunktio korotettuna potenssiin

**Huomaa:** Katso myös **e^()**, sivu 62.

Esimerkki:

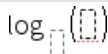
$$e^1 \quad e$$

---

$$e^1. \quad 2.71828182846$$

## Logaritmimalli

**ctrl** **10<sup>x</sup>** **painike**



Laskee määritetyn kantaisen logaritmin. 10-kantaista logaritmia laskettaessa kantaluku jätetään pois.

**Huomaa:** Katso myös **log()**, sivu 115.

Esimerkki:

$$\log_{10}(2.) \quad 0.5$$

## Paloittain määritellyn funktion malli (2-osainen)

**Katalogi >**



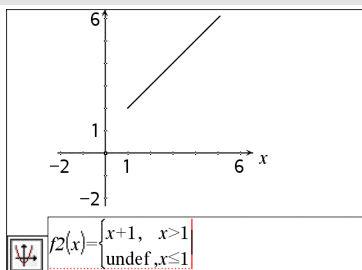
Esimerkki:

Voit luoda lausekkeita ja ehtoja 2-osaiselle paloittain määritellylle funktiolle. Lisää osa napsauttamalla mallia ja toista malli.

**Huomaa:** Katso myös **piecwise()**, sivu 143.

## Paloittain määritellyn funktion malli (2-osainen)

Katalogi > 



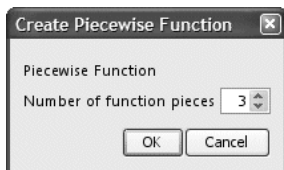
## Paloittain määritellyn funktion malli (N-osainen)

Katalogi > 

Voit luoda lausekkeita ja ehtoja  $N$ -osaiselle paloittain määritellylle funktiolle. Laskin pyytää  $N$ :n arvoa.

Esimerkki:

Katso paloittain määrittelyn funktion (2-osaisen) mallin esimerkki.



**Huomaa:** Katso myös `piecewise()`, sivu 143.

## Yhtälöparin malli

Katalogi > 



Luo kahden yhtälön ryhmän. Voit lisätä rivin olemassa olevaan yhtälöön napsauttamalla mallia ja toistamalla mallin.

**Huomaa:** Katso myös `system()`, sivu 197.

Esimerkki:

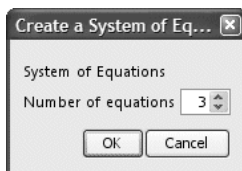
$$\text{solve}\left(\begin{cases} x+y=0 \\ x-y=5 \end{cases}, x, y\right) \quad x=\frac{5}{2} \text{ and } y=-\frac{5}{2}$$

$$\text{solve}\left(\begin{cases} y=x^2-2 \\ x+2y=-1 \end{cases}, x, y\right) \\ x=-\frac{3}{2} \text{ and } y=\frac{1}{4} \text{ or } x=1 \text{ and } y=-1$$

## N-osaisen yhtälöryhmän malli

Katalogi > 

Voit luoda  $N$  yhtälöä sisältävän yhtälöryhmän. Laskin pyytää  $N$ :n arvoa.



Esimerkki:

Katso yhtälöparin (2 yhtälöä) mallin esimerkki.

**Huomaa:** Katso myös `system()`, sivu 197.

## Itseisarvon malli

Katalogi > 



**Huomaa:** Katso myös `abs()`, sivu 8.

Esimerkki:

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

## dd°mm'ss.ss'' -malli

Katalogi > 



Voit syöttää kulmia muodossa **dd°mm'ss.ss''**, jossa **dd** on desimaaliasteiden lukumäärä, **mm** on minuuttimäärä, ja **ss.ss** on sekuntimäärä.

Esimerkki:

$$30^{\circ}15'10'' \quad \frac{10891 \cdot \pi}{64800}$$

## Matriisimalli (2 x 2)

Katalogi > 



Luo 2 x 2 -matriisin.

Esimerkki:

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \cdot a \quad \begin{bmatrix} a & 2 \cdot a \\ 3 \cdot a & 4 \cdot a \end{bmatrix}$$

## Matriisimalli (1 x 2)

Katalogi > 



Esimerkki:

$$\text{crossP}([1 \ 2], [3 \ 4]) \quad [0 \ 0 \ -2]$$



## Matriisimalli (2 x 1)

Katalogi > 

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

Esimerkki:

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

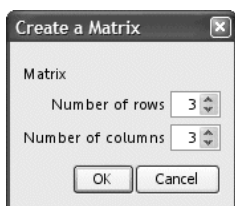
## Matriisimalli (m x n)

Katalogi > 

Malli tulee näkyviin määritettyäsi rivien ja sarakkeiden lukumäärän syöttöruutuun.

Esimerkki:

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



**Huomaa:** Jos luot paljon rivejä ja sarakkeita sisältävän matriisin, voi kestää jonkin aikaa, ennen kuin matriisi tulee näkyviin.

## Summan malli ( $\Sigma$ )

Katalogi > 

$$\sum_{i=0}^{\infty} (0) = 0$$

Esimerkki:

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

**Huomaa:** Katso myös  $\Sigma()$  (`sumSeq`), sivu 243.

## Tulon malli (II)

Katalogi > 

$$\frac{d}{dx} \left( \prod_{n=1}^x a_n \right)$$

Esimerkki:

$$\frac{d}{dx} \left( \prod_{n=1}^x \frac{1}{n} \right) = \frac{1}{120}$$

**Huomaa:** Katso myös **II()** (**prodSeq**), sivu 242.

## Ensimmäisen derivaatan malli

Katalogi > 

$$\frac{d}{dx} (x^3)$$

Esimerkki:

$$\frac{d}{dx} (x^3) = 3 \cdot x^2$$

$$\frac{d}{dx} (x^3)|_{x=3} = 27$$

Ensimmäisen derivaatan mallia voi käyttää myös laskettaessa ensimmäinen derivaatta pisteessä.

**Huomaa:** Katso myös **d()** (**derivaatta**), sivu 239.

## Toisen derivaatan malli

Katalogi > 

$$\frac{d^2}{dx^2} (x^3)$$

Esimerkki:

$$\frac{d^2}{dx^2} (x^3) = 6 \cdot x$$

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

Toisen derivaatan mallia voi käyttää myös laskettaessa toinen derivaatta pisteessä.

**Huomaa:** Katso myös **d()** (**derivaatta**), sivu 239.

## N:n:n derivaatan malli

Katalogi > 

$$\frac{d^n}{dx^n} (x^3)$$

Esimerkki:

$$\frac{d^3}{dx^3} (x^3)|_{x=3} = 6$$

$n$ :n:n derivaatan mallia voidaan käyttää laskettaessa  $n$ :s derivaatta.

## N:nen derivaatan malli

Katalogi > 

Huomaa: Katso myös **d()** (derivaatta), sivu 239.

## Määrätyn integraalin malli

Katalogi > 

$$\int_a^b f(x) dx$$

Huomaa: Katso myös **f()** integraali(), sivu 228.

Esimerkki:

$$\int_a^b x^2 dx = \frac{b^3}{3} - \frac{a^3}{3}$$

## määrittämättömän integraalin malli

Katalogi > 

$$\int f(x) dx$$

Huomaa: Katso myös **f()** integral(), sivu 228.

Esimerkki:

$$\int x^2 dx = \frac{x^3}{3}$$

## Raja-arvon malli

Katalogi > 

$$\lim_{x \rightarrow a} f(x)$$

Vasemman puolen raja-arvon saat painikkeella  $-$  tai  $(-)$ . Oikean puolen raja-arvon saat painikkeella  $+$ .

Huomaa: Katso myös **limit()**, sivu 104.

Esimerkki:

$$\lim_{x \rightarrow 5} (2 \cdot x + 3) = 13$$

# Luettelo aakkosjärjestyksessä

Komennot, joiden nimiä ei voi järjestää aakkosjärjestykseen (esimerkiksi +, ! ja >), on esitetty tämän kappaleen lopussa alkaen sivulta (sivu 228). Ellei toisin ole mainittu, kaikki tämän kappaleen esimerkit on suoritettu laskimen oletustilassa, eikä mitään muuttujia ole määritetty.

## A

### abs()

Katalogi > 

**abs(LausI)**⇒*lauseke*

$$\left\{ \left\{ \frac{\pi}{2}, \frac{\pi}{3} \right\} \right\}$$

**abs(ListaI)**⇒*lista*

**abs(MatriisiI)**⇒*matriisi*

$$2-3 \cdot i$$

$$\sqrt{13}$$

Laskee argumentin itseisarvon.

$$|z|$$

$$|z|$$

$$|x+y \cdot i|$$

$$\sqrt{x^2+y^2}$$

**Huomaa:** Katso myös **Itseisarvon malli**, sivu 4.

Jos argumentti on kompleksiluku, määrittää luvun moduulin.

**Huomaa:** Kaikkia määrittämättömiä muuttujia käsitellään reaali muuttujina.

### amortTbI()

Katalogi > 

**amortTbI(NPmt,N,I,PV, [Pmt], [FV], [PpY], [CpY], [PmtAt], [pyörArvo])**⇒*matriisi*

amortTbI(12,60,10,5000,,12,12)

Lainan lyhennysfunktio, joka laskee lyhennystaulukon tiettyjen TVM-argumenttien perusteella.

*NPmt* on taulukon maksuerien lukumäärä. Taulukko alkaa ensimmäisestä maksuerästä.

*N, I, PV, Pmt, FV, PpY, CpY* ja *PmtAt* on kuvattu TVM-argumenttien taulukossa, sivu 211.

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

- Jos jätät argumentin *Pmt* pois, sen oletusarvoksi tulee ***Pmt=tvmpmt(N,I,PV,FV,PpY,CpY,PmtAt)***.
- Jos jätät argumentin *FV* pois, sen

oletusarvoksi tulee  $FV=0$ .

- Argumenttien  $PpY$ ,  $CpY$  ja  $PmtAt$  oletusarvot ovat samat kuin TVM-funktioilla.

*pyörArvo* määrittää pyörityksessä käytettävien desimaalien määrän.  
Oletusarvo=2.

Tulosmatriisin sarakkeet ovat seuraavassa järjestyksessä: maksuerän numero, koron määrä, pääoman lyhennysmäärä ja velkasaldo.

Rivillä  $n$  näkyvä saldo on maksuerän  $n$  jälkeen jäljellä oleva velkasaldo.

Voit käyttää tulosmatriisia syötteenä muissa lyhennyslaskutoimituksissa  $\Sigma\text{Int}()$  ja  $\Sigma\text{Prn}()$ , sivu 244, sekä **bal()**, sivu 17.

**and**

*BooleanLaus1 and*

*BooleanLaus2* ⇒ *Boolean lausekeBooleanLista1*

**and** *BooleanLista2* ⇒ *Boolean listaBooleanMatriisi1*

**and** *BooleanMatriisi2* ⇒ *Boolean matriisi*

Määrittää totuusarvon tosi tai epätosi tai antaa vastauksena sievennetyn muodon alkuperäisestä syötteestä.

*Kokonaisluku1*

**and** *Kokonaisluku2* ⇒ *kokonaisluku*

Vertaa kahta reaalikokonaislukua bitti bitiltä and-operaation avulla. Sisäisesti kumpikin kokonaisluku muunnetaan etumerkilliseksi, 64 bitin binaariluvuksi. Kun vastaavia bittejä verrataan, tulos on 1, jos kumpikin bitti on 1. Muussa tapauksessa tulos on 0. Laskettu arvo edustaa bittituloksia, ja se näky kantalukutilan mukaisesti.

$$\begin{array}{l} x \geq 3 \text{ and } x \geq 4 \qquad \qquad \qquad x \geq 4 \\ \{x \geq 3, x \leq 0\} \text{ and } \{x \geq 4, x \leq 2\} \qquad \{x \geq 4, x \leq 2\} \end{array}$$

Heksadesimaalisessa kantalukutilassa:

$$\begin{array}{l} 0h7AC36 \text{ and } 0h3D5F \qquad \qquad \qquad 0h2C16 \end{array}$$

**Tärkeää:** Nolla, ei O-kirjain.

Binaarisessa kantalukutilassa:

$$\begin{array}{l} 0b100101 \text{ and } 0b100 \qquad \qquad \qquad 0b100 \end{array}$$

Desimaalisessa kantalukutilassa:

$$\begin{array}{l} 37 \text{ and } 0b100 \qquad \qquad \qquad 4 \end{array}$$

Kokonaisluvut voi syöttää minkä tahansa luvun kantalukuna. Binaarisen syötteen edelle tulee merkitä etumerkki 0b ja heksadesimaalisen syötteen edelle 0h. Jos etumerkkiä ei ole, kokonaislukuja käsitellään desimaalilukuina (kantaluku 10).

Jos syötät desimaalikokonaisluvun, joka on liian suuri etumerkilliselle, 64 bitin binaarimuodolle, laskin käyttää symmetristä modulo-operaatiota, jotta arvo saadaan oikealle alueelle.

**Huomaa:** Binaarisessa syötteenä voi olla korkeintaan 64 numeroa (etuliittettä 0b ei lasketa). Heksadesimaalisessa syötteenä voi olla korkeintaan 16 numeroa.

## angle()

**angle(Lausl)** ⇒ lauseke

Laskee argumentin kulman tulkiten argumentin kompleksiluvuksi.

**Huomaa:** Kaikkia määrittämättömiä muuttujia käsitellään reaali muuttujina.

Astekulmatilassa:

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

Graadikulmatilassa:

$$\text{angle}(0+3\cdot i) \quad 100$$

Radiaanikulmatilassa:

$$\text{angle}(1+i) \quad \frac{\pi}{4}$$

$$\text{angle}(z) \quad \frac{\pi \cdot (\text{sign}(z) - 1)}{2}$$

$$\text{angle}(x+i\cdot y) \quad \frac{\pi \cdot \text{sign}(y)}{2} - \tan^{-1}\left(\frac{x}{y}\right)$$

$$\text{angle}\left(\left\{1+2\cdot i, 3+0\cdot i, 0-4\cdot i\right\}\right) \quad \left\{\frac{\pi}{2} - \tan^{-1}\left(\frac{1}{2}\right), 0, \frac{\pi}{2}\right\}$$

**angle(Listal)** ⇒ lista

**angle(Matriisil)** ⇒ matriisi

Laskee listan tai matriisin *Listal*:n tai *Matriisil*:n elementtien kulumista tulkiten jokaisen elementin kompleksiluvuksi, joka edustaa kaksiulotteista suorakulmakoordinaattipistettä.

**ANOVA** *Lista1,Lista2[,Lista3,...,Lista20]*  
 [,Lippu]

Suorittaa yksisuuntaisen varianssianalyysin 2-20 perusjoukon keskiarvon vertailua varten. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 191.)

*Lippu=0* datalle, *Lippu=1* tilastoille

Tulosmuuttuja	Kuvaus
stat.F	F-tilaston arvo
stat.PVal	Alin merkitsevyytaso, jolla nollahypoteesi voidaan hylätä
stat.df	Ryhmiin vapausasteet
stat.SS	Ryhmiin neliöiden summa
stat.MS	Ryhmiin keskineliöt
stat.dfError	Virheiden vapausasteet
stat.SSError	Virheiden neliöiden summa
stat.MSError	Virheiden keskineliö
stat.sp	Poolattu keskihajonta
stat.xbarlist	Listojen syötteen keskiarvo
stat.CLowerList	95 %:n luottamusväli jokaisen syötelistan keskiarvolle
stat.CUpperList	95 %:n luottamusväli jokaisen syötelistan keskiarvolle

## ANOVA2way

**ANOVA2way** *Lista1,Lista2*  
 [,Lista3,...,Lista10][,TasoRivi]

Laskee kaksisuuntaisen varianssianalyysin 2-10 perusjoukon keskiarvojen vertaamiseksi. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 191.)

*TasoRivi=0* lohkolle

*TasoRivi=2,3,...,Pit-1*, kahdelle tekijälle, jossa  $Pit = \text{pituus(List1)} = \text{pituus(List2)} = \dots = \text{pituus(List10)}$  ja  $Pit / TasoRivi \in \{2,3,\dots\}$

Tulokset: Lohkomuoto

Tulosmuuttuja	Kuvaus
stat.F	F-tilasto, saraketekijän F-tilasto
stat.PVal	Alin merkitsevyystaso, jolla nollahypoteesi voidaan hylätä
stat.df	Saraketekijän vapausasteet
stat.SS	Saraketekijän neliöiden summa
stat.MS	Saraketekijän keskineliöt
stat.FBlock	F-tilasto, tekijän F-tilasto
stat.PValBlock	Pienin todennäköisyys, jolla nollahypoteesi voidaan hylätä
stat.dfBlock	Tekijän vapausasteet
stat.SSBlock	Tekijän neliöiden summa
stat.MSBlock	Tekijän keskineliöt
stat.dfError	Virheiden vapausasteet
stat.SSError	Virheiden neliöiden summa
stat.MSError	Virheiden keskineliöt
stat.s	Virheen keskihajonta

#### SARAKETEKIJÄN tulokset

Tulosmuuttuja	Kuvaus
stat.Fcol	F-tilasto, saraketekijän F-tilasto
stat.PValCol	Saraketekijän todennäköisyysarvo
stat.dfCol	Saraketekijän vapausasteet
stat.SSCol	Saraketekijän neliöiden summa
stat.MSCol	Saraketekijän keskineliöt

#### RIVITEKIJÄN tulokset

Tulosmuuttuja	Kuvaus
stat.FRow	F-tilasto, rivitekijän F-tilasto
stat.PValRow	Rivitekijän todennäköisyysarvo
stat.dfRow	Rivitekijän vapausasteet
stat.SSRow	Rivitekijän neliöiden summa
stat.MSRow	Rivitekijän keskineliöt





## VUOROVAIKUTUKSEN tulokset

Tulosmuuttuja	Kuvaus
stat.FInteract	F-tilasto, vuorovaikutuksen F-tilasto
stat.PValInteract	Vuorovaikutuksen todennäköisyysarvo
stat.dfInteract	Vuorovaikutuksen vapausasteet
stat.SSInteract	Vuorovaikutuksen neliöiden summa
stat.MSInteract	Vuorovaikutuksen keskineliöt

## VIRHEIDEN tulokset

Tulosmuuttuja	Kuvaus
stat.dfError	Virheiden vapausasteet
stat.SSError	Virheiden neliöiden summa
stat.MSError	Virheiden keskineliöt
s	Virheen keskihajonta

### ans

  **painikkeet**

**ans**⇒*arvo*

56 56

Näyttää viimeksi sievennetyn lausekkeen tuloksen.

56+4 60



60+4 64

### approx()

**Katalogi** > 

**approx**(*Lausl*)⇒*lauseke*

Määrittää argumentin sievennetyn arvon lausekkeena, joka sisältää desimaaliarvoja, mikäli mahdollista, riippumatta nykyisestä **Automaattinen tai likimääräinen** -tilasta.

Tämä vastaa argumentin syöttämistä ja painikkeen   painamista.

$\text{approx}\left(\frac{1}{3}\right)$  0.333333

$\text{approx}\left(\left\{\frac{1}{3}, \frac{1}{9}\right\}\right)$  {0.333333, 0.111111}

$\text{approx}(\{\sin(\pi), \cos(\pi)\})$  {0., -1.}

$\text{approx}(\left[\sqrt{2}, \sqrt{3}\right])$  [1.41421 1.73205]

$\text{approx}\left(\left[\frac{1}{3}, \frac{1}{9}\right]\right)$  [0.333333 0.111111]

**approx**(*Listal*)⇒*lista*

**approx**(*Matriisi l*)⇒*matriisi*

$\text{approx}(\{\sin(\pi), \cos(\pi)\})$  {0., -1.}

$\text{approx}(\left[\sqrt{2}, \sqrt{3}\right])$  [1.41421 1.73205]

**approx()**Katalogi > 

Määrittää listan tai *matriisin*, jossa jokainen elementti on laskettu desimaaliarvoksi, mikäli mahdollista.

**► approxFraction()**Katalogi > 

*Laus* ► **approxFraction**(*Tol*) ⇒ *lauseke*

$$\frac{1}{2} + \frac{1}{3} + \tan(\pi) \qquad 0.833333$$

*Lista* ► **approxFraction**(*Tol*) ⇒ *lista*

$$0.8333333333333333 \text{ ► } \mathbf{approxFraction}(5.E-14)$$

*Matriisi* ► **approxFraction**(*Tol*) ⇒ *matriisi*

$$\frac{5}{6}$$

Laskee syötteen murtolukuna käyttäen toleranssia *Tol*. Jos operaattori *Tol* jätetään pois, laskin käyttää toleranssia 5.E-14.

$$\{\pi, 1.5\} \text{ ► } \mathbf{approxFraction}(5.E-14)$$

$$\left\{ \frac{5419351}{1725033}, \frac{3}{2} \right\}$$

**Huomaa:** Voit syöttää tämän funktion tietokoneen näppäimistöltä kirjoittamalla `@>approxFraction(...)`.

**approxRational()**Katalogi > 

**approxRational**(*Laus*[, *tol*]) ⇒ *lauseke*

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

**approxRational**(*Lista*[, *tol*]) ⇒ *lista*

$$\mathbf{approxRational}(\{0.2, 0.33, 4.125\}, 5.E-14)$$

**approxRational**(*Matriisi*[, *tol*]) ⇒ *matriisi*

$$\left\{ \frac{1}{5}, \frac{33}{100}, \frac{33}{8} \right\}$$

Laskee argumentin murtolukuna käyttäen toleranssia *tol*. Jos operaattori *Tol* jätetään pois, laskin käyttää toleranssia 5.E-14.

**arccos()**Katso  $\cos^{-1}()$ , sivu 34.**arccosh()**Katso  $\cosh^{-1}()$ , sivu 35.

**arccot()**Katso  $\cot^{-1}()$ , sivu 36.**arccoth()**Katso  $\coth^{-1}()$ , sivu 37.**arccsc()**Katso  $\csc^{-1}()$ , sivu 40.**arccsch()**Katso  $\operatorname{csch}^{-1}()$ , sivu 41.**arcLen()**Katalogi > **arcLen(Laus1, Muutt, Alku, Loppu)**  
 $\Rightarrow$  lausekeLaskee *Laus1*:n kaaren pituuden alusta *Alku* loppuun *Loppu* muuttujan *Muutt* suhteen.

Kaaren pituus lasketaan kokonaislukuna käyttäen oletuksena funktioiden määrittämistä.

**arcLen(Lista1, Muutt, Alku, Loppu)**  $\Rightarrow$  listaLaskee listan jokaisen *List1*:n elementin kaaren pituuden alusta *Alku* loppuun *Loppu* muuttujan *Muutt* suhteen.

$$\frac{\operatorname{arcLen}(\cos(x), x, 0, \pi)}{\operatorname{arcLen}(f(x), x, a, b)} = \frac{3.8202}{\int_a^b \sqrt{\left(\frac{d}{dx}(f(x))\right)^2 + 1} dx}$$

$$\frac{\operatorname{arcLen}(\{\sin(x), \cos(x)\}, x, 0, \pi)}{\{3.8202, 3.8202\}}$$

**arcsec()**Katso  $\sec^{-1}()$ , sivu 170.**arcsech()**Katso  $\operatorname{sech}^{-1}()$ , sivu 171.

**arcsin()**Katso  $\sin^{-1}()$ , sivu 182.**arcsinh()**Katso  $\sinh^{-1}()$ , sivu 183.**arctan()**Katso  $\tan^{-1}()$ , sivu 198.**arctanh()**Katso  $\tanh^{-1}()$ , sivu 200.**augment()**Katalogi > **augment(Lista1, Lista2)** ⇒ listaLuo uuden listan, joka on *Lista2* liitettynä *Lista1*:n loppuun.

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

**augment(Matriisi1, Matriisi2)** ⇒ matriisiLuo uuden matriisin, joka on *Matriisi2* liitettynä *Matriisi1*:een. Kun käytetään merkkiä " , ", matriiseiden rivimäärien on oltava samat, ja *Matriisi2* liitetään *Matriisi1*:een uusina sarakkeina. Ei muuta *Matriisi1*:ä eikä *Matriisi2*:a.

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

**avgRC()**Katalogi > **avgRC(Laus1, Muutt [=Arvo] [, Askel])** ⇒ lauseke

avgRC( $f(x), x, h$ )	$\frac{f(x+h)-f(x)}{h}$
-----------------------	-------------------------

**avgRC(Laus1, Muutt [=Arvo] [, Lista1])** ⇒ lista

avgRC( $\sin(x), x, h   x=2$ )	$\frac{\sin(h+2)-\sin(2)}{h}$
--------------------------------	-------------------------------

**avgRC(Lista1, Muutt [=Arvo] [, Askel])** ⇒ lista

avgRC( $x^2-x+2, x$ )	$2 \cdot (x-0.4995)$
-----------------------	----------------------

avgRC( $x^2-x+2, x, 0.1$ )	$2 \cdot (x-0.45)$
----------------------------	--------------------

**avgRC(Matriisi1, Muutt [=Arvo] [, Askel])** ⇒ matriisi

avgRC( $x^2-x+2, x, 3$ )	$2 \cdot (x+1)$
--------------------------	-----------------

Laskee erotusosamäärän eteenpäin (keskimääräisen muutosnopeuden).

*Lausl* voi olla käyttäjän määrittämä funktionimi (katso **Func**).

Kun *Arvo* määritetään, se ohittaa mahdolliset aikaisemmat muuttujamäärytykset tai mahdolliset muuttujan nykyiset "|" -sijoitukset.

*Askel* on askeleen arvo. Jos *Askel* jätetään pois, sen oletusarvo on 0.001.

Huomaa, että samankaltaisessa funktiossa **centralDiff()** käytetään keskeiserotusosamäärää.

## B

### bal()

**bal**(*NPmt*,*N*,*I*,*PV*, [*Pmt*], [*FV*], [*PpY*], [*CpY*], [*PmtAt*], [*pyörArvo*]) $\Rightarrow$ *arvo*

**bal**(*NPmt*,*amortTable*) $\Rightarrow$ *arvo*

Lyhennysfunktio, joka laskee määritetyn maksuerän jälkeen jäljellä olevan velkasaldon.

*N*, *I*, *PV*, *Pmt*, *FV*, *PpY*, *CpY* ja *PmtAt* on kuvattu TVM-argumenttien taulukossa, sivu 211.

*NPmt* määrittää sen maksuerän numeron, jonka jälkeen velkasaldo halutaan laskea.

*N*, *I*, *PV*, *Pmt*, *FV*, *PpY*, *CpY* ja *PmtAt* on kuvattu TVM-argumenttien taulukossa, sivu 211.

- Jos jätät argumentin *Pmt* pois, sen oletusarvoksi tulee  $Pmt = \text{tvmPmt}(N, I, PV, FV, PpY, CpY, PmtAt)$ .
- Jos jätät argumentin *FV* pois, sen oletusarvoksi tulee  $FV = 0$ .
- Argumenttien *PpY*, *CpY* ja *PmtAt*

bal(5,6,5.75,5000,,12,12)	833.11
<i>tbl</i> :=amortTbl(6,6,5.75,5000,,12,12)	
0	0. 0. 5000.
1	-23.35 -825.63 4174.37
2	-19.49 -829.49 3344.88
3	-15.62 -833.36 2511.52
4	-11.73 -837.25 1674.27
5	-7.82 -841.16 833.11
6	-3.89 -845.09 -11.98
bal(4, <i>tbl</i> )	1674.27

oletusarvot ovat samat kuin TVM-funktioilla.

*pyör*:Arvo määrittää pyöristyksessä käytettävien desimaalien määrän.

Oletusarvo=2.

**bal**(*NPmt*,*amortTable*) laskee maksueränumeron *NPmt* jälkeen jäljellä olevan velkasaldon lyhennystaulukon *amortTable* perusteella. *amortTable*-argumentin on oltava matriisi, joka on kohdassa **amortTbl()** kuvatun muotoinen, katso sivu 8.

**Huomaa:** Katso myös  $\Sigma\text{Int}()$  ja  $\Sigma\text{Prn}()$ , sivu 244.

## ► Base2 (►Kantaluku2)

*Kokonaisluku1* ► **Base2** ⇒ *kokonaisluku*

256►Base2

0b100000000

**Huomaa:** Voit syöttää tämän operaattorin tietokoneen näppäimistöltä kirjoittamalla @>**Base2**.

0h1F►Base2

0b11111

Muuttaa *Kokonaisluku1*:n binaariluvuksi. Binaariluvuissa on aina etuliite 0b ja heksadesimaaliluvuissa etuliite 0h.

Ilman etuliitettä *Kokonaisluku1*:ä käsitellään desimaalilukuna (kantaluku 10). Vastaus näkyy binaarilukuna kantalukutilasta riippumatta.

Negatiiviset luvut näytetään kahden komplementteina. Esimerkki:

-1 näkyy muodossa

0hFFFFFFFFFFFFFFFF heksadesimaalisessa kantalukutilassa

0b111...111 (64 ykköstä) binaarisessa kantalukutilassa

-2<sup>63</sup> näkyy muodossa

0h8000000000000000 heksadesimaalisessa kantalukutilassa

0b100...000 (63 zeros) binaarisessa kantalukutilassa

Jos syötät desimaalikonaisluvun, joka on etumerkillisen, 64 bitin binaarimuodon lukualueen ulkopuolella, laskin käyttää symmetristä modulo-operaatiota, jotta arvo saadaan oikealle alueelle. Tarkastele seuraavassa esitettyjä esimerkkejä lukualueen ulkopuolella olevista arvoista.

$2^{63}$  muuttuu muotoon  $-2^{63}$  ja näkyy muodossa  
 0h8000000000000000 heksadesimaalisessa kantelukutilassa  
 0b100...000 (63 zeros) binaarisessa kantelukutilassa

$2^{64}$  muuttuu muotoon 0 ja näkyy  
 0h0 heksadesimaalisessa kantelukutilassa  
 0b0 binaarisessa kantelukutilassa

$-2^{63} - 1$  muuttuu muotoon  $2^{63} - 1$  ja näkyy muodossa  
 0h7FFFFFFFFFFFFFFF heksadesimaalisessa kantelukutilassa  
 0b111...111 (64 ykköstä) binaarisessa kantelukutilassa

## ► Base10 (► Kantaluku10)

*Kokonaisluku1*

► Base10 ⇒ *kokonaisluku*

0b10011 ► Base10	19
0h1F ► Base10	31

**Huomaa:** Voit syöttää tämän operaattorin tietokoneen näppäimistöltä kirjoittamalla @>Base10.

Muuttaa *Kokonaisluku1*:n desimaaliluvuksi (kantaluku 10). Binaarisen syötteen edellä tulee aina olla etumerkki 0b ja heksadesimaalisen syötteen edellä 0h.

0b *binaariluku*  
 0h *heksadesimaaliluku*

Nolla, ei O-kirjain, jonka perässä on b tai h.

Binaariluvussa voi olla enintään 64 numeroa. Heksadesimaaliluvussa voi olla enintään 16 numeroa.

Ilman etuliitettä *Kokonaisluku1*:ä käsitellään desimaalilukuna. Vastaus näkyy desimaalilukuna kantalukutilasta riippumatta.

## ►Base16 (►Kantaluku16)

*Kokonaisluku1*

►Base16⇒*kokonaisluku*

256►Base16

0h100

0b111100001111►Base16

0hFOF

**Huomaa:** Voit syöttää tämän operaattorin tietokoneen näppäimistöltä kirjoittamalla @>Base16.

Muuttaa *Kokonaisluku1*:n heksadesimaaliluvuksi. Binaariluvuissa on aina etuliite 0b ja heksadesimaaliluvuissa etuliite 0h.

0b *binaariluku*

0h *heksadesimaaliluku*

Nolla, ei O-kirjain, jonka perässä on b tai h.

Binaariluvussa voi olla enintään 64 numeroa. Heksadesimaaliluvussa voi olla enintään 16 numeroa.

Ilman etuliitettä *Kokonaisluku1*:ä käsitellään desimaalilukuna (kantaluku 10). Vastaus näkyy heksadesimaalilukuna kantalukutilasta riippumatta.

Jos syötät desimaalikokonaisluvun, joka on liian suuri etumerkilliselle, 64 bitin binaarimuodolle, laskin käyttää symmetristä modulo-operaatiota, jotta arvo saadaan oikealle alueelle.



Jos syötät desimaalikonaisluvun, joka on etumerkillisen, 64 bitin binaarimuodon lukualueen ulkopuolella, laskin käyttää symmetristä modulo-operaatiota, jotta arvo saadaan oikealle alueelle. Lisätietoja, katso ►Base2, sivu 18.

**binomCdf()**

**binomCdf( $n,p$ )**⇒*lista*

**binomCdf( $n,p,alaraja,yläraja$ )**⇒*luku*, jos *alaraja* ja *yläraja* ovat lukuja, *lista*, jos *alaraja* ja *yläraja* ovat listoja

**binomCdf( $n,p,yläraja$ )**kun  $P(0 \leq X \leq yläraja)$ ⇒*luku*, jos *yläraja* on luku, *lista*, jos *yläraja* on lista

Laskee kumulatiivisen todennäköisyyden diskreetille binomiselle jakaumalle, jossa toistojen määrä on  $n$  ja jokaisen toiston onnistumistodennäköisyys on  $p$ .

Kun  $P(X \leq yläraja)$ , aseta *alaraja*=0

**binomPdf()**

**binomPdf( $n,p$ )**⇒*lista*

**binomPdf( $n,p,XVal$ )**⇒*luku*, jos *XVal* on luku, *lista*, jos *XVal* on lista

Laskee todennäköisyyden diskreetille binomiselle jakaumalle, jossa toistojen määrä on  $n$  ja jokaisen toiston onnistumistodennäköisyys on  $p$ .

**C****ceiling()**

**ceiling(*Lausl*)**⇒*kokonaisluku*

ceiling(.456)

1.

Laskee lähimmän kokonaisluvun, joka on  $\geq$  argumentti.

Argumentti voi olla reaali- tai kompleksiluku.

**Huomaa:** Katso myös `floor()`.

`ceiling(Lista1)`  $\Rightarrow$  lista

`ceiling(Matriisi1)`  $\Rightarrow$  matriisi

Laskee listan tai matriisin jokaisen elementin ylärajasta.

<code>ceiling({ -3.1, 1.2, 5 })</code>	<code>{ -3., 1, 3 }</code>
<code>ceiling(<math>\begin{bmatrix} 0 &amp; -3.2 \cdot i \\ 1.3 &amp; 4 \end{bmatrix}</math>)</code>	$\begin{bmatrix} 0 & -3 \cdot i \\ 2. & 4 \end{bmatrix}$

## centralDiff()

`centralDiff(Laus1, Muutt [=Arvo] [,Askel])`  $\Rightarrow$  lauseke

`centralDiff(Laus1, Muutt [,Askel]) | Muutt=Arvo`  $\Rightarrow$  lauseke

`centralDiff(Laus1, Muutt [=Arvo] [,Lista])`  $\Rightarrow$  lista

`centralDiff(Lista1, Muutt [=Arvo] [,Askel])`  $\Rightarrow$  lista

`centralDiff(Matriisi1, Muutt [=Arvo] [,Askel])`  $\Rightarrow$  matriisi

Laskee numeerisen derivaatan käyttäen keskeiserotusosamäärän kaavaa.

Kun *Arvo* määritetään, se ohittaa mahdolliset aikaisemmat muuttujamäärytykset tai mahdolliset muuttujan nykyiset " | " -sijoitukset.

*Askel* on askeleen arvo. Jos *Askel* jätetään pois, sen oletusarvo on 0.001.

*Lista1*:tä tai *Matriisi1*:tä käytettäessä operaatio mapataan listan arvojen tai matriisin elementtien suhteen.

**Huomaa:** Katso myös `avgRC()` ja `d()`.

<code>centralDiff(cos(x), x, h)</code>	$\frac{-\cos(x-h) - \cos(x+h)}{2 \cdot h}$
<code>lim(centralDiff(cos(x), x, h)) h→0</code>	$-\sin(x)$
<code>centralDiff(x<sup>3</sup>, x, 0.01)</code>	$3 \cdot (x^2 + 0.000033)$
<code>centralDiff(cos(x), x)   x = <math>\frac{\pi}{2}</math></code>	$-1.$
<code>centralDiff(x<sup>2</sup>, x, {0.01, 0.1})</code>	$\{2 \cdot x, 2 \cdot x\}$

**cFactor(LausI[,Muutt])** ⇒ lauseke

**cFactor(ListaI[,Muutt])** ⇒ lista

**cFactor(MatriisiI[,Muutt])** ⇒ matriisi

**cFactor(LausI)** jakaa *LausI*:n kaikki muuttujat supistaen ne yhteisellä nimittäjällä.

*LausI*:ä jaetaan tekijöihin mahdollisimman paljon kohti lineaarisia rationaalitekijöitä, vaikka tästä saataisiin uusia ei-reaalilukuja. Tämä vaihtoehto on sopiva, jos haluat jakaa lausekkeen tekijöihin useamman kuin yhden muuttujan suhteen.

**cFactor(LausI,Muutt)** jakaa *LausI*:n tekijöihin muuttujan *Muutt* suhteen.

*LausI*:ä jaetaan tekijöihin mahdollisimman paljon kohti tekijöitä, jotka ovat lineaarisia muuttujassa *Muutt*, sisältäen mahdollisesti ei-reaalisia vakioita, vaikka tästä saataisiin irrationaalisia vakioita tai alalausekkeita, joissa on muita irrationaalisia muuttujia.

Tekijät ja niiden termit lajitellaan siten, että *Muutt* on päämuuttuja. Muuttujan *Muutt* samanlaiset potenssit kerätään jokaisessa tekijässä. Muuttujan *Muutt* tulee olla mukana, jos vain kyseistä muuttujaa halutaan jakaa tekijöihin ja jos irrationaalilausekkeet ovat hyväksyttäviä kaikissa muissa muuttujissa, jotta muuttujaa *Muutt* voitaisiin jakaa enemmän tekijöihin. Toimenpiteessä voi esiintyä jonkin verran satunnaista muiden muuttujien tekijöihin jakamista.

$\text{cFactor}(a^3 \cdot x^2 + a \cdot x^2 + a^3 + a, x)$	$a \cdot (a^2 + 1) \cdot (x - i) \cdot (x + i)$
$\text{cFactor}\left(x^2 + \frac{4}{9}\right)$	$\frac{(3 \cdot x - 2 \cdot i) \cdot (3 \cdot x + 2 \cdot i)}{9}$
$\text{cFactor}(x^2 + 3)$	$x^2 + 3$
$\text{cFactor}(x^2 + a)$	$x^2 + a$

$\text{cFactor}(a^3 \cdot x^2 + a \cdot x^2 + a^3 + a, x)$	$a \cdot (a^2 + 1) \cdot (x - i) \cdot (x + i)$
$\text{cFactor}(x^2 + 3, x)$	$(x + \sqrt{3} \cdot i) \cdot (x - \sqrt{3} \cdot i)$
$\text{cFactor}(x^2 + a, x)$	$(x + \sqrt{a} \cdot i) \cdot (x + \sqrt{a} \cdot i)$

## cFactor()

Katalogi >

**Auto or Approximate (Automaattinen tai likimääräinen)** -tilan Auto (Automaattinen) -asetuksessa muuttujan *Muutt* mukanaolo sallii myös likiarvoistamisen liukulukuvakioilla, kun irrationaalisia kertoimia ei voida ilmaista täsmällisen tiiviisti sisäänrakennetuilla termeillä. Vaikka muuttujia olisi vain yksi, muuttujan *Muutt* mukanaolo voi tuottaa täydellisemmän tekijöihin jakamisen.

$$\frac{\text{cFactor}(x^5+4\cdot x^4+5\cdot x^3-6\cdot x-3)}{x^5+4\cdot x^4+5\cdot x^3-6\cdot x-3}$$
$$\frac{\text{cFactor}(x^5+4\cdot x^4+5\cdot x^3-6\cdot x-3,x)}{(x-0.964673)\cdot(x+0.611649)\cdot(x+2.12543)\cdot(x^2+0.964673x+0.611649)}$$

Jos haluat nähdä koko vastauksen, paina  $\blacktriangle$  ja siirrä sen jälkeen osoitinta painikkeilla  $\blacktriangleleft$  ja  $\blacktriangleright$ .

**Huomaa:** Katso myös **factor()**.

## char()

Katalogi >

**char(Kokonaisluku)** $\Rightarrow$ merkki

Näyttää vastauksena merkkijonon, joka sisältää kämmenlaitteen merkkisarjasta olevan merkin, jonka tunnusnumero on *Kokonaisluku*. Kokonaisluvun *Kokonaisluku* sallittu alue on 0–65535.

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

## charPoly()

Katalogi >

**charPoly**

(*neliömatriisi*, *Muutt*) $\Rightarrow$ polynomilauseke

$$m := \begin{bmatrix} 1 & 3 & 0 \\ 2 & -1 & 0 \\ -2 & 2 & 5 \end{bmatrix} \quad \begin{bmatrix} 1 & 3 & 0 \\ 2 & -1 & 0 \\ -2 & 2 & 5 \end{bmatrix}$$

**charPoly**

(*neliömatriisi*, *Laus*) $\Rightarrow$ polynomilauseke

$$\text{charPoly}(m, x) \quad -x^3+5\cdot x^2+7\cdot x-35$$

**charPoly**

(*neliömatriisi1*, *Matriisi2*) $\Rightarrow$ polynomilauseke

$$\text{charPoly}(m, x^2+1) \quad -x^6+2\cdot x^4+14\cdot x^2-24$$

$$\text{charPoly}(m, m) \quad 0$$

Laskee *neliömatriisin* karakteristisen polynomin. Lausekkeen  $n \times n$  matriisi *A* karakteristinen polynomi, merkitään  $p_A(\lambda)$ , on polynomi, joka on määritetty lausekkeella

$$p_A(\lambda) = \det(\lambda \cdot \mathbf{I} - A)$$

jossa *I* tarkoittaa identtistä matriisia  $n \times n$ .

*neliömatriisi1:n* ja *neliömatriisi2:n* on oltava samankokoiset.

 $\chi^2$ 2way $\chi^2$ 2way *ObsMatriisi*chi22way *ObsMatriisi*

Laskee  $\chi^2$ -testin tarkasteltavan matriisiin *ObsMatriisi* sisältämän kaksisuuntaisen lukemataulukon arvojen välisestä assosiaatiosta. Tulosten yhteenvedo tallentuu *stat.results*-muuttujaan. (Katso sivu 191.)

Lisätietoja matriisissa olevien tyhjien elementtien vaikutuksesta, katso Tyhjät elementitsivulla sivu 272.

Tulosmuuttuja	Kuvaus
stat. $\chi^2$	Khin neliö -tilasto: summa (tarkasteltava - odotettu) <sup>2</sup> /odotettu
stat.PVal	Alin merkitsevyytaso, jolla nollahypoteesi voidaan hylätä
stat.df	Khin neliö -tilastojen vapausasteet
stat.ExpMat	Odotetun elementtilukemataulukon matriisi, oletuksena nollahypoteesi
stat.CompMat	Elementtien Khin neliö -tilastokontribuutioiden matriisi

 $\chi^2$ Cdf()

$\chi^2$ Cdf(*alaraja*,*yläraja*,*df*) $\Rightarrow$ luku, jos *alaraja* ja *yläraja* ovat lukuja, *lista*, jos *alaraja* ja *yläraja* ovat listoja

chi2Cdf(*alaraja*,*yläraja*,*df*) $\Rightarrow$ luku, jos *alaraja* ja *yläraja* ovat lukuja, *lista*, jos *alaraja* ja *yläraja* ovat listoja

Laskee  $\chi^2$ -jakauman todennäköisyyden *alarajan* ja *ylärajan* väliltä määritetyille vapausasteelle *df*.

Kun  $P(X \leq \textit{yläraja})$ , aseta *alaraja*= 0.

## $\chi^2$ Cdf()

Katalogi > 

Lisätietoja listassa olevien tyhjiä elementtien vaikutuksesta, katso Tyhjät elementitsivulla sivu 272.

## $\chi^2$ GOF

Katalogi > 

$\chi^2$ GOF *obsLista,expLista,df*

**chi2GOF** *obsLista,expLista,df*

Suorittaa testin, jolla varmistetaan, että otoksen data on tiettyä jakaumaa vastaavasta perusjoukosta. *obsList* on lukemalista, ja sen tulee sisältää kokonaislukuja. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 191.)

Lisätietoja listassa olevien tyhjiä elementtien vaikutuksesta, katso Tyhjät elementitsivulla sivu 272.

Tulosmuuttuja	Kuvaus
stat. $\chi^2$	Khin neliö -tilasto: $\text{sum}((\text{tarkasteltava} - \text{odotettu})^2 / \text{odotettu})$
stat.PVal	Alin merkitsevyytaso, jolla nollahypoteesi voidaan hylätä
stat.df	Khin neliö -tilastojen vapausasteet
stat.CompList	Elementtien Khin neliö -tilastokontribuutiot

## $\chi^2$ Pdf()

Katalogi > 

$\chi^2$ Pdf(*XArvo,df*) $\Rightarrow$ luku, jos *XArvo* on luku, lista, jos *XArvo* on lista

**chi2Pdf**(*XArvo,df*) $\Rightarrow$ luku, jos *XArvo* on luku, lista, jos *XArvo* on lista

Laskee  $\chi^2$ -jakauman todennäköisyysfunktio (pdf) määritetyllä *XArvon* arvolla määritetylle vapausasteelle *df*.

Lisätietoja listassa olevien tyhjiin elementtien vaikutuksesta, katso Tyhjät elementisivulla sivu 272.

## clearAZ

Katalogi > 

## clearAZ

5 → b	5
-------	---

Poistaa kaikki yksikirjaimiset muuttujat nykyiseltä tehtäväalueelta.

b	5
---	---

Clear.AZ	Done
----------	------

Jos yksi tai useampia muuttujia on lukittu, tämä komento aiheuttaa virheilmoituksen ja poistaa vain lukitsemattomat muuttujat. Katso **unLock**, sivu 214.

b	b
---	---

## ClrErr

Katalogi > 

## ClrErr

Esimerkki **ClrErr**-komennosta, katso esimerkki 2 **Try**-komennon kohdalla, sivu 207.

Poistaa virhetilan ja nolaa järjestelmän muuttujan *errCode*.

**Else**-lauseessa lohkoissa **Try...Else...EndTry** tulee käyttää komentoa **ClrErr** tai **PassErr**. Jos virhe on tarkoitus käsitellä tai jättää huomiotta, käytä komentoa **ClrErr**. Jos et tiedä, mitä tehdä virheen suhteen, lähetä se seuraavaan virheenkäsittelijään käyttämällä komentoa **PassErr**. Jos odottavia **Try...Else...EndTry**-virheenkäsittelijöitä ei ole enää, virheen valintaikkuna tulee näkyviin normaalisti.

**Huomaa:** Katso myös **PassErr**, sivu 142, ja **Try**, sivu 207.

**Huomaa esimerkkiä syöttäessäsi:** Ohjeet monirivisten ohjelmien ja funktion määritysten syöttämisestä löytyvät tuotteen ohjekirjan Laskin-osiosta.

**colAugment()**Katalogi > **colAugment**(*Matriisi1*,  
*Matriisi2*) $\Rightarrow$ *matriisi*

Luo uuden matriisin, joka on *Matriisi2* liitettyinä *Matriisi1*:een. Matriiseiden sarakemäärän on oltava sama, ja *Matriisi2* liitetään *Matriisi1*:een uusina riveinä. Ei muuta *Matriisi1*:ä eikä *Matriisi2*:a.

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

**colDim()**Katalogi > **colDim**(*Matriisi*) $\Rightarrow$ *lauseke*

Laskee *Matriisin* sisältämien sarakkeiden lukumäärän.

**Huomaa:** Katso myös **rowDim()**.

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

**colNorm()**Katalogi > **colNorm**(*Matriisi*) $\Rightarrow$ *lauseke*

Laskee maksimiarvon *Matriisin* sarakkeissa olevien elementtien itseisarvojen summista.

**Huomaa:** Määrittämättömät matriisielementit eivät ole sallittuja. Katso myös **rowNorm()**.

$\begin{bmatrix} 1 & -2 & 3 \\ 4 & 5 & -6 \end{bmatrix} \rightarrow mat$	$\begin{bmatrix} 1 & -2 & 3 \\ 4 & 5 & -6 \end{bmatrix}$
$\text{colNorm}(mat)$	9

**comDenom()**Katalogi > **comDenom**(*LausI*[,*Muutt*]) $\Rightarrow$ *lauseke***comDenom**(*ListaI*[,*Muutt*]) $\Rightarrow$ *lista***comDenom**(*MatriisiI*  
[,*Muutt*]) $\Rightarrow$ *matriisi*

**comDenom**(*LausI*) supistaa täydellisesti lavennetun osoittajan täydellisesti lavennetulla nimittäjällä.

$\text{comDenom}\left(\frac{y^2+y}{(x+1)^2}+y^2+y\right)$
$\frac{x^2 \cdot y^2 + x^2 \cdot y + 2 \cdot x \cdot y^2 + 2 \cdot x \cdot y + 2 \cdot y^2 + 2 \cdot y}{x^2 + 2 \cdot x + 1}$



**comDenom(Laus1,Muutt)** supistaa osoittajan ja nimittäjän, jotka on lavennettu muuttujalla *Muutt*. Termit ja niiden tekijät lajitellaan siten, että *Muutt* on päämuuttuja. Muuttujan *Muutt* samanlaiset potenssit kerätään.

Toimenpiteessä voi esiintyä jonkin verran kerättyjen kertoimien satunnaista tekijöihin jakamista. Verrattuna siihen, että muuttuja *Muutt* jätettäisiin pois, tämä toiminto säästää usein aikaa, muistia ja näyttötilaa, ja samalla lausekkeesta tulee ymmärrettävämpi. Lisäksi tulokseen kohdistuvat seuraavat operaatiot ovat nopeampia eivätkä kuluta muistia yhtä todennäköisesti.

Jos muuttujaa *Muutt* ei ole *Laus1*:ssä, **comDenom(Laus1,Muutt)** supistaa laventamattoman osoittajan laventamattomalla nimittäjällä. Tällaiset tulokset säästävät yleensä vielä enemmän aikaa, muistia ja näyttötilaa. Tällaiset osittain tekijöihin jaetut tulokset nopeuttavat myös seuraavia tulokseen kohdistuvia operaatioita eivätkä kuluta muistia läheskään yhtä todennäköisesti.

Vaikka nimittäjää ei olisi, **comden**-funktio on usein nopea tapa suorittaa osittainen tekijöihin jako, mikäli **factor()** on liian hidas tai käyttää liikaa muistia.

**Vinkki:** Syötä tämä **comden()**-funktion määrittäminen ja kokeile sitä rutiininomaisesti vaihtoehtona funktioille **comDenom()** ja **factor()**.

$$\text{comDenom}\left(\frac{y^2+y}{(x+1)^2}+y^2+y,x\right) \\ \frac{x^2 \cdot y \cdot (y+1) + 2 \cdot x \cdot y \cdot (y+1) + 2 \cdot y \cdot (y+1)}{x^2 + 2 \cdot x + 1}$$

$$\text{comDenom}\left(\frac{y^2+y}{(x+1)^2}+y^2+y,y\right) \\ \frac{y^2 \cdot (x^2+2 \cdot x+2) + y \cdot (x^2+2 \cdot x+2)}{x^2 + 2 \cdot x + 1}$$

$$\text{Define } \text{comden}(\text{exprn}) = \text{comDenom}(\text{exprn}, \text{abc}) \\ \text{Done}$$

$$\text{comden}\left(\frac{y^2+y}{(x+1)^2}+y^2+y\right) \frac{(x^2+2 \cdot x+2) \cdot y \cdot (y+1)}{(x+1)^2}$$

$$\text{comden}(1234 \cdot x^2 \cdot (y^3-y) + 2468 \cdot x \cdot (y^2-1)) \\ 1234 \cdot x \cdot (x \cdot y + 2) \cdot (y^2-1)$$

## completeSquare ()

**completeSquare(ExprOrEqn, Var)** ⇒ lauseke tai yhtälö

**completeSquare(ExprOrEqn, Var^Power)** ⇒ lauseke tai yhtälö

**completeSquare(ExprOrEqn, Var1, Var2 [...])** ⇒ lauseke tai yhtälö

$$\text{completeSquare}(x^2+2 \cdot x+3,x) \quad (x+1)^2+2$$

$$\text{completeSquare}(x^2+2 \cdot x-3,x) \quad (x+1)^2-4$$

$$\text{completeSquare}(x^6+2 \cdot x^3+3,x^3) \quad (x^3+1)^2+2$$

## completeSquare ()

Katalogi > 

**completeSquare**(ExprOrEqn, {Var1, Var2 [...]} ) ⇒ lauseke tai yhtälö

Muuntaa muotoa  $a \cdot x^2 + b \cdot x + c$  olevan toisen asteen polynomilausekkeen muotoon  $a \cdot (x-h)^2 + k$

- tai -

Muuntaa muotoa  $a \cdot x^2 + b \cdot x + c$  olevan toisen asteen yhtälön muotoon  $a \cdot (x-h)^2 + k$

Ensimmäisen argumentin on oltava toisen asteen lauseke tai yhtälö vakio muodossa toisen argumentin suhteen.

Toisen argumentin on oltava yhden muuttujan termi tai yhden muuttujan termi korotettuna rationaaliseen potenssiin  $x$ ,  $y^2$  tai  $z^{1/3}$ .

Kolmas ja neljäs syntaksi yrittävät neliöksi täydentämisen muuttujien *Var1*, *Var2* [... ] suhteen.

completeSquare( $x^2+4 \cdot x+y^2+6 \cdot y+3=0, x, y$ )	
	$(x+2)^2+(y+3)^2=10$

completeSquare( $3 \cdot x^2+2 \cdot y+7 \cdot y^2+4 \cdot x=3, \{x, y\}$ )	
	$3 \cdot \left(x+\frac{2}{3}\right)^2+7 \cdot \left(y+\frac{1}{7}\right)^2=\frac{94}{21}$

completeSquare( $x^2+2 \cdot x \cdot y, x, y$ )	$(x+y)^2-y^2$
---	---------------

## conj()

Katalogi > 

**conj**(LausI) ⇒ lauseke

**conj**(ListaI) ⇒ lista

**conj**(MatriisiI) ⇒ matriisi

Laskee argumentin liittokompleksiluvun.

**Huomaa:** Kaikkia määrittämättömiä muuttujia käsitellään reaali muuttujina.

conj( $1+2 \cdot i$ )	$1-2 \cdot i$
conj( $\begin{pmatrix} 2 & 1-3 \cdot i \\ -i & -7 \end{pmatrix}$ )	$\begin{pmatrix} 2 & 1+3 \cdot i \\ i & -7 \end{pmatrix}$
conj( $z$ )	$\bar{z}$
conj( $x+i \cdot y$ )	$x-y \cdot i$

## constructMat()

Katalogi > 

**constructMat**

(  
Laus

,  
Muutt1

, Muutt2, numRivit, numSarakeet)

⇒ matriisi

constructMat( $\frac{1}{i+j}, i, j, 3, 4$ )	$\begin{pmatrix} 1 & 1 & 1 & 1 \\ 2 & 3 & 4 & 5 \\ 1 & 1 & 1 & 1 \\ 3 & 4 & 5 & 6 \\ 1 & 1 & 1 & 1 \\ 4 & 5 & 6 & 7 \end{pmatrix}$
---	--

Laskee matriisiin argumentteihin perustuen.

*Laus* on lauseke muuttujissa *Muutt1* ja *Muutt2*. Tuloksena olevan matriisin elementit muodostetaan sieventämällä *Laus* jokaisella *Muutt1*:n ja *Muutt2*:n lisätyllä arvolla.

*Muutt1*:ä lisätään automaattisesti välillä **1 - numRivit**. Kullakin rivillä *Muutt2*:a lisätään välillä **1 - numSarakeet**.

## CopyVar

**CopyVar** *Muutt1*, *Muutt2*

Define $a(x)=\frac{1}{x}$	Done
---------------------------	------

**CopyVar** *Muutt1*., *Muutt2*.

Define $b(x)=x^2$	Done
-------------------	------

**CopyVar** *Muutt1*, *Muutt2* kopioi muuttujan *Muutt1* arvon muuttujaan *Muutt2* ja luo tarvittaessa *Muutt2*:n. Muuttujalla *Muutt1* on oltava arvo.

CopyVar a,c: c(4)	$\frac{1}{4}$
-------------------	---------------

CopyVar b,c: c(4)	16
-------------------	----

Jos *Muutt1* on olemassa olevan käyttäjän määrittämän funktion nimi, kopioi kyseisen funktion määrittämisen funktioon *Muutt2*. Funktio *Muutt1* on määritettävä.

*Muutt1*:n on oltava muuttujien nimeämissääntöjen mukainen tai epäsuora lauseke, joka sieventyy näitä vaatimuksia vastaavaksi muuttujan nimeksi.

**CopyVar** *Muutt1*., *Muutt2*. kopioi kaikki *Muutt1*:n jäsenet. muuttujaryhmä *Var2*:een. ryhmä, *Muutt2*:n luominen. tarvittaessa.

<i>aa.a</i> :=45	45																
<i>aa.b</i> :=6.78	6.78																
CopyVar <i>aa</i> ., <i>bb</i> .	Done																
getVarInfo()	<table border="1"> <tr> <td><i>aa.a</i></td> <td>"NUM"</td> <td>"{}"</td> <td>0</td> </tr> <tr> <td><i>aa.b</i></td> <td>"NUM"</td> <td>"{}"</td> <td>0</td> </tr> <tr> <td><i>bb.a</i></td> <td>"NUM"</td> <td>"{}"</td> <td>0</td> </tr> <tr> <td><i>bb.b</i></td> <td>"NUM"</td> <td>"{}"</td> <td>0</td> </tr> </table>	<i>aa.a</i>	"NUM"	"{}"	0	<i>aa.b</i>	"NUM"	"{}"	0	<i>bb.a</i>	"NUM"	"{}"	0	<i>bb.b</i>	"NUM"	"{}"	0
<i>aa.a</i>	"NUM"	"{}"	0														
<i>aa.b</i>	"NUM"	"{}"	0														
<i>bb.a</i>	"NUM"	"{}"	0														
<i>bb.b</i>	"NUM"	"{}"	0														

*Muutt1*. tulee olla olemassa olevan muuttujaryhmän nimi, kuten tilastollinen *stat.nn* vastausta tai muuttujaa, jotka on luotu funktiolla **LibShortcut()**. Jos *Muutt2*. on jo olemassa, komento korvaa kaikki jäsenet, jotka ovat yhteisiä kummallekin ryhmälle, ja lisää jäsenet, joita ei vielä ole olemassa. Jos yksi tai useampia muuttujan *Muutt2*. jäseniä on lukittu, kaikki muuttujan *Var2*. jäsenet pysyvät muuttumattomina.

**corrMat()**

**corrMat(Lista1,Lista2[,...[,Lista20]])**

Laskee korrelaatiomatriisin laajennetulle matriisille [*Lista1*, *Lista2*, ..., *Lista20*].

**► cos**

*Laus* ► **cos**

**Huomaa:** Voit syöttää tämän operaattorin tietokoneen näppäimistöltä kirjoittamalla @>**cos**.

$$\frac{(\sin(x))^2 \blacktriangleright \cos}{1 - (\cos(x))^2}$$

Näyttää *Laus*:n kulman kosinin. Tämä on näytön muunnosoperaattori. Sitä voidaan käyttää vain syöterivin lopussa.

► **cos** alentaa kaikkia lausekkeen  $\sin(\dots)$  modulo  $1 - \cos(\dots)^2$  potensseja, siten että jäljelle jäävien lausekkeen  $\cos(\dots)$  potenssien eksponentit ovat alueella (0, 2). Tulos ei täten sisällä lauseketta  $\sin(\dots)$ , jos ja vain jos  $\sin(\dots)$  esiintyy lausekkeessa korotettuna vain parillisiin potensseihin.

**Huomaa:** Tätä muunnosoperaattoria ei tueta aste- eikä graadikulmatilassa. Ennen kuin käytät sitä, varmista, että kulmatila on asetettu radiaaneiksi ja että *Laus* ei sisällä eksplisiittisiä viittauksia aste- tai graadikulmiin.

**cos(LausI)** ⇒ lauseke

**cos(ListaI)** ⇒ lista

**cos(LausI)** määrittää argumentin kosinin lausekkeena.

**cos(ListaI)** määrittää listan kaikkien *Listai*:n sisältämien elementtien kosineista.

**Huomaa:** Argumentti tulkitaan aste-, graadi- tai radiaanikulmaksi käytössä olevan kulmatila-asetuksen mukaisesti. Voit ohittaa kulmatilan väliaikaisesti painikkeilla °, G tai r.

Astekulmatilassa:

$$\begin{array}{l} \cos\left(\frac{\pi}{4}\right) \qquad \qquad \qquad \frac{\sqrt{2}}{2} \\ \hline \cos(45) \qquad \qquad \qquad \frac{\sqrt{2}}{2} \\ \hline \cos(\{0,60,90\}) \qquad \qquad \qquad \left\{1, \frac{1}{2}, 0\right\} \end{array}$$

Graadikulmatilassa:

$$\cos(\{0,50,100\}) \qquad \qquad \qquad \left\{1, \frac{\sqrt{2}}{2}, 0\right\}$$

Radiaanikulmatilassa:

$$\begin{array}{l} \cos\left(\frac{\pi}{4}\right) \qquad \qquad \qquad \frac{\sqrt{2}}{2} \\ \hline \cos(45^\circ) \qquad \qquad \qquad \frac{\sqrt{2}}{2} \end{array}$$

**cos(neliömatriisiI)** ⇒ neliömatriisi

Laskee *neliömatriisiI*:n matriisikosinin. Tämä ei ole sama kuin kunkin elementin kosinin laskeminen.

Kun skaalarista funktiota  $f(A)$  käytetään *neliömatriisiI*:een (A), tulos lasketaan algoritmile:

Radiaanikulmatilassa:

$$\cos \begin{pmatrix} 1 & 5 & 3 \\ 4 & 2 & 1 \\ 6 & -2 & 1 \end{pmatrix} \begin{bmatrix} 0.212493 & 0.205064 & 0.121389 \\ 0.160871 & 0.259042 & 0.037126 \\ 0.248079 & -0.090153 & 0.218972 \end{bmatrix}$$

Laske A:n ominaisarvot ( $\lambda_i$ ) ja ominaisvektorit ( $V_i$ ).

*neliömatriisiI*:n on oltava diagonalisoitavissa. Lisäksi siinä ei voi olla symbolisia muuttujia, joille ei ole määritetty arvoa.

Matriiseista:

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

Tällöin  $A = X B X^{-1}$  ja  $f(A) = X f(B) X^{-1}$ .

Esimerkiksi,  $\cos(A) = X \cos(B) X^{-1}$ , jossa:

$\cos(B) =$

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

Kaikki laskut suoritetaan liukulukuaritmetiikalla.

## cos<sup>-1</sup>()

cos<sup>-1</sup>(LausI) ⇒ lauseke

cos<sup>-1</sup>(ListaI) ⇒ lista

cos<sup>-1</sup>(LausI) määrittää lausekkeena kulman, jonka kosini on LausI.

cos<sup>-1</sup>(ListaI) laskee listan ListaI:n jokaisen elementin käänteiskosineista.

**Huomaa:** Vastaus lasketaan aste-, graadi- tai radiaanikulmana käytössä olevan kulmatila-asetuksen mukaisesti.

**Huomaa:** Voit syöttää tämän funktion näppäimistöä kirjoittamalla **arccos** (...).

cos<sup>-1</sup>(neliömatriisiI) ⇒ neliömatriisi

Laskee neliömatriisiI:n matriisin käänteiskosinin. Tämä ei ole sama kuin kunkin elementin käänteiskosinin laskeminen. Laskentamenetelmä on kuvattu kohdassa cos().

Astekulmatilassa:

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

Graadikulmatilassa:

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

Radiaanikulmatilassa:

$$\cos^{-1}\{0, 0, 2, 0, 5\} = \left\{ \frac{\pi}{2}, 1.36944, 1.0472 \right\}$$

Radiaanikulmatilassa ja suorakulmakompleksimuodossa:

**cos<sup>-1</sup>()**

trig painike

*neliömatriisi*1:n on oltava diagonalisoitavissa. Vastaus sisältää aina liukulukuja.

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

$$\begin{bmatrix} 1.73485+0.064606\cdot i & -1.49086+2.10514 \\ -0.725533+1.51594\cdot i & 0.623491+0.778369 \\ -2.08316+2.63205\cdot i & 1.79018-1.27182\cdot i \end{bmatrix}$$

Jos haluat nähdä koko vastauksen, paina ▲ ja siirrä sen jälkeen osoitinta painikkeilla ◀ ja ▶.

**cosh()**

Katalogi &gt;

**cosh**(*LausI*)⇒*lauseke*

Astekulmatilassa:

**cosh**(*ListaI*)⇒*lista*

$$\cosh\left(\left(\frac{\pi}{4}\right)_r\right) \quad \cosh(45)$$

**cosh**(*LausI*) määrittää argumentin hyperbolisen kosinin lausekkeena.

**cosh**(*ListaI*) määrittää listan *ListaI*:n kunkin elementin hyperbolisista kosineista.

**cosh**(*neliömatriisiI*)⇒*neliömatriisi*

Radiaanikulmatilassa:

Laskee *neliömatriisiI*:n matriisin hyperbolisen kosinin. Tämä ei ole sama kuin kunkin elementin hyperbolisen kosinin laskeminen. Laskentamenetelmä on kuvattu kohdassa **cos()**.

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

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

*neliömatriisi*1:n on oltava diagonalisoitavissa. Vastaus sisältää aina liukulukuja.

**cosh<sup>-1</sup>()**

Katalogi &gt;

**cosh<sup>-1</sup>**(*LausI*)⇒*lauseke*

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

**cosh<sup>-1</sup>**(*ListaI*)⇒*lista*

$$\cosh^{-1}(\{1,2,1,3\}) \quad \{0,1.37286,\cosh^{-1}(3)\}$$

**cosh<sup>-1</sup>**(*LausI*) määrittää argumentin käänteisen hyperbolisen kosinin lausekkeena.

**cosh<sup>-1</sup>**(*ListaI*) määrittää listan *ListaI*:n kunkin elementin käänteisistä hyperbolisista kosineista.

**Huomaa:** Voit syöttää tämän funktion näppäimistöltä kirjoittamalla `arcosh` (...).

`cosh-1(neliomatriisiI)` ⇒ neliomatriisi

Laskee *neliomatriisiI*:n matriisin käänteisen hyperbolisen kosinin. Tämä ei ole sama kuin kunkin elementin käänteisen hyperbolisen kosinin laskeminen. Laskentamenetelmä on kuvattu kohdassa `cos()`.

*neliomatriisiI*:n on oltava diagonalisoitavissa. Vastaus sisältää aina liukulukuja.

Radiaanikulmatilassa ja suorakulmakompleksimuodossa:

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


---

2.52503+1.73485·i	-0.009241-1.4908i
0.486969-0.725533·i	1.66262+0.623491i
-0.322354-2.08316·i	1.26707+1.79018i

Jos haluat nähdä koko vastauksen, paina **▲** ja siirrä sen jälkeen osoitinta painikkeilla **◀** ja **▶**.

`cot(LausI)` ⇒ lauseke

`cot(ListaI)` ⇒ lista

Laskee *LausI*:n kotangentin tai määrittää listan *Listai*:n kaikkien elementtien kotangenteista.

**Huomaa:** Argumentti tulkitaan aste-, graadi- tai radiaanikulmaksi käytössä olevan kulmatila-asetuksen mukaisesti. Voit ohittaa kulmatilan väliaikaisesti painikkeilla °, G tai r.

Astekulmatilassa:

---


$$\cot(45)$$


---

 1

Graadikulmatilassa:

---


$$\cot(50)$$


---

 1

Radiaanikulmatilassa:

---


$$\cot(\{1,2,1,3\}) \left\{ \frac{1}{\tan(1)}, -0.584848, \frac{1}{\tan(3)} \right\}$$


---

`cot-1(LausI)` ⇒ lauseke

`cot-1(ListaI)` ⇒ lista

Laskee kulman, jonka kotangenti on *LausI*, tai määrittää listan, joka sisältää *Listai*:n kunkin elementin käänteiskotangentit.

Astekulmatilassa:

---


$$\cot^{-1}(1)$$


---

 45.

Graadikulmatilassa:

---


$$\cot^{-1}(1)$$


---

 50.

Radiaanikulmatilassa:



**cot<sup>-1</sup>()**

trig painike

**Huomaa:** Vastaus lasketaan aste-, graadi- tai radiaanikulmana käytössä olevan kulmatila-asetuksen mukaisesti.

cot <sup>-1</sup> (1)	$\frac{\pi}{4}$
-----------------------	-----------------

**Huomaa:** Voit syöttää tämän funktion näppäimistöltä kirjoittamalla **arccot** (...).

**coth()**

Katalogi &gt;

**coth**(*Laus1*)⇒*lauseke*

coth(1.2)	1.19954
-----------	---------

**coth**(*List1*)⇒*lista*

coth({1,3,2})	$\left\{ \frac{1}{\tanh(1)}, 1.00333 \right\}$
---------------	--

Laskee *Laus1*:n hyperbolisen kotangentin tai määrittää listan *List1*:n kaikkien elementtien hyperbolisista kotangenteista.

**coth<sup>-1</sup>()**

Katalogi &gt;

**coth<sup>-1</sup>**(*Laus1*)⇒*lauseke*

coth <sup>-1</sup> (3.5)	0.293893
--------------------------	----------

**coth<sup>-1</sup>**(*List1*)⇒*lista*

coth <sup>-1</sup> ({-2,2,1,6})	$\left\{ \frac{-\ln(3)}{2}, 0.518046, \frac{\ln\left(\frac{7}{5}\right)}{2} \right\}$
---------------------------------	---

Laskee *Laus1*:n käänteisen hyperbolisen kotangentin tai määrittää listan, joka sisältää *List1*:n kaikkien elementtien käänteiset hyperboliset kotangentit.

**Huomaa:** Voit syöttää tämän funktion näppäimistöltä kirjoittamalla **arccoth** (...).

**count()**

Katalogi &gt;

**count**(*Arvo1* tai *List1* [,*Arvo2* tai *List2* [...]])⇒*arvo*

count(2,4,6)	3
--------------	---

count({2,4,6})	3
----------------	---

Laskee elementtien kokonaismäärän argumenteille, jotka sieventyvät numeroarvoiksi.

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

count( $\frac{1}{2}, 3+4\cdot i$ , undef, "hello", $x+5$ , sign(0))	2
---	---

Argumentit voivat olla lausekkeita, arvoja, listoja tai matriiseja. Argumenttien datatyypit voivat olla erilaisia, ja argumentit voivat olla erikokoisia.

Listan, matriisin tai solualueen jokainen elementti sievennetään, jotta voidaan määrittää, kuuluuko se laskettavaan lukumäärään.

Listat & Taulukot -sovelluksessa voit käyttää solualueita argumenttien tilalla.

Tyhjiä elementtejä ei huomioida. Lisätietoja tyhjistä elementeistä, katso sivu 272.

Viimeisessä esimerkissä lukumäärään lasketaan mukaan vain  $1/2$  ja  $3+4*i$ . Muut argumenteista, olettaen että  $x$  on määrittämätön, eivät sievenny numeroarvoiksi.

## countif()

**countif(Lista, Kriteerit) ⇒ arvo**

Laskee niiden *Listan* sisältämien elementtien kokonaismäärän, jotka vastaavat määritettyjä kriteereitä *Kriteerit*.

*Kriteeri* voi olla:

- Arvo, lauseke tai merkkijono. Jos kriteerinä käytetään esimerkiksi lukua **3**, laskee lukumäärään vain ne *Listan* elementit, jotka sievennyvät arvoksi 3.
- Boolean lauseke, joka sisältää symbolin **?** kunkin elementin paikanpitäjänä. Esimerkiksi lauseke **?<5** laskee lukumäärään vain ne *Listan* elementit, jotka ovat alle 5.

Listat & Taulukot -sovelluksessa voit käyttää solualueita *Listan* tilalla.

Listassa olevia tyhjiä elementtejä ei huomioida. Lisätietoja tyhjistä elementeistä, katso sivu 272.

**Huomaa:** Katso myös **sumIf()**, sivu 196, ja **frequency()**, sivu 80.

---

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

---

Laskee niiden elementtien lukumäärän, jotka ovat yhtä kuin 3.

---

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

---

Laskee niiden elementtien lukumäärän, jotka ovat yhtä kuin "def".

---

countIf({x<sup>-2</sup>,x<sup>-1</sup>,1,x,x<sup>2</sup>},x) 1

---

Laskee niiden elementtien lukumäärän, jotka ovat yhtä kuin  $x$ ; tässä esimerkissä oletetaan, että muuttuja  $x$  on määrittämätön.

---

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

---

Laskee lukumäärään 1:n ja 3:n.

---

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

---

Laskee lukumäärään 3:n, 5:n ja 7:n.

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

Laskee lukumäärään 1:n, 3:n, 7:n ja 9:n.

**cPolyRoots()**

**cPolyRoots**(*Poly*, *Muutt*) ⇒ *lista*

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

**cPolyRoots**(*Kertoinlista*) ⇒ *lista*

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

Ensimmäinen syntaksi, **cPolyRoots** (*Poly*, *Muutt*), laskee polynomin *Poly* kompleksisten juurten listan muuttujan *Muutt* suhteen.

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

*Poly* on oltava polynomi yhdessä muuttujassa.

$$\text{cPolyRoots}(\{1, 2, 1\}) \quad \{-1, -1\}$$

Toinen syntaksi, **cPolyRoots** (*Kertoinlista*), laskee kompleksisten juurten listan kertoimille, jotka sisältyvät *Kertoinlistaan*.

**Huomaa:** Katso myös **polyRoots()**, sivu 147.

**crossP()**

**crossP**(*Lista1*, *Lista2*) ⇒ *lista*

$$\text{crossP}(\{a1, b1\}, \{a2, b2\}) \\ \{0, 0, a1 \cdot b2 - a2 \cdot b1\}$$

Määrittää listan *Lista1*:n ja *Lista2*:n ristitulosta.

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

*Lista1*:n ja *Lista2*:n on oltava samankokoiset, ja koon on oltava joko 2 tai 3.

**crossP**(*Vektori1*, *Vektori2*) ⇒ *vektori*

$$\text{crossP}(\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}, \begin{bmatrix} -3 & 6 & -3 \\ 0 & 0 & -2 \end{bmatrix})$$

Laskee rivi- tai sarakevektorin (argumenteista riippuen), joka on *Vektori1*:n ja *Vektori2*:n ristitulo.

Sekä *Vektori1*:n että *Vektori2*:n on oltava rivivektoreita tai sarakevektoreita. Vektoreiden on oltava samankokoiset, ja koon tulee olla joko 2 tai 3.

**csc()**

trig painike

**csc(Laus1)** ⇒ lauseke

Astekulmatilassa:

$$\text{csc}(45) \quad \sqrt{2}$$

**csc(Lista1)** ⇒ lista

Graadikulmatilassa:

$$\text{csc}(50) \quad \sqrt{2}$$

Laskee *Laus1*:n kosekantin tai määrittää listan, joka sisältää *Lista1*:n kaikkien elementtien kosekantit.

Radiaanikulmatilassa:

$$\text{csc}\left(\left\{1, \frac{\pi}{2}, \frac{\pi}{3}\right\}\right) \quad \left\{\frac{1}{\sin(1)}, 1, \frac{2\sqrt{3}}{3}\right\}$$

**csc<sup>-1</sup>()**

trig painike

**csc<sup>-1</sup>(Laus1)** ⇒ lauseke

Astekulmatilassa:

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

**csc<sup>-1</sup>(Lista1)** ⇒ lista

Graadikulmatilassa:

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

Laskee kulman, jonka kosekanti on *Laus1*, tai määrittää listan, joka sisältää *Lista1*:n kunkin elementin käännteiskosekantit.

**Huomaa:** Vastaus lasketaan aste-, graadi- tai radiaanikulmana käytössä olevan kulmatila-asetuksen mukaisesti.

Radiaanikulmatilassa:

$$\text{csc}^{-1}\left(\{1, 4, 6\}\right) \quad \left\{\frac{\pi}{2}, \sin^{-1}\left(\frac{1}{4}\right), \sin^{-1}\left(\frac{1}{6}\right)\right\}$$

**Huomaa:** Voit syöttää tämän funktion näppäimistöltä kirjoittamalla **arcscsc** (...).

**csch()**

Katalogi &gt;

**csch(Laus1)** ⇒ lauseke

$$\text{csch}(3) \quad \frac{1}{\sinh(3)}$$

**csch(Lista1)** ⇒ lista

$$\text{csch}\left(\{1, 2, 1, 4\}\right) \quad \left\{\frac{1}{\sinh(1)}, 0.248641, \frac{1}{\sinh(4)}\right\}$$

Laskee *Laus1*:n hyperbolisen kosekantin tai määrittää listan, joka sisältää *Lista1*:n kaikkien elementtien hyperboliset kosekantit.

**csch<sup>-1</sup>()**

Katalogi &gt;

**csch<sup>-1</sup>(LausI)** ⇒ lauseke

csch <sup>-1</sup> (1)	sinh <sup>-1</sup> (1)
------------------------	------------------------

**csch<sup>-1</sup>(ListaI)** ⇒ lista

csch <sup>-1</sup> ({1,2,1,3})	{sinh <sup>-1</sup> (1), 0.459815, sinh <sup>-1</sup> ( $\frac{1}{3}$ )}
--------------------------------	--

Laskee *LausI*:n käänteisen hyperbolisen kosekantin tai määrittää listan, joka sisältää *ListaI*:n kaikkien elementtien käänteiset hyperboliset kosekantit.

**Huomaa:** Voit syöttää tämän funktion näppäimistöltä kirjoittamalla **arcscsch** (...).

**cSolve()**

Katalogi &gt;

**cSolve(Yhtälö, Muutt)** ⇒ Boolean lauseke

cSolve( $x^3 = -1, x$ )	
-------------------------	--

**cSolve(Yhtälö, Muutt=Arvaus)** ⇒ Boolean lauseke

$$x = \frac{1}{2} + \frac{\sqrt{3}}{2} \cdot i \text{ or } x = \frac{1}{2} - \frac{\sqrt{3}}{2} \cdot i \text{ or } x = -1$$

**cSolve(Epäyhtälö, Muutt)** ⇒ Boolean lauseke

solve( $x^3 = -1, x$ )	$x = -1$
------------------------	----------

Määrittää kompleksiyhtälön tai epäyhtälön mahdollisia ratkaisuja muuttujalle *Muutt*. Tavoitteena on tuottaa kaikkien reaalisten ja ei-reaalisten ratkaisujen ehdotuksia. Vaikka *Yhtälö* olisi reaalinen, **cSolve()** sallii ei-reaaliset vastaukset reaalituloksen kompleksilukumuodossa.

**cSolve()** asettaa määrittäjäjoukon väliaikaisesti kompleksilukumuotoon yhtälön ratkaisemisen ajaksi, vaikka nykyinen määrittäjäjoukko olisi reaalinen. Kompleksilukujen määrittäjäalueella murtolukueksponenteissa, joiden nimittäjä on pariton luku, käytetään perus- eikä reaalityyppiä. Tämän vuoksi **solve()**-funktion ratkaisut yhtälöille, joihin liittyy tällaisia murtopotensseja, eivät välttämättä ole **cSolve()**-funktion ratkaisujen alasarja.

cSolve( $x^{\frac{1}{3}} = -1, x$ )	false
-------------------------------------	-------

solve( $x^{\frac{1}{3}} = -1, x$ )	$x = -1$
------------------------------------	----------

**cSolve()**-funktion ratkaisu aloitetaan eksakteilla symbolisilla menetelmillä. **cSolve()** käyttää tarvittaessa myös iteratiivista likimääräistä kompleksipolynomin tekijöihin jakamista.

**Huomaa:** Katso myös **cZeros()**, **solve()** ja **zeros()**.

**cSolve**(*Yht1* and *Yht2* [*and...*],  
*MuuttTaiArvaus1*, *MuuttTaiArvaus2* [,  
... ]) ⇒ *Boolean lauseke*

**cSolve**(*Yhtälöryhmä*, *MuuttTaiArvaus1*,  
*MuuttTaiArvaus2* [, ...])  
⇒ *Boolean lauseke*

Laskee mahdollisia kompleksiratkaisuja samanaikaisille algebrallisille yhtälöille, joissa jokainen *MuuttTaiArvaus* määrittää ratkaistavan muuttujan.

Voit halutessasi määrittää muuttujan ensimmäisen arvauksen. Jokaisen *muuttTaiArvaus*-komennon on oltava muodossa:

*muuttuja*  
– tai –  
*muuttuja* = *reaaliluku* tai *ei-reaaliluku*

Esimerkiksi  $x$  kelpaa ja samoin  $x=3+i$ .

Jos kaikki yhtälöt ovat polynomeja, ja jos ET määrittä yhtään ensimmäistä arvausta, **cSolve()** käyttää leksikaalista Gröbner/Buchbergerin eliminaatiomenetelmää yrittäessään määrittää **kaikki** kompleksiratkaisut.

Kompleksiratkaisut voivat sisältää sekä reaali- että ei-reaaliratkaisuja kuten oikealla olevassa esimerkissä.

Desimaalien näyttötötilassa Kiinteä 2:

$$\frac{\text{exact}(\text{cSolve}(x^5+4x^4+5x^3-6x-3=0,x))}{x \cdot (x^4+4x^3+5x^2-6)=3}$$

$$\text{cSolve}(\text{Ans},x)$$

$$x=-1.11+1.07 \cdot i \text{ or } x=-1.11-1.07 \cdot i \text{ or } x=-2.1$$

Jos haluat nähdä koko vastauksen, paina ▲ ja siirrä sen jälkeen osoitinta painikkeilla ◀ ja ▶.

$$\text{cSolve}(u \cdot v - u = v \text{ and } v^2 = -u, \{u, v\})$$

$$u = \frac{1}{2} + \frac{\sqrt{3}}{2} \cdot i \text{ and } v = \frac{1}{2} - \frac{\sqrt{3}}{2} \cdot i \text{ or } u = \frac{1}{2} - \frac{\sqrt{3}}{2} \cdot i$$

Jos haluat nähdä koko vastauksen, paina ▲ ja siirrä sen jälkeen osoitinta painikkeilla ◀ ja ▶.

Samanaikaisissa polynomiyhtälöissä voi olla ylimääräisiä muuttujia, joilla ei ole arvoja, vaan ne edustavat tiettyjä numeerisia arvoja, jotka voidaan korvata myöhemmin.

Voit ottaa mukaan myös ratkaisumuuttujia, jotka eivät esiinny yhtälöissä. Nämä ratkaisut osoittavat, miten ratkaisujen sarjat voivat sisältää mielivaltaisia vakioita, jotka ovat muotoa  $ck$ , jossa  $k$  on kokonaislukualue väliltä 1-255.

Polynomisarjoissa laskun suoritus aika tai muistin käyttö voivat riippua merkittävästi ratkaisumuuttujien järjestyksestä. Jos ensimmäinen valintasi kuluttaa muistia, tai et jaksaa odottaa vastausta, yritä järjestää muuttujat uudelleen yhtälöihin ja/tai *muuttTaiArvaus*-listaan.

Jos et ota mukaan arvauksia, ja jos jokin yhtälöstä on ei-polynominen minkä tahansa muuttujan suhteen, mutta kaikki yhtälöt ovat lineaarisia kaikissa ratkaisumuuttujissa, **cSolve()** käyttää Gaussin eliminointia yrittäessään määrittää kaikki ratkaisut.

Jos sarja ei ole polynominen kaikilta muuttujiltaan eikä lineaarinen ratkaisumuuttujiltaan, **cSolve()** määrittää korkeintaan yhden ratkaisun käyttäen likimääräistä iteratiivista menetelmää. Tässä ratkaisumuuttujien lukumäärän on oltava sama kuin yhtälöiden lukumäärä, ja kaikkien muiden yhtälöiden sisältämien muuttujien on sievennyttävä luvuiksi.

Ei-reaalinen arvaus on usein välttämätön ei-reaalisen ratkaisun määrittämiseksi. Suppenemista varten arvauksen on mahdollisesti oltava melko lähellä ratkaisua.

$$\text{cSolve}(u \cdot v - u = c \cdot v \text{ and } v^2 = -u, \{u, v\})$$

$$u = \frac{-(\sqrt{4 \cdot c - 1} \cdot i + 1)^2}{4} \text{ and } v = \frac{\sqrt{4 \cdot c - 1} \cdot i + 1}{2} \circ \rightarrow$$

$$\text{cSolve}(u \cdot v - u = v \text{ and } v^2 = -u, \{u, v, w\})$$

$$u = \frac{1}{2} + \frac{\sqrt{3}}{2} \cdot i \text{ and } v = \frac{1}{2} - \frac{\sqrt{3}}{2} \cdot i \text{ and } w = c \cdot d \circ \rightarrow$$

$$\text{cSolve}(u + v = e^w \text{ and } u - v = i, \{u, v\})$$




$$u = \frac{e^w + i}{2} \text{ and } v = \frac{e^w - i}{2}$$

$$\text{cSolve}(e^z = w \text{ and } w = z^2, \{w, z\})$$

$$w = 0.494866 \text{ and } z = 0.703467$$

$$\text{cSolve}(e^z = w \text{ and } w = z^2, \{w, z = 1 + i\})$$

$$w = 0.149606 + 4.8919 \cdot i \text{ and } z = 1.58805 + 1.5402 \cdot i$$

Jos haluat nähdä koko vastauksen, paina  ja siirrä sen jälkeen osoitinta painikkeilla  ja .

**CubicReg**  $X, Y$ , [*Frekv*] [, *Luokka*,  
*Sisällytä*]]

Laskee 3. asteen polynomiregression  $y = a \cdot x^3 + b \cdot x^2 + c \cdot x + d$  listaista  $X$  ja  $Y$  frekvenssillä *Frekv*. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 191.)

Kaikkien listojen on oltava samankokoisia *Sisällytä*-listaa lukuunottamatta.

$X$  ja  $Y$  ovat riippumattomien ja riippuvien muuttujien listoja.

*Frekv* on valinnainen frekvenssiarvojen lista. Jokainen *Frekv*:n elementti määrittää kunkin vastaavan datapisteen  $X$  ja  $Y$  esiintymisfrekvenssin. Oletusarvo on 1. Kaikkien elementtien on oltava kokonaislukuja  $\geq 0$ .

*Luokka* on luokkakoodien lista vastaavalle  $X$ - ja  $Y$ -dataalle.

*Sisällytä* on yhden tai usemman luokkakoodin lista. Vain ne datayksiköt, joiden luokkakoodi sisältyy tähän listaan, ovat mukana laskutoimituksessa.

Lisätietoja listassa olevien tyhjiin elementtien vaikutuksesta, katso Tyhjet elementitsivulla sivu 272.

Tulosmuuttuja	Kuvaus
stat.RegEqn	Regressioyhtälö: $a \cdot x^3 + b \cdot x^2 + c \cdot x + d$ .
stat.a, stat.b, stat.c, stat.d	Regressiokertoimet.
stat.R <sup>2</sup>	Määrittyskerroin.
stat.Resid	Regressioyhtälön jäännökset
stat.XReg	Muokatun $X$ Listan sisältämä datapisteiden lista, jota käytetään regressiossa komentojen <i>Frekv</i> , <i>Luokkalista</i> ja <i>Sisällytä luokat</i> rajoitusten mukaisesti.



Tulosmuuttuja	Kuvaus
stat.YReg	Muokatun <i>Y Lista</i> :n sisältämä datapisteiden lista, jota käytetään regressiossa komentojen <i>Frekv</i> , <i>Luokkalista</i> ja <i>Sisällytä luokat</i> rajoitusten mukaisesti.
stat.FreqReg	Komentoja <i>stat.XReg</i> ja <i>stat.YReg</i> vastaava frekvenssiliista.

## cumulativeSum()

Katalogi >

**cumulativeSum(Lista1)⇒lista**

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

Laskee listan *Listal*:n sisältämien elementtien kumulatiivisista summista alkaen elementistä 1.

**cumulativeSum(Matriisi1)⇒matriisi**

1	2	→ <i>m1</i>	1	2
3	4		3	4
5	6		5	6
cumulativeSum( <i>m1</i> )			1	2
			4	6
			9	12

Laskee matriisiin *Matriisi1*:n sisältämien elementtien kumulatiivisista summista. Jokainen elementti on ylhäältä alas ulottuvan sarakkeen kumulatiivinen summa.

Tyhjä elementti listassa *Listal* tai matriisissa *Matriisi1* tuottaa tyhjän elementin tuloksena olevaan listaan tai matriisiin. Lisätietoja tyhjästä elementeistä, katso sivu 272.

## Cycle

Katalogi >

### Cycle

Funktio, joka laskee yhteen kokonaisluvut väliiltä 1-100 ohittaen luvun 50.

Siirtää ohjauksen välittömästi nykyisen silmukan (**For**, **While** tai **Loop**) seuraavaan iteraatioon.

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

Done

---

*g()*      5000

**Cycle** ei ole sallittu näiden kolmen silmukkarakenteen (**For**, **While** tai **Loop**) ulkopuolella.

**Huomaa esimerkkiä syöttäessäsi:** Ohjeet monirivisten ohjelmien ja funktion määritysten syöttämisestä löytyvät tuotteen ohjekirjan Laskin-osiosta.

Vektori ►Cylind

[2 2 3]►Cylind

$$\left[ 2\sqrt{2} \quad \angle \frac{\pi}{4} \quad 3 \right]$$

**Huomaa:** Voit syöttää tämän operaattorin tietokoneen näppäimistöltä kirjoittamalla @>Cylind.

Näyttää rivi- tai sarakevektorin sylinterin muodossa  $[r, \angle, z]$ .

Vektorissa on oltava täsmälleen kolme elementtiä. Se voi olla joko rivi tai sarake.

## cZeros()

**cZeros(Laus, Muutt)**⇒*lista*

Desimaalien näyttötilassa Kiinteä 3:

Määrittää listan muuttujan *Muutt* mahdollisista reaali- ja ei-reaaliarvoista, joiden tuloksena *Laus*=0. **cZeros()** suorittaa tämän seuraavasti: **explist(cSolve(Laus=0,Muutt),Muutt)**. Muilta osin **cZeros()** on samanlainen kuin **zeros()**.

$$\begin{aligned} & \text{cZeros}(x^5+4\cdot x^4+5\cdot x^3-6\cdot x-3,x) \\ & \{-1.114+1.073\cdot i; 1.114-1.073\cdot i; -2.125, -0.612, 0\} \end{aligned}$$

Jos haluat nähdä koko vastauksen, paina ▲ ja siirrä sen jälkeen osoitinta painikkeilla ◀ ja ▶.

**Huomaa:** Katso myös **cSolve()**, **solve()** ja **zeros()**.

**cZeros({Laus1, Laus2 [, ... ]}, {MuuttTaiArvaus1, MuuttTaiArvaus2 [, ... ]})**⇒*matriisi*

Laskee mahdollisia kohtia, joissa lausekkeet ovat samanaikaisesti nolla. Jokainen *MuuttTaiArvaus* määrittää ratkaistavan tuntemattoman arvon.

Voit halutessasi määrittää muuttujan ensimmäisen arvauksen. Jokaisen *muuttTaiArvaus*-komennon on oltava muodossa:

*muuttuja*

– tai –

*muuttuja* = *reaaliluku tai ei-reaaliluku*

Esimerkiksi  $x$  kelpaa ja samoin  $x=3+i$ .

Jos kaikki lausekkeet ovat polynomeja, ja ET määrittää ensimmäisiä arvauksia, **cZeros()** käyttää leksikaalista Gröbner/Buchbergerin eliminaatiomenetelmää yrittäessään määrittää **kaikki** kompleksiset nollakohdat.

Kompleksiset nollakohdat voivat sisältää sekä reaalisia että ei-reaalisia nollakohtia, kuten oikealla olevassa esimerkissä.

Jokainen tulosmatriisin rivi edustaa vaihtoehtoista nollakohtaa, jossa komponentit on järjestetty samalla tavalla kuin *MuuttTaiArvaus*-listassa. Jos haluat määrittää rivin juuren, indeksoi matriisi *[riveittäin]*.

$$\text{cZeros}\left(\{u \cdot v - u - v, v^2 + u\}, \{u, v\}\right)$$

$$\begin{bmatrix} 0 & 0 \\ \frac{1}{2} - \frac{\sqrt{3}}{2} \cdot i & \frac{1}{2} + \frac{\sqrt{3}}{2} \cdot i \\ \frac{1}{2} + \frac{\sqrt{3}}{2} \cdot i & \frac{1}{2} - \frac{\sqrt{3}}{2} \cdot i \end{bmatrix}$$

Määritä rivin 2 juuri:

$$\text{Ans}[2] \quad \left[ \frac{1}{2} - \frac{\sqrt{3}}{2} \cdot i \quad \frac{1}{2} + \frac{\sqrt{3}}{2} \cdot i \right]$$

Samanaikaisissa polynomeissa voi olla ylimääräisiä muuttujia, joilla ei ole arvoja, vaan ne edustavat tiettyjä numeerisia arvoja, jotka voidaan korvata myöhemmin.

Voit ottaa mukaan myös tuntemattomia muuttujia, jotka eivät esiinny lausekkeissa. Nämä nollakohdat osoittavat, miten nollakohtien sarjat voivat sisältää mielivaltaisia vakioita, jotka ovat muotoa  $ck$ , jossa  $k$  on kokonaislukuliite väliiltä 1-255.

$$\text{cZeros}\left(\{u \cdot v - u - c \cdot v^2, v^2 + u\}, \{u, v\}\right)$$

$$\begin{bmatrix} 0 & 0 \\ -(c-1)^2 & -(c-1) \end{bmatrix}$$

Polynomisarjoissa laskutoimituksen suoritus aika tai muistin käyttö voivat riippua merkittävästi tuntemattomien muuttujien järjestyksestä. Jos ensimmäinen valintasi kuluttaa muistia, tai et jaksa odottaa vastausta, yritä järjestää muuttujat uudelleen lausekkeisiin ja/tai *MuuttTaiArvaus*-listaan.

$$\text{cZeros}\left(\{u \cdot v - u - v, v^2 + u\}, \{u, v, w\}\right)$$

$$\text{cZero}\left(\{u \cdot (v-1) - v, w + v^2\}, \{u, v, w\}\right)$$

$$\begin{bmatrix} 0 & 0 & c\# \\ \frac{1}{2} - \frac{\sqrt{3}}{2} \cdot i & \frac{1}{2} + \frac{\sqrt{3}}{2} \cdot i & c\# \\ \frac{1}{2} + \frac{\sqrt{3}}{2} \cdot i & \frac{1}{2} - \frac{\sqrt{3}}{2} \cdot i & c\# \end{bmatrix}$$

Jos et ota mukaan arvauksia, ja jokin lausekkeista on ei-polynominen missä tahansa muuttujassa, mutta kaikki muuttujat ovat lineaarisia kaikissa tuntemattomissa muuttujissa, **cZeros()** käyttää Gaussin eliminointia yrittäessään määrittää kaikki nollakohtat.

$$\left| \text{cZeros} \left( \left\{ u+v-e^w, u-v-i \right\}, \left\{ u, v \right\} \right) \right. \\ \left. \left[ \frac{e^{w+i}}{2} \quad \frac{e^{w-i}}{2} \right] \right.$$

Jos sarja ei ole polynominen kaikilta muuttujiltaan eikä lineaarinen tuntemattomilta muuttujiltaan, **cZeros()** määrittää korkeintaan yhden nollakohtan käyttäen likimääräistä iteratiivista menetelmää. Tässä tuntemattomien muuttujien lukumäärän on oltava sama kuin lausekkeiden lukumäärä, ja kaikkien muiden lausekkeiden sisältämien muuttujien on sievennyttävä luvuiksi.

$$\left| \text{cZeros} \left( \left\{ e^z-w, w-z^2 \right\}, \left\{ w, z \right\} \right) \right. \\ \left. \left[ 0.494866 \quad -0.703467 \right] \right.$$

Ei-reaalinen arvaus on usein välttämätön ei-reaalisen nollakohtan määrittämiseksi. Suppenemista varten arvauksen on mahdollisesti oltava melko lähellä nollakohtaa.

$$\left| \text{cZeros} \left( \left\{ e^{-z}-w, w-z^2 \right\}, \left\{ w, z=1+i \right\} \right) \right. \\ \left. \left[ 0.149606+4.8919 \cdot i \quad 1.58805+1.54022 \cdot i \right] \right.$$

## D

### dbd()

**dbd(pvm1,pvm2)⇒arvo**

Laskee *pvm1*:n ja *pvm2*:n välissä olevien päivien lukumäärän käyttäen todellisten päivien laskentamenetelmää.

*pvm1* ja *pvm2* voivat olla lukuja tai lukulistoja, jotka ovat vakiokalenterin päivämääräalueen sisällä. Jos sekä *pvm1* että *pvm2* ovat listoja, niiden on oltava samanpituiset.

*pvm1*:n ja *pvm2*:n on oltava vuosien 1950 ja 2049 välillä.

Voit syöttää päivämäärät kahdessa eri muodossa. Desimaalipisteen paikka on erilainen näissä päivämäärien esitystavoissa.

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

MM.DDYY (Yhdysvalloissa yleisesti käytetty esitystapa)  
DDMM.YY (Euroopassa yleisesti käytetty esitystapa)

## ►DD

*Lausl* ►DD⇒*arvoListal*

►DD⇒*listaMatriisil*

►DD⇒*matriisi*

**Huomaa:** Voit syöttää tämän operaattorin tietokoneen näppäimistöltä kirjoittamalla @>DD.

Laskee vastaavan desimaaliluvun asteina ilmaistulle argumentille. Argumentti on luku, lista tai matriisi, jonka kulmatilasetus tulkitsee graadeina, radiaaneina tai asteina.

Astekulmatilassa:

$(1.5^\circ)$ ►DD	1.5°
$(45^\circ 22' 14.3")$ ►DD	45.3706°
$(\{45^\circ 22' 14.3", 60^\circ 0' 0"\})$ ►DD	{45.3706°, 60°}

Graadikulmatilassa:

1►DD	$\frac{9}{10}$
------	----------------

Radiaanikulmatilassa:

$(1.5)$ ►DD	85.9437°
-------------	----------

## ►Decimal

*Lauseke1*►Decimal⇒*lauseke*

*Listal*►Decimal⇒*lauseke*

*Matriisil*►Decimal⇒*lauseke*

**Huomaa:** Voit syöttää tämän operaattorin tietokoneen näppäimistöltä kirjoittamalla @>Decimal.

Näyttää argumentin desimaalimuodossa. Tätä operaattoria voi käyttää ainoastaan syöterivin lopussa.

$\frac{1}{3}$ ►Decimal	0.333333
------------------------	----------

## Define (Määritä)

Define *Muutt* = *Lauseke*

Define *Funktio*(*Param1*, *Param2*, ...) =

*Lauseke*

Määrittää muuttujan *Muutt* tai käyttäjän määrittämän funktion *Funktio*.

Parametrit, kuten *Param1*, toimivat paikanpitäjinä argumenttien syöttämiseksi funktioon. Kun haet käyttäjän määrittämän funktion, sinun on annettava parametreja vastaavat argumentit (esimerkiksi arvoja tai muuttujia). Kun funktio haetaan, se sieventää *Lausekkeen* annettujen argumenttien perusteella.

*Muutt* ja *Funktio* eivät voi olla järjestelmän muuttujan tai sisäänrakennetun funktion tai komennon nimenä.

**Huomaa:** Seuraava **Define**-funktion muoto on vastaava kuin lausekkeen sieventäminen: *lauseke* → *Funktio* (*Param1*, *Param2*).

**Define** *Funktio*(*Param1*, *Param2*, ...) = **Func**  
*Lohko*  
**EndFunc**

**Define** *Ohjelma*(*Param1*, *Param2*, ...) = **Prgm**  
*Lohko*  
**EndPrgm**

Tässä muodossa käyttäjän määrittämä funktio tai ohjelma voi suorittaa useista lausekkeista koostuvan lohkon.

*Lohko* voi olla joko yksi lauseke tai eri riveillä olevien lausekkeiden sarja. *Lohko* voi sisältää myös lausekkeita ja ohjeita (kuten **If**, **Then**, **Else** ja **For**).

**Huomaa esimerkkiä syöttäessäsi:** Ohjeet monirivisten ohjelmien ja funktion määrittysten syöttämisestä löytyvät tuotteen ohjekirjan Laskin-osiosta.

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

Define $g(x,y)=\text{Func}$	Done
If $x > y$ Then	
Return $x$	
Else	
Return $y$	
EndIf	
EndFunc	
$g(3,-7)$	3

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

**Huomaa:** Katso myös **Define LibPriv**, sivu 51, ja **Define LibPub**, sivu 51.

**Define LibPriv (Määritä LibPriv)**

**Define LibPriv** *Muutt = Lauseke*  
**Define LibPriv** *Funktio*(*Param1, Param2,*  
*...*) = *Lauseke*

**Define LibPriv** *Funktio*(*Param1, Param2,*  
*...*) = **Func**  
*Lohko*  
**EndFunc**

**Define LibPriv** *Ohjelma*(*Param1, Param2,*  
*...*) = **Prgm**  
*Lohko*  
**EndPrgm**

Tämä komento toimii muuten samalla tavalla kuin **Define** paitsi, että se määrittää yksityisen kirjastomuuttujan, -funktion tai -ohjelman. Yksityiset funktiot ja ohjelmat eivät ole katalogissa.

**Huomaa:** Katso myös **Define**, sivu 49, ja **Define LibPub**, sivu 51.

**Define LibPub (Määritä LibPub)**

**Define LibPub** *Muutt = Lauseke*  
**Define LibPub** *Funktio*(*Param1, Param2,*  
*...*) = *Lauseke*

**Define LibPub** *Funktio*(*Param1, Param2,*  
*...*) = **Func**  
*Lohko*  
**EndFunc**

**Define LibPub** *Ohjelma* (*Param1, Param2,*  
*...*) = **Prgm**  
*Lohko*  
**EndPrgm**

Tämä komento toimii muuten samalla tavalla kuin **Define** paitsi, että se määrittää julkisen kirjastomuuttujan, -funktion tai -ohjelman. Julkiset funktiot ja ohjelmat näkyvät katalogissa sen jälkeen, kun kirjasto on tallennettu ja näyttö on päivitetty.

**Huomaa:** Katso myös **Define**, sivu 49, ja **Define LibPriv**, sivu 51.

## deltaList()

Katso  $\Delta$ List(), sivu 111.

## deltaTmpCnv()

Katso  $\Delta$ tmpCnv(), sivu 205.

## DelVar

**DelVar** *Muutt1* [, *Muutt2*] [, *Muutt3*] ...

$2 \rightarrow a$	2
-------------------	---

**DelVar** *Muutt*.

$(a+2)^2$	16
-----------	----

Poistaa määritetyn muuttujan tai muuttujaryhmän muistista.

DelVar <i>a</i>	<i>Done</i>
-----------------	-------------

Jos yksi tai useampia muuttujia on lukittu, tämä komento aiheuttaa virheilmoituksen ja poistaa vain lukitsemattomat muuttujat. Katso **unLock**, sivu 214.

$(a+2)^2$	$(a+2)^2$
-----------	-----------

**DelVar** *Muutt*. poistaa kaikki *Muutt:n* jäsenet. muuttujaryhmä (kuten tilastollinen *stat.nn* tulosta tai muuttujaa, jotka on luotu funktiolla **LibShortcut()**). Piste (.) tässä **DelVar**-komennon muodossa rajoittaa funktion muuttujaryhmän poistamiseen; komento ei vaikuta yksinkertaiseen muuttujaan *Muutt*.

<i>aa.a:=45</i>	45
-----------------	----

<i>aa.b:=5.67</i>	5.67
-------------------	------

<i>aa.c:=78.9</i>	78.9
-------------------	------

getVarInfo()	<table border="1"> <tr> <td><i>aa.a</i></td> <td>"NUM"</td> <td>"{:}"</td> </tr> <tr> <td><i>aa.b</i></td> <td>"NUM"</td> <td>"{:}"</td> </tr> <tr> <td><i>aa.c</i></td> <td>"NUM"</td> <td>"{:}"</td> </tr> </table>	<i>aa.a</i>	"NUM"	"{:}"	<i>aa.b</i>	"NUM"	"{:}"	<i>aa.c</i>	"NUM"	"{:}"
<i>aa.a</i>	"NUM"	"{:}"								
<i>aa.b</i>	"NUM"	"{:}"								
<i>aa.c</i>	"NUM"	"{:}"								

DelVar <i>aa</i> .	<i>Done</i>
--------------------	-------------

getVarInfo()	"NONE"
--------------	--------



**delVoid()**

Katalogi &gt;

**delVoid(Lista1)** ⇒ lista $\text{delVoid}(\{1, \text{void}, 3\})$   $\{1, 3\}$ 

Antaa tuloksena listan, jossa on listan *Listal* sisältö, ja kaikki tyhjät elementit on poistettu.

Lisätietoja tyhjästä elementeistä, katso sivu 272.

**derivative()**Katso *d()*, sivu 239.**deSolve()**

Katalogi &gt;

**deSolve(1.astTai2.astODE, Muutt, riippuvaMuutt)** ⇒ yleinen ratkaisu
$$\text{deSolve}(y''+2\cdot y'+y=x^2, x, y)$$

$$y=(c3\cdot x+c4)\cdot e^{-x}+x^2-4\cdot x+6$$

$$\text{right}(Ans)\rightarrow temp \quad (c3\cdot x+c4)\cdot e^{-x}+x^2-4\cdot x+6$$

$$\frac{d^2}{dx^2}(temp)+2\cdot \frac{d}{dx}(temp)+temp-x^2 \quad 0$$

$$\text{DelVar temp} \quad Done$$

Ratkaisee yhtälön, joka määrittää eksplisiittisesti tai implisiittisesti yleisratkaisun 1. tai 2. asteen tavalliselle differentiaaliyhtälölle (ODE). ODE:ssa:

- Käytä jaottoman merkkiä (näppäin ) viittaamaan riippuvan muuttujan 1. derivaattaan riippumattomaan muuttujaan nähden.
- Käytä kahta jaottoman merkkiä viittaamaan vastaavaan toiseen derivaattaan.

Jaottoman merkkiä käytetään vain deSolve()-funktion derivaatoissa. Muissa tapauksissa käytetään merkintää **d()**.

1. asteen yhtälön yleisratkaisu sisältää mielivaltaisen vakion muotoa *ck*, jossa *k* on kokonaislukuliite väliltä 1-255. 2. asteen yhtälön ratkaisu sisältää kaksi tällaista vakiota.

Käytä **solve()**-funktioita implisiittisessä ratkaisussa, jos haluat yrittää muuntaa sen yhdeksi tai useammaksi ekvivalenttiseksi eksplisiittiseksi ratkaisuksi.

$$\text{deSolve}(y'=(\cos(y))^2, x, x, y)$$

$$\tan(y)=\frac{x^2}{2}+c4$$

Kun vertaata vastauksia oppikirjan tai käsikirjan ratkaisuihin, huomaa, että erilaiset menetelmät tuovat mielivaltaisia vakioita eri kohtiin laskutoimituksessa, mistä voi olla tuloksena erilaisia yleisratkaisuja.

**deSolve(1.astODEandalkuehto, Muutt, riippuvaMuutt)** ⇒ tietty ratkaisu

Laskee tietyn ratkaisun, joka täyttää 1.astODE:n ja alkuehdon vaatimukset. Tämä on yleensä helpompaa kuin yleisratkaisun määrittäminen, alkuarvojen korvaaminen, mielivaltaisen vakion ratkaiseminen ja sen jälkeen arvon korvaaminen yleisratkaisuun.

*alkuehto* on yhtälö, joka on muotoa:

*riippuvaMuutt* (*riippumatonAlkuarvo*) = *riippuvaAlkuarvo*

*riippumatonAlkuarvo* ja *riippuvaAlkuarvo* voivat olla muuttujia, kuten  $x_0$  ja  $y_0$ , joilla ei ole tallennettuja arvoja. Implisiittinen derivointi voi helpottaa implisiittisten ratkaisujen tarkistamista.

**deSolve**  
(2.astODEandalkuehto1andalkuehto2, Muutt, riippuvaMuutt) ⇒ tietty ratkaisu

Antaa tietyn ratkaisun, joka sopii 2. ast ODE:hen ja jolla on määritetty riippuvan muuttujan arvo ja sen ensimmäinen derivaatta yhdessä pisteessä.

Käytä *alkuehto1*:lle muotoa:

*riippuvaMuutt* (*riippumatonAlkuarvo*) = *riippuvaAlkuarvo*

Käytä *alkuehto2*:lle muotoa:

*riippuvaMuutt* (*riippumatonAlkuarvo*) = *1.derivaatanAlkuarvo*

$$\text{solve}(Ans,y) \quad y = \tan^{-1}\left(\frac{x^2 + 2 \cdot c4}{2}\right) + n3 \cdot \pi$$

$$Ans|c4=c-1 \text{ and } n3=0 \quad y = \tan^{-1}\left(\frac{x^2 + 2 \cdot (c-1)}{2}\right)$$

$$\sin(y) = (y \cdot e^x + \cos(y)) \cdot y' \rightarrow ode$$

$$\sin(y) = (e^x \cdot y + \cos(y)) \cdot y'$$

$$\text{deSolve}(ode \text{ and } y(0)=0, x, y) \rightarrow soln$$

$$\frac{-(2 \cdot \sin(y) + y^2)}{2} = (e^x - 1) \cdot e^{-x} \cdot \sin(y)$$

$$soln|x=0 \text{ and } y=0 \quad \text{true}$$

$$ode|y' = \text{impDif}(soln, x, y) \quad \text{true}$$

$$\text{DelVar } ode, soln \quad Done$$

$$\text{deSolve}\left(y'' = y^{-2} \text{ and } y(0)=0 \text{ and } y'(0)=0, t, y\right)$$

$$\frac{3}{2 \cdot y^4} = t$$

$$\text{solve}\left(\frac{3}{2 \cdot y^4} = t, y\right)$$

$$y = \frac{1}{3 \cdot 3^{\frac{1}{3}} \cdot 2^{\frac{2}{3}} \cdot t^{\frac{4}{3}}} \text{ and } t \geq 0$$

## deSolve()

Katalogi >

**deSolve(2.astODEandreunaehto1and reunaehto2, Muutt, riippuvaMuutt)**⇒*tiety ratkaisu*

Laskee tietyn ratkaisun, joka sopii 2.astODE:lle ja jolla on määritetyt arvot kahdessa eri pisteessä.

deSolve( $y''=x$  and  $y(0)=1$  and  $y'(2)=3,x,y$ )

$$y = \frac{x^3}{6} + x + 1$$

deSolve( $y''=2 \cdot y'$  and  $y(3)=1$  and  $y'(4)=2,x,y$ )

$$y = e^{2 \cdot x - 8} - e^{-2 + 1}$$

$$\text{deSolve}\left(w'' - \frac{2 \cdot w'}{x} + \left(9 + \frac{2}{x^2}\right); w = x \cdot e^x \text{ and } w\left(\frac{\pi}{6}\right) = 0 \text{ and } w\left(\frac{\pi}{3}\right) = 0, x, w\right)$$
$$w = \frac{x \cdot e^x}{(\ln(e))^2 + 9} + \frac{e^{\frac{\pi}{3}} \cdot x \cdot \cos(3 \cdot x)}{(\ln(e))^2 + 9} - \frac{e^{\frac{\pi}{6}} \cdot x \cdot \sin(3 \cdot x)}{(\ln(e))^2 + 9}$$

## det()

Katalogi >

**det(neliömatriisi[, Toleranssi])**⇒*lauseke*

Laskee neliömatriisin determinantin.

Valinnaisesti kaikkia matriisielementtejä käsitellään nollana, jos niiden itseisarvo on pienempi kuin *Toleranssi*. Tätä toleranssia käytetään vain, jos matriisissa on liukulukusyötteitä eikä se sisällä symbolisia muuttujia, joille ei ole määritetty arvoa. Muussa tapauksessa *Toleranssia* ei huomioida.

- Jos käytät painikkeita   tai **Automaattinen tai likimääräinen** -tilan valintaa Approximate (Likimääräinen), laskut suoritetaan liukulukuaritmetiikalla.
- Jos *Toleranssi* jätetään pois tai sitä ei käytetä, oletusarvoinen toleranssi lasketaan seuraavasti:

$$5E-14 \cdot \max(\text{dim}(\text{neliömatriisi}) \cdot \text{rowNorm}(\text{neliömatriisi}))$$

$$\det\left(\begin{bmatrix} a & b \\ c & d \end{bmatrix}\right) \quad a \cdot d - b \cdot c$$

$$\det\left(\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}\right) \quad -2$$

$$\det\left(\text{identity}(3) - x \cdot \begin{bmatrix} 1 & -2 & 3 \\ -2 & 4 & 1 \\ -6 & -2 & 7 \end{bmatrix}\right) \quad -(98 \cdot x^3 - 55 \cdot x^2 + 12 \cdot x - 1)$$

$$\begin{bmatrix} 1. \text{E}20 & 1 \\ 0 & 1 \end{bmatrix} \rightarrow \text{matI} \quad \begin{bmatrix} 1. \text{E}20 & 1 \\ 0 & 1 \end{bmatrix}$$

$$\det(\text{matI}) \quad 0$$

$$\det(\text{matI}, 1) \quad 1. \text{E}20$$

**diag()**

Katalogi &gt;

**diag(Lista)** ⇒ matriisi  
**diag(rivimatriisi)** ⇒ matriisi  
**diag(sarakematriisi)** ⇒ matriisi

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

Laskee matriisin, joka sisältää arvot argumenttilistassa tai matriisin sen päälävistäjässä.

**diag(neliömatriisi)** ⇒ rivimatriisi

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

Laskee rivimatriisin, joka sisältää elementit *neliömatriisin* päälävistäjästä.

*neliömatriisi*:n on oltava neliö.

**dim()**

Katalogi &gt;

**dim(Lista)** ⇒ kokonaisluku

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

Laskee *Listan* mitat.

**dim(Matriisi)** ⇒ lista

dim( $\begin{pmatrix} 1 & -1 \\ 2 & -2 \\ 3 & 5 \end{pmatrix}$ )	{3,2}
--	-------

Laskee matriisin mitat kahden elementin listana {rivit, sarakkeet}.

**dim(Merkkijono)** ⇒ kokonaisluku

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

Laskee merkkijonon *Merkkijono* sisältämien merkkien lukumäärän.

**Disp**

Katalogi &gt;

**Disp lausTaiMerkkijono1 [, lausTaiMerkkijono2] ...**

Näyttää *Laskin*-sovelluksen historiatietojen sisältämät argumentit. Argumentit näytetään peräkkäin, ja erotinmerkkeinä käytetään ohuita välilyöntejä.

Käyttökelpoisia pääasiassa ohjelmissa ja funktioissa, jotta välilaskutoimitusten näyttäminen voidaan varmistaa.

**Huomaa esimerkkiä syöttäessäsi:** Ohjeet monirivisten ohjelmien ja funktion määritysten syöttämisestä löytyvät tuotteen ohjekirjan *Laskin*-osiosta.

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

**DispAt** *int, lauseke1* [, *lauseke2* ...] ...

Komennolla **DispAt** voidaan määritellä rivi, jolla määrätty lauseke tai merkkijono näytetään ruudulla.

Rivinumero voidaan määritellä lausekkeeksi.

Huomaa, että rivin numero ei viittaa koko ruutuun vaan alueeseen, joka seuraa välittömästi komentoa/ohjelmaa.

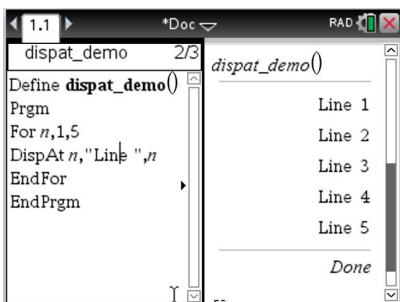
Tämä komento mahdollistaa ohjauspaneelin kaltaisen tuotoksen ohjelmista, joissa lausekkeen arvo tai anturin lukema päivitetään samalle riville.

Komentoja **DispAt** ja **Disp** voidaan käyttää samassa ohjelmassa.

**Huomaa:** Suurin sallittu numero on asetettu luvuksi 8, koska se vastaa koko näyttöä täynnä rivejä kannettavassa laitteessa – kunhan riveillä ei ole kaksiulotteisia matemaattisia lausekkeita. Rivien täsmällinen määrä riippuu näytetyn tiedon sisällöstä.

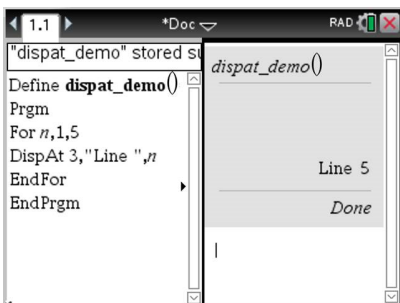
DispAt

#### Esimerkki



```

1.1 | *Doc | RAD |
-----|-----|
disp_at_demo 2/3 | disp_at_demo()
Define disp_at_demo()
Prgm
For n,1,5
DispAt n,"Line ",n
EndFor
EndPrgm
  
```



```

1.1 | *Doc | RAD |
-----|-----|
"disp_at_demo" stored in memory | disp_at_demo()
Define disp_at_demo()
Prgm
For n,1,5
DispAt 3,"Line ",n
EndFor
EndPrgm
  
```

#### Havainnollistavia esimerkkejä:

Define z()=	Ulostulo
Prgm	z()
For n,1,50	Iteraatio 1:
DispAt 1,"N: ",n	Rivi 1: N:1
Disp "Hello"	Rivi 2: Hello
EndFor	Iteraatio 2:
EndPrgm	Rivi 1: N:2
	Rivi 1: Hello
	Rivi 3: Hello
	Iteraatio 2:
	Rivi 1: N:3

	Rivi 2: Hello Rivi 3: Hello Rivi 4: Hello
Define z1() Prgm For n,1,50 DispAt 1,"N: ",n EndFor  For n,1,50 Disp "Hello" EndFor EndPrgm	z1()  Rivi 1: N:3 Rivi 2: Hello Rivi 3: Hello Rivi 4: Hello Rivi 5: Hello

## Virhetilat:

Virheviestit	Kuvaus
DisplAt-rivinumeron on oltava lukujen 1 ja 8 välillä	Lauseke arvioi rivinumeron välin 1–8 (mukaan lukien) ulkopuolella
Liian vähän argumentteja	Toiminnosta tai komennosta puuttuu yksi tai useampi argumentti
Ei argumentteja	Sama kuin nykyinen "syntaksivirhe" - dialogi
Liian monta argumenttia	Rajoita argumenttia. Sama virhe kuin Disp.
Virheellinen tietotyyppi	Ensimmäisen argumentin on oltava numero.
Mitätön: DispAt mitätön	"Hello World" Datatyyppivirhe on mitätöity (jos soittopyyntö on määritetty)
Muunnosoperaattori: DispAt 2_ft @> _m, "Hello World	<b>CAS:</b> Datatyyppi Virhe annetaan (jos takaisinkutsu on määritetty) <b>Numeerinen</b> Tulos arvioidaan ja jos tulos on kelpuutettu argumentti, DispAt tulostaa merkkijonon tuloslinjalla.

Laus ►DMS

Astekulmatilassa:

Arvo ►DMS

**Huomaa:** Voit syöttää tämän operaattorin tietokoneen näppäimistöltä kirjoittamalla @>DMS.

$$\{45.371\} \blacktriangleright \text{DMS} \quad 45^\circ 22' 15.6''$$

$$\{\{45.371, 60\}\} \blacktriangleright \text{DMS} \quad \{45^\circ 22' 15.6'', 60^\circ\}$$

Tulkitsee argumentin kulmana ja näyttää vastaavan DMS-luvun (DDDDDD°MM'SS.ss"). DMS-muoto (asteet, minuutit, sekunnit) on kuvattu kohdissa °, ', " sivulla sivu 248 .

**Huomaa:** ►DMS muuntaa radiaanit asteiksi, kun sitä käytetään radiaanitilassa. Jos syöteen perässä on asteen merkki °, muunnosta ei suoriteta. Voit käyttää komentoa ►DMS ainoastaan syöterivin lopussa.

**domain()**

**Luettelo >**

**domain(Laus1, Muut) ⇒ lauseke**

Antaa vastauksena *Laus1* määrittelyjoukon verrattuna *Muut*.

**domain()** voi käyttää funktioiden arvoalueiden tarkasteluun. Se on rajattu todelliseen ja äärelliseen määrittelyjoukkoon.

Tällä toiminnolla on rajoituksia johtuen tietokonealgebran sieventämis- ja ratkaisualgoritmien puutteista.

Tiettyjä funktioita ei voi käyttää argumentteina **domain()**, -funktioille riippumatta siitä, ilmaantuvatko ne eksplisiitisti tai käyttäjän määrittämien muuttujien ja funktioiden puitteissa. Seuraavassa esimerkissä lauseketta ei voi sieventää, sillä  $f()$  on kielletty funktio.

$\text{domain}\left(\frac{1}{x+y}, y\right)$	$-\infty < y < -x \text{ or } -x < y < \infty$
$\text{domain}\left(\frac{x+1}{x^2+2 \cdot x}, x\right)$	$x \neq -2 \text{ and } x \neq 0$
$\text{domain}\left(\left(\sqrt{x}\right)^2, x\right)$	$0 \leq x < \infty$
$\text{domain}\left(\frac{1}{x+y}, y\right)$	$-\infty < y < -x \text{ or } -x < y < \infty$

$$\text{domain}\left(\left[\begin{array}{c} x \\ \frac{1}{t} \\ dt, x \\ 1 \end{array}\right]\right) \blacktriangleright \text{domain}\left(\left[\begin{array}{c} x \\ \frac{1}{t} \\ dt, x \\ 1 \end{array}\right]\right)$$

**dominantTerm(LausI, Muutt [, Piste])** ⇒ lauseke

**dominantTerm(LausI, Muutt [, Piste]) | Muutt > Piste** ⇒ lauseke

**dominantTerm(LausI, Muutt [, Piste]) | Muutt < Piste** ⇒ lauseke

Laskee dominanttitermin *LausI*:n potenssarjaesityksestä, kun lauseke on lavennettu *Piste*ellä. Dominanttitermi on se, jonka suuruus kasvaa nopeimmin lähellä arvoa *Muutt = Piste*. Lausekkeen (*Muutt - Piste*) tuloksena olevalla potenssilla voi olla negatiivinen ja/tai murtolukueksponentti. Tämän potenssin kertoin voi sisältää lausekkeen (*Muutt - Piste*) logaritmeja ja muita *Muutt*:n funktioita, joita hallitsevat kaikki lausekkeen (*Muutt - Piste*) potenssit, joilla on sama eksponentin etumerkki.

*Pisteen* oletusarvo on 0. *Piste* voi olla ∞ tai -∞, jolloin dominanttitermi on termi, jolla on suurin *Muutt*:n eksponentti eikä pienin *Muutt*:n eksponentti.

**dominantTerm(...)** antaa tuloksena “**dominantTerm(...)**”, ellei se pysty määrittämään tällaista esitystä, kuten olennaisille erikoispisteille, esim. **sin** (1/z), kun z=0, e<sup>-1/z</sup>, kun z=0, tai e<sup>z</sup>, kun z = ∞ tai -∞.

$\text{dominantTerm}(\tan(\sin(x)) - \sin(\tan(x)), x)$	$\frac{x^7}{30}$
$\text{dominantTerm}\left(\frac{1 - \cos(x-1)}{(x-1)^3}, x, 1\right)$	$\frac{1}{2 \cdot (x-1)}$
$\text{dominantTerm}\left(x^{-2} \cdot \tan\left(\frac{1}{x^3}\right), x\right)$	$\frac{1}{x^3}$
$\text{dominantTerm}(\ln(x^x - 1) \cdot x^{-2}, x)$	$\frac{\ln(x \cdot \ln(x))}{x^2}$

$\text{dominantTerm}\left(e^{\frac{-1}{z}}, z\right)$	$e$
$\text{dominantTerm}\left(\left(1 + \frac{1}{n}\right)^n, n, \infty\right)$	$e$
$\text{dominantTerm}\left(\tan^{-1}\left(\frac{1}{x}\right), x, 0\right)$	$\frac{\pi \cdot \text{sign}(x)}{2}$
$\text{dominantTerm}\left(\tan^{-1}\left(\frac{1}{x}\right), x, x > 0\right)$	$\frac{\pi}{2}$



Jos sarjassa tai yhdellä sen derivaatoista on hyppypäjätkuvuus kohdassa *Piste*, tulos sisältää todennäköisesti alalausekkeita, jotka ovat muotoa sign (...) tai abs(...) reaalille kehitelmän muuttujalle tai  $(-1)^{\text{floor}(\dots \text{angle}(\dots) \dots)}$  kompleksille kehitelmän muuttujalle, joka on merkkiin “\_” päättyvä muuttuja. Jos tarkoituksesi on käyttää dominanttitermiä vain *Pisteen* toisella puolella oleville arvoille, siinä tapauksessa liitä funktioon **dominantTerm(...)** sopiva lauseke, “| *Muutt* > *Piste*”, “| *Muutt* < *Piste*”, “| *Muutt* ≥ *Piste*” tai “*Muutt* ≤ *Piste*”, jotta saat yksinkertaisemman vastauksen.

**dominantTerm()** jakautuu 1. argumentin listoihin ja matriiseihin.

**dominantTerm()** on hyödyllinen funktio, kun haluat selvittää mahdollisimman yksinkertaisen lausekkeen, joka on asymptoottinen toisen lausekkeen suhteen, esim. *Muutt* → *Piste*.

**dominantTerm()** on hyödyllinen myös silloin, kun sarjan ensimmäisen ei-nolla-termin astetta ei tiedetä, etkä halua arvilla iteratiivisesti tai interaktiivisesti ohjelmasilmukan avulla.

**Huomaa:** Katso myös **series()**, sivu 174.

dotP()

**dotP(Lista1, Lista2) ⇒ lauseke**

Laskee kahden listan “pistetulon”.

$\text{dotP}(\{a, b, c\}, \{d, e, f\})$	$a \cdot d + b \cdot e + c \cdot f$
$\text{dotP}(\{1, 2\}, \{5, 6\})$	17

**dotP(Vektori1, Vektori2) ⇒ lauseke**

Laskee kahden vektorin “pistetulon”.

$\text{dotP}([a \ b \ c], [d \ e \ f])$	$a \cdot d + b \cdot e + c \cdot f$
$\text{dotP}([1 \ 2 \ 3], [4 \ 5 \ 6])$	32

Kummankin on oltava rivivektoreita, tai kummankin on oltava sarakevektoreita.

E

**e^()****e^x** painike**e^(Lausl)**⇒lausekeLaskee **e:n** arvon korotettuna **Lausl:n** potenssiin.**Huomaa:** Katso myös **e eksponenttimalli**, sivu 2.**Huomaa:** Painikkeen **e^x** painaminen, jotta näkyviin saadaan **e^**, on eri asia kuin näppäimistön merkin **E** painaminen.Voit syöttää kompleksiluvun  $re^{i\theta}$  polaarissa muodossa. Käytä tätä muotoa kuitenkin vain radiaanikulmatilassa; aste- tai graadikulmatilassa se aiheuttaa määrittäjävirheen (Domain).**e^(Listal)**⇒listaLaskee **e:n** arvon korotettuna **Listal:n** jokaisen elementin potenssiin.**e^(neliomatriisi1)**⇒neliomatriisiLaskee **neliomatriisi1:n** matriisiekspotentin. Tämä ei ole sama kuin laskettaessa **e** korotettuna kunkin elementin mukaiseen potenssiin. Laskentamenetelmä on kuvattu kohdassa **cos()**.**neliomatriisi1:n** on oltava diagonalisoitavissa. Vastaus sisältää aina liukulukuja.

$e^1$	$e$
$e^1.$	2.71828
$e^{3^2}$	$e^9$

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

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

**eff()****Katalogi** > **eff(nimelliskorko, CpY)**⇒arvoTalouseläntätoiminto, joka muuntaa nimelliskorkokannan **nimelliskorko** efektiiviseksi vuosikoroksi, kun **CpY** määritetään korkojaksojen lukumääräksi vuodessa.**nimelliskoron** on oltava reaalityluku, ja **CpY:n** on oltava reaalityluku  $> 0$ .

$eff(5.75,12)$	5.90398
----------------	---------

Huomaa: Katso myös `nom()`, sivu 132.

## eigVc()

eigVc(neliömatriisi)⇒matriisi

Suorakulmakompleksimuodossa:

Laskee matriisin, joka sisältää ominaisvektorit reaaliselle tai kompleksiselle *neliömatriisille*, jossa jokainen vastauksen sarake vastaa ominaisarvoa. Huomaa, että ominaisvektori ei ole yksilöllinen; sitä voidaan skaalata millä tahansa vakiokertoimella. Ominaisvektorit ovat normaalimuotoisia, mikä tarkoittaa, että, jos  $V = [x_1, x_2, \dots, x_n]$ , tällöin:

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

*neliömatriisia* tasapainotetaan ensin similaarimuunnoksilla, kunnes rivi- ja sarakenormit ovat mahdollisimman lähellä samaa arvoa. Sen jälkeen *neliömatriisi* sievennetään Hessenbergin ylämatriisimuotoon ja ominaisvektorit lasketaan Schurin tekijöihin jaon menetelmällä.

$$\begin{bmatrix} -1 & 2 & 5 \\ 3 & -6 & 9 \\ 2 & -5 & 7 \end{bmatrix} \rightarrow mI \quad \begin{bmatrix} -1 & 2 & 5 \\ 3 & -6 & 9 \\ 2 & -5 & 7 \end{bmatrix}$$

eigVc(mI)

$$\begin{bmatrix} -0.800906 & 0.767947 & 0.57389 \\ 0.484029 & 0.573804+0.052258 \cdot i & 0.57389 \\ 0.352512 & 0.262687+0.096286 \cdot i & 0.2626 \end{bmatrix}$$

Jos haluat nähdä koko vastauksen, paina ▲ ja siirrä sen jälkeen osoitinta painikkeilla ◀ ja ▶.

## eigVl()

eigVl(neliömatriisi)⇒lista

Suorakulmakompleksimuodossa:

Laskee listan reaalisien tai kompleksien *neliömatriisien* ominaisarvoista.

*neliömatriisia* tasapainotetaan ensin similaarimuunnoksilla, kunnes rivi- ja sarakenormit ovat mahdollisimman lähellä samaa arvoa. Sen jälkeen *neliömatriisi* sievennetään Hessenbergin ylämatriisimuotoon ja ominaisarvot lasketaan Hessenbergin ylämatriisista.

$$\begin{bmatrix} -1 & 2 & 5 \\ 3 & -6 & 9 \\ 2 & -5 & 7 \end{bmatrix} \rightarrow mI \quad \begin{bmatrix} -1 & 2 & 5 \\ 3 & -6 & 9 \\ 2 & -5 & 7 \end{bmatrix}$$

eigVl(mI)

$$\{-4.40941, 2.20471+0.763006 \cdot i, 2.20471-0.763006 \cdot i\}$$

Jos haluat nähdä koko vastauksen, paina ▲ ja siirrä sen jälkeen osoitinta painikkeilla ◀ ja ▶.

## ElseIf

Katalogi > **If** *BooleanLaus1* Then*Lohko1***ElseIf** *BooleanLaus2* Then*Lohko2*

⋮

**ElseIf** *BooleanLausN* Then*LohkoN***EndIf**

⋮

**Huomaa esimerkkiä syöttäessäsi:** Ohjeet monirivisten ohjelmien ja funktion määrittysten syöttämisestä löytyvät tuotteen ohjekirjan Laskin-osiosta.

Define  $g(x)$ =FuncIf  $x \leq -5$  Then

Return 5

ElseIf  $x > -5$  and  $x < 0$  ThenReturn  $-x$ ElseIf  $x \geq 0$  and  $x \neq 10$  ThenReturn  $x$ ElseIf  $x = 10$  Then

Return 3

EndIf

EndFunc

*Done*

## EndFor

Katso For, sivu 78.

## EndFunc

Katso Func, sivu 82.

## EndIf

Katso If, sivu 94.

## EndLoop

Katso Loop, sivu 118.

## EndPrgm

Katso Prgm, sivu 148.

## euler ()

Katalogi > 

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

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

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

Käyttää Eulerin menetelmää  
järjestelmän ratkaisuun

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

muuttujalla *depVar*(*Var0*)=*depVar0*  
välillä [*Var0*,*VarMax*]. Laskee matriisin,  
jonka ensimmäinen rivi määrittelee *Var*  
tulosarvot ja jonka toinen rivi  
määrittelee ensimmäisen  
ratkaisukomponentin arvon vastaavilla  
*Var*-arvoilla jne.

*Expr* on oikea puoli, joka määrittelee  
tavallisen differentiaaliyhtälön (ODE).

*SystemOfExpr* on oikeiden puolten  
ryhmä, joka määrittelee ODE-yhtälöiden  
ryhmän (vastaa riippuvien muuttujien  
järjestystä kohdassa *ListOfDepVars*).

*ListOfExpr* on oikeiden puolten luettelo,  
joka määrittelee ODE-yhtälöiden ryhmän  
(vastaa riippuvien muuttujien järjestystä  
kohdassa *ListOfDepVars*).

Differentiaaliyhtälö:

$$y' = 0.001 \cdot y \cdot (100 - y) \text{ ja } y(0) = 10$$

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

Jos haluat nähdä koko vastauksen, paina ▲ ja  
siirrä sen jälkeen osoitinta painikkeilla ◀ ja ▶.

Vertaile yllä olevaa tulosta CAS:n tarkkaan  
tulokseen, joka on saatu käyttämällä deSolve  
()- ja seqGEN()-funktioita:

$$\text{deSolve}\{y' = 0.001 \cdot y \cdot (100 - y) \text{ and } y(0) = 10, t, y\}$$

$$y = \frac{100 \cdot (1.10517)^t}{(1.10517)^t + 9}$$

$$\text{seqGen}\left(\frac{100 \cdot (1.10517)^t}{(1.10517)^t + 9}, t, y, \{0, 100\}\right)$$

$$\{10., 10.9367, 11.9494, 13.0423, 14.2189\}$$

Yhtälöryhmä:

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

kun  $y1(0) = 2$  ja  $y2(0) = 5$

euler( $\begin{cases} y1 + 0.1 \cdot y1 \cdot y2 \\ 3 \cdot y2 - y1 \cdot y2 \end{cases}, t, \{y1, y2\}, \{0, 5\}, \{2, 5\}, 1$ )					
0.	1.	2.	3.	4.	5.
2.	1.	1.	3.	27.	243.
5.	10.	30.	90.	90.	-2070.

*Var* on riippumaton muuttuja.

*ListOfDepVars* on riippuvien muuttujien luettelo.

{*Var0*, *VarMax*} on kahden elementin lista, joka määrittää funktion integroinnin muuttujasta *Var0* muuttujaan *VarMax*.

*ListOfDepVars0* on riippuvien muuttujien alkuehtoien luettelo.

*VarStep* nolasta eroava numero niin, että  $\text{sign}(\text{VarStep}) = \text{sign}(\text{VarMax} - \text{Var0})$  ja ratkaisut lasketaan  $\text{Var0} + i \cdot \text{VarStep}$  kaikille  $i=0,1,2,\dots$  niin, että  $\text{Var0} + i \cdot \text{VarStep}$  on alueella [*var0*, *VarMax*] (muuttujalla *VarMax* ei ehkä ole ratkaisuarvoa).

*eulerStep* on positiivinen kokonaisluku (oletus 1), joka määrittelee Eulerin vaiheiden määrän tulosarvojen välillä. Eulerin menetelmän käyttämä varsinainen vaihemäärä on  $\text{VarStep} / \text{eulerStep}$ .

## eval()

## Laitevalikko

**eval(*Expr*)** ⇒ *string*

**eval()** on validi vain TI-Innovator™ Hub ohjelmointikomentojen komentoargumenteissa **Get**, **GetStr**, ja **Send**. Ohjelmisto käsittelee lausekkeen *Expr* ja korvaa **eval()**-ilmauksen lopputuloksella merkijoukkona

Argumentin *Expr* on sievennyttävä reaalitylukuksi.

Aseta RGB-ledin sininen väri puolelle intensiteetille.

<i>lum</i> :=127	127
Send "SET COLOR.BLUE eval( <i>lum</i> )"	Done

Palauta sininen väri OFF-tilaan.

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

**eval()**-argumentin on sievennyttävä reaalitylukuksi.

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

Ohjelmoi punainen väri voimistumaan

```

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

```

Suorita ohjelma.

<i>fadein()</i>	<i>Done</i>
<i>n:=0.25</i>	0.25
<i>m:=8</i>	8
<i>n·m</i>	2.
Send "SET COLOR.BLUE ON TIME eval(n·m)"	<i>Done</i>
<i>iostr.SendAns</i>	"SET COLOR.BLUE ON TIME 2"

Vaikka **eval()** ei näytä tulostaan, voi tuloksena saatavaa laitekomentojonoa katsoa komennon suorittamisen jälkeen tarkastamalla jonkin seuraavista erikoismuuttujista.

*iostr.SendAns*  
*iostr.GetAns*  
*iostr.GetStrAns*

**Huomio:** Katso myös **Get** (sivu 84), **GetStr** (sivu 91), ja **Send** (sivu 171).

## exact()

Katalogi > 


**exact(Lausl [, Toleranssi])** ⇒ lauseke  
**exact(Listal [, Toleranssi])** ⇒ lista  
**exact(Matriisil [, Toleranssi])** ⇒ matriisi

Laskee täsmällisen tilan aritmetiikalla argumentin vastaavan rationaaliluvun, mikäli mahdollista.

*Toleranssi* määrittää muunnoksen toleranssin; oletusarvo on 0 (nolla).

exact(0.25)	$\frac{1}{4}$
exact(0.333333)	$\frac{333333}{1000000}$
exact(0.333333,0.001)	$\frac{1}{3}$
exact(3.5·x+y)	$\frac{7 \cdot x}{2} + y$
exact({0.2,0.33,4.125})	$\left\{ \frac{1}{5}, \frac{33}{100}, \frac{33}{8} \right\}$

## Exit

Katalogi > 

### Exit

Funktion listaus:

Poistuu nykyisestä **For**-, **While**- tai **Loop**-lohkosta.

Exit-komento ei ole sallittu näiden kolmen silmukkarakenteen (**For**, **While** tai **Loop**) ulkopuolella.

**Huomaa esimerkkiä syöttäessäsi:** Ohjeet monirivisten ohjelmien ja funktion määritysten syöttämisestä löytyvät tuotteen ohjekirjan Laskin-osiosta.

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

**Laus ►exp**

Näyttää *Laus*:n *e*:n luonnollisen eksponentin arvolla. Tämä on näytön muunnosoperaattori. Sitä voidaan käyttää vain syöterivin lopussa.

**Huomaa:** Voit syöttää tämän operaattorin tietokoneen näppäimistöltä kirjoittamalla @>exp.

$\frac{d}{dx}(e^x + e^{-x})$	$2 \cdot \sinh(x)$
$2 \cdot \sinh(x) \blacktriangleright \text{exp}$	$e^x - e^{-x}$

**exp(LausI) ⇒ lauseke**

Laskee *e*:n arvon korotettuna *LausI*:n potenssiin.

**Huomaa:** Katso myös **e** eksponenttimalli, sivu 2.

Voit syöttää kompleksiluvun  $re^{i\theta}$  polaarisisä muodossa. Käytä tätä muotoa kuitenkin vain radiaanikulmatilassa; aste- tai graadikulmatilassa se aiheuttaa määrittäjäjoukkovirheen (Domain).

**exp(ListaI) ⇒ lista**

Laskee *e*:n arvon korotettuna *Listal*:n jokaisen elementin potenssiin.

$e^1$	$e$
$e^1.$	2.71828
$e^{3^2}$	$e^9$
$e^{\{1,1,0.5\}}$	$\{e, 2.71828, 1.64872\}$



## exp()

 painike

**exp(neliömatriisi I) ⇒ neliömatriisi**

Laskee *neliömatriisi I*:n matriisiekspONENTIN. Tämä ei ole sama kuin laskettaessa **e** korotettuna kunkin elementin mukaiseen potenssiin. Laskentamenetelmä on kuvattu kohdassa **cos()**.

*neliömatriisi I*:n on oltava diagonalisoitavissa. Vastaus sisältää aina liukulukuja.

	1	5	3	782.209	559.617	456.509
	4	2	1	680.546	488.795	396.521
e	6	-2	1	524.929	371.222	307.879

## exp▶lista()

Katalogi > 

**exp▶list(Laus, Muutt) ⇒ lista**

Tutkii, onko lausekkeessa *Laus* yhtälöitä, jotka on erotettu sanalla "or", ja laskee listan, joka sisältää yhtälöiden oikeat puolet, jotka ovat muotoa *Muutt=Laus*. Tällä tavoin voit saada helpolla tavalla joitakin ratkaisuarvoja, jotka on upotettu funktioiden **solve()**, **cSolve()**, **fMin()** ja **fMax()** vastauksiin.

**Huomaa:** **exp▶list()** ei ole välttämätön funktioiden **zeros** ja **cZeros()** kanssa, koska ne laskevat suoraan ratkaisulistan.

Voit syöttää tämän funktion näppäimistöltä kirjoittamalla **exp@>list (...)**.

$\text{solve}(x^2-x-2=0,x)$	$x=-1$ or $x=2$
$\text{exp}\blacktriangleright\text{list}(\text{solve}(x^2-x-2=0,x),x)$	$\{-1,2\}$

## expand()

Katalogi > 

**expand(Laus I [, Muutt]) ⇒ lauseke**  
**expand(Lista I [, Muutt]) ⇒ lista**  
**expand(Matriisi I [, Muutt]) ⇒ matriisi**

**expand(Laus I)** laskee *Laus I*:n arvon lavennettuna kaikkien muuttujiensa suhteen. Lavennus on polynomilavennus polynomeille ja osamurtolukulavennus rationaalilukulausekkeille.

$\text{expand}(x+y+1)^2$	$x^2+2\cdot x\cdot y+2\cdot x+y^2+2\cdot y+1$
$\text{expand}\left(\frac{x^2-x+y^2-y}{x^2\cdot y^2-x^2\cdot y-x\cdot y^2+x\cdot y}\right)$	$\frac{1}{x-1}-\frac{1}{x}-\frac{1}{y-1}-\frac{1}{y}$

Funktion **expand()** tehtävä on muuntaa *Laus1* yksinkertaisten termien summaksi ja/tai erotukseksi. Funktion **factor()** tehtävä sen sijaan on muuntaa *Laus1* yksinkertaisten tekijöiden tuloksi ja/tai osamääräksi.

**expand(Laus1, Muutt)** laskee *Laus1*:n arvon lavennettuna muuttujan *Muutt* suhteen. Muuttujan *Muutt* samanlaiset potenssit kerätään. Termit ja niiden tekijät lajitellaan siten, että *Muutt* on päämuuttuja. Kerättyjen kertoimien satunnaista tekijöihin jakamista tai laventumista voi esiintyä jonkin verran. Verrattuna siihen, että muuttuja *Muutt* jätettäisiin pois, tämä toiminto säästää usein aikaa, muistia ja näyttötilaa, ja samalla lausekkeesta tulee ymmärrettävämpi.

Vaikka muuttujia olisi vain yksi, muuttujan *Muutt* käytön ansiosta nimittäjää voidaan mahdollisesti jakaa täydellisemmin tekijöihinsä lavennettaessa murtolukua osittain.

Vinkki: Rationaalilausekkeissa **propFrac()** on nopeampi, mutta vähemmän äärimmäinen vaihtoehto kuin **expand()**.

**Huomaa:** Katso myös **comDenom()**, jossa käsitellään lavennetulla nimittäjällä lavennettua osoittajaa.

**expand(Laus1, [Muutt])** jakaa myös logaritmit ja murtopotenssit muuttujasta *Muutt* riippumatta. Jos logaritmeja ja murtopotensseja halutaan jakaa enemmän, tarvitaan mahdollisesta epäyhtälöehtoja, jotta voidaan varmistaa, että jotkin tekijät ovat ei-negatiivisia.

**expand(Laus1, [Muutt])** jakaa myös itseisarvot, **sign()**, ja eksponentit muuttujasta *Muutt* riippumatta.

$$\begin{array}{l} \text{expand}\left((x+y+1)^2, y\right) \quad y^2+2\cdot y\cdot(x+1)+(x+1)^2 \\ \text{expand}\left((x+y+1)^2, x\right) \quad x^2+2\cdot x\cdot(y+1)+(y+1)^2 \\ \text{expand}\left(\frac{x^2-x+y^2-y}{x^2\cdot y^2-x^2\cdot y-x\cdot y^2+x\cdot y}, y\right) \\ \frac{1}{y-1}-\frac{1}{y}+\frac{1}{x\cdot(x-1)} \\ \text{expand}(Ans, x) \quad \frac{1}{x-1}-\frac{1}{x}+\frac{1}{y\cdot(y-1)} \end{array}$$

$$\begin{array}{l} \text{expand}\left(\frac{x^3+x^2-2}{x^2-2}\right) \quad \frac{2\cdot x}{x^2-2}+x+1 \\ \text{expand}(Ans, x) \quad \frac{1}{x-\sqrt{2}}+\frac{1}{x+\sqrt{2}}+x+1 \end{array}$$

$$\begin{array}{l} \frac{\ln(2\cdot x\cdot y)+\sqrt{2\cdot x\cdot y}}{\text{expand}(Ans)} \quad \frac{\ln(2\cdot x\cdot y)+\sqrt{2\cdot x\cdot y}}{\ln(x\cdot y)+\sqrt{2\cdot x\cdot y}+\ln(2)} \\ \text{expand}(Ans), y \geq 0 \\ \frac{\ln(x)+\sqrt{2\cdot x}\cdot\sqrt{y}+\ln(y)+\ln(2)}{\text{sign}(x\cdot y)+|x\cdot y|+e^{2\cdot x+y}} \\ \frac{e^{2\cdot x+y}+\text{sign}(x\cdot y)+|x\cdot y|}{\text{expand}(Ans)} \\ \frac{\text{sign}(x)\cdot\text{sign}(y)+|x|\cdot|y|+(e^x)^2\cdot e^y}{\text{expand}(Ans)} \end{array}$$

**Huomaa:** Katso myös **tExpand()**, jossa käsitellään trigonometrista kulma-summa- ja monikulmalavennusta.

## expr()

**expr(Merkkijono)** ⇒ lauseke

Määrittää *Merkkijonon* sisältämän merkkijonon lausekkeena ja suorittaa toimenpiteen välittömästi.

expr("1+2+x^2+x")	$x^2+x+3$
expr("expand((1+x)^2)")	$x^2+2\cdot x+1$
"Define cube(x)=x^3" → funcstr	"Define cube(x)=x^3"
expr(funcstr)	Done
cube(2)	8

## ExpReg

**ExpReg**  $X, Y$  [, [Frekv][, Luokka, Sisällytä]]

Laskee eksponentiaalisen regressiony =  $a \cdot (b)^x$  listoista  $X$  ja  $Y$  frekvenssillä *Frekv*. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 191.)

Kaikkien listojen on oltava samankokoisia *Sisällytä*-listaa lukuunottamatta.

$X$  ja  $Y$  ovat riippumattomien ja riippuvien muuttujien listoja.

*Frekv* on valinnainen frekvenssiarvojen lista. Jokainen *Frekv*:n elementti määrittää kunkin vastaavan datapisteen  $X$  ja  $Y$  esiintymisfrekvenssin. Oletusarvo on 1. Kaikkien elementtien on oltava kokonaislukuja  $\geq 0$ .

*Luokka* on luokkakoodien lista vastaavalle  $X$ - ja  $Y$ -datalle.

*Sisällytä* on yhden tai usemman luokkakoodin lista. Vain ne datayksiköt, joiden luokkakoodi sisältyy tähän listaan, ovat mukana laskutoimituksessa.

Lisätietoja listassa olevien tyhjien elementtien vaikutuksesta, katso Tyhjät elementtisivulla sivu 272.

Tulosmuuttuja	Kuvaus
stat.RegEqn	Regressioyhtälö: $a \cdot (b)^x$
stat.a, stat.b	Regressiokertoimet
stat.r <sup>2</sup>	Muunnettujen tietojen lineaarimäärittelyn kerroin
stat.r	Muunnettujen tietojen korrelaatiokerroin ( $x, \ln(y)$ )
stat.Resid	Eksponentiaalimalliin liittyvät jäännökset
stat.ResidTrans	Muunnettujen tietojen lineaariseen sovitukseen liittyvät jäännökset
stat.XReg	Muokatun <i>X Lista</i> :n sisältämä datapisteiden lista, jota käytetään regressiossa komentojen <i>Frekv</i> , <i>Luokkalista</i> ja <i>Sisällytä luokat</i> rajoitusten mukaisesti
stat.YReg	Muokatun <i>Y Lista</i> :n sisältämä datapisteiden lista, jota käytetään regressiossa komentojen <i>Frekv</i> , <i>Luokkalista</i> ja <i>Sisällytä luokat</i> rajoitusten mukaisesti
stat.FreqReg	Komentoja <i>stat.XReg</i> ja <i>stat.YReg</i> vastaava frekvenssiliista

## F

### factor()

Katalogi > 

**factor(LausI[, Muutt])** ⇒ lauseke

**factor(ListaI[, Muutt])** ⇒ lista

**factor(MatriisiI[, Muutt])** ⇒ matriisi

**factor(LausI)** jakaa *LausI*:n kaikki muuttujat tekijöihin yhteisen nimityksen suhteen.

*LausI* :ä jaetaan tekijöihin mahdollisimman paljon lineaaristen rationaalilukutekijöiden suuntaan ilman, että uusia ei-reaalisia alalausekkeita syntyy. Tämä vaihtoehto on sopiva, jos haluat jakaa lausekkeen tekijöihin useamman kuin yhden muuttujan suhteen.

**factor(LausI, Muutt)** laskee *LausI*:n jaettuna tekijöihin muuttujan *Muutt* suhteen.

$$\frac{\text{factor}(a^3 \cdot x^2 - a \cdot x^2 - a^3 + a)}{a \cdot (a-1) \cdot (a+1) \cdot (x-1) \cdot (x+1)}$$

$$\frac{\text{factor}(x^2+1)}{x^2+1}$$

$$\frac{\text{factor}(x^2-4)}{(x-2) \cdot (x+2)}$$

$$\frac{\text{factor}(x^2-3)}{x^2-3}$$

$$\frac{\text{factor}(x^2-a)}{x^2-a}$$

$$\frac{\text{factor}(a^3 \cdot x^2 - a \cdot x^2 - a^3 + a, x)}{a \cdot (a^2-1) \cdot (x-1) \cdot (x+1)}$$

$$\frac{\text{factor}(x^2-3, x)}{(x+\sqrt{3}) \cdot (x-\sqrt{3})}$$

$$\frac{\text{factor}(x^2-a, x)}{(x+\sqrt{a}) \cdot (x-\sqrt{a})}$$

*Lausl* :ä jaetaan mahdollisimman paljon kohti reaalisia tekijöitä, jotka ovat lineaarisia muuttujassa *Muutt*, vaikka tästä syntyisi irrationaalisia vakioita tai alalausekkeita, jotka ovat irrationaalisia muissa muuttujissa.

Tekijät ja niiden termit lajitellaan siten, että *Muutt* on päämuuttuja. Muuttujan *Muutt* samanlaiset potenssit kerätään jokaisessa tekijässä. Muuttujan *Muutt* tulee olla mukana, jos vain kyseistä muuttujaa halutaan jakaa tekijöihin ja jos irrationaalilausekkeet ovat hyväksyttäviä kaikissa muissa muuttujissa, jotta muuttujaa *Muutt* voitaisiin jakaa enemmän tekijöihin. Toimenpiteessä voi esiintyä jonkin verran satunnaista muiden muuttujien tekijöihin jakamista.

#### Auto or Approximate (Automaattinen tai likimääräinen) -tilan Auto

(Automaattinen) -asetuksessa muuttujan *Muutt* mukanaolo sallii likiarvoistamisen liukulukukertoimilla, kun irrationaalisia kertoimia ei voida ilmaista täsmällisen tiiviisti sisäänrakennetuilla termeillä. Vaikka muuttujia olisi vain yksi, muuttujan *Muutt* mukanaolo voi tuottaa täydellisemmän tekijöihin jakamisen.

**Huomaa:** Katso myös `comDenom()`, jossa on kuvattu nopea tapa suorittaa osittainen tekijöihin jako, kun `factor()` ei ole tarpeeksi nopea tai käyttää liikaa muistia.

**Huomaa:** Katso myös `cFactor()`, jossa on kuvattu täydellinen tekijöihin jako kompleksikertoimiksi etsittäessä lineaarisia tekijöitä.



$$\frac{\text{factor}(x^5+4\cdot x^4+5\cdot x^3-6\cdot x-3)}{x^5+4\cdot x^4+5\cdot x^3-6\cdot x-3}$$

$$\frac{\text{factor}(x^5+4\cdot x^4+5\cdot x^3-6\cdot x-3,x)}{(x-0.964673)\cdot(x+0.611649)\cdot(x+2.12543)\cdot(x^2$$

**factor**(*rationaaliluku*) laskee rationaaliluvun, joka on jaettu jaottomiin tekijöihin. Sekalukujen kohdalla laskenta-aika pitenee eksponentiaalisesti toiseksi suurimman tekijän sisältämien numeroiden määrän suhteen. Esimerkiksi 30-numeroisen kokonaisluvun tekijöihin jakaminen voi kestää pitempään kuin vuorokauden ja 100-numeroisen luvun pitempään kuin vuosisadan.

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

Pysäytä laskenta käsin,

- **Kämmenlaite:** Pidä -painiketta painettuna ja paina toistuvasti -painiketta.
- **Windows®:** Pidä **F12**-näppäintä pohjassa ja paina toistuvasti **Enter**.
- **Macintosh®:** Pidä **F5**-näppäintä pohjassa ja paina toistuvasti **Enter**.
- **iPad®:** Sovellus näyttää kehotuksen. Voit jatkaa odottamista tai peruuttaa.

Jos haluat pelkästään määrittää, onko jokin luku jaoton, käytä sen sijaan kommentoa **isPrime()**. Se on paljon nopeampi, erityisesti jos *rationaaliluku* ei ole jaoton, ja jos toiseksi suurimmassa tekijässä on enemmän kuin viisi numeroa.

### FCdf

(*alaraja,yläraja,dfOsoitt,dfNimitt*)⇒*luku*,  
jos *alaraja* ja *yläraja* ovat lukuja, *lista*, jos  
*alaraja* ja *yläraja* ovat listoja

### FCdf

(*alaraja,yläraja,dfOsoitt,dfNimitt*)⇒*luku*,  
jos *alaraja* ja *yläraja* ovat lukuja, *lista*, jos  
*alaraja* ja *yläraja* ovat listoja

Laskee F-jakauman todennäköisyyden *alarajan* ja *ylärajan* välillä määritellylle *dfOsoittajalle* (vapausaste) ja *dfNimittäjälle*.

Aseta  $P(X \leq \text{yläraja})$ :lle *alaraja* = 0.

## Fill

Fill *Laus*, *matriisiMuutt* ⇒ *matriisi*

Korvaa muuttujan *matriisiMuutt* jokaisen elementin lausekkeella *Laus*.

*matriisiMuutt*ujan on oltava valmiiksi olemassa.

Fill *Laus*, *listaMuutt* ⇒ *lista*

Korvaa muuttujan *listaMuutt* jokaisen elementin lausekkeella *Laus*.

*listaMuutt*ujan on oltava valmiiksi olemassa.

$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$ → <i>amatrix</i>	$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$
Fill 1.01, <i>amatrix</i>	Done
<i>amatrix</i>	$\begin{bmatrix} 1.01 & 1.01 \\ 1.01 & 1.01 \end{bmatrix}$
$\{1,2,3,4,5\}$ → <i>alist</i>	$\{1,2,3,4,5\}$
Fill 1.01, <i>alist</i>	Done
<i>alist</i>	$\{1.01,1.01,1.01,1.01,1.01\}$

## FiveNumSummary

FiveNumSummary *X*[,*[Frekv]* [,*Luokka,Sisällytä*]]

Antaa lyhennetyn version 1 muuttujan tilastoista listalle *X*. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 191.)

*X* edustaa datan sisältävää listaa.

*Frekv* on valinnainen frekvenssiarvojen lista. Jokainen *Frekv*:n elementti määrittää kunkin vastaavan *X*:n arvon esiintymisfrekvenssin. Oletusarvo on 1. Kaikkien elementtien on oltava kokonaislukuja 0.

*Luokka* on numeeristen luokkakoodien lista vastaaville *X*:n arvoille.

*Sisällytä* on yhden tai usemman luokkakoodin lista. Vain ne datayksiköt, joiden luokkakoodi sisältyy tähän listaan, ovat mukana laskutoimituksessa.

Tyhjä elementti jossakin listassa  $X$ , *Frekv* tai *Luokka* saa aikaan, että kaikkien listojen vastaava elementti on tyhjä. Lisätietoja tyhjistä elementeistä, katso sivu 272.

Tulosmuuttuja	Kuvaus
stat.MinX	x:n arvojen minimi
stat.Q <sub>1</sub> X	x:n ensimmäinen neljännes
stat.MedianX	x:n mediaani
stat.Q <sub>3</sub> X	x:n kolmas neljännes
stat.MaxX	x:n arvojen maksimi

**floor()**

**floor(Lausl)** ⇒ kokonaisluku

$$\text{floor}(-2.14)$$

-3.

Laskee suurimman kokonaisluvun, joka on  $\leq$  argumentti. Tämä funktio on identtinen funktion **int()** kanssa.

Argumentti voi olla reaali- tai kompleksiluku.

**floor(Lista l)** ⇒ lista

**floor(Matriisi l)** ⇒ matriisi

$$\text{floor}\left(\left\{\frac{3}{2}, 0, -5.3\right\}\right) \quad \{1, 0, -6\}$$

Määrittää listan tai matriisin jokaisen elementin alarajasta.

$$\text{floor}\left(\begin{array}{cc} 1.2 & 3.4 \\ 2.5 & 4.8 \end{array}\right) \quad \begin{array}{cc} 1. & 3. \\ 2. & 4. \end{array}$$

**Huomaa:** Katso myös **ceiling()** ja **int()**.

**fMax()**

**fMax(Laus, Muutt)** ⇒ Boolean lauseke  
**fMax(Laus, Muutt, alaraja)**

$$\text{fMax}\left(1-(x-a)^2-(x-b)^2, x\right) \quad x = \frac{a+b}{2}$$

**fMax(Laus, Muutt, alaraja, ylärajaja)**

$$\text{fMax}\left(.5 \cdot x^3 - x - 2, x\right) \quad x = \infty$$

**fMax(Laus, Muutt) | alaraja ≤ Muutt**



$\leq$ yläraja

Laskee Boolean lausekkeen, joka määrittää mahdollisia arvoja muuttujalle *Muutt*, joilla saadaan suurin lausekkeen *Laus* arvo tai jotka määrittävät sen pienimmän ylärajan.

Voit käyttää sijoitusoperaattoria ("|") rajoittaaksesi ratkaisuväliä ja/tai määrittääksesi muita ehtoja.

$$fMax(0.5 \cdot x^3 - x - 2, x) | x \leq 1 \quad x = 0.816497$$

**Auto or Approximate (Automaattinen tai likimääräinen)** -tilan Approximate (Likimääräinen) -asetuksessa **fMax()** etsii iteratiivisesti yhtä likimääräistä paikallista maksimiarvoa. Tämä on usein nopeampi menetelmä, erityisesti jos rajoitat haun operaattorilla "|" suhteellisen pienelle välille, joka sisältää täsmälleen yhden paikallisen maksimin.

**Huomaa:** Katso myös **fMin()** ja **max()**.

**fMin(Laus, Muutt)**  $\Rightarrow$  Boolean lauseke

**fMin(Laus, Muutt, alaraja)**

**fMin(Laus, Muutt, alaraja, yläraja)**

**fMin(Laus, Muutt) | alaraja  $\leq$  Muutt  $\leq$  yläraja**

Laskee Boolean lausekkeen, joka määrittää mahdollisia arvoja muuttujalle *Muutt*, joilla saadaan pienin lausekkeen *Laus* arvo tai jotka määrittävät sen suurimman alarajan.

Voit käyttää sijoitusoperaattoria ("|") rajoittaaksesi ratkaisuväliä ja/tai määrittääksesi muita ehtoja.

$$fMin(1 - (x-a)^2 - (x-b)^2, x) \quad x = -\infty \text{ or } x = \infty$$

$$fMin(0.5 \cdot x^3 - x - 2, x) | x \geq 1 \quad x = 1.$$

**Auto or Approximate (Automaattinen tai likimääräinen)** -tilan Approximate (Likimääräinen) -asetuksessa **fMin()** etsii iteratiivisesti yhtä likimääräistä paikallista minimiarvoa. Tämä on usein nopeampi menetelmä, erityisesti jos rajoitat haun operaattorilla "**|**" suhteellisen pienelle välille, joka sisältää täsmälleen yhden paikallisen minimin.

**Huomaa:** Katso myös **fMax()** ja **min()**.

**For**

**For** *Muutt, Matala, Korkea [, Askel]*  
*Lohko*

**EndFor**

Suorittaa *Lohkon* sisältämät lausekkeet iteratiivisesti jokaiselle muuttujan *Muutt* arvolle, *Matalasta Korkeaan* kohdassa *Askel* määritetyin portain.

*Muutt* ei saa olla järjestelmän muuttuja.

*Askel* voi olla positiivinen tai negatiivinen. Oletusarvo on 1.

*Lohko* voi olla joko yksi lauseke tai sarja lausekkeita, jotka on erotettu toisistaan kaksoispisteellä (:).

**Huomaa esimerkkiä syöttäessäsi:** Ohjeet monirivisten ohjelmien ja funktion määrittysten syöttämisestä löytyvät tuotteen ohjekirjan Laskin-osiosta.

Define $g()$ =Func	<i>Done</i>
Local <i>tempsum,step,i</i>	
$0 \rightarrow tempsum$	
$1 \rightarrow step$	
For $i,1,100,step$	
$tempsum+i \rightarrow tempsum$	
EndFor	
EndFunc	
$g()$	5050

**format()**

**format(Laus[,  
muotoMerkkijono])** $\Rightarrow$ merkkijono

Määrittää lausekkeen *Laus* merkkijonona muotoilumallin perusteella.

*Laus* on voitava sieventää luvuksi.

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

*muoto* Merkkijono on merkkijono, ja sen tulee olla muodossa: "F[n]", "S[n]", "E[n]", "G[n][c]", jossa [ ] ilmaisevat valinnaisia osia.

F[n]: Kiinteä muoto. n on desimaalipisteen jälkeen näytettävien numeroiden lukumäärä.

S[n]: Kymmenpotenssimuoto. n on desimaalipisteen jälkeen näytettävien numeroiden lukumäärä.

E[n]: Tekninen esitystapa. n on ensimmäisen merkitsevän numeron jälkeen näytettävien numeroiden lukumäärä. Eksponentti säätyy kolmella kerrolliseksi, ja desimaalipiste siirtyy 0, 1 tai 2 numeroa oikealle.

G[n][c]: Muuten sama kuin kiinteä muoto, mutta erottaa myös juuren vasemmalla puolella olevat numerot kolmen ryhmiin. c määrittää ryhmän erotusmerkin, ja sen oletusarvo on pilkku. Jos c on piste, juuri näytetään pilkkuna.

[Rc]: Mihin tahansa edellä mainituista määrittäjistä voidaan liittää Rc-juurilippu, jossa c on yksi merkki, joka määrittää korvauksen kohteen juuripisteestä.

## fPart()

**fPart**(*Lausl*) ⇒ *lauseke*

**fPart**(*Lista1*) ⇒ *lista*

**fPart**(*Matriisi1*) ⇒ *matriisi*

fPart(-1.234)	-0.234
---------------	--------

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

Laskee argumentin murtolukuosan.

Kun kyseessä on lista tai matriisi, laskee elementtien murtolukuosat.

Argumentti voi olla reaali- tai kompleksiluku.

$F\text{Pdf}(XArvo,dfOsoitt,dfNimitt)\Rightarrow luku$ , jos  $XArvo$  on luku, *lista*, jos  $XArvo$  on lista

Laskee F-jakauman todennäköisyyden  $XArvon$  kohdalle määritetyille  $dfOsoittajalle$  (vapausasteet) ja  $dfNimittäjälle$ .

## freqTable▶list()

### freqTable▶list

$(Lista1, frekvKokonaislukuLista)\Rightarrow lista$

Laskee listan, joka sisältää *Lista1*:n elementit lavennettuina *frekvKokonaislukuListan* määrittämien frekvenssien mukaisesti. Tätä funktiota voidaan käyttää laadittaessa frekvenssitaulukkoa Data & Tilastot -sovelluksessa.

*Lista1* voi olla mikä tahansa kelvollinen lista.

*frekvKokonaislukuListan* on oltava samankokoinen kuin *Lista1* ja sen tulee sisältää ainoastaan ei-negatiivisia kokonaislukuelementtejä. Jokainen elementti määrittää kuinka monta kertaa *Lista1*-elementti toistetaan tuloslistassa. Nolla-arvo sulkee pois vastaavan *Lista1*-elementin.

**Huomaa:** Voit syöttää tämän funktion tietokoneen näppäimistöltä kirjoittamalla `freqTable@>list(...)`.

Tyhjiä elementtejä ei huomioida. Lisätietoja tyhjästä elementistä, katso sivu 272.

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

## frequency()

$frequency(Lista1,lokerotLista)\Rightarrow lista$

$datalist=\{1,2,e,3,\pi,4,5,6,"hello",7\}$	
$\{1,2,2.71828,3,3.14159,4,5,6,"hello",7\}$	
$frequency(datalist,\{2.5,4.5\})$	$\{2,4,3\}$

Luo listan, joka sisältää *Lista1*:n elementtien lukumäärät. Lukumäärät perustuvat alueisiin (lokeroihin), jotka määritetään kohtaan *lokerotLista*.

Jos *lokerotLista* on  $\{b(1), b(2), \dots, b(n)\}$ , määritetyt alueet ovat  $\{? \leq b(1), b(1) < ? \leq b(2), \dots, b(n-1) < ? \leq b(n), b(n) > ?\}$ . Tuloksena oleva lista on yhden elementin pitempi kuin *lokerotLista*.

Jokainen vastauksen elementti vastaa niiden *Lista1*:n elementtien lukumäärää, jotka ovat kyseisen lokeron alueella. Funktion **countIf()** termeillä ilmaistuna vastaus on  $\{\text{countIf}(\text{list}, ? \leq b(1)), \text{countIf}(\text{list}, b(1) < ? \leq b(2)), \dots, \text{countIf}(\text{list}, b(n-1) < ? \leq b(n)), \text{countIf}(\text{list}, b(n) > ?)\}$ .

Niitä *Lista1*:n elementtejä, joita ei voi "lokeroida", ei huomioida. Tyhjiä elementtejä ei myöskään huomioida. Lisätietoja tyhjistä elementeistä, katso sivu 272.

Listat & Taulukot -sovelluksessa voit käyttää solualueita kummankin argumentin tilalla.

**Huomaa:** Katso myös **countIf()**, sivu 38.

Vastauksen selitys:

**2** *Datalistan* elementtiä on  $\leq 2.5$

**4** *Datalistan* elementtiä on  $> 2.5$  ja  $\leq 4.5$

**3** *Datalistan* elementtiä on  $> 4.5$

Elementti "hei" on merkkijono, jota ei voi sijoittaa mihinkään määritetyistä lokeroista.

## FTest\_2Samp

**FTest\_2Samp** *Lista1, Lista2[, Frekv1 [, Frekv2[, Hypot]]]*

**FTest\_2Samp** *Lista1, Lista2[, Frekv1[, Frekv2 [, Hypot]]]*

(*Datalistan* syöte)

**FTest\_2Samp** *sx1, n1, sx2, n2[, Hypot]*

**FTest\_2Samp** *sx1, n1, sx2, n2[, Hypot]*

(Yhteenvetotilaston syöte)

Suorittaa kahden otoksen F -testin. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 191.)

Kun  $H_a: \sigma_1 > \sigma_2$ , aseta *Hypot*>0

Kun  $H_a: \sigma_1 \neq \sigma_2$  (oletus), aseta *Hypot*=0

Kun  $H_a: \sigma_1 < \sigma_2$ , aseta *Hypot*<0

Lisätietoja listassa olevien tyhjen elementtien vaikutuksesta, katso Tyhjät elementitsivulla sivu 272.

Tulosmuuttuja	Kuvaus
stat.F	Laskettu $F$ -tilasto datasekvenssille
stat.PVal	Alin merkitsevyystaso, jolla nollahypoteesi voidaan hylätä
stat.dfNumer	osoittajan vapausasteet = $n_1 - 1$
stat.dfDenom	nimittäjän vapausasteet = $n_2 - 1$
stat.sx1, stat.sx2	Otoksen keskihajonnat <i>Lista 1</i> :n ja <i>Lista 2</i> :n sisältämille datasekvensseille
stat.x1_bar stat.x2_bar	Otoksen keskiarvot <i>Lista 1</i> :n ja <i>Lista 2</i> :n sisältämille datasekvensseille
stat.n1, stat.n2	Otosten koko

## Func

## Func

*Lohko*

## EndFunc

Malli käyttäjän määrittämän funktion luomista varten.

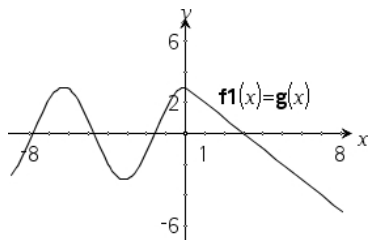
*Lohko* voi olla yksi lauseke tai sarja lausekkeita, jotka on erotettu toisistaan kaksoispisteellä (:), tai sarja eri riveillä olevia lausekkeita. Funktio voi käyttää **Return**-ohjetta tietyn vastauksen laskemiseen.

**Huomaa esimerkkiä syöttäessäsi:** Ohjeet monirivisten ohjelmien ja funktion määrittysten syöttämisestä löytyvät tuotteen ohjekirjan Laskin-osiosta.

Määritä paloittain määrittely funktio:

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

Funktion  $g(x)$  kuvaajan piirtämisen tulos



**gcd()**Katalogi > **gcd**(*Arvo1*, *Arvo2*) $\Rightarrow$ *lauseke* $\text{gcd}(18,33)$  3

Laskee kahden argumentin suurimman yhteisen jakajan. Kahden murtoluvun **gcd** on niiden osoittajien **gcd** jaettuna nimittäjien **lcm**:llä.

Auto or Approximate (Automaattinen tai likimääräinen) -tilassa murtoluvun liukulukujen **gcd** on 1.0.

**gcd**(*Lista1*, *Lista2*) $\Rightarrow$ *lista* $\text{gcd}(\{12,14,16\},\{9,7,5\})$  {3,7,1}

Laskee *Lista1*:n ja *Lista2*:n toisiaan vastaavien elementtien suurimmat yhteiset jakajat.

**gcd**(*Matriisi1*, *Matriisi2*) $\Rightarrow$ *matriisi* $\text{gcd}\left(\begin{pmatrix} 2 & 4 \\ 6 & 8 \end{pmatrix}, \begin{pmatrix} 4 & 8 \\ 12 & 16 \end{pmatrix}\right)$   $\begin{pmatrix} 2 & 4 \\ 6 & 8 \end{pmatrix}$ 

Laskee *Matriisi1* :n ja *Matriisi2*:n toisiaan vastaavien elementtien suurimmat yhteiset jakajat.

**geomCdf()**Katalogi > 

**geomCdf**(*p*,*alaraja*,*yläraja*) $\Rightarrow$ *luku*, jos *alaraja* ja *yläraja* ovat lukuja, *lista*, jos *alaraja* ja *yläraja* ovat listoja

**geomCdf**(*p*,*yläraja*)kun  $P(1 \leq X \leq \text{yläraja}) \Rightarrow$ *luku*, jos *yläraja* on luku, *lista*, jos *yläraja* on lista

Laskee kumulatiivisen geometrisen todennäköisyyden *alarajalta ylärajalle* määritetyllä onnistumistodennäköisyydellä *p*.

Kun  $P(X \leq \text{yläraja})$ , aseta *alaraja* = 1.

**geomPdf()**Katalogi > 

**geomPdf**(*p*,*XArvo*) $\Rightarrow$ *luku*, jos *XArvo* on luku, *lista*, jos *XArvo* on lista

Laskee diskreetin jakauman todennäköisyyden  $XArvo$ :n, eli ensimmäisen onnistuneen kokeen järjestysnumeron kohdalla, määritetyllä onnistumistodennäköisyyllä p.

**Get**

**Get** [*kehotemerkkijono*,] *var*[, *statusVar*]

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

Ohjelmointikomento: Noutaa arvon liitetystä TI-Innovator™ Hub ja sijoittaa arvon muuttujaan *var*.

Arvo täytyy kysyä:

- Etukäteen **Send "READ ..."** - komennolla.  
— tai —
- Upottamalla **"READ ..."** kysy valinnaisena *kehotemerkkijono*argumenttina. Tämä menetelmä antaa sinun käyttää yksittäistä komentoa kysyäksesi arvoa ja hakeaksesi sen.

Tapahtuu implisiittinen yksinkertaistus. Esimerkiksi vastaanotettu merkkijono "123" tulkitaan numeeriseksi arvoksi. Säilyttääksesi merkkijonon käytä toimintoa **GetStr** toiminnon **Get** sijaan.

Jos sisällytät valinnaisen argumentin *statusVar*, sille määrätään arvo toimenpiteen onnistumisen perusteella. Arvo nolla merkitsee, ettei tietoa ole vastaanotettu.

Järjestyksessä toisessa syntaksissa *func* (-)argumentti sallii ohjelman tallentaa vastaanotetun merkkijonon funktiomääritelmänä. Tämä syntaksi toimii ikään kuin ohjelma suorittaisi komennon:

**Laitevalikko**

Esimerkki: Kysy laitteen sisäänrakennetun valaistusanturin tämänhetkinen arvo. Käytä komentoa **Get** noutaaksesi arvon ja sijoittaaksesi se muuttujaan *lightval*.

Send "READ BRIGHTNESS"	Done
Get <i>lightval</i>	Done
<i>lightval</i>	0.347922

Upota READ-kysely komentoon **Get**.

Get "READ BRIGHTNESS" <i>lightval</i>	Done
<i>lightval</i>	0.378441



Määrittele  $func(arg1, \dots, argn) =$   
vastaanotettu merkkijono

Sen jälkeen ohjelma voi käyttää  
määriteltyä funktiota  $func()$ .

**Huomio:** Komentoa **Get** voi käyttää  
käyttäjän määrittelemän ohjelman  
sisällä, mutta ei funktion sisällä.

**Huomio:** Katso myös **GetStr**, sivu 91 ja  
**Send** sivu 171.

## getDenom()

Katalogi >

$getDenom(Laus1) \Rightarrow lauseke$

Muuttaa argumentin lausekkeeksi, jolla  
on sievennetty yhteinen nimittäjä, ja  
laskee sen jälkeen lausekkeen nimittäjän.

$getDenom\left(\frac{x+2}{y-3}\right)$	$y-3$
$getDenom\left(\frac{2}{7}\right)$	7
$getDenom\left(\frac{1}{x} + \frac{y^2+y}{y^2}\right)$	$x \cdot y$

## getKey()

Katalogi >

$getKey([0|1]) \Rightarrow returnString$

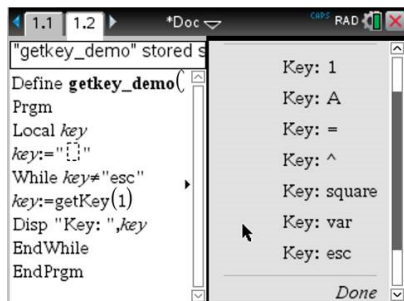
**Kuvaus:**  $getKey()$  – mahdollistaa TI-  
Basic -ohjelmalle näppäimistön –  
kannettava laite, pöytätietokone ja  
emulaattori pöytätietokoneella.

**Esimerkki:**

- alas painettu näppäin :=  $getKey()$   
palauttaa näppäimen tai tyhjän  
merkkijonon, jos mitään näppäintä  
ei ole painettu. Tämä komento  
palaa välittömästi.
- alas painettu näppäin :=  $getKey(1)$   
odottaa, kunnes jotakin näppäintä  
painetaan. Tämä komento  
pysäyttää ohjelman suorituksen,  
kunnes jotakin näppäintä on  
painettu.

$getKey()$

**Esimerkki:**



**Näppäimen pitäminen painettuna:**

<b>Kannettava laite/Emulaattorinäppäin</b>	<b>Pöytätietokone</b>	<b>Palauta arvo</b>
Esc	Esc	"esc"
Kosketuslevy – Yläpainallus	N/A	"ylös"
Päällä	N/A	"päävalikko"
Scratchapps	N/A	muistilehtiö
Kosketuslevy – Vasen painallus	N/A	"vasen"
Kosketuslevy – Keskipainallus	N/A	"keskus"
Kosketuslevy – Oikea painallus	N/A	"oikea"
Doc	N/A	"doc"
Tabulaattori	Tabulaattori	"Tabulaattori"
Kosketuslevy – Alapainallus	Nuoli alaspäin	"alas"
Päävalikko	N/A	"päävalikko"
Ctrl	Ctrl	ei palautusta
Vaihto	Vaihto	ei palautusta
Var	N/A	"var"
Poista	N/A	"poista"
=	=	"="
Trigonometria	N/A	Trigonometria
0:sta 9:ään	0-9	"0" ... 9
Sapluunat	N/A	"sapluuna"
Lista	N/A	"lista"
^	^	"^"
X <sup>2</sup>	N/A	"neliö"
/ (jakonäppäin)	/	"/"
* (kertonäppäin)	*	"**"
e <sup>x</sup>	N/A	"eksponentti"

Kannettava laite/Emulaattorinäppäin	Pöytätietokone	Palauta arvo
10^x	N/A	"10voima"
+	+	"+"
-	-	"_"
(	(	"{"
)	)	"}"
.	.	". "
(-)	N/A	"-" (negatiivinen merkki)
Syötä	Syötä	"syötä"
ee	N/A	"E" (tieteellinen merkintä E)
a – z	a–z	alfa = kirjainmerkki painettu (pieni kirjain) ("a" – "z")
vaihto a–z	vaihto a–z	alfa = kirjainmerkki painettu "A" - "Z"
		Huomaa: Ctrl-vaihto lukitsee isot kirjaimet
?!	N/A	"?!"
pii	N/A	"pij"
Lippu	N/A	ei palautusta
,	,	","
Palautus	N/A	Palautus
Välilyönti	Välilyönti	Välilyönti
Ei pääsyä	Näppäimet erikoismerkeille, kuten @,!,^, etc.	Kirjainmerkki on palautettu
N/A	Toimintonäppäimet	Kirjainmerkkejä ei ole palautettu
N/A	Erityiset näytön kontrollinäppäimet	Kirjainmerkkejä ei ole palautettu
Ei pääsyä	Muita näytön näppäimiä, jotka eivät ole käytettävissä	Sama kirjainmerkki, jonka saat Notesista (ei

Kannettava laite/Emulaattorinäppäin	Pöytätietokone	Palauta arvo
	laskimessa, kun getKey() odottaa näppäimen painallusta. ({, },, ;, ...)	matemaattisessa kentässä)

**Huomaa:** On tärkeää huomata, että ohjelman **getKey()** läsnäolo muuttaa ohjelmaa tiettyjen tapahtumien käsittelyä. Joitakin näistä kuvataan jäljempänä.

**Lopeta ohjelma ja käsittele tapahtuma** – aivan kuin jos käyttäjä poistuisi ohjelmasta painamalla **ON** -näppäintä

"Tuki" jäljempänä tarkoittaa – Järjestelmä toimii odotetusti – ohjelma jatkuu.

Tapahtuma	laite	Työpöytä – TI-Nspire™ Ohjelmisto opiskelijoille
Pikakysely	Lopeta ohjelma, käsittele tapahtuma	Sama kuin kannettava laite (TI-Nspire™ Student Software, TI-Nspire™ Navigator™ NC Teacher Software – vain)
Etätiedoston hallinta  (Sisältää lähetyksen "Exit Press 2 Test" -tiedoston toisesta kämmenlaitteesta tai työpöydän kannettavasta laitteesta)	Lopeta ohjelma, käsittele tapahtuma	Sama kuin kannettava laite. (TI-Nspire™ Student Software, TI-Nspire™ Navigator™ NC Teacher Software-vain)
Lopeta oppitunti	Lopeta ohjelma, käsittele tapahtuma	Tuki (TI-Nspire™ Student Software, TI-Nspire™ Navigator™ NC Teacher Software-vain)

Tapahtuma	Laite	Työpöytä – TI-Nspire™ Kaikki versiot
TI-Innovator™ Hub Yhdistä / katkaise	Tuki – onnistuu komennolla TI-Innovator™ Hub. Kun olet lopettanut ohjelman, TI-Innovator™ Hub työskentelee edelleen kämmenlaitteen kanssa.	Sama kuin kannettava laite

**getLangInfo()**

**Katalogi >** 

**getLangInfo()** ⇒ *merkkijono*

getLangInfo()

"en"

Antaa merkkijonon, joka vastaa parhaillaan käytössä olevan kielen lyhyttä nimeä. Voit käyttää sitä esimerkiksi ohjelmassa tai funktiossa nykyisen kielen määrittämiseen.

englanti = "en"  
 tanska = "da"  
 saksa = "de"  
 suomi = "fi"  
 ranska = "fr"  
 italia = "it"  
 hollanti = "nl"  
 flaami = "nl\_BE"  
 norja = "no"  
 portugali = "pt"  
 espanja = "es"  
 ruotsi = "sv"

## getLockInfo()

**getLockInfo**(*Muutt*)⇒*arvo*

Määrittää muuttujan *Muutt* nykyisen lukittu/lukitsematon-tilan.

*arvo* =0: *Muutt* on lukitsematon tai sitä ei ole olemassa.

*arvo* =1: *Muutt* on lukittu eikä sitä voi muuttaa tai poistaa.

Katso **Lock**, sivu 114, ja **unLock**, sivu 214.

<i>a</i> :=65	65
Lock <i>a</i>	Done
getLockInfo( <i>a</i> )	1
<i>a</i> :=75	"Error: Variable is locked."
DelVar <i>a</i>	"Error: Variable is locked."
Unlock <i>a</i>	Done
<i>a</i> :=75	75
DelVar <i>a</i>	Done

## getMode()

**getMode**  
(*TilanNimiKokonaisluku*)⇒*arvo*

**getMode(0)**⇒*lista*

**getMode**(*TilanNimiKokonaisluku*)  
 laskee arvon, joka vastaa *TilanNimiKokonaisluku*-tilan nykyistä asetusta.

getMode(0)	{ 1,7,2,1,3,1,4,1,5,1,6,1,7,1,8,1 }
getMode(1)	7
getMode(8)	1

**getMode(0)** laskee listan, joka sisältää lukupareja. Jokainen pari koostuu tilaa kuvaavasta kokonaisluvusta ja asetusta kuvaavasta kokonaisluvusta.

Tilat ja niiden asetukset on esitetty alla olevassa taulukossa.

Jos tallennat asetukset komennolla **getMode(0)** → *muutt*, voit käyttää komentoa **setMode(muutt)** funktiossa tai ohjelmassa ja tallentaa asetukset näin väliaikaisesti pelkästään funktion tai ohjelman suorituksen ajaksi. Katso **setMode()**, sivu 176.

Tilan nimi	Tilaa vastaava kokonaisluku	Asetuksia vastaavat kokonaisluvut
Näytettävät numerot	1	1=Liukuva, 2=Liukuva1, 3=Liukuva2, 4=Liukuva3, 5=Liukuva4, 6=Liukuva5, 7=Liukuva6, 8=Liukuva7, 9=Liukuva8, 10=Liukuva9, 11=Liukuva10, 12=Liukuva11, 13=Liukuva12, 14=Kiinteä0, 15=Kiinteä1, 16=Kiinteä2, 17=Kiinteä3, 18=Kiinteä4, 19=Kiinteä5, 20=Kiinteä6, 21=Kiinteä7, 22=Kiinteä8, 23=Kiinteä9, 24=Kiinteä10, 25=Kiinteä11, 26=Kiinteä12
Kulma	2	1=Radiaani, 2=Aste, 3=Graadi
EkspONENTTIMUOTO	3	1=Normaali, 2=Kymmenpotenssi, 3=Tekninen
Reaali- tai kompleksiluku	4	1=Reaali, 2=Suorakulma, 3=Polaarinen
Automaattinen tai likimääräinen.	5	1=Automaattinen, 2=Likimääräinen, 3=Täsmällinen
Vektorimuoto	6	1=Suorakulma, 2=Sylinteri, 3=Pallo
Kantaluku	7	1=Desimaali, 2=Heksagonaalinen, 3=Binaarinen
Yksikköjärjestelmä	8	1=SI, 2=Eng/US

## getNum()

Katalogi > 

**getNum**(*Lausl*) $\Rightarrow$ *lauseke*

Muuttaa argumentin lausekkeeksi, jolla on sievennetty yhteinen nimittäjä, ja laskee sen jälkeen lausekkeen osoittajan.

$\text{getNum}\left(\frac{x+2}{y-3}\right)$	$x+2$
$\text{getNum}\left(\frac{2}{7}\right)$	2
$\text{getNum}\left(\frac{1}{x}+\frac{1}{y}\right)$	$x+y$

## GetStr

Hub-valikko

**GetStr** [*kehotemerkkijono,*] *var*[, *statusVar*] Katso esimerkit kohdasta **Get**.

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

Ohjelmointikomento: Toimii samalla tavalla kuin **Get**-komento, mutta vastaanotettu arvo tulkitaan aina merkkijonoksi. **Get**-komento kuitenkin tulkitsee vastauksen lausekkeeksi, jollei sitä merkitä lainausmerkkien ("" ) sisään.

**Huomio:** Katso myös **Get**, sivu 84 ja **Send** sivu 171.

## getType()

Katalogi > 

**getType**(*var*) $\Rightarrow$ *merkkijono*

Antaa tulokseksi merkkijonon, joka ilmoittaa muuttujan *var* datatyyppin.

Jos muuttujaa *var* ei ole määritelty, tulokseksi tulee merkkijono "EI MITÄÄN".

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

**getVarInfo()** ⇒ matriisi tai merkkijono

**getVarInfo(LibNameString)** ⇒ matriisi tai merkkijono

**getVarInfo()** laskee tietomatriisin (muuttujan nimi, tyyppi, kirjaston käytettävyyys ja lukittu/lukitsematon-tila) kaikille nykyisessä tehtävässä määritetyille muuttujille ja kirjasto-objekteille.

Jos yhtään muuttujaa ei ole määritetty, **getVarInfo()** antaa vastauksena merkkijonon "NONE".

**getVarInfo(KirjNimiMerkkijono)** antaa tuloksena tietomatriisin kaikista kirjastossa *KirjNimiMerkkijono* määritetyistä kirjasto-objekteista. *KirjNimiMerkkijono* on oltava merkkijono (lainausmerkkien sisällä oleva teksti) tai merkkijonomuuttuja.

Jos kirjastoa *KirjNimiMerkkijono* ei ole olemassa, esiintyy virhe.

Huomaa vasemmanpuoleinen esimerkki, jossa funktion **getVarInfo()** vastaus on määritetty muuttujaan *vs*. Jos muuttujan *vs* riviä 2 tai riviä 3 yritetään näyttää, tuloksena on "Kelpaamaton lista tai matriisi" -virhe, koska vähintään yksi näiden rivien elementeistä (esimerkiksi muuttuja *b*) sieventyy uudelleen matriisiksi.

Tämä virhe voi esiintyä myös käytettäessä *Ans*-muuttujaa funktion **getVarInfo()** tuloksen uudelleenlaskennassa.

Järjestelmä antaa edellä mainitun virheen, koska ohjelmiston nykyinen versio ei tue yleistettyä matriisirakennetta, jossa matriisin elementti voi olla joko matriisi tai lista.

getVarInfo()	"NONE"												
Define x=5	Done												
Lock x	Done												
Define LibPriv y={1,2,3}	Done												
Define LibPub z(x)=3*x <sup>2</sup> -x	Done												
getVarInfo()	<table border="1"> <tr> <td>x</td> <td>"NUM"</td> <td>"{}"</td> <td>1</td> </tr> <tr> <td>y</td> <td>"LIST"</td> <td>"LibPriv"</td> <td>0</td> </tr> <tr> <td>z</td> <td>"FUNC"</td> <td>"LibPub"</td> <td>0</td> </tr> </table>	x	"NUM"	"{}"	1	y	"LIST"	"LibPriv"	0	z	"FUNC"	"LibPub"	0
x	"NUM"	"{}"	1										
y	"LIST"	"LibPriv"	0										
z	"FUNC"	"LibPub"	0										
getVarInfo(tmp3)	"Error: Argument must be a string"												
getVarInfo("tmp3")	<table border="1"> <tr> <td>volcyI2</td> <td>"NONE"</td> <td>"LibPub"</td> <td>0</td> </tr> </table>	volcyI2	"NONE"	"LibPub"	0								
volcyI2	"NONE"	"LibPub"	0										

a:=1	1												
b:=[1 2]	[1 2]												
c:=[1 3 7]	[1 3 7]												
vs:=getVarInfo()	<table border="1"> <tr> <td>a</td> <td>"NUM"</td> <td>"{}"</td> <td>0</td> </tr> <tr> <td>b</td> <td>"MAT"</td> <td>"{}"</td> <td>0</td> </tr> <tr> <td>c</td> <td>"MAT"</td> <td>"{}"</td> <td>0</td> </tr> </table>	a	"NUM"	"{}"	0	b	"MAT"	"{}"	0	c	"MAT"	"{}"	0
a	"NUM"	"{}"	0										
b	"MAT"	"{}"	0										
c	"MAT"	"{}"	0										
vs[1]	[1 "NUM" "{}" 0]												
vs[1,1]	1												
vs[2]	"Error: Invalid list or matrix"												
vs[2,1]	[1 2]												



**Goto tunnusnimi**

Siirtää ohjauksen tunnuksen  
*tunnusnimi*.

*tunnusnimi* on määritettävä samassa  
funktiossa käyttäen **Lbl**-ohjetta.

**Huomaa** esimerkkiä syöttäessäsi: Ohjeet  
monirivisten ohjelmien ja funktion  
määritysten syöttämisestä löytyvät  
tuotteen ohjekirjan Laskin-osiosta.

```
Define g() $\Rightarrow$ Func
  Local temp,i
  0  $\rightarrow$  temp
  1  $\rightarrow$  i
  Lbl top
  temp + i  $\rightarrow$  temp
  If i < 10 Then
  i + 1  $\rightarrow$  i
  Goto top
EndIf
Return temp
EndFunc
```

Done

---

*g()* 55

**►Grad**

*Laus1* ► **Grad**  $\Rightarrow$  lauseke

Muuttaa *Laus1*:n graadikulmaan.

**Huomaa:** Voit syöttää tämän  
operaattorin tietokoneen näppäimistöä  
kirjoittamalla **@>Grad**.

Astekulmatilassa:

---

(1.5) ► Grad (1.66667)<sup>g</sup>

Radiaanikulmatilassa:

---

(1.5) ► Grad (95.493)<sup>g</sup>

/

**identity()**

**identity(kokonaisluku)**  $\Rightarrow$  matriisi

Laskee identiteettimatriisin, jonka koko  
on *kokonaisluku*.

*Kokonaisluvun* on oltava positiivinen  
kokonaisluku.

---

identity(4)	1	0	0	0
	0	1	0	0
	0	0	1	0
	0	0	0	1

---

**Jos** *BooleanLaus*  
*Ilmaisut*

Define $g(x)=$ Func	<i>Done</i>
If $x<0$ Then	
Return $x^2$	
EndIf	
EndFunc	

**Jos** *BooleanLaus Niin*  
*Lohko*

**EndIf**

Jos *BooleanLaus* on tosi, suorittaa yhden lausekkeen *Lauseke* tai lausekkeiden lohkon *Lohko* ennen suorituksen jatkamista.

Jos *BooleanLaus* on epätosi, jatkaa suoritusta suorittamatta lauseketta tai lausekkeiden lohkoa.

Lohko voi olla joko yksi lauseke tai sarja lausekkeita, jotka on erotettu toisistaan kaksoispisteellä (:) merkki.

**Huomaa esimerkkiä syöttäessäsi:** Ohjeet monirivisten ohjelmien ja funktion määritysten syöttämisestä löytyvät tuotteen ohjekirjan Laskin-osiosta.

**Jos** *BooleanLaus, niin*  
*Lohko1*

**Tai**

*Lohko2*

**EndIf**

Jos *BooleanLaus* on tosi, suorittaa *Lohko1*:n ja sen jälkeen jättää väliin *Lohko2*:n.

Jos *BooleanLaus* on epätosi, ohittaa *Lohko1*:n, mutta suorittaa *Lohko2*:n.

*Lohko1* ja *Lohko2* voivat olla yksi lauseke.

$g(-2)$	4
---------	---

Define $g(x)=$ Func	<i>Done</i>
If $x<0$ Then	
Return $-x$	
Else	
Return $x$	
EndIf	
EndFunc	

$g(12)$	12
$g(-12)$	12

**Jos** *BooleanLaus1*, niin

*Lohko1*

**Jos taas** *BooleanLaus2*, niin

*Lohko2*

:

**Jos taas** *BooleanLausN*, niin

*LohkoN*

**EndIf**

Sallii haarautumisen *Jos BooleanLaus1* on tosi, suorittaa *Lohko1*:n *Jos BooleanLaus1* on epätosi, laskee *BooleanLaus2*:n jne.

Define  $g(x)=\text{Func}$

If  $x < 5$  Then

Return 5

ElseIf  $x > 5$  and  $x < 0$  Then

Return  $-x$

ElseIf  $x \geq 0$  and  $x \neq 10$  Then

Return  $x$

ElseIf  $x = 10$  Then

Return 3

EndIf

EndFunc

*Done*

$g(-4)$	4
$g(10)$	3

## ifFn()

**ifFn**(*BooleanLaus*, *Arvo\_Jos\_tosi* [, *Arvo\_Jos\_epätosi* [, *Arvo\_Jos\_tuntematon*]])  
 $\Rightarrow$  *lauseke, lista tai matriisi*

Laskee *BooleanLaus* (jokaiselle *BooleanLaus* ) elementille) ja antaa tuloksen noudattaen seuraavia sääntöjä:

- *BooleanLaus* voi testata yksittäisen arvon, listan tai matriisin.
- Jos jokin *BooleanLaus* elementti on tosi, laskee vastaavan elementin lausekkeesta *Arvo\_Jos\_tosi*.
- Jos jokin *BooleanLaus* elementti on epätosi, laskee vastaavan elementin lausekkeesta *Arvo\_Jos\_epätosi*. Jos jätät pois lausekkeen *Arvo\_Jos\_epätosi*, laskee määrittelemättömäksi.
- Jos *BooleanLaus* elementti ei ole tosi eikä epätosi, laskee vastaavan elementin *Arvo\_If\_unknown*. Jos jätät pois *Arvo\_If\_unknown*, laskee määrittelemättömäksi.
- Jos funktion **josFn()** toinen, kolmas tai neljäs argumentti on yksi lauseke, Boolean testiä sovelletaan jokaiseen sijaintiin Boolean lausekkeessa *BooleanLaus*.

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

**1:n** testiarvo on alle 2.5, joten sen vastaava

*Arvo\_Jos\_tosi* -elementti arvolle **5** kopioidaan vastausten listaan.

**2:n** testiarvo on alle 2.5, joten sitä vastaava

*Arvo\_Jos\_tosi*-elementti arvolle **6** kopioidaan vastausten listaan.

**3:n** testiarvo ei ole alle 2.5, joten sitä vastaava

*Arvo\_Jos\_epätosi*-elementti **10** kopioidaan vastausten listaan.

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

*Arvo\_Jos\_tosi* on yksittäinen arvo ja vastaa mitä tahansa valittua sijaintia.

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

Arvoa *Arvo\_Jos\_epätosi* ei ole määritelty. Käytetään merkintää EiMäär

**ifFn()**

Luettelo &gt;

**Huomaa:** Jos sievennetty *BooleanLaus* ilmaisu sisältää listan tai matriisin, kaikkien muiden lista- tai matriisiargumenttien on oltava samansuuruisia, ja myös tuloksen on oltava samansuuruinen.

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


---


$$\{6, "err"\}$$

Yksi elementti valittu lausekkeesta *Arvo\_Jos\_tosi*. Yksi elementti valittu lausekkeesta *Arvo\_Jos\_epätosi*.

**imag()**

Luettelo &gt;

**imag(Expr1) ⇒ lauseke**

Laskee argumentin imaginaarisen osan.

**Huomaa:** Kaikkia määrittelemättömiä muuttujia käsitellään reaali muuttujina. Katso myös *real()*, page 157

**imag(Lista1) ⇒ lista**

Laskee listan alkutekijöiden imaginaarisista osista.

**imag(Matriisi1) ⇒ matriisi**

Laskee matriisin alkutekijöiden imaginaarisista osista.

$\text{imag}(1+2 \cdot i)$	2
$\text{imag}(z)$	0
$\text{imag}(x+iy)$	y

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

$$\text{imag}\left(\begin{bmatrix} a & b \\ i \cdot c & i \cdot d \end{bmatrix}\right) \quad \begin{bmatrix} 0 & 0 \\ c & d \end{bmatrix}$$

**impDif()**

Luettelo &gt;

**impDif(Yhtälö, Muutt, riippuvaMuutt [,Aste]) ⇒ lauseke**

jossa luokan *Aste* oletusarvo on 1

Laskee implisiittisen derivaatan yhtälöille, joissa yksi muuttuja määritellään implisiittisesti toisen suhteen

$$\text{impDif}(x^2+y^2=100, x, y) \quad \frac{-x}{y}$$

**Epäsuora operaattori**

Katso #(), sivu 245.

## inString()

Luettelo > 

**inString**(*srcMerkkijono*, *alaMerkkijono* [, *Alku*]) ⇒ kokonaisluku

<code>inString("Hello there", "the")</code>	7
<code>inString("ABCEFG", "D")</code>	0

Laskee merkin paikan merkkijonossa *srcMerkkijono*, jossa merkkijonon *alaMerkkijono* ensimmäinen esiintyminen alkaa.

*Alku*, jos se sisältyy, määrää merkin paikan siinä merkkijonossa *srcMerkkijono*, josta haku alkaa. Oletusarvo = 1 (*srcMerkkijonon* ensimmäinen merkki).

Jos *srcMerkkijono* ei sisällä *alaMerkkijonoa* tai *Alku* on *srcMerkkijonon* pituus, vastaus on nolla.

## int()

Luettelo > 

**int**(*Laus*) ⇒ kokonaisluku

<code>int(-2.5)</code>	-3.
<code>int([-1.234 0 0.37])</code>	[-2. 0 0.]

**int**(*Lista1*) ⇒ lista

**int**(*Matriisi1*) ⇒ matriisi

Laskee suurimman kokonaisluvun, joka on pienempi tai yhtä suuri kuin argumentti. Tämä funktio on identtinen funktion pohjan **floor()** kanssa.

Argumentti voi olla reaali- tai kompleksiluku.

Kun kyseessä on lista tai matriisi, laskee kunkin elementin suurimman kokonaisluvun.

## intDiv()

Luettelo > 

**intDiv**(*Luku1*, *Luku2*) ⇒ kokonaisluku

**intDiv**(*Lista1*, *Lista2*) ⇒ lista

**intDiv**(*Matriisi1*, *Matriisi2*) ⇒ matriisi

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

Laskee lausekkeen  $Luku1 \div Luku2$  etumerkillisen kokonaislukuosan.

Laskee sarjoille ja matriiseille lausekkeen (argumentti 1 ÷ argumentti 2) etumerkillisen kokonaislukuosan kullekin elementtiparille.

**Interpoloi**(*xArvo*, *xList*, *yList*, *yPrimeList*) ⇒ lista

Tällä toiminnolla suoritetaan seuraavaa:

Kun ilmoitetaan *xList*, *yList*=**f**(*xList*) ja *yPrimeList*=**f'**(*xList*) jollekin tuntemattomalle funktiolle **f**, käytetään kuutiointerpolanttia funktion **f** määrittelemiseksi arvolla *xArvo*.

Oletetaan, että *xList* on monotonisesti kasvavien tai laskevien numeroiden lista, mutta tämän toiminnon tuloksena saattaa olla arvo, vaikka se ei olisikaan sitä. Tämä toiminto käy läpi listan *xList* etsien väliä [*xList*[*i*], *xList*[*i*+1]], joka sisältää arvon *xValue*. Jos se löytää tällaisen välin, se palauttaa interpoloidun arvon funktiolle **f**(*xValue*); muuten se antaa tuloksen **määrittelemätön**.

Sarjojen *xList*, *yList* ja *yPrimeList* on oltava samansuuruiset ≥ 2 ja sisällettävä lausekkeita, jotka sieventyvät luvuiksi.

*xValue* voi olla määrittelemätön muuttuja, luku tai lukuluettelo.

Differentiaaliyhtälö:

$$y' = -3y + 6t + 5 \text{ and } y(0) = 5$$

$$rk := rk23(-3 \cdot y + 6 \cdot t + 5, t, y, \{0, 10\}, 5, 1)$$

0.	1.	2.	3.	4.
5.	3.19499	5.00394	6.99957	9.00593

Jos haluat nähdä koko vastauksen, paina ▲ ja siirrä sen jälkeen osoitinta painikkeilla ◀ ja ▶.

Käytä interpolate()-funktioita laskeaksesi funktion arvot *x*-arvolistalle:

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

invχ<sup>2</sup>(*Ala*,*df*)

invChi2(*Ala*,*df*)

## inv $\chi^2$ ()

Luettelo > 

Laskee käänteisen kumulatiivisen  $\chi^2$  (chi-neliö) -todennäköisyysfunktion, joka on määritelty vapauden asteella,  $df$  annettulle käyrän alapuoliselle alalle  $Ala$ .

## invF()

Luettelo > 

**invF**( $Ala, dfOsoitt, dfNimitt$ )

**vakioF**( $Area, dfOsoitt, dfNimitt$ )

Laskee käänteisen kumulatiivisen F-jakaumafunktion, jolle on määritelty  $dfOsoitt$  ja  $dfNimitt$ , annettulle käyrän alapuoliselle alueelle  $Ala$ .

## invBinom()

Luettelo > 

**invBinom**  
( $CumulativeProb, NumTrials, Prob, OutputForm$ ) $\Rightarrow$  asteikkomuoto tai matriisi

Käänteinen binomi. Johtuen kokeiden ( $NumTrials$ ) ja kunkin kokeen todennäköisyydestä onnistua ( $Prob$ ), tämä toiminto laskee onnistumisten minimimäärän, ksiten, että arvo  $k$  on suurempi tai yhtä suuri kuin kumuloituva todennäköisyys ( $CumulativeProb$ ).

$OutputForm=0$ , näyttää tuloksen asteikkomuodossa (oletus).

$OutputForm=1$ , näyttää tuloksen matriisina.

Esimerkki: Mary ja Kevin pelaavat noppapeliä. Maryn on arvattava, miten monta kertaa numero 6 enintään esiintyy 30 heittoa kohti. Jos numero 6 esiintyy yhtä monta kertaa tai vähemmän, Mary voittaa. Lisäksi, mitä pienempi on hänen arvaamansa määrä, sitä suuremmat ovat hänen voittonsa. Mikä on pienin määrä, jonka Mary voi arvata, jos hän haluaa voittamisen todennäköisyyden olevan suurempi kuin 77 %?

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

## invBinomN()

Luettelo > 

**invBinomN**( $CumulativeProb, Prob, NumSuccess, OutputForm$ ) $\Rightarrow$  asteikkomuoto tai matriisi

Esimerkki: Monique harjoittelee koriin heittoa koripallossa. Hän tietää kokemuksesta, että hänen mahdollisuutensa tehdä kori on 70 %. Hän suunnittelee harjoittelevansa, kunnes hän on tehnyt 50 koria. Kuinka montaa koria hänen on yritettävä varmistaakseen, että todennäköisyys tehdä ainakin 50 koria on enemmän kuin 0,99?

## invBinomN()

Luettelo > 

$N:n$  suhteen käänteisesti binominen. Johtuen menestyksen todennäköisyydestä kussakin kokeessa (*Prob*), onnistumisten määrä (*NumSuccess*), tämä funktio laskee kokeiden vähimmäismäärän  $N$  siten, että arvo,  $N$ , on vähemmän tai yhtä suuri kuin annettu kumulatiivinen todennäköisyys (*CumulativeProb*).

*OutputForm=0*, näyttää tuloksen asteikkomuodossa (oletus).

*OutputForm=1*, näyttää tuloksen matriisina.

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

## invNorm()

Luettelo > 

invNorm(*Ala* [,  $\mu$  [,  $\sigma$ ]])

Laskee käänteisen kumulatiivisen normaalijakaumafunktion annetulle alalle *Ala*, joka on normaalijakaumakäyrän alapuolella ja jonka määräävät  $\mu$  ja  $\sigma$ .

## invt()

Luettelo > 

invt(*Ala*, *df*)

Laskee käänteisen kumulatiivisen student  $t$ -todennäköisyysfunktion, jonka määräävät vapausaste, *df* ja annettu alue *Ala* käyrän alapuolella.

## iPart()

Luettelo > 

iPart(*Luku*)  $\Rightarrow$  kokonaisluku

iPart(*Listal*)  $\Rightarrow$  lista

iPart(*Matriisi1*)  $\Rightarrow$  matriisi

iPart(-1.234)	-1.
iPart( $\left\{ \frac{3}{2}, -2.3, 7.003 \right\}$ )	{1, -2., 7.}

Laskee argumentin kokonaisosan.

Laskee sarjoille ja matriiseille kunkin elementin kokonaisosan.

Argumentti voi olla reaali- tai kompleksiluku.



**irr()**Luettelo > **irr**(*CF0*,*CFLista* [,*CFFrekv*]) ⇒ *arvo*

Talouseläytötoiminto, joka laskee investoinnin sisäisen korkokannan.

*CF0* on kassavirta alussa aikana 0; arvon on oltava kokonaisluku.

*CFLista* on lista kassavirtamääristä alun kassavirran *CF0* jälkeen.

*CFFrekv* on valinnainen lista, jossa kukin elementti määrää esiintymisfrekvenssin ryhmitetyille (peräkkäiselle) kassavirtamäärälle, joka on *CFFrekv*:n vastaava alkutekijä. Oletusarvo on 1; jos syötät arvoja, niiden on oltava positiivisia kokonaislukuja ja < 10 000.

**Huomaa:** Katso myös **mirr()**, sivu 124.

<i>list1</i> := { 6000, -8000, 2000, -3000 }	{ 6000, -8000, 2000, -3000 }
<i>list2</i> := { 2, 2, 2, 1 }	{ 2, 2, 2, 1 }
<b>irr</b> (5000, <i>list1</i> , <i>list2</i> )	-4.64484

**isPrime()**Luettelo > **isPrime**(*Luku*) ⇒ *Boolean vakiolauseke*

Laskee totuusarvon tosi tai epätosi ilmaistakseen, onko *luku* kokonaisluku ≥ 2, joka on tasan jaollinen vain itsellään ja ykkösellä (1)

Jos *Luku* on pitempi kuin 306 numeroa, eikä siinä ole tekijöitä ≤ 1021, kaava **isPrime**(*Luku*) näyttää virheilmoituksen.

Jos haluat pelkästään määrätä, onko *Luku* jaoton, käytä komentoa **isPrime()** funktion **factor()** sijasta. Se on paljon nopeampi, erityisesti jos *Lukuei* ole jaoton ja sen toiseksi suurimmassa tekijässä on enemmän kuin viisi numeroa.

**Huomaa esimerkkiä syöttäessäsi:** Ohjeet monirivisten ohjelmien ja funktion määrittysten syöttämisestä löytyvät tuotteen ohjekirjan Laskin-osiosta.

<b>isPrime</b> (5)	true
<b>isPrime</b> (6)	false

Funktio, jolla etsitään seuraava jaoton luku määrätyn luvun jälkeen\_

Define <i>nextprim</i> ( <i>n</i> ) = Func	<i>Done</i>
Loop	
<i>n</i> + 1 → <i>n</i>	
If <b>isPrime</b> ( <i>n</i> )	
Return <i>n</i>	
EndLoop	
EndFunc	
<i>nextprim</i> (7)	11

**isVoid()**Luettelo > 

**isVoid**(*Muutt*)  $\Rightarrow$  *Boolean vakiolauseke*  
**isVoid**(*Laus*)  $\Rightarrow$  *Boolean vakiolauseke*  
**isVoid**(*Lista*)  $\Rightarrow$  *lista Boolean vakiolausekkeista*

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

Laskee totuusarvon tosi tai epätosi ilmaisten, onko argumentti tyhjä datatyyppi.

Lisätietoja tyhjistä elementeistä, katso sivu 272.

**L****Lbl**Katalogi > **Lbl** *tunnusnimi*

Määrittää funktion sisällä tunnuksen, jonka nimi on *tunnusnimi*.

Ohjeella **Siirry** *tunnusnimi* voit siirtää ohjauksen kyseistä tunnusta välittömästi seuraavaan ohjaukseen.

*tunnusnimellä* on samat nimeämissäännöt kuin muuttujan nimellä.

**Huomaa esimerkkiä syöttäessäsi:** Ohjeet monirivisten ohjelmien ja funktion määritysten syöttämisestä löytyvät tuotteen ohjekirjan Laskin-osiosta.

Define $g()$ = Func	<i>Done</i>
Local $temp, i$	
$0 \rightarrow temp$	
$1 \rightarrow i$	
Lbl $top$	
$temp + i \rightarrow temp$	
If $i < 10$ Then	
$i + 1 \rightarrow i$	
Goto $top$	
EndIf	
Return $temp$	
EndFunc	
$g()$	55

**lcm()**Katalogi > 

**lcm**(*Luku1*, *Luku2*)  $\Rightarrow$  *lauseke*  
**lcm**(*Lista1*, *Lista2*)  $\Rightarrow$  *lista*  
**lcm**(*Matriisi1*, *Matriisi2*)  $\Rightarrow$  *matriisi*

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

Laskee kahden argumentin pienimmän yhteisen jaettavan. Kahden murtoluvun **lcm** on niiden osoittajien **lcm** jaettuna niiden nimittäjien **gcd**:llä. Murtoluvun liukulukujen **lcm** on niiden tulo.

Kun kyseessä on kaksi listaa tai matriisia, laskee vastaavien elementtien pienimmät yhteiset jakajat.

## left()

**left**(*lähdemerkkijono*[, *Num*])⇒*merkkijono*

left("Hello",2) "He"

Määrittää vasemmanpuoleisimmat *Num*-merkit, jotka sisältyvät merkkijonoon *lähdemerkkijono*.

Jos jätät pois komennon *Num*, määrittää kaiken merkkijonosta *lähdemerkkijono*.

**left**(*Listal*[, *Num*])⇒*lista*

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

Määrittää vasemmanpuoleisimmat *Num*-elementit, jotka sisältyvät listaan *Listal*.

Jos jätät pois komennon *Num*, määrittää kaiken listasta *Listal*.

**left**(*Vertailu*)⇒*lauseke*

left(x<3) x

Laskee yhtälön tai epäyhtälön vasemman puolen.

## libShortcut()

**libShortcut**(*KirjNimiMerkkijono*, *PikavalNimiMerkkijono* [, *KirjYksitLippu*])⇒*muuttujalista*

Luo muuttujaryhmän nykyiseen ongelmaan, joka sisältää viittauksia kaikkiin määritetyn kirjastoasiakirjan *kirjNimiMerkkijono* sisältämiin objekteihin. Lisää myös ryhmän jäsenet Muuttujat-valikkoon. Tällöin voit viitata kuhunkin objektiin käyttäen sen komentoa *PikavalNimimerkkijono*.

Aseta *KirjYksitLippu*=0, kun haluat sulkea pois yksityiset kirjasto-objektit (oletusarvo)

Aseta *KirjYksitLippu*=1, kun haluat sisällyttää yksityiset kirjasto-objektit

Tässä esimerkissä edellytetään asianmukaisesti tallennettua ja päivitettyä kirjastoasiakirjaa nimeltä *linalg2*, joka sisältää objektit *clearmat*, *gauss1* ja *gauss2*.

```
getVarInfo("linalg2")
┌──────────┬──────────┬──────────┬──────────┬──────────┬──────────┐
│ clearmat │ "FUNC"   │ "LibPub"  │           │           │           │
│ gauss1   │ "PRGM"   │ "LibPriv" │           │           │           │
│ gauss2   │ "FUNC"   │ "LibPub"  │           │           │           │
└──────────┴──────────┴──────────┴──────────┴──────────┴──────────┘
libShortcut("linalg2","la")
┌──────────┬──────────┬──────────┬──────────┬──────────┬──────────┐
│ la.clearmat │ la.gauss2 │           │           │           │           │
└──────────┴──────────┴──────────┴──────────┴──────────┴──────────┘
libShortcut("linalg2","la",1)
┌──────────┬──────────┬──────────┬──────────┬──────────┬──────────┐
│ la.clearmat │ la.gauss1 │ la.gauss2 │           │           │           │
└──────────┴──────────┴──────────┴──────────┴──────────┴──────────┘
```

Muuttujaryhmän kopioiminen, katso **CopyVar** sivulla sivu 31.  
Muuttujaryhmän poistaminen, katso **DelVar** sivulla sivu 52.

## limit() tai lim()

**limit**(*LausI*, *Muutt*, *Piste*

[, *Suunta*]) ⇒ *lauseke*

**raja-arvo**(*ListaI*, *Muutt*, *Piste* [, *Suunta*]) ⇒ *lista*

**raja-arvo**(*MatriisiI*, *Muutt*, *Piste* [, *Suunta*]) ⇒ *matriisi*

Laskee pyydetyn raja-arvon.

**Huomaa:** Katso myös **Raja-arvomalli**, sivu 7.

*Suunta*: negatiivinen=vasemmalta, positiivinen=oikealta, muu=molemmat. (Jos ohje jätetään pois, *Suunta* on oletusarvoisesti 'molemmat'.)

Positiivisen  $\infty$ :n ja negatiivisen  $\infty$ :n kohdalla olevat raja-arvot muunnetaan aina äärellisen puolen yksipuolisiksi raja-arvoiksi.

Tilanteesta riippuen **limit()** antaa vastauksena itsensä tai 'undef' silloin, kun se ei pysty määrittämään yksilöllistä raja-arvoa. Tämä ei välttämättä tarkoita, että yksilöllistä raja-arvoa ei ole olemassa. undef tarkoittaa, että vastaus on joko tuntematon luku, jonka suuruus on äärellinen tai ääretön, tai se on tällaisten lukujen koko sarja.

**limit()** käyttää menetelmiä, kuten L'Hopitalin sääntöä, joten on olemassa yksilöllisiä raja-arvoja, joita se ei pysty määrittämään. Jos *LausI* sisältää muita määrittämättömiä muuttujia kuin *Muutt*, joudut mahdollisesti rajoittamaan niitä saadaksesi suppeamman tuloksen.

$\lim_{x \rightarrow 5} (2 \cdot x + 3)$	13
$\lim_{x \rightarrow 0^+} \left( \frac{1}{x} \right)$	$\infty$
$\lim_{x \rightarrow 0} \left( \frac{\sin(x)}{x} \right)$	1
$\lim_{h \rightarrow 0} \left( \frac{\sin(x+h) - \sin(x)}{h} \right)$	$\cos(x)$
$\lim_{n \rightarrow \infty} \left( \left( 1 + \frac{1}{n} \right)^n \right)$	e

$\lim_{x \rightarrow \infty} (a^x)$	undef
$\lim_{x \rightarrow \infty} (a^x)   a > 1$	$\infty$
$\lim_{x \rightarrow \infty} (a^x)   a > 0 \text{ and } a < 1$	0

Raja-arvot voivat olla hyvin herkkiä pyöristysvirheille. Mikäli mahdollista, vältä **Auto or Approximate (Automaattinen tai Likimääräinen)** -tilan Approximate (Likimääräinen) -asetusta ja likiarvoista luvut laskiessasi raja-arvoja. Muussa tapauksessa ne raja-arvot, joiden pitäisi olla nolla tai jotka ovat äärettömiä, eivät todennäköisesti ole tätä, ja raja-arvot, joiden pitäisi olla äärellisiä ei-nolla-arvoja, eivät välttämättä ole sitä.

**LinRegBx**  $X, Y, [Frequ]$ ,  $[Luokka, Sisällytä]$

Laskee lineaarisen regressiony =  $a + b \cdot x$  listoista  $X$  ja  $Y$  frekvenssillä  $Frequ$ . Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 191.)

Kaikkien listojen on oltava samankokoisia *Sisällytä*-listaa lukuunottamatta.

$X$  ja  $Y$  ovat riippumattomien ja riippuvien muuttujien listoja.

*Frequ* on valinnainen frekvenssiarvojen lista. Jokainen *Frequ*:n elementti määrittää kunkin vastaavan datapisteen  $X$  ja  $Y$  esiintymisfrekvenssin. Oletusarvo on 1. Kaikkien elementtien on oltava kokonaislukuja 0.

*Luokka* on luokkakoodien lista vastaavalle  $X$ - ja  $Y$ -datalle..

*Sisällytä* on yhden tai usemman luokkakoodin lista. Vain ne datayksiköt, joiden luokkakoodi sisältyy tähän listaan, ovat mukana laskutoimituksessa.

Lisätietoja listassa olevien tyhjen elementtien vaikutuksesta, katso Tyhjät elementtsivulla sivu 272.

Tulosmuuttuja	Kuvaus
stat.RegEqn	Regressioyhtälö: $a+b \cdot x$
stat.a, stat.b	Regressiokertoimet
stat.r <sup>2</sup>	Määrittyskerroin
stat.r	Korrelaatiokerroin
stat.Resid	Regressioyhtälön jäännökset
stat.XReg	Muokatus <i>Y Lista</i> :n sisältämä datapisteiden lista, jota käytetään regressiossa komentojen <i>Frekv</i> , <i>Luokkalista</i> ja <i>Sisällytä luokat rajoitusten mukaisesti</i>
stat.YReg	Muokatus <i>Y Lista</i> :n sisältämä datapisteiden lista, jota käytetään regressiossa komentojen <i>Frekv</i> , <i>Luokkalista</i> ja <i>Sisällytä luokat rajoitusten mukaisesti</i>
stat.FreqReg	Komentoja <i>stat.XReg</i> ja <i>stat.YReg</i> vastaava frekvenssilista

## LinRegMx

Katalogi > 

**LinRegMx** *X*,*Y*,[*Frekv*],[*Luokka*,*Sisällytä*]]

Laskee lineaarisen regression  $y = m \cdot x + b$  listoista *X* ja *Y* frekvenssillä *Frekv*. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 191.)

Kaikkien listojen on oltava samankokoisia *Sisällytä*-listaa lukuunottamatta.

*X* ja *Y* ovat riippumattomien ja riippuvien muuttujien listoja.

*Frekv* on valinnainen frekvenssiarvojen lista. Jokainen *Frekv*:n elementti määrittää kunkin vastaavan datapisteen *X* ja *Y* esiintymisfrekvenssin. Oletusarvo on 1. Kaikkien elementtien on oltava kokonaislukuja 0.

*Luokka* on luokkakoodien lista vastaavalle *X*- ja *Y*-datalle.

*Sisällytä* on yhden tai usemman luokkakoodin lista. Vain ne datayksiköt, joiden luokkakoodi sisältyy tähän listaan, ovat mukana laskutoimituksessa.

Lisätietoja listassa olevien tyhjen elementtien vaikutuksesta, katso Tyhjät elementtisivulla sivu 272.

Tulosmuuttuja	Kuvaus
stat.RegEqn	Regressioyhtälö: $m \cdot x + b$
stat.m, stat.b	Regressiokertoimet
stat.r <sup>2</sup>	Määrittyskerroin
stat.r	Korrelaatiokerroin
stat.Resid	Regressioyhtälön jäännökset
stat.XReg	Muokatus <i>Y Lista</i> :n sisältämä datapisteiden lista, jota käytetään regressiossa komentojen <i>Freqv</i> , <i>Luokkalista</i> ja <i>Sisällytä luokat rajoitusten mukaisesti</i>
stat.YReg	Muokatus <i>Y Lista</i> :n sisältämä datapisteiden lista, jota käytetään regressiossa komentojen <i>Freqv</i> , <i>Luokkalista</i> ja <i>Sisällytä luokat rajoitusten mukaisesti</i>
stat.FreqReg	Komentoja <i>stat.XReg</i> ja <i>stat.YReg</i> vastaava frekvenssilista

## LinRegtIntervals

**LinRegtIntervals**  $X, Y, F[, 0, CTaso]]]$

Kulmakerroin. Laskee tason C luottamusvälin kulmakertoimelle.

**LinRegtIntervals**  $X, Y, F[, 1, Xarvo[, CTaso]]]$

Vaste. Laskee ennustetun  $y$ :n arvon, tason C ennustevälin yhdelle havainnolle ja tason C luottamusvälin keskiarvovasteelle.

**Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu sivu 191.)**

Kaikkien listojen on oltava samankokoisia.

$X$  ja  $Y$  ovat riippumattomien ja riippuvien muuttujien listoja.

$F$  on valinnainen frekvenssiarvojen lista. Jokainen  $F$ :n elementti määrittää kunkin vastaavan datapisteen  $X$  ja  $Y$  esiintymisfrekvenssin. Oletusarvo on 1. Kaikkien elementtien on oltava kokonaislukuja  $\geq 0$ .

Lisätietoja listassa olevien tyhjiin elementtien vaikutuksesta, katso Tyhjät elementitsivulla sivu 272.

Tulosmuuttuja	Kuvaus
stat.RegEqn	Regressioyhtälö: $a+b \cdot x$
stat.a, stat.b	Regressiokertoimet
stat.df	Vapausasteet
stat.r <sup>2</sup>	Määrittyskerroin
stat.r	Korrelaatiokerroin
stat.Resid	Regressioyhtälön jäännökset

Vain Kulmakerroin-tyyppi

Tulosmuuttuja	Kuvaus
[stat.CLower, stat.CUpper]	Kulmakertoimen luottamusväli
stat.ME	Luottamusvälin virhemarginaali
stat.SESlope	Kulmakertoimen keskivirhe
stat.s	Keskivirhe suoran ympärillä

Vain Vaste-tyyppi

Tulosmuuttuja	Kuvaus
[stat.CLower, stat.CUpper]	Keskivasteen luottamusväli
stat.ME	Luottamusvälin virhemarginaali
stat.SE	Keskivasteen keskivirhe
[stat.LowerPred, stat.UpperPred]	Yhden havainnon ennusteväli
stat.MEPred	Ennustevälin virhemarginaali



Tulosmuuttuja	Kuvaus
stat.SEPred	Ennusteen keskivirhe
stat.ŷ	$a + b \cdot X_{\text{Arvo}}$

## LinRegtTest

Katalogi > 

### LinRegtTest $X, Y, \text{Frekv}[, \text{Hypot}]$

Laskee lineaarisen regression  $X$ - ja  $Y$ -listoista ja suorittaa  $t$ -testin kulmakertoimen  $\beta$  ja korrelaatiokerroimen  $\rho$  arvosta yhtälölle  $y = \alpha + \beta x$ . Testaa nollahypoteesia  $H_0: \beta = 0$  (vastaavasti,  $\rho = 0$ ) johonkin kolmesta vaihtoehdoisesta hypoteesista.

Kaikkien listojen on oltava samankokoisia.

$X$  ja  $Y$  ovat riippumattomien ja riippuvien muuttujien listoja.

*Frekv* on valinnainen frekvenssarvojen lista. Jokainen *Frekv*:n elementti määrittää kunkin vastaavan datapisteen  $X$  ja  $Y$  esiintymisfrekvenssin. Oletusarvo on 1. Kaikkien elementtien on oltava kokonaislukuja 0.

*Hypot* on valinnainen arvo, joka määrittää yhden kolmesta hypoteesista, johon nollahypoteesia ( $H_0: \beta = \rho = 0$ ) testataan.

Kun  $H_a: \beta \neq 0$  ja  $\rho \neq 0$  (oletus), aseta *Hypot*=0  
 Kun  $H_a: \beta < 0$  ja  $\rho < 0$ , aseta *Hypot*<0  
 Kun  $H_a: \beta > 0$  ja  $\rho > 0$ , aseta *Hypot*>0

Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu sivu 191.)

Lisätietoja listassa olevien tyhjen elementtien vaikutuksesta, katso Tyhjät elementitsivulla sivu 272.

Tulosmuuttuja	Kuvaus
stat.RegEqn	Regressioyhtälö: $a + b \cdot x$
stat.t	$t$ -tilasto merkitsevyydestille
stat.PVal	Alin merkitsevyytaso, jolla nollahypoteesi voidaan hylätä

Tulosmuuttuja	Kuvaus
stat.df	Vapausasteet
stat.a, stat.b	Regressiokertoimet
stat.s	Keskivirhe suoran ympärillä
stat.SESlope	Kulmakertoimen keskivirhe
stat.r <sup>2</sup>	Määrittyskerroin
stat.r	Korrelaatiokerroin
stat.Resid	Regressioyhtälön jäännökset

## linSolve()

Katalogi > 

**linSolve**(Lineaariyhtälöryhmä, Muutt1, Muutt2, ...)⇒lista

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

**linSolve**(Lineaariyht1 ja Lineaariyht2 ja ..., Muutt1, Muutt2, ...)⇒lista

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

**linSolve**({Lineaariyht1, Lineaariyht2, ...}, Muutt1, Muutt2, ...)⇒lista

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

**linSolve**(Lineaariyhtälöryhmä, {Muutt1, Muutt2, ...})⇒lista

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

**linSolve**(Lineaariyht1 ja Lineaariyht2 ja ..., {Muutt1, Muutt2, ...})⇒lista

**linSolve**({Lineaariyht1, Lineaariyht2, ...}, {Muutt1, Muutt2, ...})⇒lista

Laskee ratkaisulistan muuttujille *Muutt1*, *Muutt2*, ...

Ensimmäisen argumentin sievennyksen tuloksena on oltava lineaariyhtälöryhmä tai yksi lineaariyhtälö. Muussa tapauksessa esiintyy argumenttinvirhe.

Esimerkiksi yhtälön **linSolve**(x=1 and x=2,x) sieventäminen antaa tuloksena virheilmoituksen Argumenttinvirhe.

## $\Delta$ List()

Katalogi >  $\Delta$ List(Lista1)  $\Rightarrow$  lista $\Delta$ List({20,30,45,70})

{10,15,25}

**Huomaa:** Voit syöttää tämän funktion näppäimistöltä kirjoittamalla **deltaList (...)**.

Määrittää listan, joka sisältää *Listal*:n peräkkäisten elementtien väliset erotukset. Jokainen *Listal*:n elementti vähennetään *Listal*:n seuraavasta elementistä. Tuloksena oleva lista on aina yhden elementin lyhyempi kuin alkuperäinen *Listal*.

## listmat()

Katalogi > listmat(Lista [, elementtiäRivillä])  $\Rightarrow$  matriisi

listmat({1,2,3})

1	2	3
---	---	---

listmat({1,2,3,4,5},2)

1	2
3	4
5	0

Laskee matriisin, joka on täytetty rivi riviltä *Listan* elementeillä.

*elementtiäRivillä*, mikäli sisällytetään, määrittää elementtien määrän rivillä. Oletusarvo on *Listan* elementtien määrä (yksi rivi).

Jos *Listal* ei täytä tulosmatriisia, siihen lisätään nollia.

**Huomaa:** Voit syöttää tämän funktion näppäimistöltä kirjoittamalla **list@>mat (...)**.

## ln

Katalogi > Laus ln  $\Rightarrow$  lauseke $\left(\log_{10}(x)\right) \ln$  $\frac{\ln(x)}{\ln(10)}$ 

Aiheuttaa sen, että lausekkeen *Laus* syöte muunnetaan luonnollisia logaritmeja (ln) sisältäväksi lausekkeeksi.

**Huomaa:** Voit syöttää tämän operaattorin tietokoneen näppäimistöltä kirjoittamalla **@>ln**.

**ln()**

ctrl ex painikkeet

**ln(LausI)**⇒*lauseke* $\ln(2.)$  0.693147**ln(ListaI)**⇒*lista*

Määrittää argumentin luonnollisen logaritmin.

Jos kompleksilukumuodon tila on Reali:

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

Jos kyseessä on lista, määrittää elementtien luonnolliset logaritmit.

Jos kompleksilukumuodon tila on Suorakulma:

 $\ln(\{-3,1.2,5\})$  { $\ln(3)+\pi \cdot i, 0.182322, \ln(5)$ }**ln(neliömatriisiI)**⇒*neliömatriisi*Määrittää *neliömatriisiI*:n matriisin luonnollisen logaritmin. Tämä ei ole sama kuin kunkin elementin luonnollisen logaritmin laskeminen.

Radiaanikulmatilassa ja suorakulmakompleksimuodossa:

Laskentamenetelmä on kuvattu kohdassa **cos()**.
$$\ln \begin{pmatrix} 1 & 5 & 3 \\ 4 & 2 & 1 \\ 6 & -2 & 1 \end{pmatrix}$$
$$\begin{bmatrix} 1.83145+1.73485 \cdot i & 0.009193-1.49086 \\ 0.448761-0.725533 \cdot i & 1.06491+0.623491 \cdot i \\ -0.266891-2.08316 \cdot i & 1.12436+1.79018 \cdot i \end{bmatrix}$$
*neliömatriisiI*:n on oltava diagonalisoitavissa. Vastaus sisältää aina liukulukuja.

Jos haluat nähdä koko vastauksen, paina ▲ ja siirrä sen jälkeen osoitinta painikkeilla ◀ ja ▶.

**LnReg**

Katalogi &gt;

**LnReg** *X*, *Y*, [*Frekv*] [, *Luokka*, *Sisällytä*]Laskee logaritmisin regression  $y = a+b \cdot \ln(x)$  listoista *X* ja *Y* frekvenssillä *Frekv*. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 191.)Kaikkien listojen on oltava samankokoisia *Sisällytä*-listaa lukuunottamatta.*X* ja *Y* ovat riippumattomien ja riippuvien muuttujien listoja.*Frekv* on valinnainen frekvenssiarvojen lista. Jokainen *Frekv*:n elementti määrittää kunkin vastaavan datapisteen *X* ja *Y* esiintymisfrekvenssin. Oletusarvo on 1. Kaikkien elementtien on oltava kokonaislukuja 0.

*Luokka* on luokkakoodien lista vastaavalle  $X$ - ja  $Y$ -datalle.

*Sisällytä* on yhden tai usemman luokkakoodin lista. Vain ne datayksiköt, joiden luokkakoodi sisältyy tähän listaan, ovat mukana laskutoimituksessa.

Lisätietoja listassa olevien tyhjiä elementtien vaikutuksesta, katso Tyhjät elementtisivulla sivu 272.

Tulosmuuttuja	Kuvaus
stat.RegEqn	Regressioyhtälö: $a+b \cdot \ln(x)$
stat.a, stat.b	Regressiokertoimet
stat.r <sup>2</sup>	Muunnettujen tietojen lineaarimäärittelyn kerroin
stat.r	Muunnettujen tietojen korrelaatiokerroin ( $\ln(x)$ , $y$ )
stat.Resid	Logaritmimalliin liittyvät jäännökset
stat.ResidTrans	Muunnettujen tietojen lineaariseen sovitukseen liittyvät jäännökset
stat.XReg	Muokatus <i>Y Lista</i> :n sisältämä datapisteiden lista, jota käytetään regressiossa komentojen <i>Frekv</i> , <i>Luokkalista</i> ja <i>Sisällytä luokat rajoitusten mukaisesti</i>
stat.YReg	Muokatus <i>Y Lista</i> :n sisältämä datapisteiden lista, jota käytetään regressiossa komentojen <i>Frekv</i> , <i>Luokkalista</i> ja <i>Sisällytä luokat rajoitusten mukaisesti</i>
stat.FreqReg	Komentoja <i>stat.XReg</i> ja <i>stat.YReg</i> vastaava frekvenssilista

**Local** *Muutt1*[, *Muutt2*] [, *Muutt3*] ...

Määrittää määritetyt *muuttujat* paikallisina muuttujina. Nämä muuttujat ovat olemassa vain funktion sievennyksen aikana, ja ne poistetaan, kun funktion suoritus päättyy.

**Huomaa:** Paikalliset muuttujat säästävät muistia, koska ne ovat olemassa vain väliaikaisesti. Lisäksi ne eivät häiritse mitään olemassa olevia globaalien muuttujien arvoja. Paikallisia muuttujia on käytettävä **For**-silmukoissa sekä tallennettaessa arvoja väliaikaisesti monirivisessä funktiossa, sillä globaalien muuttujien modifiointeja ei sallita funktiossa.

**Huomaa esimerkkiä syöttäessäsi:** Ohjeet monirivisten ohjelmien ja funktion määritysten syöttämisestä löytyvät tuotteen ohjekirjan Laskin-osiosta.

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

**Lock****Lock***Muutt1*[, *Muutt2*] [, *Muutt3*] ...  
**Lock***Muutt*.

Lukitsee määritetyt muuttujat tai muuttujaryhmän. Lukittuja muuttujia ei voi muokata eikä poistaa.

Et voi lukita tai vapauttaa järjestelmän muuttujaa *Ans*, etkä voi lukita järjestelmän muuttujaryhmiä *stat*. tai *tvM*.

**Huomaa:** Lukitse-komento (**Lock**) tyhjentää toimintojen Kumoa/Tee uudelleen historian, kun sitä käytetään lukitsemattomiin muuttujiin.

Katso **unLock**, sivu 214, ja **getLockInfo()**, sivu 89.

<i>a</i> :=65	65
Lock <i>a</i>	<i>Done</i>
getLockInfo( <i>a</i> )	1
<i>a</i> :=75	"Error: Variable is locked."
DelVar <i>a</i>	"Error: Variable is locked."
Unlock <i>a</i>	<i>Done</i>
<i>a</i> :=75	75
DelVar <i>a</i>	<i>Done</i>

## log()

**ctrl** **10<sup>x</sup>** painikkeet

**log(Laus1[,Laus2])**⇒*lauseke***log(Lista1[,Laus2])**⇒*lista*

Laskee ensimmäisen argumentin kantaluku--*Laus2*:n logaritmin.

**Huomaa:** Katso myös **Logaritmimalli**, sivu 2.

Kun kyseessä on lista, laskee elementtien kantaluku--*Laus2*:n logaritmin.

Jos toinen argumentti jätetään pois, kantalukuna käytetään lukua 10.

**log(neliömatriisi1[,Laus])**⇒*neliömatriisi*

Laskee matriisin kantaluku--*Laus*:n logaritmin *neliömatriisi1*:lle. Tämä ei ole sama kuin kunkin elementin kantaluku--*Laus*-logaritmin laskeminen. Laskentamenetelmä on kuvattu kohdassa **cos()**.

*neliömatriisi1*:n on oltava diagonalisoitavissa. Vastaus sisältää aina liukulukuja.

Jos kantalukuargumentti jätetään pois, kantalukuna käytetään lukua 10.

$\log_{10} (2.)$	0.30103
$\log_4 (2.)$	0.5
$\log_3 (10) - \log_3 (5)$	$\log_3 (2)$

Jos kompleksilukumuodon tila on Reali:

$\log_{10} (\{-3,1.2,5\})$	Error: Non-real result
----------------------------	------------------------

Jos kompleksilukumuodon tila on Suorakulma:

$\log_{10} (\{-3,1.2,5\})$
$\left\{ \log_{10} (3) + 1.36438 \cdot i, 0.079181, \log_{10} (5) \right\}$

Radiaanikulmatilassa ja suorakulmakompleksimuodossa:

$\log_{10} \begin{pmatrix} 1 & 5 & 3 \\ 4 & 2 & 1 \\ 6 & -2 & 1 \end{pmatrix}$
$0.795387 + 0.753438 \cdot i$ 0.003993 - 0.6474i
0.194895 - 0.315095 · i 0.462485 + 0.2707i
-0.115909 - 0.904706 · i 0.488304 + 0.7774i

Jos haluat nähdä koko vastauksen, paina ▲ ja siirrä sen jälkeen osoitinta painikkeilla ◀ ja ▶.

## logbase

**Katalogi >**

*Laus* ▶**logbase(Laus1)**⇒*lauseke*

Aiheuttaa sen, että syötteenä oleva lauseke sievennetään lausekkeeksi, joka käyttää kantaluku--*Laus1*:ä.

**Huomaa:** Voit syöttää tämän operaattorin tietokoneen näppäimistöltä kirjoittamalla @>**logbase (...)** .

$\log_3 (10) - \log_5 (5)$ ▶ <b>logbase(5)</b>	$\log_5 \left( \frac{10}{3} \right)$
	$\log_5 (3)$

**Logistic  $X, Y$ , [*Frekv*] [, *Luokka*, *Sisällytä*]**

Laskee logistisen regressioyhtälön  $y = c/(1+a \cdot e^{-bx})$  listoista  $X$  ja  $Y$  frekvenssillä *Frekv*.

Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 191.)

Kaikkien listojen on oltava samankokoisia *Sisällytä*-listaa lukuunottamatta.

$X$  ja  $Y$  ovat riippumattomien ja riippuvien muuttujien listoja.

*Frekv* on valinnainen frekvenssiarvojen lista. Jokainen *Frekv*:n elementti määrittää kunkin vastaavan datapisteen  $X$  ja  $Y$  esiintymisfrekvenssin. Oletusarvo on 1. Kaikkien elementtien on oltava kokonaislukuja 0.

*Luokka* on luokkakoodien lista vastaavalle  $X$ - ja  $Y$ -datalle.

*Sisällytä* on yhden tai usemman luokkakoodin lista. Vain ne datayksiköt, joiden luokkakoodi sisältyy tähän listaan, ovat mukana laskutoimituksessa.

Lisätietoja listassa olevien tyhjiä elementtien vaikutuksesta, katso Tyhjät elementtsivulla sivu 272.

Tulosmuuttuja	Kuvaus
stat.RegEqn	Regressioyhtälö: $c/(1+a \cdot e^{-bx})$
stat.a, stat.b, stat.c	Regressiokertoimet
stat.Resid	Regressioyhtälön jäännökset
stat.XReg	Muokatun <i>Y Lista</i> :n sisältämä datapisteiden lista, jota käytetään regressiossa komentojen <i>Frekv</i> , <i>Luokkalista</i> ja <i>Sisällytä</i> luokat rajoitusten mukaisesti
stat.YReg	Muokatun <i>Y Lista</i> :n sisältämä datapisteiden lista, jota käytetään regressiossa komentojen <i>Frekv</i> , <i>Luokkalista</i> ja <i>Sisällytä</i> luokat rajoitusten mukaisesti
stat.FreqReg	Komentoja <i>stat.XReg</i> ja <i>stat.YReg</i> vastaava frekvenssilista



**LogisticD**  $X, Y$  [, [*Iteraatiot*] , [*Frekv*] [, *Luokka, Sisällytä*] ]

Laskee logistisen regression  $y = (c/(1+a \cdot e^{-bx})+d)$  listoista  $X$  ja  $Y$  frekvenssillä *Frekv* käyttäen tiettyä *Iteraatioiden* määrää. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 191.)

Kaikkien listojen on oltava samankokoisia *Sisällytä*-listaa lukuunottamatta.

$X$  ja  $Y$  ovat riippumattomien ja riippuvien muuttujien listoja.

*Iteraatiot* on valinnainen arvo, joka määrittää ratkaisun yrityskertojen enimmäismäärän. Mikäli se jätetään pois, käytetään arvoa 64. Suuremmilla arvoilla saadaan tyypillisesti parempi tarkkuus, mutta suoritus aika on pitempi ja päin vastoin.

*Frekv* on valinnainen frekvenssiarvojen lista. Jokainen *Frekv*:n elementti määrittää kunkin vastaavan datapisteen  $X$  ja  $Y$  esiintymisfrekvenssin. Oletusarvo on 1. Kaikkien elementtien on oltava kokonaislukuja 0.

*Luokka* on luokkakoodien lista vastaavalle  $X$ - ja  $Y$ -datalle.

*Sisällytä* on yhden tai usemman luokkakoodin lista. Vain ne datayksiköt, joiden luokkakoodi sisältyy tähän listaan, ovat mukana laskutoimituksessa.

Lisätietoja listassa olevien tyhjien elementtien vaikutuksesta, katso Tyhjät elementtsivulla sivu 272.

Tulosmuuttuja	Kuvaus
stat.RegEqn	Regressioyhtälö: $c/(1+a \cdot e^{-bx})+d$
stat.a, stat.b, stat.c, stat.d	Regressiokertoimet
stat.Resid	Regressioyhtälön jäännökset

Tulosmuuttuja	Kuvaus
stat.XReg	Muokatun <i>X Listan</i> sisältämä datapisteiden lista, jota käytetään todellisesti regressiossa komentojen <i>Frekv</i> , <i>Luokka Lista</i> ja <i>Sisällyttä Luokat rajoitusten mukaisesti</i>
stat.YReg	Muokatun <i>Y Listan</i> sisältämä datapisteiden lista, jota käytetään todellisesti regressiossa komentojen <i>Frekv</i> , <i>Luokka Lista</i> ja <i>Sisällyttä Luokat rajoitusten mukaisesti</i>
stat.FreqReg	Komentoja <i>stat.XReg</i> ja <i>stat.YReg</i> vastaava <i>frekvenssilista</i>

## Loop (Silmukka)

Katalogi > 

### Loop

*Lohko*

### EndLoop

Suorittaa toistuvasti *Lohkon* sisältämät lausekkeet. Huomaa, että silmukkaa suoritetaan loputtomasti, ellei **Goto**- tai **Exit**-ohjetta suoriteta *Lohkon* sisällä.

*Lohko* on lausekkeiden sarja, jotka on erotettu toisistaan kaksoispisteellä (:).

**Huomaa esimerkkiä syöttäessäsi:** Ohjeet monirivisten ohjelmien ja funktion määrittysten syöttämisestä löytyvät tuotteen ohjekirjan Laskin-osiosta.

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

**LU** *Matriisi*, *lMatriisi*, *uMatriisi*, *pMatriisi*[*Tol*]

Laskee Doolittlen LU (ala-ylä)-dekomponoinnin reaali- tai kompleksimatriisista. Alakolmiomatriisi tallentuu muuttujaan *lMatriisi*, yläkolmiomatriisi muuttujaan *uMatriisi* ja permutaatiomatriisi (joka kuvaa laskennan aikana tehdyt rivien vaihdot) muuttujaan *pMatriisi*.

$lMatriisi \cdot uMatriisi = pMatriisi \cdot matriisi$

Valinnaisesti kaikkia matriisielementtejä käsitellään nollana, jos niiden itseisarvo on pienempi kuin *Tol*. Tätä toleranssia käytetään vain, jos matriisissa on liukulukusyötteitä eikä se sisällä symbolisia muuttujia, joille ei ole määritetty arvoa. Muussa tapauksessa *Tol*-komentoa ei huomioida.

- Jos käytät painikkeita **ctrl** **enter** tai **Automaattinen tai likimääräinen** -tilan valintaa Approximate (Likimääräinen), laskut suoritetaan liukulukuaritmetiikalla.
- Jos *Tol* jätetään pois tai sitä ei käytetä, oletusarvoinen toleranssi lasketaan seuraavasti:  
 $5E-14 \cdot \max(\dim(Matriisi)) \cdot \text{riviNorm}(Matriisi)$

**LU**-dekomponointialgoritmi käyttää osittaista rivien vaihtoa (pivoting).

## M

### mat▶list()

**mat▶list**(*Matriisi*) $\Rightarrow$ *lista*

Luo listan, joka on täytetty *Matriisin* elementeillä. Elementit kopioidaan *Matriisista* rivi riviltä.

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

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

<i>lower</i>	$\begin{bmatrix} 1 & 0 & 0 \\ \frac{5}{6} & 1 & 0 \\ \frac{1}{2} & \frac{1}{2} & 1 \end{bmatrix}$
--------------	---

<i>upper</i>	$\begin{bmatrix} 6 & 12 & 18 \\ 0 & 4 & 16 \\ 0 & 0 & 1 \end{bmatrix}$
--------------	--

<i>perm</i>	$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$
-------------	---

$\begin{bmatrix} m & n \\ o & p \end{bmatrix} \rightarrow m1$	$\begin{bmatrix} m & n \\ o & p \end{bmatrix}$
---	--

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

<i>lower</i>	$\begin{bmatrix} 1 & 0 \\ \frac{m}{o} & 1 \end{bmatrix}$
--------------	--

<i>upper</i>	$\begin{bmatrix} o & p \\ 0 & n - \frac{m \cdot p}{o} \end{bmatrix}$
--------------	--

<i>perm</i>	$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$
-------------	--

**mat▶list**( $\{1, 2, 3\}$ )  $\Rightarrow$   $\{1, 2, 3\}$

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

**mat▶list**(*m1*)  $\Rightarrow$   $\{1, 2, 3, 4, 5, 6\}$

**Huomaa:** Voit syöttää tämän funktion tietokoneen näppäimistöltä kirjoittamalla `mat@>list(...)`.

**max()**

**max(Laus1, Laus2)**⇒*lauseke*

$$\max(2.3, 1.4) \quad 2.3$$

**max(Lista1, Lista2)**⇒*lista*

$$\max(\{1, 2\}, \{-4, 3\}) \quad \{1, 3\}$$

**max(Matriisi1, Matriisi2)**⇒*matriisi*

Laskee kahden argumentin maksimiarvon. Jos argumentteina on kaksi listaa tai matriisia, laskee listan tai matriisin, joka sisältää vastaavien elementtien kunkin parin maksimiarvon.

**max(Lista)**⇒*lauseke*

$$\max(\{0, 1, -7, 1.3, 0.5\}) \quad 1.3$$

Laskee *lista*:n maksimielementin.

**max(Matriisi1)**⇒*matriisi*

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

Luo rivivektorin, joka sisältää *Matriisi1*:n jokaisen sarakkeen maksimielementin.

Tyhjiä elementtejä ei huomioida.

Lisätietoja tyhjistä elementeistä, katso sivu 272.

**Huomaa:** Katso myös **fMax()** ja **min()**.

**mean()**

**mean(Lista[, frekvLista])**⇒*lauseke*

$$\text{mean}(\{0.2, 0.1, -0.3, 0.4\}) \quad 0.26$$

Laskee *Listan* sisältämien elementtien keskiarvon.

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

Jokainen *frekvListan* elementti näyttää *Listan* vastaavien elementtien peräkkäisten esiintymien lukumäärän.

**mean(Matriisi[, frekvMatriisi])**⇒*matriisi*

Suorakulmavektorimuodossa:

Luo rivivektorin kaikkien *Matriisi1*:n sarakkeiden keskiarvoista.

**mean()**Katalogi > 

Jokainen *frekvMatriisin* elementti näyttää *Matriisi1*:n vastaavien elementtien peräkkäisten esiintymien lukumäärän.

Tyhjiä elementtejä ei huomioida. Lisätietoja tyhjistä elementeistä, katso sivu 272.

mean	$\begin{pmatrix} 0.2 & 0 \\ -1 & 3 \\ 0.4 & -0.5 \end{pmatrix}$	$[-0.133333 \quad 0.833333]$
mean	$\begin{pmatrix} \frac{1}{5} & 0 \\ -1 & 3 \\ \frac{2}{5} & \frac{-1}{2} \end{pmatrix}$	$\begin{bmatrix} \frac{-2}{15} & \frac{5}{6} \end{bmatrix}$
mean	$\begin{pmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{pmatrix}, \begin{pmatrix} 5 & 3 \\ 4 & 1 \\ 6 & 2 \end{pmatrix}$	$\begin{bmatrix} \frac{47}{15} & \frac{11}{3} \end{bmatrix}$

**median()**Katalogi > 

**median(Lista[, frekvLista])** ⇒ *lauseke*

Laskee *Listan* elementtien mediaanin.

Jokainen *frekvListan* elementti näyttää *Listan* vastaavien elementtien peräkkäisten esiintymien lukumäärän.

**median(MatriisiI[, frekvMatriisi])** ⇒ *matriisi*

Luo rivivektorin, joka sisältää *Matriisi1*:n sarakkeiden mediaanit.

Jokainen *frekvMatriisin* elementti näyttää *Matriisi1*:n vastaavien elementtien peräkkäisten esiintymien lukumäärän.

**Huomaa:**

- Kaikkien listan tai matriisin syötteiden tulee sieventyä luvuiksi.
- Listassa tai matriisissa olevia tyhjiä elementtejä ei huomioida. Lisätietoja tyhjistä elementeistä, katso sivu 272.

median	$\{0.2, 0, 1, -0.3, 0.4\}$	0.2
median	$\begin{pmatrix} 0.2 & 0 \\ 1 & -0.3 \\ 0.4 & -0.5 \end{pmatrix}$	$[0.4 \quad -0.3]$

**MedMed**Katalogi > 

**MedMed X,Y [, Frekv] [, Luokka, Sisällytä]**

Laskee mediaani-mediaani-suorany =  $(m \cdot x + b)$  listoista  $X$  ja  $Y$  frekvenssillä *Frekv.* Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 191.)

Kaikkien listojen on oltava samankokoisia *Sisällytä*-listaa lukuunottamatta.

$X$  ja  $Y$  ovat riippumattomien ja riippuvien muuttujien listoja.

*Frekv* on valinnainen frekvenssiarvojen lista. Jokainen *Frekv*:n elementti määrittää kunkin vastaavan datapisteen  $X$  ja  $Y$  esiintymisfrekvenssin. Oletusarvo on 1. Kaikkien elementtien on oltava kokonaislukuja 0.

*Luokka* on luokkakoodien lista vastaavalle  $X$ - ja  $Y$ -datalle.

*Sisällytä* on yhden tai usemman luokkakoodin lista. Vain ne datayksiköt, joiden luokkakoodi sisältyy tähän listaan, ovat mukana laskutoimituksessa.

Lisätietoja listassa olevien tyhjiä elementtien vaikutuksesta, katso Tyhjät elementtsivulla sivu 272.

Tulosmuuttuja	Kuvaus
stat.RegEqn	Mediaani-mediaani-suoran yhtälö: $m \cdot x + b$
stat.m, stat.b	Mallin kertoimet
stat.Resid	Mediaani-mediaani-suoran jäännökset
stat.XReg	Muokatus <i>Y Lista</i> :n sisältämä datapisteiden lista, jota käytetään regressiossa komentojen <i>Frekv</i> , <i>Luokkalista</i> ja <i>Sisällytä luokat rajoitusten mukaisesti</i>
stat.YReg	Muokatus <i>Y Lista</i> :n sisältämä datapisteiden lista, jota käytetään regressiossa komentojen <i>Frekv</i> , <i>Luokkalista</i> ja <i>Sisällytä luokat rajoitusten mukaisesti</i>
stat.FreqReg	Komentoja <i>stat.XReg</i> ja <i>stat.YReg</i> vastaava frekvenssilista

**mid()**Katalogi > **mid(lähdemerkkijono, Alku[, Count])** ⇒ merkkijono

Laskee *Count*:n merkit merkkijonosta *lähdemerkkijono* aloittaen merkistä numero *Alku*.

Jos *Count* jätetään pois, tai jos se on suurempi kuin *lähdemerkkijono*, laskee kaikki merkit *lähdemerkkijonosta* aloittaen merkistä numero *Alku*.

*Count*:n on oltava  $\geq 0$ . Jos *Count* = 0, antaa vastauksena tyhjän merkkijonon.

**mid(lähdelista, Alku [, Count])** ⇒ lista

Laskee *Count*:n elementit *lähdelistasta* aloittaen elementistä numero *Alku*.

Jos *Count* jätetään pois, tai jos se on suurempi kuin *lähdelista*, laskee kaikki elementit *lähdelistasta* aloittaen elementistä numero *Alku*.

*Count*:n on oltava  $\geq 0$ . Jos *Count* = 0, antaa vastauksena tyhjän listan.

**mid(lähdeMerkkijonoLista, Alku[, Count])** ⇒ lista

Laskee *Count*:n merkkijonot merkkijonolistasta *lähdeMerkkijonoLista* aloittaen elementistä numero *Alku*.

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

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

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

**min()**Katalogi > **min(Laus1, Laus2)** ⇒ lauseke**min(Lista1, Lista2)** ⇒ lista**min(Matriisi1, Matriisi2)** ⇒ matriisi

Laskee kahden argumentin minimiarvon. Jos argumentteina on kaksi listaa tai matriisia, laskee listan tai matriisin, joka sisältää vastaavien elementtien kunkin parin minimiarvon.

**min(Lista)** ⇒ lauseke

Laskee *Listan* minimielementin.

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

min({0,1,-7,1.3,0.5})	-7
-----------------------	----

**min()**Katalogi > **min(MatriisiI)**⇒*matriisi*

$$\min \begin{pmatrix} 1 & -3 & 7 \\ -4 & 0 & 0.3 \end{pmatrix} \quad [-4 \quad -3 \quad 0.3]$$

Luo rivivektorin, joka sisältää *MatriisiI*:n jokaisen sarakkeen minimielementin.

**Huomaa:** Katso myös **fMin()** ja **max()**.

**mirr()**Katalogi > **mirr**

(*tal.arvo, uud.invest.arvo, CF0, CFLista*  
[, *CFFrekv*])

$$\begin{aligned} list1 &:= \{6000, -8000, 2000, -3000\} \\ &\quad \{6000, -8000, 2000, -3000\} \\ list2 &:= \{2, 2, 2, 1\} && \{2, 2, 2, 1\} \\ \min(4.65, 12, 5000, list1, list2) & \quad 13.41608607 \end{aligned}$$

Talouselaskentatoiminto, joka laskee investoinnin modifioidun sisäisen korkokannan.

*tal.arvo* on kassavirtamääristä maksettava korkoprosentti.

*uud.invest.arvo* on korkokanta, jolla kassavirrat investoidaan uudelleen.

*CF0* on alkuperäinen kassavirta aikana 0; arvon on oltava kokonaisluku.

*CFLista* on lista kassavirtamääristä ensimmäisen kassavirran *CF0* jälkeen.

*CFFrekv* on valinnainen lista, jossa jokainen elementti määrittää esiintymisfrekvenssin ryhmitetyille (peräkkäiselle) kassavirtamäärälle, joka on *CFListan* vastaava elementti. Oletusarvo on 1; jos syötät arvoja, niiden on oltava positiivisia kokonaislukuja < 10,000.

**Huomaa:** Katso myös **irr()**, sivu 101.



**mod()**

Katalogi &gt;

**mod**(*Laus1*, *Laus2*) ⇒ *lauseke*

mod(7,0)	7
----------	---

**mod**(*Lista1*, *Lista2*) ⇒ *lista*

mod(7,3)	1
----------	---

**mod**(*Matriisi1*, *Matriisi2*) ⇒ *matriisi*

mod(-7,3)	2
-----------	---

Laskee ensimmäisen argumentin modulo toinen argumentti identtisten yhtälöiden määrittelyn mukaisesti:

mod(7,-3)	-2
-----------	----

$$\text{mod}(x,0) = x$$

$$\text{mod}(x,y) = x - y \text{ floor}(x/y)$$

Kun toinen argumentti on ei-nolla, vastaus on periodinen tässä argumentissa. Vastaus on joko nolla tai samanmerkinen kuin toinen argumentti.

mod(-7,-3)	-1
------------	----

Jos argumentteina on kaksi listaa tai matriisia, laskee listan tai matriisin, joka sisältää vastaavien elementtien kunkin parin modulon (jakojäännöksen).

mod({12,-14,16},{9,7,-5})	{3,0,-4}
---------------------------	----------

**Huomaa:** Katso myös **remain()**, sivu 161

**mRow()**

Katalogi &gt;

**mRow**(*Laus*, *Matriisi1*, *Indeksi*) ⇒ *matriisi*

mRow( $\begin{pmatrix} -1 \\ 3 \end{pmatrix}, \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}, 2)$	$\begin{bmatrix} 1 & 2 \\ -1 & -4 \\ & 3 \end{bmatrix}$
---	---

Luo kopion *Matriisi1*:stä, jossa jokainen rivin *Indeksi* elementti *Matriisi1*:ssä on kerrottu lausekkeella *Laus*.

**mRowAdd()**

Katalogi &gt;

**mRowAdd**(*Laus*, *Matriisi1*, *Indeksi1*, *Indeksi2*) ⇒ *matriisi*

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

Luo kopion *Matriisi1*:stä, jossa jokainen rivin *Indeksi2* elementti *Matriisi1*:ssä on korvattu seuraavasti:

mRowAdd( $n, \begin{bmatrix} a & b \\ c & d \end{bmatrix}, 1, 2)$	$\begin{bmatrix} a & b \\ a \cdot n + c & b \cdot n + d \end{bmatrix}$
---	--

*Laus* · rivi *Indeksi1* + rivi *Indeksi2*

**MultReg**

Katalogi &gt;

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

Laskee listan  $Y$  moninkertaisen lineaarisen regression listojen  $X1, X2, \dots, X10$  suhteen. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 191.)

Kaikkien listojen on oltava samankokoisia.

Lisätietoja listassa olevien tyhjiin elementtien vaikutuksesta, katso Tyhjät elementitsivulla sivu 272.

Tulosmuuttuja	Kuvaus
stat.RegEqn	Regressioyhtälö: $b_0 + b_1 \cdot x_1 + b_2 \cdot x_2 + \dots$
stat.b0, stat.b1, ...	Regressiokertoimet
stat.R <sup>2</sup>	Moninkertaisen määrittelyn kerroin
stat.ŷLista	$\hat{y}$ Lista = $b_0 + b_1 \cdot x_1 + \dots$
stat.Resid	Regressioyhtälön jäännökset

## MultRegIntervals

**MultRegIntervals**  $Y, X1[,X2[,X3,...$   
 $[,X10]]], XArvoLista[,CTaso]$

Laskee ennustetun  $y$ :n arvon, tason  $C$  ennustevälin yhdelle havainnolle ja tason  $C$  luottamusvälin keskiarvovasteelle.

Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 191.)

Kaikkien listojen on oltava samankokoisia.

Lisätietoja listassa olevien tyhjiin elementtien vaikutuksesta, katso Tyhjät elementitsivulla sivu 272.

Tulosmuuttuja	Kuvaus
stat.RegEqn	Regressioyhtälö: $b_0 + b_1 \cdot x_1 + b_2 \cdot x_2 + \dots$
stat.ŷ	Pisteen $A$ arvio: $\hat{y} = b_0 + b_1 \cdot x_1 + \dots$ <i>XArvoListalle</i>
stat.dfError	Virheen vapausasteet
stat.CLower, stat.CUpper	Keskiarvovasteen luottamusväli

Tulosmuuttuja	Kuvaus
stat.ME	Luottamusvälin virhemarginaali
stat.SE	Keskiarvovasteen keskivirhe
stat.LowerPred, stat.UpperrPred	Yhden havainnon ennusteväli
stat.MEPred	Ennustevälin virhemarginaali
stat.SEPred	Ennusteen keskivirhe
stat.bList	Regressiokertoimien lista, {b0,b1,b3,...}
stat.Resid	Regressioyhtälön jäännökset

## MultRegTests

Katalogi > 

### MultRegTests $Y, X1[,X2[,X3,...[,X10]]]$

Moninkertaisen lineaarisen regression testi laskee lineaarisen regression tietystä datasta ja määrittää kertoimille globaalin  $F$ -testin tilastot sekä  $t$ -testin tilastot.

Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu sivu 191.)

Lisätietoja listassa olevien tyhjen elementtien vaikutuksesta, katso Tyhjät elementitsivulla sivu 272.

### Tulokset

Tulosmuuttuja	Kuvaus
stat.RegEqn	Regressioyhtälö: $b_0+b_1 \cdot x_1+b_2 \cdot x_2+ \dots$
stat.F	Globaalien $F$ -testin tilasto
stat.PVal	Globaaliin $F$ -tilastoon liittyvä P-arvo
stat.R <sup>2</sup>	Moninkertaisen määrittämisen kerroin
stat.AdjR <sup>2</sup>	Moninkertaisen määrittämisen säädetty kerroin
stat.s	Virheen keskijajonta
stat.DW	Durbin-Watsonin tilasto; käytetään määrittäessä, esiintyykö mallissa ensimmäisen asteen automaattista korrelaatiota
stat.dfReg	Regression vapausasteet

Tulosmuuttuja	Kuvaus
stat.SSReg	Regression neliöiden summa
stat.MSReg	Regression keskineliö
stat.dfError	Virheen vapausasteet
stat.SSError	Virheen neliöiden summa
stat.MSError	Virheen keskineliö
stat.bList	{b0,b1,...} Kertoimien lista
stat.tList	t-tilastojen lista, yksi kullekin bListan sisältämälle kertoimelle
stat.PList	P-arvojen lista kullekin t-tilastolle
stat.SEList	Keskivirheiden lista bListan sisältämille kertoimille
stat.yList	$\hat{y}$ Lista = $b_0 + b_1 \cdot x_1 + \dots$
stat.Resid	Regressioyhtälön jäännökset
stat.sResid	Standardoidut jäännökset; saadaan jakamalla jäännös keskihajonnalla
stat.CookDist	Cookin etäisyys; jäännökseen ja tuottosuhteeseen perustuvan havainnon vaikutus
stat.Leverage	Miten kaukana riippumattoman muuttujan arvot ovat niiden keskiarvoista

## N

### nand (ei-ja)

  näppäimet

*BooleanLaus1 nand BooleanLaus2* antaa vastauksena *Boolean*

*lausekkeen BooleanList1*

*nand BooleanList2* antaa vastauksena

*Boolean listan BooleanMatriisi1*

*nand BooleanMatriisi2* antaa

vastauksena *Boolean matriisin*

$$\frac{x \geq 3 \text{ and } x \geq 4}{x \geq 4}$$

$$\frac{x \geq 3 \text{ nand } x \geq 4}{x < 4}$$

Antaa vastauksena loogisen **and** operaation negaation kahdesta argumentista. Antaa vastauksena totuusarvon tosi, epätosi tai yhtälön sievennetyn muodon.

Listojen ja matriisien kohdalla vastauksena on vertailuja elementti elementiltä.

*Kokonaisluku1 nand*

*Kokonaisluku2* ⇒ *kokonaisluku*

Vertaa kahta reaalikokonaislukua bitti bitiltä **nand**-operaation avulla. Sisäisesti kumpikin kokonaisluku muunnetaan etumerkilliseksi, 64 bitin binaariluvuksi. Kun vastaavia bittejä verrataan, tulos on 0, jos kumpikin bitti on 1. Muussa tapauksessa tulos on 1. Laskettu arvo edustaa bittituloksia ja se näkyy kantalukutilan mukaisesti.

Kokonaisluvut voi syöttää minkä tahansa luvun kantalukuna. Binaarisen syötteen edelle tulee merkitä etumerkki 0b ja heksadesimaalisen syötteen edelle 0h. Jos etumerkkiä ei ole, kokonaislukuja käsitellään desimaalilukuina (kantaluku 10).

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

## nCr()

Katalogi >

*nCr(Laus1, Laus2)* ⇒ *lauseke*

Kokonaisluvulle *Laus1* ja *Laus2* sekä  $Laus1 \geq Laus2 \geq 0$ , **nCr()** on *Laus1*:n asioiden kombinaatioiden lukumäärä, joita otetaan *Laus2*:n verran kerrallaan. (Tästä käytetään myös nimitystä binomikerroin.) Kumpikin argumentti voi olla kokonaisluku tai symbolinen lauseke.

$nCr(z,3)$	$\frac{z \cdot (z-2) \cdot (z-1)}{6}$
$Ans z=5$	10
$nCr(z,c)$	$\frac{z!}{c! \cdot (z-c)!}$
$\frac{Ans}{nPr(z,c)}$	$\frac{1}{c!}$

*nCr(Laus, 0)* ⇒ 1

*nCr(Laus, negKokonaisluku)* ⇒ 0

*nCr(Laus, posKokonaisluku)* ⇒ *Laus* · (*Laus*-1) ... (*Laus*-*posKokonaisluku*+1) / *posKokonaisluku*!

*nCr(Laus, eiKokonaisluku)* ⇒ *lauseke*!  
 ((  
*Laus*  
 -*eiKokonaisluku*)! · *eiKokonaisluku*!)

*nCr(Lista1, Lista2)* ⇒ *lista*

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

## nCr()

Katalogi > 

Laskee listan kombinaatioista kahden listan sisältämiin vastaaviin elementtipareihin perustuen. Argumenttien on oltava samankokoisia listoja.

**nCr**(*Matriisi1*, *Matriisi2*) $\Rightarrow$ *matriisi*

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

Laskee matriisiin kombinaatioista kahden matriisin sisältämiin vastaaviin elementtipareihin perustuen. Argumenttien on oltava samankokoisia matriiseja.

## nDerivative()

Katalogi > 

**nDerivative**(*Laus1*, *Muutt*=*Arvo* [, *Aste*]) $\Rightarrow$ *arvo*

$$\text{nDerivative}(|x|, x=1) \quad 1$$

**nDerivative**(*Laus1*, *Muutt*[, *Aste*]) | *Muutt*=*Arvo* $\Rightarrow$ *arvo*

$$\text{nDerivative}(|x|, x)|_{x=0} \quad \text{undef}$$

$$\text{nDerivative}(\sqrt{x-1}, x)|_{x=1} \quad \text{undef}$$

Laskee numeerisen derivaatan käyttäen automaattisia derivointimenetelmiä.

Kun *Arvo* määritetään, se ohittaa mahdolliset aikaisemmat muuttujamäärytykset tai mahdolliset muuttujan nykyiset "|" -sijoitukset.

*Derivaatan Asteen* on oltava **1** tai **2**.

## newList()

Katalogi > 

**newList**(*numElementit*) $\Rightarrow$ *lista*

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

Antaa tuloksena listan, jonka koko on *numElementit*. Jokainen elementti on nolla.

## newMat()

Katalogi > 

**newMat**(*numRivit*, *numSarakkeet*) $\Rightarrow$ *matriisi*

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

Antaa tuloksena nollamatriisin, jonka koko on *numRivit* ja *numSarakkeet*.

**nfMax()**Katalogi > 

**nfMax(Laus, Muutt)**⇒arvo  
**nfMax(Laus, Muutt, alaraja)**⇒arvo  
**nfMax(Laus, Muutt, alaraja, yläraja)**⇒arvo  
**nfMax(Laus, Muutt) alaraja≤Muutt**  
 ≤yläraja⇒arvo

$\text{nfMax}(x^2 - 2 \cdot x - 1, x)$	-1.
$\text{nfMax}(0.5 \cdot x^3 - x - 2, x, -5, 5)$	5.

Laskee muuttujan *Muutt* numeerisen ehdotusarvon, jossa lausekkeen *Laus* paikallinen maksimi esiintyy.

Jos ilmoitat *alarajan* ja *ylärajan*, funktio etsii suljetulta väliltä [*alaraja*,*yläraja*] paikallista maksimia.

**Huomaa:** Katso myös **fMax()** ja **d()**.

**nfMin()**Katalogi > 

**nfMin(Laus, Muutt)**⇒arvo  
**nfMin(Laus, Muutt, alaraja)**⇒arvo  
**nfMin(Laus, Muutt, alaraja, yläraja)**⇒arvo  
**nfMin(Laus, Muutt) alaraja≤Muutt**  
 ≤yläraja⇒arvo

$\text{nfMin}(x^2 + 2 \cdot x + 5, x)$	-1.
$\text{nfMin}(0.5 \cdot x^3 - x - 2, x, -5, 5)$	-5.

Laskee muuttujan *Muutt* numeerisen ehdotusarvon, jossa lausekkeen *Laus* paikallinen minimi esiintyy.

Jos ilmoitat *alarajan* ja *ylärajan*, funktio etsii suljetulta väliltä [*alaraja*,*yläraja*] paikallista minimia.

**Huomaa:** Katso myös **fMin()** ja **d()**.

**nlnt()**Katalogi > 

**nlnt(Laus1, Muutt, Ala, Ylä)**⇒lauseke

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

Jos integroitava funktio *LausI* ei sisällä muita muuttujia kuin *Muutt*, ja jos *Ala* ja *Ylä* ovat vakioita, positiivinen  $\infty$  tai negatiivinen  $-\infty$ , tällöin **nInt()** laskee likiarvon lausekkeesta  $\int(LausI, Var, Ala, Ylä)$ . Tämä likiarvo on integrandin välillä  $Ala < Muutt < Ylä$  olevien joidenkin otosarvojen painotettu keskiarvo.

Tavoitteena on kuusi merkitsevää numeroa. Adaptiivinen algoritmi päättyy, kun näyttää todennäköiseltä, että tavoite on saavutettu, tai kun näyttää epätodennäköiseltä, että lisäotokset tuottaisivat merkittävää parannusta.

Näkyviin tulee viesti (Questionable accuracy (Kyseenalainen tarkkuus)), kun näyttää siltä, että tavoitetta ei ole saavutettu.

Sijoita **nInt()**-komentoa sisäkkäin, jos haluat suorittaa moninkertaisen numeerisen integroinnin. Integroinnin raja-arvot voivat riippua niiden ulkopuolella olevista integrointimuuttujista.

**Huomaa:** Katso myös **j()**, sivu 228.

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

$$\int_{\pi}^{\pi+10^{-12}} \cos(x) dx \quad -\sin\left(\frac{1}{100000000000}\right)$$

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

## nom()

**nom(efektiivinenKorko, CpY) ⇒ arvo**

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

Talouseläintaloustoiminto, joka muuntaa efektiivisen vuosikoron *efektiivinenKorko* nimelliskoroksi, kun *CpY* määritetään korkojaksojen lukumääräksi vuodessa.

*efektiivinenKorko* on oltava reaaliluku, ja *CpY*:n on oltava reaaliluku  $> 0$ .

**Huomaa:** Katso myös **eff()**, sivu 62.



*BooleanLaus1* **nor** *BooleanLaus2* antaa vastauksena *Boolean lausekkeen BooleanList1*  
**nor** *BooleanList2* antaa vastauksena *Boolean listan BooleanMatriisi1*  
**nor** *BooleanMatriisi2* antaa vastauksena *Boolean matriisin*

$x \geq 3$ or $x \geq 4$	$x \geq 3$
$x \geq 3$ nor $x \geq 4$	$x < 3$

Antaa vastauksena loogisen **or** operaation negaation kahdesta argumentista. Antaa vastauksena totuusarvon tosi, epätosi tai yhtälön sievennetyn muodon.

Listojen ja matriisien kohdalla vastauksena on vertailuja elementti elementiltä.

*Kokonaisluku1*  
**nor** *Kokonaisluku2*  $\Rightarrow$  *kokonaisluku*

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

Vertaa kahta reaalikokonaislukua bitti bitiltä **nor**-operaation avulla. Sisäisesti kumpikin kokonaisluku muunnetaan etumerkilliseksi, 64 bitin binaariluvuksi. Kun vastaavia bittejä verrataan, tulos on 1, jos kumpikin bitti on 1. Muussa tapauksessa tulos on 0. Laskettu arvo edustaa bittituloksia ja se näkyy kantelukutilan mukaisesti.

Kokonaisluvut voi syöttää minkä tahansa luvun kantelukuna. Binaarisen syötteen edelle tulee merkitä etumerkki 0b ja heksadesimaalisen syötteen edelle 0h. Jos etumerkkiä ei ole, kokonaislukuja käsitellään desimaalilukuina (kantaluku 10).

**norm()**

Katalogi &gt;

**norm(Matriisi)**⇒*lauseke*

$\text{norm}\begin{pmatrix} a & b \\ c & d \end{pmatrix}$	$\sqrt{a^2+b^2+c^2+d^2}$
---	--------------------------

**norm(Vektori)**⇒*lauseke*

$\text{norm}\begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}$	$\sqrt{30}$
---	-------------

Laskee Frobeniusin normin.

$\text{norm}\begin{pmatrix} 1 & 2 \end{pmatrix}$	$\sqrt{5}$
--	------------

$\text{norm}\begin{pmatrix} 1 \\ 2 \end{pmatrix}$	$\sqrt{5}$
---	------------

**normalLine()**

Katalogi &gt;

**normalLine***(Lausl,Muutt,Piste)*⇒*lauseke*

$\text{normalLine}(x^2,x,1)$	$\frac{3}{2} \frac{x}{2}$
------------------------------	---------------------------

**normalLine***(Lausl,Muutt=Piste)*⇒*lauseke*

$\text{normalLine}((x-3)^2-4,x,3)$	$x=3$
------------------------------------	-------

Määrittää normaalisuoran *Lausl*:n kuvaamasta käyrästä pisteessä, joka on määritetty kohtaan *Muutt=Piste*.

$\text{normalLine}\left(\frac{1}{x^3},x=0\right)$	0
---	---

$\text{normalLine}(\sqrt{ x },x=0)$	undef
-------------------------------------	-------

Varmista, että riippumatonta muuttujaa ei ole määritetty. Esimerkiksi, jos  $f_1(x):=5$  ja  $x:=3$ , tällöin **normalLine**( $f_1(x),x,2$ ) antaa vastauksen "epätosi".

**normCdf()**

Katalogi &gt;

**normCdf(alaraja,yläraja[,μ[,σ]])**⇒*luku*, jos *yläraja* ja *alaraja* ovat lukuja, *lista*, jos *alaraja* ja *yläraja* ovat listoja

Laskee normaalijakauman todennäköisyyden *alarajan* ja *ylärajan* välillä määritetyille  $\mu$ :lle (oletus=0) ja  $\sigma$ :lle (oletus=1).

Kun  $P(X \leq \text{yläraja})$ , aseta *alaraja* =  $-\infty$ .

**normPdf()**

Katalogi &gt;

**normPdf(XArvo[,μ[,σ]])**⇒*luku*, jos *XArvo* on luku, *lista*, jos *XArvo* on lista

Laskee normaalijakauman pistetodennäköisyysfunktion määrittelyssä  $X$ Arvossa määritetyille  $\mu$ :lle ja  $\sigma$ :lle.

## not (ei)

**not** *BooleanLaus*  $\Rightarrow$  *Boolean lauseke*

Määrittää totuusarvon tosi, epätosi tai argumentin sievennetyn muodon.

**not** *Kokonaisluku1*  $\Rightarrow$  *kokonaisluku*

Laskee reaalikokonaisluvun ykkösen komplementin. Sisäisesti *Kokonaisluku1* muunnetaan etumerkilliseksi, 64-bittiseksi binaariluvuksi. Jokaisen bitin arvo vaihtuu (0:sta tulee 1 ja päin vastoin) ykkösen komplementille. Tulokset näytetään kantalukutilan mukaisesti.

Voit syöttää kokonaisluvun minkä tahansa luvun kantalukuna. Binaarisen syötteen edelle tulee merkitä etumerkki 0b ja heksadesimaalisen syötteen edelle 0h. Ilman etuliitettä kokonaislukua käsitellään desimaalilukuna (kantaluku 10).

Jos syötät desimaalikokonaisluvun, joka on etumerkillisen, 64 bitin binaarimuodon lukualueen ulkopuolella, laskin käyttää symmetristä modulo-operaatiota, jotta arvo saadaan oikealle alueelle. Lisätietoja, katso ►**Base2**, sivu 18.

not( $2 \geq 3$ )	true
not( $x < 2$ )	$x \geq 2$
not not <i>innocent</i>	<i>innocent</i>

Heksadesimaalisessa kantalukutilassa:

not 0h7AC36	0hFFFFFFFF853C9
-------------	-----------------

Binaarisessa kantalukutilassa:

0b100101 ►Base10	37
not 0b100101	
0b11111111111111111111111111111111 ►	
not 0b100101 ►Base10	-38

Jos haluat nähdä koko vastauksen, paina ▲ ja siirrä sen jälkeen osoitinta painikkeilla ◀ ja ▶.

**Huomaa:** Binaarisessa syötteessä voi olla korkeintaan 64 numeroa (etuliitettä 0b ei lasketa). Heksadesimaalisessa syötteessä voi olla korkeintaan 16 numeroa.

**nPr()**

Katalogi &gt;

**nPr(Laus1, Laus2) ⇒ lauseke**

Kokonaisluvulle *Laus1* ja *Laus2* sekä  $Laus1 \geq Laus2 \geq 0$ , **nPr()** on *Laus1*:n asioiden permutaatioiden lukumäärä, joita otetaan *Laus2*:n verran kerrallaan. Kumpikin argumentti voi olla kokonaisluku tai symbolinen lauseke.

nPr(z,3)	$z \cdot (z-2) \cdot (z-1)$
Ans z=5	60
nPr(z,-3)	$\frac{1}{(z+1) \cdot (z+2) \cdot (z+3)}$
nPr(z,c)	$\frac{z!}{(z-c)!}$
Ans·nPr(z-c,-c)	1

**nPr(Laus, 0) ⇒ 1**

**nPr(Laus, negKokonaisluku) ⇒ 1 / ((Laus+1) · (Laus+2) ... (lauseke - negKokonaisluku))**

**nPr(Laus, posKokonaisluku) ⇒ Laus · (Laus-1) ... (Laus-posKokonaisluku+1)**

**nPr(Laus, eiKokonaisluku) ⇒ Laus! / (Laus-eiKokonaisluku)!**

**nPr(Lista1, List2) ⇒ lista**

Luo listan permutaatioista kahden listan sisältämiin vastaaviin elementtipareihin perustuen. Argumenttien on oltava samankokoisia listoja.

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

**nPr(Matriisi1, Matriisi2) ⇒ matriisi**

Luo matriisin permutaatioista kahden matriisin sisältämiin vastaaviin elementtipareihin perustuen. Argumenttien on oltava samankokoisia matriiseja.

nPr( $\begin{bmatrix} 6 & 5 \\ 4 & 3 \end{bmatrix}, \begin{bmatrix} 2 & 2 \\ 2 & 2 \end{bmatrix}$ )	$\begin{bmatrix} 30 & 20 \\ 12 & 6 \end{bmatrix}$
---	---

**npv()**

Katalogi &gt;

**npv(Korkoprosentti, CFO, CFLista [CFFrekvi])**

Talouseläintaloudet, joka laskee nettonykyarvon; tulevien ja poistuvien kassavirtojen nykyisten arvojen summan. Jos npv:n tulos on positiivinen, investointi on kannattava.

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

*Korkoprosentti* on prosentti, jolla kassavirtoja (rahan kustannusta) vähennetään yhden jakson aikana.

*CF0* on alkuperäinen kassavirta aikana 0; arvon on oltava kokonaisluku.

*CFList*a on lista kassavirtamääristä ensimmäisen kassavirran *CF0* jälkeen.

*CFFrekv* on lista, jossa jokainen elementti määrittää esiintymisfrekvenssin ryhmitetyille (peräkkäiselle) kassavirtamäärälle, joka on *CFList*an vastaava elementti. Oletusarvo on 1; jos syötät arvoja, niiden on oltava positiivisia kokonaislukuja < 10,000.

## nSolve()

**nSolve**(*Yhtälö*,*Muutt*[=*Arvaus*]) $\Rightarrow$ luku tai virhe\_merkkijono

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

**nSolve**(*Yhtälö*,*Muutt*[=*Arvaus*],*alaraja*) $\Rightarrow$ luku tai virhe\_merkkijono

$\text{nSolve}(x^2=4,x=-1)$	-2.
-----------------------------	-----

$\text{nSolve}(x^2=4,x=1)$	2.
----------------------------	----

**nSolve**(*Yhtälö*,*Muutt*[=*Arvaus*],*alaraja*,*yläraja*) $\Rightarrow$ luku tai virhe\_merkkijono

**Huomaa:** Jos ratkaisuja on useita, voit yrittää löytää tietyn ratkaisun käyttämällä apuna arvausta.

**nSolve**(*Yhtälö*,*Muutt*[=*Arvaus*]) | *alaraja* $\leq$ *Muutt* $\leq$ *yläraja* $\Rightarrow$ luku tai virhe\_merkkijono

Etsii iteratiivisesti yhtä likimääräistä numeerista ratkaisua *Yhtälön* yhdelle muuttujalle. Määritä muuttuja seuraavasti:

*muuttuja*

– tai –

*muuttuja* = reaali-luku

Esimerkiksi x kelpaa ja samoin x=3.

## nSolve()

Katalogi > 

**nSolve()** on usein paljon nopeampi kuin **solve()** tai **zeros()**, erityisesti jos operaattoria "**|**" käytetään haun rajaamisessa pienelle välille, joka sisältää täsmälleen yhden yksinkertaisen ratkaisun.

**nSolve()** yrittää määrittää joko yhden pisteen, jossa jäännös on nolla, tai kaksi toisiaan suhteellisen lähellä olevaa pistettä, jossa jäännöksen etumerkki on vastakkainen ja jäännöksen magnitudi ei ole liian suuri. Jos funktio ei pysty määrittämään tätä kohtuullisella otospisteiden määrällä, se antaa vastauksena merkkijonon "no solution found" (yhtään ratkaisua ei löydy).

**Huomaa:** Katso myös **cSolve()**, **cZeros()**, **solve()**, ja **zeros()**.

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

## O

## OneVar

Katalogi > 

**OneVar** [1,]X[,][Frekv][,Luokka,Sisällytä]]

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

Laskee yhden muuttujan tilaston enintään 20 listasta. Tulosten yhteenvedo tallentuu *stat.results*-muuttujaan. (Katso sivu 191.)

Kaikkien listojen on oltava samankokoisia *Sisällytä*-listaa lukuunottamatta.

*X*-argumentit ovat datalistoja.

*Frekv* on valinnainen frekvenssiarvojen lista. Jokainen *Frekv*:n elementti määrittää kunkin vastaavan *X*:n arvon esiintymisfrekvenssin. Oletusarvo on 1. Kaikkien elementtien on oltava kokonaislukuja 0.

*Luokka* on numeeristen luokkakoodien lista vastaaville *X*:n arvoille.

*Sisällyttä* on yhden tai usemman luokkakoodin lista. Vain ne datayksiköt, joiden luokkakoodi sisältyy tähän listaan, ovat mukana laskutoimituksessa.

Tyhjä elementti jossakin listassa  $X$ , *Frekv* tai *Luokka* saa aikaan, että kaikkien listojen vastaava elementti on tyhjä. Tyhjä elementti jossakin listassa  $X1$ - $X20$  saa aikaan, että kaikkien listojen vastaava elementti on tyhjä. Lisätietoja tyhjistä elementeistä, katso sivu 272.

Tulosmuuttuja	Kuvaus
stat. $\bar{x}$	x:n arvojen keskiarvo
stat. $\Sigma x$	x:n arvojen summa
stat. $\Sigma x^2$	$x^2$ :n arvojen summa
stat.sx	x:n otoksen keskihajonta
stat. x	x:n perusjoukon keskihajonta
stat.n	Datapisteiden lukumäärä
stat.MinX	x:n arvojen minimi
stat.Q <sub>1</sub> X	x:n ensimmäinen neljännes
stat.MedianX	x:n mediaani
stat.Q <sub>3</sub> X	x:n kolmas neljännes
stat.MaxX	x:n arvojen maksimi
stat.SSX	x:n keskiarvon poikkeamien neliöiden summa

**or (tai)**

*BooleanLaus1* **or** *BooleanLaus2* antaa vastauksena *Boolean* lausekkeen *BooleanList1*

**or** *BooleanList2* antaa vastauksena

*Boolean* listan *BooleanMatriisi1*

**or** *BooleanMatriisi2* antaa vastauksena

*Boolean* matriisin

$$x \geq 3 \text{ or } x \geq 4$$

$$x \geq 3$$

Määrittää totuusarvon tosi tai epätosi tai antaa vastauksena sievennetyn muodon alkuperäisestä syötteestä.

Antaa vastauksen tosi, jos jompikumpi tai molemmat lausekkeet ovat tosia. Antaa vastauksen epätosi, jos kumpikin lauseke on epätosi.

**Huomaa:** Katso xor.

**Huomaa esimerkkiä syöttäessäsi:** Ohjeet monirivisten ohjelmien ja funktion määritysten syöttämisestä löytyvät tuotteen ohjekirjan Laskin-osiosta.

*Kokonaisluku1*

**or***Kokonaisluku2*⇒*kokonaisluku*

Vertaa kahta reaalikokonaislukua bitti bitiltä or-operaation avulla. Sisäisesti kumpikin kokonaisluku muunnetaan etumerkilliseksi, 64 bitin binaariluvuksi. Kun vastaavia bittejä verrataan, tulos on 1, jos jompikumpi bitti on 1; tulos on 0 vain, jos kumpikin bitti on 0. Laskettu arvo edustaa bittituloksia, ja se näkyy kantelukutilan mukaisesti.

Kokonaisluvut voi syöttää minkä tahansa luvun kantelukuna. Binaarisen syötteen edelle tulee merkitä etumerkki 0b ja heksadesimaalisen syötteen edelle 0h. Jos etumerkkiä ei ole, kokonaislukuja käsitellään desimaalilukuina (kantaluksi 10).

Jos syötät desimaalikokonaisluvun, joka on liian suuri etumerkilliselle, 64 bitin binaarimuodolle, laskin käyttää symmetristä modulo-operaatiota, jotta arvo saadaan oikealle alueelle.

Define $g(x)=$ Func	Done
If $x \leq 0$ or $x \geq 5$	
Goto end	
Return $x \cdot 3$	
Lbl end	
EndFunc	
$g(3)$	9
$g(0)$	A function did not return a value

Heksadesimaalisessa kantelukutilassa:

0h7AC36 or 0h3D5F	0h7BD7F
-------------------	---------

**Tärkeää:** Nolla, ei O-kirjain.

Binaarisessa kantelukutilassa:

0b100101 or 0b100	0b100101
-------------------	----------

**Huomaa:** Binaarisessa syötteessä voi olla korkeintaan 64 numeroa (etuliitettä 0b ei lasketa). Heksadesimaalisessa syötteessä voi olla korkeintaan 16 numeroa.



Jos syötät desimaalikonaisluvun, joka on etumerkillisen, 64 bitin binaarimuodon lukualueen ulkopuolella, laskin käyttää symmetristä modulo-operaatiota, jotta arvo saadaan oikealle alueelle. Lisätietoja, katso ▶**Base2**, sivu 18.

**Huomaa:** Katso **xor**.

## ord()

**ord**(*Merkkijono*) ⇒ *kokonaisluku*

**ord**(*Lista1*) ⇒ *lista*

Määrittää merkkijonon *Merkkijono* ensimmäisen merkin numerokoodin tai luo listan jokaisen listaelementin ensimmäisistä merkeistä.

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

## P

## P ▶ Rx()

P ▶ **Rx**(*rLaus*, *θLaus*) ⇒ *lauseke*

P ▶ **Rx**(*rLista*, *θLista*) ⇒ *lista*

P ▶ **Rx**(*rMatriisi*, *θMatriisi*) ⇒ *matriisi*

Määrittää paria (*r*, *θ*) vastaavan x-koordinaatin.

**Huomaa:** Argumentti *θ* tulkitaan joko aste-, graadi- tai radiaanikulmaksi valittuna olevan kulmatilan mukaisesti. Jos argumentti on lauseke, voit ohittaa kulmatila-asetuksen väliaikaisesti käyttämällä merkintää °, G tai r.

**Huomaa:** Voit syöttää tämän funktion tietokoneen näppäimistöltä kirjoittamalla **P@>Rx** (...).

Radiaanikulmatilassa:

<code>P▶Rx(r,θ)</code>	$\cos(\theta) \cdot r$
<code>P▶Rx(4,60°)</code>	2
<code>P▶Rx({-3,10,1.3}, {π/3, π/4, 0})</code>	$\left\{ \frac{-3}{2}, 5\sqrt{2}, 1.3 \right\}$

## P ▶ Ry()

P ▶ **Ry**(*rLaus*, *θLaus*) ⇒ *lauseke*

Radiaanikulmatilassa:

P► Ry(*rLista*,  $\theta$ *Lista*) $\Rightarrow$ *lista*  
 P► Ry(*rMatriisi*,  $\theta$ *Matriisi*) $\Rightarrow$ *matriisi*

Määrittää paria (*r*,  $\theta$ ) vastaavan y-koordinaatin.

**Huomaa:** Argumentti  $\theta$  tulkitaan joko aste-, radiaani- tai graadikulmaksi valittuna olevan kulmatilan mukaisesti. Jos argumentti on lauseke, voit ohittaa kulmatila-asetuksen väliaikaisesti käyttämällä merkintää °, G tai R.

**Huomaa:** Voit syöttää tämän funktion tietokoneen näppäimistöltä kirjoittamalla P@>Ry (...).

P►Ry( <i>r</i> , $\theta$ )	$\sin(\theta) \cdot r$
P►Ry(4,60°)	$2 \cdot \sqrt{3}$
P►Ry( $\{-3,10,1.3\}, \left\{ \frac{\pi}{3}, \frac{\pi}{4}, 0 \right\}$ )	$\left\{ \frac{-3 \cdot \sqrt{3}}{2}, -5 \cdot \sqrt{2}, 0 \right\}$

## PassErr

### PassErr

Ohittaa virheen siirtyen seuraavalle tasolle.

Jos järjestelmän muuttuja *errCode* on nolla, **PassErr** ei tee mitään.

**Else**-lauseessa lohossa **Try...Else...EndTry** tulee käyttää komentoa **ClrErr** tai **PassErr**. Jos virhe on tarkoitus käsitellä tai jättää huomiotta, käytä komentoa **ClrErr**. Jos et tiedä, mitä tehdä virheen suhteen, lähetä se seuraavaan virheenkäsittelijään käyttämällä komentoa **PassErr**. Jos odottavia **Try...Else...EndTry**-virheenkäsittelijöitä ei ole enää, virheen valintaikkuna tulee näkyviin normaalisti.

**Huomaa:** Katso myös **ClrErr**, sivu 27, ja **Try**, sivu 207.

**Huomaa esimerkkiä syöttäessäsi:** Ohjeet monirivisten ohjelmien ja funktion määritysten syöttämisestä löytyvät tuotteen ohjekirjan Laskin-osiosta.

Esimerkki **PassErr**-komennosta, katso esimerkki 2 **Try**-komennon kohdalla, sivu 207.

## piecewise()

Katalogi > 

**piecewise**(*Laus1* [, *Ehto1* [, *Laus2* [, *Ehto2* [, ... ]]]])

Laatii määritelmät paloittain määritellylle funktiolle listan muodossa. Voit luoda paloittain määriteltyjen funktioiden määrittäjiä myös mallin avulla.

**Huomaa:** Katso myös **Paloittain määritellyn funktion malli**, sivu 3.

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

## poissCdf()

Katalogi > 

**poissCdf**( $\lambda$ , *alaraja*, *yläraja*) $\Rightarrow$  luku, jos *alaraja* ja *yläraja* ovat lukuja, *lista*, jos *alaraja* ja *yläraja* ovat listoja

**poissCdf**( $\lambda$ , *yläraja*)(kun  $P(0 \leq X \leq \text{yläraja}) \Rightarrow$  luku, jos *yläraja* on luku, *lista*, jos *yläraja* on lista

Laskee kumulatiivisen todennäköisyyden diskreetille Poissonin jakaumalle, jolla on määritetty keskiarvo  $\lambda$ .

Kun  $P(X \leq \text{yläraja})$ , aseta *alaraja*=0

## poissPdf()

Katalogi > 

**poissPdf**( $\lambda$ , *XArvo*) $\Rightarrow$  luku, jos *XArvo* on luku, *lista*, jos *XArvo* on lista

Laskee todennäköisyyden diskreetille Poissonin jakaumalle, jolla on määritetty keskiarvo  $\lambda$ .

## ► Polar (►Polaarinen)

Katalogi > 

*Vektori* ► **Polar**

**Huomaa:** Voit syöttää tämän operaattorin tietokoneen näppäimistöltä kirjoittamalla @>Polar.

[1 3.]►Polar	[3.16228 1.24905]
[x y.]►Polar	$\left[ \sqrt{x^2+y^2} \quad \angle \frac{\pi \cdot \text{sign}(y)}{2} - \tan^{-1}\left(\frac{x}{y}\right) \right]$

Näyttää *vektorin* polaarisisä muodossa  $[r \angle \theta]$ . Vektorin on oltava kooltaan 2, ja se voi olla rivi tai sarake.

**Huomaa:** ► **Polar** on näyttömuodon ohje, ei muunnosfunktio. Voit käyttää komentoa ainoastaan syöterivin lopussa, eikä se päivitä *ans*:n arvoa.

**Huomaa:** Katso myös ► **Rect**, sivu 158.

*kompleksiArvo* ► **Polaarinen**

Näyttää *kompleksiVektorin* polaarisisä muodossa.

- Astekulmatilassa vastauksena on  $(r \angle \theta)$ .
- Radiaanikulmatilassa vastauksena on  $re^{i\theta}$ .

*kompleksiArvolla* voi olla mikä tahansa kompleksilukumuoto. Syöte  $re^{i\theta}$  aiheuttaa kuitenkin virheen astekulmatilassa.

**Huomaa:** Polaarisisä syötteessä  $(r \angle \theta)$  on käytettävä sulkeita.

Radiaanikulmatilassa:

$$\begin{array}{l} (3+4 \cdot i) \text{►Polar} \qquad e^{i \cdot \left( \frac{\pi}{2} - \tan^{-1} \left( \frac{3}{4} \right) \right)} \cdot 5 \\ \left( 4 \angle \frac{\pi}{3} \right) \text{►Polar} \qquad e^{\frac{i \cdot \pi}{3}} \cdot 4 \end{array}$$

Graadikulmatilassa:

$$(4 \cdot i) \text{►Polar} \qquad (4 \angle 100)$$

Astekulmatilassa:

$$(3+4 \cdot i) \text{►Polar} \qquad \left( 5 \angle 90 - \tan^{-1} \left( \frac{3}{4} \right) \right)$$

## polyCoeffs()

**polyCoeffs**(*Poly* [,*Muutt*]) ⇒ lista

Laatii listan polynomin *Poly* kertoimista muuttujan *Muutt* suhteen.

*Poly* on oltava polynomilauseke muuttujassa *Muutt*. Suositeltavaa on, ettet jätä muuttujaa *Muutt* pois, ellei *Poly* ole lauseke yhdessä muuttujassa.

$$\text{polyCoeffs}(4 \cdot x^2 - 3 \cdot x + 2, x) \qquad \{4, -3, 2\}$$

$$\text{polyCoeffs}((x-1)^2 \cdot (x+2)^3) \qquad \{1, 4, 1, -10, -4, 8\}$$

Laventaa polynomin ja valitsee *x*:n poisjätetyille muuttujalle *Muutt*.

$\text{polyCoeffs}\left(\left(x+y+z\right)^2, x\right)$	$\{1, 2 \cdot (y+z), (y+z)^2\}$
$\text{polyCoeffs}\left(\left(x+y+z\right)^2, y\right)$	$\{1, 2 \cdot (x+z), (x+z)^2\}$
$\text{polyCoeffs}\left(\left(x+y+z\right)^2, z\right)$	$\{1, 2 \cdot (x+y), (x+y)^2\}$

## polyDegree()

**polyDegree**(*Poly* [, *Muutt*])  $\Rightarrow$  arvo

Määrittää polynomilausekkeen *Poly* asteen muuttujan *Muutt* suhteen. Jos jätät muuttujan *Muutt* pois, **polyDegree** ()-funktio valitsee oletusarvon polynomien *Poly* sisältämistä muuttujista.

*Poly* on oltava polynomilauseke muuttujassa *Muutt*. Suositeltavaa on, ettet jätä muuttujaa *Muutt* pois, ellei *Poly* ole lauseke yhdessä muuttujassa.

$\text{polyDegree}(5)$	0
$\text{polyDegree}(\ln(2)+\pi, x)$	0
Vakiopolynomit	
$\text{polyDegree}(4 \cdot x^2 - 3 \cdot x + 2, x)$	2
$\text{polyDegree}((x-1)^2 \cdot (x+2)^3)$	5
$\text{polyDegree}\left(\left(x+y^2+z^3\right)^2, x\right)$	2
$\text{polyDegree}\left(\left(x+y^2+z^3\right)^2, y\right)$	4
$\text{polyDegree}\left(\left(x-1\right)^{10000}, x\right)$	10000

Asteesta voidaan ottaa juuri, vaikka kertoimista ei voida. Tämä johtuu siitä, että asteesta voidaan ottaa juuri lauantamatta polynomia.

## polyEval()

**polyEval**(*Lista1*, *Laus1*)  $\Rightarrow$  lauseke  
**polyEval**(*Lista1*, *Lista2*)  $\Rightarrow$  lauseke

$\text{polyEval}(\{a, b, c\}, x)$	$a \cdot x^2 + b \cdot x + c$
$\text{polyEval}(\{1, 2, 3, 4\}, 2)$	26
$\text{polyEval}(\{1, 2, 3, 4\}, \{2, -7\})$	$\{26, -262\}$

**polyEval()**Katalogi > 

Tulkitsee ensimmäisen argumentin laskeva-asteisen polynomin kertoimeksi ja antaa vastauksena polynomin, josta on laskettu toisen argumentin arvo.

**polyGcd()**Katalogi > **polyGcd(Laus1,Laus2)**⇒*lauseke*

Laskee kahden argumentin suurimman yhteisen jakajan.

*Laus1*:n ja *Laus2*:n on oltava polynomilausekkeita.

Lista-, matriisi- ja Boolean argumentteja ei sallita.

$\text{polyGcd}(100,30)$	10
$\text{polyGcd}(x^2-1,x-1)$	$x-1$
$\text{polyGcd}(x^3-6\cdot x^2+11\cdot x-6,x^2-6\cdot x+8)$	$x-2$

**polyQuotient()**Katalogi > **polyQuotient(Poly1,Poly2 [,Muutt])**⇒*lauseke*

Laskee polynomin *Poly1* osamäärän jaettuna polynomilla *Poly2* määritetyn muuttujan *Muutt* suhteen.

*Poly1*:n ja *Poly2*:n on oltava polynomilausekkeita muuttujassa *Muutt*. Suositeltavaa on, ettet jätä muuttujaa *Muutt* pois, elleivät *Poly1* ja *Poly2* ole lausekkeita yhdessä ja samassa muuttujassa.

$\text{polyQuotient}(x-1,x-3)$	1
$\text{polyQuotient}(x-1,x^2-1)$	0
$\text{polyQuotient}(x^2-1,x-1)$	$x+1$
$\text{polyQuotient}(x^3-6\cdot x^2+11\cdot x-6,x^2-6\cdot x+8)$	$x$
$\text{polyQuotient}((x-y)\cdot(y-z),x+y+z,x)$	$y-z$
$\text{polyQuotient}((x-y)\cdot(y-z),x+y+z,y)$	$2\cdot x-y+2\cdot z$
$\text{polyQuotient}((x-y)\cdot(y-z),x+y+z,z)$	$-(x-y)$

**polyRemainder()**Katalogi > **polyRemainder(Poly1,Poly2 [,Muutt])**⇒*lauseke*

$\text{polyRemainder}(x-1,x-3)$	2
$\text{polyRemainder}(x-1,x^2-1)$	$x-1$
$\text{polyRemainder}(x^2-1,x-1)$	0

## polyRemainder()

Katalogi > 

Laskee jakojäännöksen polynomista  $Poly1$ , joka on jaettu polynomilla  $Poly2$  määritetyn muuttujan  $Muutt$  suhteen.

$Poly1$ :n ja  $Poly2$ :n on oltava polynomilausekkeita muuttujassa  $Muutt$ . Suositeltavaa on, ettei jätä muuttujaa  $Muutt$  pois, elleivät  $Poly1$  ja  $Poly2$  ole lausekkeita yhdessä ja samassa muuttujassa.

$$\frac{\text{polyRemainder}((x-y)\cdot(y-z),x+y+z,x)}{-(y-z)\cdot(2\cdot y+z)}$$
$$\frac{\text{polyRemainder}((x-y)\cdot(y-z),x+y+z,y)}{-2\cdot x^2-5\cdot x\cdot z-2\cdot z^2}$$
$$\frac{\text{polyRemainder}((x-y)\cdot(y-z),x+y+z,z)}{(x-y)\cdot(x+2\cdot y)}$$

## polyRoots()

Katalogi > 

**polyRoots( $Poly, Muutt$ )**  $\Rightarrow$  lista

**polyRoots( $Kertoinlista$ )**  $\Rightarrow$  lista

Ensimmäinen syntaksi, **polyRoots( $Poly, Muutt$ )**, laskee polynomin  $Poly$  reaalitylukujuurten listan muuttujan  $Muutt$  suhteen. Jos reaalitylukujuurta ei ole, tuloksena on tyhjä lista: { }.

$Poly$  on oltava polynomi yhdessä muuttujassa.

Toinen syntaksi, **polyRoots( $Kertoinlista$ )**, askee reaalitylukujuurten listan kertoimille, jotka sisältyvät  $Kertoinlistaan$ .

**Huomaa:** Katso myös **cPolyRoots()**, sivu 39.

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

## PowerReg

Katalogi > 

**PowerReg**  $X, Y$  [,  $Frekv$ ] [,  $Luokka$ ,  $Sisällyttä$ ]]

Laskee potenssiregression  $y = (a \cdot x)^b$  listoista  $X$  ja  $Y$  frekvenssillä  $Frekv$ . Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 191.)

Kaikkien listojen on oltava samankokoisia  $Sisällyttä$ -listaa lukuunottamatta.

$X$  ja  $Y$  ovat riippumattomien ja riippuvien muuttujien listoja.

*Frekv* on valinnainen frekvenssiarvojen lista. Jokainen *Frekv*:n elementti määrittää kunkin vastaavan datapisteen  $X$  ja  $Y$  esiintymisfrekvenssin. Oletusarvo on 1. Kaikkien elementtien on oltava kokonaislukuja 0.

*Luokka* on luokkakoodien lista vastaavalle  $X$ - ja  $Y$ -datalle.

*Sisällytä* on yhden tai usemman luokkakoodin lista. Vain ne datayksiköt, joiden luokkakoodi sisältyy tähän listaan, ovat mukana laskutoimituksessa.

Lisätietoja listassa olevien tyhjiä elementtien vaikutuksesta, katso Tyhjat elementtsivulla sivu 272.

Tulosmuuttuja	Kuvaus
stat.RegEqn	Regressioyhtälö: $a \cdot (x)^b$
stat.a, stat.b	Regressiokertoimet
stat.r <sup>2</sup>	Muunnettujen tietojen lineaarimäärittelyn kerroin
stat.r	Muunnettujen tietojen korrelaatiokerroin ( $\ln(x)$ , $\ln(y)$ )
stat.Resid	Potenssimalliin liittyvät jäännökset
stat.ResidTrans	Muunnettujen tietojen lineaariseen sovitukseen liittyvät jäännökset
stat.XReg	Muokatun <i>Y Lista</i> :n sisältämä datapisteiden lista, jota käytetään regressiossa komentojen <i>Frekv</i> , <i>Luokkalista</i> ja <i>Sisällytä luokat rajoitusten mukaisesti</i>
stat.YReg	Muokatun <i>Y Lista</i> :n sisältämä datapisteiden lista, jota käytetään regressiossa komentojen <i>Frekv</i> , <i>Luokkalista</i> ja <i>Sisällytä luokat rajoitusten mukaisesti</i>
stat.FreqReg	Komentoja <i>stat.XReg</i> ja <i>stat.YReg</i> vastaava frekvenssilista

Prgm  
*Lohko*  
EndPrgm

Laske GCD ja näytä välitulokset.



Malli käyttäjän määrittämän ohjelman luomista varten. Käytetään komennon **Define**, **Define LibPub** tai **Define LibPriv** kanssa.

*Lohko* voi olla yksi lauseke tai sarja lausekkeita, jotka on erotettu toisistaan kaksoispisteellä (:), tai sarja eri riveillä olevia lausekkeita.

**Huomaa esimerkkiä syöttäessäsi:** Ohjeet monirivisten ohjelmien ja funktion määritysten syöttämisestä löytyvät tuotteen ohjekirjan Laskin-osiosta.

```
Define proggcd(a,b)=Prgm
  Local d
  While b≠0
  d:=mod(a,b)
  a:=b
  b:=d
  Disp a, " ",b
EndWhile
Disp "GCD=",a
EndPrgm

```

---

*Done*

---

```
proggcd(4560,450)

```

---

450 60

---

60 30

---

30 0

---

GCD=30

---

*Done*

---

**prodSeq()****Katso  $\Pi()$ , sivu 242.****Product (PI)****Katso  $\Pi()$ , sivu 242.****product()**Katalogi > 

**product(Lista[, Alku[, loppu]])** ⇒ lauseke

Laskee *Listan* sisältämien elementtien tulon. *Alku* ja *Loppu* ovat valinnaisia. Ne määrittävät elementtien alueen.

**product(MatriisiI[, Alku[, loppu]])** ⇒ matriisi

Laskee rivivektorin, joka sisältää *MatriisiI*:n sarakkeiden elementtien tulot. *Alku* ja *Loppu* ovat valinnaisia. Ne määrittävät rivialueen.

product({1,2,3,4})	24
product({2,x,y})	2·x·y
product({4,5,8,9},2,3)	40
product( $\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix}$ )	[28 80 162]
product( $\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix}, 1, 2$ )	[4 10 18]

Tyhjiä elementtejä ei huomioida.  
Lisätietoja tyhjistä elementeistä, katso sivu 272.

## propFrac()

**propFrac(LausI[, Muutt])** ⇒ lauseke

**propFrac(rationaali\_luku)** laskee *rationaali\_luvun* kokonaisluvun summana ja murtolukuna, jolla on sama etumerkki ja suurempi nimittäjä kuin osoittaja.

**propFrac(rationaali\_lauseke, Muutt)** laskee aitojen suhdelukujen summan ja polynomin muuttujan *Muutt* suhteen. Muuttujan *Muutt* aste nimittäjässä on suurempi kuin muuttujan *Muutt* aste osoittajassa kussakin aidossa suhdeluvussa. Muuttujan *Muutt* samanlaiset potenssit kerätään. Termit ja niiden tekijät lajitellaan siten, että *Muutt* on päämuuttuja.

Jos *Muutt* jätetään pois, varsinaisen murtoluvun lavennus tehdään pääasiallisimmalla muuttujalla. Polynomiosan kertoimet tehdään sen jälkeen aidoiksi ensin niiden pääasiallisimman muuttujan suhteen ja niin edelleen.

Rationaalilausekkeissa **propFrac()** on nopeampi, mutta vähemmän äärimmäinen vaihtoehto kuin **expand()**.

**propFrac()**-funktion avulla voit esittää sekalukuja ja suorittaa sekalukujen yhteen- ja vähennyslaskua.

$$\text{propFrac}\left(\frac{4}{3}\right) \quad 1 + \frac{1}{3}$$

$$\text{propFrac}\left(\frac{-4}{3}\right) \quad -1 - \frac{1}{3}$$

$$\text{propFrac}\left(\frac{x^2+x+1}{x+1} + \frac{y^2+y+1}{y+1}, x\right) \quad \frac{1}{x+1} + x + \frac{y^2+y+1}{y+1}$$

$$\text{propFrac(Ans)} \quad \frac{1}{x+1} + x + \frac{1}{y+1} + y$$

$$\text{propFrac}\left(\frac{11}{7}\right) \quad 1 + \frac{4}{7}$$

$$\text{propFrac}\left(3 + \frac{1}{11} + 5 + \frac{3}{4}\right) \quad 8 + \frac{37}{44}$$



$$\text{propFrac}\left(3 + \frac{1}{11} - \left(5 + \frac{3}{4}\right)\right) \quad -2 - \frac{29}{44}$$

## Q

QR *Matriisi*, *qMatNimi*, *rMatNimi* [, *Tol*]

Suorittaa Householderin QR-dekomponoinnin reaaliuku- tai kompleksilukumatriisista. Tuloksena olevat Q- ja R-matriisit tallentuvat määritettyihin *MatNimi*-muuttujiin. Q-matriisi on jakamaton. R-matriisi on yläkolmiomatriisi.

Valinnaisesti kaikkia matriisielementtejä käsitellään nollana, jos niiden itseisarvo on pienempi kuin *Tol*. Tätä toleranssia käytetään vain, jos matriisissa on liukulukusyötteitä eikä se sisällä symbolisia muuttujia, joille ei ole määritetty arvoa. Muussa tapauksessa *Tol*-komentoa ei huomioida.

- Jos käytät painikkeita   tai **Automaattinen tai likimääräinen** -tilan valintaa Approximate (Likimääräinen), laskut suoritetaan liukulukuaritmetiikalla.
- Jos *Tol* jätetään pois tai sitä ei käytetä, oletusarvoinen toleranssi lasketaan seuraavasti:  
 $5E-14 \cdot \max(\dim(\text{Matriisi})) \cdot \text{rowNorm}(\text{Matriisi})$

QR-hajotus lasketaan numeerisesti Householderin transformaatioiden avulla. Symbolinen ratkaisu lasketaan Gram-Schmidtin menetelmällä. Matriisin *qMatNimi* sarakkeet ovat ne ortonormitetut perusvektorit, jotka sijoittuvat *matriisin* määrittämään tilaan.

Liukuluku (9.) muuttujassa m1 aiheuttaa sen, että vastaukset lasketaan liukulukumuodossa.

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix} \rightarrow m1 \quad \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$$

QR m1,qm,rm Done

$$qm \quad \begin{bmatrix} 0.123091 & 0.904534 & 0.408248 \\ 0.492366 & 0.301511 & -0.816497 \\ 0.86164 & -0.301511 & 0.408248 \end{bmatrix}$$

$$rm \quad \begin{bmatrix} 8.12404 & 9.60114 & 11.0782 \\ 0. & 0.904534 & 1.80907 \\ 0. & 0. & 0. \end{bmatrix}$$

$$\begin{bmatrix} m & n \\ o & p \end{bmatrix} \rightarrow m1 \quad \begin{bmatrix} m & n \\ o & p \end{bmatrix}$$

QR m1,qm,rm Done

$$qm \quad \begin{bmatrix} \frac{m}{\sqrt{m^2+o^2}} & \frac{-\text{sign}(m \cdot p - n \cdot o) \cdot o}{\sqrt{m^2+o^2}} \\ \frac{o}{\sqrt{m^2+o^2}} & \frac{m \cdot \text{sign}(m \cdot p - n \cdot o)}{\sqrt{m^2+o^2}} \end{bmatrix}$$

$$rm \quad \begin{bmatrix} \sqrt{m^2+o^2} & \frac{m \cdot n + o \cdot p}{\sqrt{m^2+o^2}} \\ 0 & \frac{|m \cdot p - n \cdot o|}{\sqrt{m^2+o^2}} \end{bmatrix}$$

QuadReg *X,Y* [, *Frekv*] [, *Luokka*, *Sisällyt*]

Laskee 2. asteen polynomiregression  $y = a \cdot x^2 + b \cdot x + c$  listaosta  $X$  ja  $Y$  frekvenssillä *Frekv*. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 191.)

Kaikkien listojen on oltava samankokoisia *Sisällytä*-listaa lukuunottamatta.

$X$  ja  $Y$  ovat riippumattomien ja riippuvien muuttujien listoja.

*Frekv* on valinnainen frekvenssiarvojen lista. Jokainen *Frekv*:n elementti määrittää kunkin vastaavan datapisteen  $X$  ja  $Y$  esiintymisfrekvenssin. Oletusarvo on 1. Kaikkien elementtien on oltava kokonaislukuja 0.

*Luokka* on luokkakoodien lista vastaavalle  $X$ - ja  $Y$ -datalle..

*Sisällytä* on yhden tai usemman luokkakoodin lista. Vain ne datayksiköt, joiden luokkakoodi sisältyy tähän listaan, ovat mukana laskutoimituksessa.

Lisätietoja listassa olevien tyhjen elementtien vaikutuksesta, katso Tyhjät elementtsivulla sivu 272.

Tulosmuuttuja	Kuvaus
stat.RegEqn	Regressioyhtälö: $a \cdot x^2 + b \cdot x + c$
stat.a, stat.b, stat.c	Regressiokertoimet
stat.R <sup>2</sup>	Määrityskerroin
stat.Resid	Regressioyhtälön jäännökset
stat.XReg	Muokatus <i>Y Lista</i> :n sisältämä datapisteiden lista, jota käytetään regressiossa komentojen <i>Frekv</i> , <i>Luokkalista</i> ja <i>Sisällytä luokat rajoitusten mukaisesti</i>
stat.YReg	Muokatus <i>Y Lista</i> :n sisältämä datapisteiden lista, jota käytetään regressiossa komentojen <i>Frekv</i> , <i>Luokkalista</i> ja <i>Sisällytä luokat rajoitusten mukaisesti</i>
stat.FreqReg	Komentoja <i>stat.XReg</i> ja <i>stat.YReg</i> vastaava frekvenssilista

**QuartReg** *X,Y* [, *Frekv*] [, *Luokka*, *Sisällytä*]]

Laskee 4. asteen polynomiregression  $y = a \cdot x^4 + b \cdot x^3 + c \cdot x^2 + d \cdot x + e$  listaista *X* ja *Y* frekvenssillä *Frekv*. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu sivu 191.)

Kaikkien listojen on oltava samankokoisia *Sisällytä*-listaa lukuunottamatta.

*X* ja *Y* ovat riippumattomien ja riippuvien muuttujien listoja.

*Frekv* on valinnainen frekvenssiarvojen lista. Jokainen *Frekv*:n elementti määrittää kunkin vastaavan datapisteen *X* ja *Y* esiintymisfrekvenssin. Oletusarvo on 1. Kaikkien elementtien on oltava kokonaislukuja 0.

*Luokka* on luokkakoodien lista vastaavalle *X*- ja *Y*-dataalle.

*Sisällytä* on yhden tai usemman luokkakoodin lista. Vain ne datayksiköt, joiden luokkakoodi sisältyy tähän listaan, ovat mukana laskutoimituksessa.

Lisätietoja listassa olevien tyhjiä elementtien vaikutuksesta, katso Tyhjät elementtsivulla sivu 272.

Tulosmuuttuja	Kuvaus
stat.RegEqn	Regressioyhtälö: $a \cdot x^4 + b \cdot x^3 + c \cdot x^2 + d \cdot x + e$
stat.a, stat.b, stat.c, stat.d, stat.e	Regressiokertoimet
stat.R <sup>2</sup>	Määrittyskerroin
stat.Resid	Regressioyhtälön jäännökset
stat.XReg	Muokatun <i>Y</i> Lista:n sisältämä datapisteiden lista, jota käytetään regressiossa komentojen <i>Frekv</i> , <i>Luokkalista</i> ja <i>Sisällytä</i> luokat rajoitusten mukaisesti

Tulosmuuttuja	Kuvaus
stat.YReg	Muokatun <i>Y Lista</i> :n sisältämä datapisteiden lista, jota käytetään regressiossa komentojen <i>Frekv</i> , <i>Luokkalista</i> ja <i>Sisällytä luokat rajoitusten mukaisesti</i>
stat.FreqReg	Komentoja <i>stat.XReg</i> ja <i>stat.YReg</i> vastaava frekvenssilista

## R

### R ► Pθ()

Luettelo > 

R ► Pθ (*xLaus*, *yLaus*) ⇒ lauseke

Astekulmatilassa:

R ► Pθ (*xLista*, *yLista*) ⇒ lista

$$\text{R} \blacktriangleright \text{P}\theta(x, y) \quad 90 \cdot \text{sign}(y) - \tan^{-1}\left(\frac{x}{y}\right)$$

R ► Pθ (*xMatriisi*, *yMatriisi*) ⇒ matriisi

Laskee yhtäpitävän θ-koordinaatin argumenttiparille (*x*, *y*).

Graadikulmatilassa:

$$\text{R} \blacktriangleright \text{P}\theta(x, y) \quad 100 \cdot \text{sign}(y) - \tan^{-1}\left(\frac{x}{y}\right)$$

**Huomaa:** Vastaus lasketaan aste-, graadi- tai radiaanikulmana käytössä olevan kulmatila-asetuksen mukaisesti.

Radiaanikulmatilassa:

**Huomaa:** Voit syöttää tämän funktion tietokoneen näppäimistöltä kirjoittamalla **R@>Ptheta (...)**.

$$\text{R} \blacktriangleright \text{P}\theta(3, 2) \quad \tan^{-1}\left(\frac{2}{3}\right)$$

$$\text{R} \blacktriangleright \text{P}\theta\left([3 \ -4 \ 2], \left[0 \ \frac{\pi}{4} \ 1.5\right]\right) \quad \left[0 \ \tan^{-1}\left(\frac{16}{\pi}\right) + \frac{\pi}{2} \ 0.643501\right]$$

### R ► Pr()

Luettelo > 

R ► Pr (*xLaus*, *yLaus*) ⇒ lauseke

Radiaanikulmatilassa:

R ► Pr (*xLista*, *yLista*) ⇒ lista

$$\text{R} \blacktriangleright \text{Pr}(3, 2) \quad \sqrt{13}$$

R ► Pr (*xMatriisi*, *yMatriisi*) ⇒ matriisi

$$\text{R} \blacktriangleright \text{Pr}(x, y) \quad \sqrt{x^2 + y^2}$$

Laskee yhtäpitävän r-koordinaatin argumenttiparille (*x*, *y*).

$$\text{R} \blacktriangleright \text{Pr}\left([3 \ -4 \ 2], \left[0 \ \frac{\pi}{4} \ 1.5\right]\right)$$

**Huomaa:** Voit syöttää tämän funktion tietokoneen näppäimistöltä kirjoittamalla **R@>Pr (...)**.

$$\left[3 \ \frac{\sqrt{\pi^2 + 256}}{4} \ 2.5\right]$$

**► Rad**Luettelo > *Lausl* ► *Rad* ⇒ *lauseke*

Astekulmatilassa:

Muuntaa argumentin  
radiaanikulmayksikköön.

(1.5)►Rad	(0.02618) <sup>r</sup>
-----------	------------------------

**Huomaa:** Voit syöttää tämän  
operaattorin tietokoneen näppäimistöltä  
kirjoittamalla @>Rad.

Graadikulmatilassa:

(1.5)►Rad	(0.023562) <sup>r</sup>
-----------	-------------------------

**rand()**Luettelo > **rand()** ⇒ *lauseke***rand(Kokeiden lkm)** ⇒ *lista*

Aseta satunnainen siemenluku.

**rand()** laskee satunnaisen arvon välillä 0  
ja 1.

RandSeed 1147	Done
---------------	------

rand(2)	{0.158206,0.717917}
---------	---------------------

**rand(Kokeiden lkm)** laskee listan, joka  
sisältää *Kokeiden lkm* satunnaista arvoa  
välillä 0 ja 1.**randBin()**Luettelo > **randBin(n, p)** ⇒ *lauseke***randBin(n, p, Kokeiden lkm)** ⇒ *lista*

randBin(80,0,5)	42
-----------------	----

**randBin(n, p)** laskee satunnaisen  
reaaliluvun määrätystä  
binomijakaumasta.

randBin(80,0,5,3)	{41,32,39}
-------------------	------------

**randBin(n, p, Kokeiden lkm)** laskee  
listan, joka sisältää *Kokeiden lkm*  
satunnaista reaalilukua määrätystä  
binomijakaumasta.**randInt()**Luettelo > **randInt****(**  
*Muutt.alaraja*  
*,Muutt.yläaraja)* ⇒  
*lauseke*

randInt(3,10)	5
---------------	---

randInt(3,10,4)	{9,7,5,8}
-----------------	-----------

**randInt****(**  
*Muutt.alaraja*  
*,Muutt.yläaraja*  
*,Kokeiden lkm)* ⇒

*lista***randInt**

(  
*Muutt.alaraja*  
*,Muutt.yläraja*) laskee  
 satunnaisen  
 kokonaisluvun  
 rajojen *Muutt.alaraja*  
 ja *Muutt.yläraja*  
 määrämällä  
 kokonaislukuvälillä.

**randInt**

(  
*Muutt.alaraja*  
*,Muutt.yläraja*  
*,Kokeiden lkm*)  
 laskee listan, joka  
 sisältää *Kokeiden lkm*  
 satunnaista  
 kokonaislukua  
 määrättyllä välillä.

**randMat()**

**randMat**(*numRivit*, *numSarakkeet*) ⇒  
*matriisi*

Laskee välillä -9 ja 9 määrätyle mitalle  
 kokonaislukumatriisin.

Kummankin argumentin on  
 sievennytävä kokonaisluvuksi.

RandSeed 1147	Done									
randMat(3,3)	<table border="1"> <tr><td>8</td><td>-3</td><td>6</td></tr> <tr><td>-2</td><td>3</td><td>-6</td></tr> <tr><td>0</td><td>4</td><td>-6</td></tr> </table>	8	-3	6	-2	3	-6	0	4	-6
8	-3	6								
-2	3	-6								
0	4	-6								

**Huomaa:** Tämän matriisin arvot muuttuvat  
 aina, kun painat .

**randNorm()**

**randNorm**( $\mu$ ,  $\sigma$ ) ⇒ *lauseke*  
**randNorm**( $\mu$ ,  $\sigma$ , *Kokeiden lkm*) ⇒ *lista*

**randNorm**( $\mu$ ,  $\sigma$ ) laskee desimaaliluvun  
 määrätystä normaalijakaumasta. Luku  
 voi olla mikä tahansa reaalityyppinen, mutta  
 se keskittyy voimakkaasti välille  $[\mu-3\cdot\sigma$ ,  
 $\mu+3\cdot\sigma]$ .

RandSeed 1147	Done
randNorm(0,1)	0.492541
randNorm(3,4.5)	-3.54356



## randNorm()

Luettelo > 

**randNorm**( $\mu$ ,  $\sigma$ , *Kokeiden lkm*) listan (list), joka sisältää *Kokeiden lkm* desimaalilukua määrätystä normaalijakaumasta.

## randPoly()

Luettelo > 

**randPoly**(*Muutt*, *Aste*)  $\Rightarrow$  lauseke

Laskee polynomin muuttujasta *Muutt* määrättyssä järjestyksessä *Aste*.

Kertoimet ovat satunnaisia kokonaislukuja välillä  $-9$  ja  $9$ . Johtava kerroin ei ole 0.

*Asteen* on oltava  $0-99$ .

RandSeed 1147	Done
randPoly(x,5)	$-2 \cdot x^5 + 3 \cdot x^4 - 6 \cdot x^3 + 4 \cdot x - 6$

## randSample()

Luettelo > 

**randSample**(*List*, *Kokeiden lkm* [,*noRepl*])  $\Rightarrow$  lista

Laskee satunnaisesta otoksesta listan, joka koostuu valinnoista *Kokeiden lkm* listasta *List* valinnaisena otoksen korvaaminen muulla (*noRepl*=0), tai ilman otoksen korvaamista (*noRepl*=1). Oletusarvona on otoksen korvaaminen

Define list3={1,2,3,4,5}	Done
Define list4=randSamp(list3,6)	Done
list4	{2,3,4,3,1,2}

## RandSeed

Luettelo > 

**RandSeed** *Luku*

Jos *Luku* = 0, tämä asettaa siemenluvut satunnaislukugeneraattorin tehdasasetuksiin. Jos *Luku*  $\neq$  0, sitä käytetään luomaan kaksi siemenlukua, jotka tallennetaan järjestelmän muuttujiin siemen1 ja siemen2

RandSeed 1147	Done
rand()	0.158206

## real()

Luettelo > 

**real**(*LausI*)  $\Rightarrow$  lauseke

Laskee argumentin reaali osan.

real(2+3·i)	2
real(z)	z
real(x+i·y)	x

**Huomaa:** Kaikkia määrittelemättömiä muuttujia käsitellään reaali muuttujina. Katso myös **imag()**, page 96.

**real(ListI)** ⇒ lista

Laskee kaikkien alkutekijöiden reaalisosat

**real(MatriisiI)** ⇒ matriisi

Laskee kaikkien alkutekijöiden reaalisosat.

$$\text{real}(\{a+i\cdot b, 3, i\}) \quad \{a, 3, 0\}$$

$$\text{real}\left(\begin{bmatrix} a+i\cdot b & 3 \\ c & i \end{bmatrix}\right) \quad \begin{bmatrix} a & 3 \\ c & 0 \end{bmatrix}$$

## ► Rect

**Vectr** ► **Rect**

**Huomaa:** Voit syöttää tämän operaattorin tietokoneen näppäimistöltä kirjoittamalla **@>Rect**.

Näyttää **Vektorin** suorakulmamuodossa [x, y, z] Vektorin on oltava kooltaan 2 tai 3, ja se voi olla rivi tai sarake.

**Huomaa:** ► **Rect** on näyttömuodon ohje, ei muunnosfunktio. Voit käyttää kommentoa ainoastaan syöterivin lopussa, eikä se päivitä arvoa *ans*.

**Huomaa:** Katso myös ► **Polar**, sivu 143.

**kompleksiArvo** ► **Rect**

Näyttää kompleksisen arvon **complexValues** suorakulmaisessa muodossa a+bi. Arvolla **complexValue** voi olla mikä tahansa kompleksinen muoto. Syöte  $re^{i\theta}$  aiheuttaa kuitenkin virheen astekulmatilassa.

**Huomaa:** Polaarisisä syötteessä (r ∠ θ) on käytettävä sulkeita.

$$\left(3 \angle \frac{\pi}{4} \quad \angle \frac{\pi}{6}\right) \text{►Rect}$$

$$\begin{bmatrix} 3\cdot\sqrt{2} & 3\cdot\sqrt{2} & 3\cdot\sqrt{3} \\ 4 & 4 & 2 \end{bmatrix}$$

$$\begin{bmatrix} a \angle b \quad \angle c \\ a\cdot\cos(b)\cdot\sin(c) & a\cdot\sin(b)\cdot\sin(c) & a\cdot\cos(c) \end{bmatrix}$$

Radiaanikulmatilassa:

$$\left(4 \cdot e^{\frac{\pi}{3}}\right) \text{►Rect} \quad 4 \cdot e^{\frac{\pi}{3}}$$

$$\left(4 \angle \frac{\pi}{3}\right) \text{►Rect} \quad 2+2\cdot\sqrt{3}\cdot i$$

Graadikulmatilassa:

$$\left((1 \angle 100)\right) \text{►Rect} \quad i$$

Astekulmatilassa:

$$\left((4 \angle 60)\right) \text{►Rect} \quad 2+2\cdot\sqrt{3}\cdot i$$

**Huomaa:** Näppäilläksesi merkin ∠ valitse se Luettelon symboliluettelosta.

$\text{ref}(\text{Matriisi}1[, \text{Tol}]) \Rightarrow \text{matriisi}$

Laskee matriisiin *Matriisi1* rivi-echelonmuodossa.

Valinnaisesti kaikkia matriisin elementtejä käsitellään nollana, jos niiden itseisarvo on pienempi kuin *Tol*. Tätä toleranssia käytetään vain, jos matriisissa on liukulukusyötteitä eikä se sisällä symbolisia muuttujia, joille ei ole määrätty arvoa. Muussa tapauksessa komentoa *Tol* ei huomioida.

- Jos käytät painikkeita   tai asetat tilan **Auto or Approximate**, laskenta suoritetaan liukulukuaritmetiikalla.
- Jos *Tol* jätetään pois tai sitä ei käytetä, oletusarvoinen toleranssi lasketaan seuraavasti:  
 $5E-14 \cdot \max(\dim(\text{Matriisi}1))$   
 $\bullet \text{rowNorm}(\text{Matriisi}1)$

Vältä määrittelemättömiä alkutekijöitä matriisissa *Matriisi1*. Tuloksena saattaa olla odottamattomia vastauksia.

Jos esimerkiksi *a* on määrittelemätön seuraavassa lausekkeessa, näkyviin tulee varoitusviesti, ja vastaus näyttää seuraavalta:

$$\text{ref} \left( \begin{bmatrix} a & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \right) \quad \begin{bmatrix} 1 & \frac{1}{a} & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Varoitus näkyy, koska yleistetty alkutekijä  $1/a$  ei olisi mahdollinen lausekkeelle  $a=0$ .

Voit välttää tämän tallentamalla etukäteen arvon *a*:lle tai käyttämällä rajoittavaa ("|")-operaattoria arvon korvaamiseksi, kuten on esitetty seuraavassa esimerkissä.

$$\text{ref} \left( \begin{bmatrix} -2 & -2 & 0 & -6 \\ 1 & -1 & 9 & -9 \\ -5 & 2 & 4 & -4 \end{bmatrix} \right) \quad \begin{bmatrix} 1 & \frac{-2}{5} & \frac{-4}{5} & \frac{4}{5} \\ 0 & 1 & \frac{4}{7} & \frac{11}{7} \\ 0 & 0 & 1 & \frac{-62}{71} \end{bmatrix}$$

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} \rightarrow m1$$

$$\text{ref}(m1) \quad \begin{bmatrix} a & b \\ 1 & \frac{d}{c} \\ 0 & 1 \end{bmatrix}$$

$$\text{ref} \left( \begin{bmatrix} a & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \right) | a=0 \quad \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}$$

**Huomaa:** Katso myös `rref()`, page 169.

## RefreshProbeVars

## RefreshProbeVars

Päiset mittausarvoon kaikista TI-Basic-ohjelmaan kytketyistä antureista.

StatusVar Value	Tila
<i>statusVar</i> =0	Normaali (jatka ohjelmalla)
	Vernier DataQuest™ on tiedonkeruutilassa.
<i>statusVar</i> =1	<b>Huomaa:</b> Vernier DataQuest™ -sovelluksen on oltava metrisessä tilassa, jotta tämä komento toimisi. 
<i>statusVar</i> =2	Vernier DataQuest™ -sovellusta ei ole käynnistetty.
<i>statusVar</i> =3	The Vernier DataQuest™ -sovellus on käynnistetty, mutta et ole yhdistänyt mitään antureita.

## Esimerkki

```
Lämmön määrittys() =
Prgm
© Tarkista, onko järjestelmä
valmiina
RefreshProbeVars-tila
Jos tila=0, niin
Käyttötila "valmis"
Muuttujalle n,1,50
RefreshProbeVars-tila
lämpötila:=mittari.lämpötila
Disp "Lämpötila: ",lämpötila
Jos lämpötila>30, niin
Disp "Liian kuuma"
LopetaJos
© Odota 1 sekunti otoksien
välillä
Odota 1
LopetaKoska
Tai
Disp "Ei valmis". Yritä
myöhemmin uudelleen"
EndIf
EndPrgm
```

Huomaa: Tätä voi käyttää myös TI-Innovator™<sub>i</sub>-laitteen keskiössä.

## remain()

**remain(Laus1, Laus2)** ⇒ lauseke

**remain(Lista1, Lista2)** ⇒ lista

**remain(Matriisi1, Matriisi2)** ⇒ matriisi

Laskee ensimmäisen argumentin jäännöksen toisen argumentin suhteen seuraavien identtisten yhtälöiden määritelmien mukaisesti:

$\text{remain}(x,0) = x$

$\text{remain}(x,y) = x - y \cdot \text{iPart}(x/y)$

Tämän seurauksena huomaa, että **remain(-x,y) = remain(x,y)**. Vastaus on joko nolla tai samanmerkkinen kuin ensimmäinen argumentti.

**Huomaa:** Katso myös **mod()**, sivu 125.

$\text{remain}(7,0)$	7
$\text{remain}(7,3)$	1
$\text{remain}(-7,3)$	-1
$\text{remain}(7,-3)$	1
$\text{remain}(-7,-3)$	-1
$\text{remain}(\{12,-14,16\},\{9,7,-5\})$	$\{3,0,1\}$

$\text{remain}\left(\begin{bmatrix} 9 & -7 \\ 6 & 4 \end{bmatrix}, \begin{bmatrix} 4 & 3 \\ 4 & -3 \end{bmatrix}\right)$	$\begin{bmatrix} 1 & -1 \\ 2 & 1 \end{bmatrix}$
--	---

## Pyydä

**Pyydä** *promptString*, *muutt* [, *NäytäLippu* [, *statusVar*]]

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

Ohjelmointikomento: Keskeyttää ohjelman ja näyttää valintaruudun, jossa on viesti *kehotemerkkijono* sekä syöttöruutu käyttäjän antamaa vastausta varten.

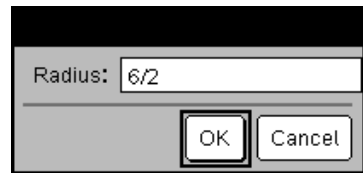
Kun käyttäjä kirjoittaa vastauksen ja klikkaa **OK**-painiketta, syöttöruudun sisältö määrätään muuttujalle *var*.

Määrittele ohjelma:

```
Määrittele pyydä_demo()=Ohjelma
Pyydä "Säde: ",r
Disp "Ala = ",pi*r^2
EndPrgm
```

Aja ohjelma ja kirjoita vastaus:

Pyydä\_demo()



Vastaus **OK**-näppäimen painamisen jälkeen:

Jos käyttäjä klikkaa **Cancel**, ohjelma etenee hyväksymättä mitään syötteitä. Ohjelma käyttää muuttujan *muutt* aikaisempaa arvoa, jos *var* on jo määritelty.

Säde: 6/2  
Ala = 28.2743

Valinnainen *NäytäLippu* -argumentti voi olla mikä tahansa lauseke.

- Jos *NäytäLippu* jätetään pois, tai jos se sieventyy arvoon **1**, kehoteviesti ja käyttäjän vastaus näkyvät laskimen historiassa.
- Jos *NäytäLippu* sieventyy arvoon **0**, kehote ja vastaus eivät näy historiassa.

Valinnainen *statusVar* argumentti laskee ohjelmalle tavan määrittellä, kuinka käyttäjä on sulkenut valintaikkunan. Huomaa, että *statusVar* vaatii *NäytäLippu* -argumentin.

- Jos käyttäjä on klikannut **OK** tai painanut **Enter** tai **Ctrl+Enter**, muuttuja *statusVar* asetetaan arvoon **1**.
- Muuten muuttujastatusVar asetetaan arvoon **0**.

*func()*-argumentin avulla ohjelma voi tallentaa käyttäjän vastauksen funktion määritelmäksi. Tämä syntaksi toimii ikään kuin käyttäjä suorittaisi komennon:

Määrittele *func(arg1, ...argn) = käyttäjän vastaus*

Sen jälkeen ohjelma voi käyttää määriteltyä funktiota *func()*. Komennon *promptString* pitäisi opastaa käyttäjää syöttämään sopiva *käyttäjän vastaus*, joka täydentää funktion määritelmän.

**Huomaa:** Voit käyttää Pyydä komento käyttäjän määrittelemässä ohjelmassa, mutta ei funktiossa.

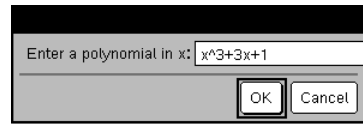
Pysäytä ohjelma, joka sisältää komennon **Request** loputtoman silmukan sisällä:

Määrittele ohjelma:

```
Määrittele polynomi()=Ohjelma
  Pyyntö "Syötä polynomi x:ään:",p(x)
  Disp "Reaalilukujuuret
ovat:",polyRoots(p(x),x)
EndPrgm
```



Aja ohjelma ja kirjoita vastaus:

polynomi()



Tulos, kun  $x^3+3x+1$  on syötetty ja valittu **OK**:

Reaalilukujuuret ovat:  $\{-0.322185\}$

- **Kämmenlaite:** Pidä -painiketta painettuna ja paina toistuvasti -painiketta.
- **Windows®:** Pidä **F12**-näppäintä pohjassa ja paina toistuvasti **Enter**.
- **Macintosh®:** Pidä **F5**-näppäintä pohjassa ja paina toistuvasti **Enter**.
- **iPad®:** Sovellus näyttää kehotuksen. Voit jatkaa odottamista tai peruuttaa.

**Huomaa:** Katso myös **RequestStr**, page 163.



## RequestStr

**RequestStr** *kehotemerkkijono, muutt[, NäytäLippu]*

Ohjelmointikomento: Toimii samalla tavalla kuin **Request**-komennon ensimmäinen syntaksi paitsi, että käyttäjän vastaus tulkitaan aina merkkijonoksi. Vastakohtaisesti **Request**-komento tulkitsee vastauksen lausekkeeksi, jollei käyttäjä merkitse sitä lainausmerkkien ("") sisään.

Huomaa: Voit käyttää **RequestStr** -komentoa käyttäjän määrittelemässä ohjelmassa, mutta ei funktiossa.

Pysäytä ohjelma, joka sisältää **RequestStr** komento äärettömässä silmukassa:

- **Kämmenlaite:** Pidä -painiketta painettuna ja paina toistuvasti -painiketta.
- **Windows®:** Pidä **F12**-näppäintä pohjassa ja paina toistuvasti **Enter**.
- **Macintosh®:** Pidä **F5**-näppäintä pohjassa ja paina toistuvasti **Enter**.
- **iPad®:** Sovellus näyttää kehotuksen. Voit jatkaa odottamista tai peruuttaa.

**Huomaa:** Katso myös **Pyydä**, page 161.

Määrittele ohjelma:

```
Määrittele requestStr_demo()=Ohjelma
RequestStr "Oma nimesi:",nimi,0
Disp "Vastauksessa on ",dim(nimi),"
merkkiä."
EndPrgm
```

Aja ohjelma ja kirjoita vastaus:

```
requestStr_demo()
```



Vastaus **OK**-näppäimen painamisen jälkeen (Huomaa, että jos *NäytäLippu*-argumentti on **0**, kehote ja vastaus eivät näy historiassa):

```
requestStr_demo()
```

Vastauksessa on 5 merkkiä.

**Return** [*Laus*]

Laskee lausekkeen *Laus* funktion tuloksena Käytetään lohkon **Func...EndFunc** sisällä.

**Huomaa:** Käytä **Return** ilman argumenttia lohkoissa **Prm...EndPrm** poistuaaksesi ohjelmasta.

**Huomaa esimerkkiä syöttäessäsi:** Ohjeet monirivisten ohjelmien ja funktion määritysten syöttämisestä löytyvät tuotteen ohjekirjan Laskin-osiosta.

```
Define factorial (nn)=
Func
Local answer,counter
1 → answer
For counter,1,nn
answer·counter → answer
EndFor
Return answer|
EndFunc
```

*factorial* (3)

6

**right()**

**oikea**(*List1* [, *Num*]) ⇒ *lista*

Laskee oikeanpuoleisimmat *Num*-alkutekijät, jotka sisältyvät listaan *List1*.

Jos jätät pois komennon *Num*, laskee kaiken listasta *List1*.

**right**(*lähdemerkkijono* [, *Num*]) ⇒ *merkkijono*

Laskee oikeanpuoleisimmat *Num*-merkit, jotka sisältyvät merkkijonoon *lähdemerkkijono*

Jos jätät pois komennon *Num*, laskee kaiken *lähdemerkkijonosta*.

**right**(*Vertailu*) ⇒ *lauseke*

Laskee yhtälön tai epäyhtälön oikean puolen.

*right*{ {1,3,-2,4},3} {3,-2,4}

*right*("Hello",2) "lo"

*right*( $x < 3$ ) 3

**rk23 ()**

**rk23**(*Laus*, *Var*, *depVar*, {*Var0*, *VarMax*}, *depVar0*, *VarStep* [, *diftol*]) ⇒ *matriisi*

**rk23**(*SystemOfExpr*, *Var*, *ListOfDepVars*, {*Var0*, *VarMax*}, *ListOfDepVars0*, *VarStep* [, *diftol*]) ⇒ *matriisic*

Differentiaaliyhtälö:

$y' = 0.001 \cdot y \cdot (100 - y)$  ja  $y(0) = 10$

*rk23*{ $0.001 \cdot y \cdot (100 - y)$ ,  $t, y$ , {0,100}, 10, 1}

0.	1.	2.	3.	4.
10.	10.9367	11.9493	13.042	14.2



**rk23**(*ListOfExpr*, *Var*, *ListOfDepVars*, {*Var0*, *VarMax*}, *ListOfDepVars0*, *VarStep*, *diftol*) ⇒ *matriisi*

Käyttää Runge-Kutta-menetelmää järjestelmän ratkaisuun

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

funktiolla *depVar*(*Var0*)=*depVar0* välillä [*Var0*,*VarMax*]. Laskee tulokseksi matriisin, jonka ensimmäinen rivi määrittelee *Var* tulosarvot muuttujan *VarStep* määrittelemällä tavalla. Toinen rivi määrää ensimmäisen ratkaisukomponentin arvon vastaavissa *Var*-arvoissa, jne.

*Expr* on oikea puoli, joka määrittelee tavallisen differentiaaliyhtälön (ODE).

*SystemOfExpr* on oikeiden puolten ryhmä, joka määrittelee ODE-yhtälöiden järjestelmän (vastaa riippuvaisten muuttujien järjestystä kohdassa *ListOfDepVars*).

*ListOfExpr* on oikeiden puolten luettelo, joka määrittelee ODE-yhtälöiden järjestelmän (vastaa riippuvaisten muuttujien järjestystä kohdassa *ListOfDepVars*).

*Var* on riippumaton muuttuja.

*ListOfDepVars* on riippuvaisten muuttujien luettelo.

{*Var0*, *VarMax*} on kahden elementin lista, joka määrää funktion integroimaan muuttujasta *Var0* to *VarMax*.

*ListOfDepVars0* on lista riippuvaisten muuttujien alkuarvoja.

Jos haluat nähdä koko vastauksen, paina ▲ ja siirrä sen jälkeen osoitinta painikkeilla ◀ ja ▶.

Sama yhtälö, jossa *diftol* on asetettu 1.E-6

$$\text{rk23}\left(0.001 \cdot y \cdot (100 - y), t, y, \{0, 100\}, 10, 1, 1.E-6\right)$$

0.	1.	2.	3.	4.
10.	10.9367	11.9495	13.0423	14.2189

Vertaile edellä olevaa tulosta CAS:n täsmälliseen tulokseen, joka on saatu käyttämällä deSolve()- ja seqGEN()-funktioita:

$$\text{deSolve}(y' = 0.001 \cdot y \cdot (100 - y) \text{ and } y(0) = 10, t, y)$$

$$y = \frac{100 \cdot (1.10517)^t}{(1.10517)^t + 9}$$

$$\text{seqGen}\left(\frac{100 \cdot (1.10517)^t}{(1.10517)^t + 9}, t, y, \{0, 100\}\right)$$

$$\{10., 10.9367, 11.9494, 13.0423, 14.2189, 15.4\}$$

Yhtälöryhmä:

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

kun  $y1(0) = 2$  and  $y2(0) = 5$

$$\text{rk23}\left(\begin{cases} y1' + 0.1 \cdot y1 \cdot y2 \\ 3 \cdot y2 - y1 \cdot y2 \end{cases}, t, \{y1, y2\}, \{0, 5\}, \{2, 5\}, 1\right)$$

0.	1.	2.	3.	4.
2.	1.94103	4.78694	3.25253	1.82848
5.	16.8311	12.3133	3.51112	6.27245

Jos *VarStep* on arvoltaan nollasta eroava numero: merkki(*VarStep*) = merkki (*VarMax-Var0*) ja ratkaisut lasketaan muuttujalla  $Var0+i*VarStep$  kaikille  $i=0,1,2,\dots$  niin, että  $Var0+i*VarStep$  on alueella [*muutt0*,*VarMax*] (muuttujalla *VarMax* ei ehkä ole ratkaisuarvoa).

jos *VarStep* sievenee nollassi, ratkaisut lasketaan "Runge-Kutta" -muuttujan *Muutt* arvoilla.

*diftol* on virhetoleranssi (oletuksena 0.001).

**root()**

**root(Laus) ⇒ juuri**

**root(Laus1, Laus2) ⇒ juuri**

**root(Laus)** laskee neliöjuuren lausekkeelle *Laus*.

**root(Laus1, Laus2)** laskee lausekkeena *Laus2* lausekkeen *Laus1* juuren. *Laus1* voi olla reaalinen tai kompleksinen liukulukuvakio, kokonaisluku tai kompleksinen rationaalilukuvakio tai yleinen symbolinen lauseke.

**Huomaa:** Katso myös **N:s juuri -malli**, sivu 2.

$\sqrt[3]{8}$	2
$\sqrt[3]{3}$	$\frac{1}{3^3}$
$\sqrt[3]{3}$ .	1.44225

**rotate()**

**rotate(Kokonaisluku1[,KiertojenLkm]) ⇒ kokonaisluku**

Binaarisessa kantaluutilassa:

```
rotate(0b11111111111111111111111111111111)
0b10000000000000000000000000000000000001P
rotate(256,1)                                0b1000000000
```

Kiertää bittejä binaarisessa kokonaisluvussa. Voit syöttää luvun *Kokonaisluku1* minä tahansa kantalukena; se muunnetaan automaattisesti 64 bitin binaarimuotoon. Jos *Kokonaisluku1* on liian suuri tälle muodolle, symmetrinen modulo-operaatio sovittaa sen alueelle sopivaksi Katso lisätietoa kohdasta ▶ **Kantaluku2**, sivu 18.

Jos haluat nähdä koko vastauksen, paina ▶ ja siirrä sen jälkeen osoitinta painikkeilla ◀ ja ▶.

**rotate()**

Luettelo &gt;

Jos *KiertojenLkm* on positiivinen, kierto on vasemmalle. Jos *KiertojenLkm* on negatiivinen, kierto on oikealle. Oletusarvo on  $-1$  (kierrä oikealle yksi bitti).

Esimerkki, kierrossa oikealle:

Jokainen bitti kiertyy oikealle.

0b00000000000001111010110000110101

Oikeanpuoleisin bitti siirtyy vasemmaisimmaksi.

tuottaa:

0b10000000000001111010110000110101

Vastaus näkyy kantalukutilan mukaisesti.

**rotate(ListaI[,KiertojenLkm])** ⇒ lista

Laskee *Listal:n* kopion, jota on kierretty oikealle tai vasemmalle *KiertojenLkm:n* elementtien määrittämän määrän. Ei muuta *Listaal*.

Jos *KiertojenLkm* on positiivinen, kierto on vasemmalle. Jos *KiertojenLkm* on negatiivinen, kierto on oikealle. Oletusarvo on  $-1$  (kierrä oikealle yksi elementti).

**rotate(MerkkijonoI[,KiertojenLkm])** ⇒ merkkijono

Laskee *MerkkijonoI:n* kopion, jonka on kiertänyt oikealle tai vasemmalle *KiertojenLkm:n* merkkien määrittämän määrän. Ei muuta *MerkkijonoI:tä*

Jos *KiertojenLkm* on positiivinen, kierto on vasemmalle. Jos *KiertojenLkm* on negatiivinen, kierto on oikealle. Oletusarvo on  $-1$  (kierrä oikealle yksi merkki).

Heksadesimaalisessa kantalukutilassa:

rotate(0h78E)	0h3C7
rotate(0h78E,-2)	0h80000000000001E3
rotate(0h78E,2)	0h1E38

**Tärkeää:** Binaariluvun edelle tulee aina merkitä etumerkki 0b ja heksadesimaaliluvun edelle 0h (nolla, ei 0-kirjain).

Desimaalisessa kantalukutilassa:

rotate({1,2,3,4})	{4,1,2,3}
rotate({1,2,3,4},-2)	{3,4,1,2}
rotate({1,2,3,4},1)	{2,3,4,1}

rotate("abcd")	"dabc"
rotate("abcd",-2)	"cdab"
rotate("abcd",1)	"bcda"

**round()**

Luettelo &gt;

**round(LausI[, numeroa])** ⇒ lauseke

round(1.234567,3)	1.235
-------------------	-------

Laskee argumentin pyöristettynä jättäen desimaalipisteen jälkeen määrätyn määrän numeroita.

## round()

Luettelo > 

*numeroiden* on oltava kokonaisluku välillä 0–12. Jos *numeroita* ei sisällytetä, laskee argumentin pyöristettynä 12 merkitykselliseen numeroon.

**Huomaa:** Näytettävät numerot -tila saattaa vaikuttaa näyttöön.

**round(Lista1[, numeroit])** ⇒ lista

Pyöristää alkutekijöiden listan jättäen desimaalipisteen jälkeen määrätyn määrän numeroita.

**round(Matriisi1[, numeroit])** ⇒ matriisi

Pyöristää alkutekijöiden matriisin jättäen desimaalipisteen jälkeen määrätyn määrän numeroita.

$$\text{round}(\{\pi, \sqrt{2}, \ln(2)\}, 4) \\ \{3.1416, 1.4142, 0.6931\}$$

$$\text{round}\left(\begin{pmatrix} \ln(5) & \ln(3) \\ \pi & e^1 \end{pmatrix}, 1\right) \quad \begin{bmatrix} 1.6 & 1.1 \\ 3.1 & 2.7 \end{bmatrix}$$

## rowAdd()

Luettelo > 

**rowAdd(Matriisi1, rIndeksi1, rIndeksi2)**  
⇒ matriisi

Laskee matriisiin *Matriisi1* kopion rivin *rIndeksi2* korvattuna rivien summalla *rIndeksi1* ja *rIndeksi2*.

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

$$\text{rowAdd}\left(\begin{pmatrix} a & b \\ c & d \end{pmatrix}, 1, 2\right) \quad \begin{bmatrix} a & b \\ a+c & b+d \end{bmatrix}$$

## rowDim()

Luettelo > 

**rowDim(Matriisi)** ⇒ lauseke

Laskee *Matriisin* sisältämien rivien lukumäärän.

**Huomaa:** Katso myös **colDim()**, sivu 28.

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix} \rightarrow m1 \quad \begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix} \\ \text{rowDim}(m1) \quad 3$$

## rowNorm()

Luettelo > 

**rowNorm(Matriisi)** ⇒ lauseke

Laskee *Matriisin* riveillä olevien elementtien itseisarvojen summien maksimin.

**Huomaa:** Kaikkien matriisien alkutekijöiden on sievennyttävä luvuiksi. Katso myös **colNorm()**, sivu 28.

$$\text{rowNorm}\left(\begin{pmatrix} -5 & 6 & -7 \\ 3 & 4 & 9 \\ 9 & -9 & -7 \end{pmatrix}\right) \quad 25$$

## rowSwap()

Luettelo >

**rowSwap**(*Matriisi1*, *rIndeksi1*,  
*rIndeksi2*) ⇒ *matriisi*

Laskee matriisiin *Matriisi1*, jossa rivien *rIndeksi1* ja *rIndeksi2* paikkoja on vaihdettu.

$\begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix} \rightarrow mat$	$\begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix}$
rowSwap(mat,1,3)	$\begin{bmatrix} 5 & 6 \\ 3 & 4 \\ 1 & 2 \end{bmatrix}$

## rref()

Luettelo >

**rref**(*Matriisi1*[, *Tol*]) ⇒ *matriisi*

Laskee matriisiin *Matriisi1* pelkistetyn rivi-echelon-muodon.

$rref\left(\begin{bmatrix} -2 & -2 & 0 & -6 \\ 1 & -1 & 9 & -9 \\ -5 & 2 & 4 & -4 \end{bmatrix}\right)$	$\begin{bmatrix} 1 & 0 & 0 & \frac{66}{71} \\ 0 & 1 & 0 & \frac{147}{71} \\ 0 & 0 & 1 & \frac{-62}{71} \end{bmatrix}$
---	---

Valinnaisesti kaikkia matriisin elementtejä käsitellään nollana, jos niiden itseisarvo on pienempi kuin *Tol*. Tätä toleranssia käytetään vain, jos matriisissa on liukulukusyötteitä eikä se sisällä symbolisia muuttujia, joille ei ole määritelty arvoa. Muussa tapauksessa kommentoa *Tol* ei huomioida.

$rref\left(\begin{bmatrix} a & b \\ c & d \end{bmatrix}\right)$	$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$
---	--

- Jos käytät painikkeita tai **Auto or Approximate** -tilan valintaa Approximate (Likimääräinen), laskenta suoritetaan liukulukuaritmetiikalla.
- Jos *Tol* jätetään pois tai sitä ei käytetä, oletusarvoinen toleranssi lasketaan seuraavasti:  
 $5E-14 \cdot \max(\dim(\text{Matriisi1}))$   
 $\bullet \text{rowNorm}(\text{Matriisi1})$

**Huomaa:** Katso myös **ref()**, page 159.

## S

### sec()

-painike

**sec**(*Laus1*) ⇒ *lauseke*

Astekulmatilassa:

**sec**(*Lista1*) ⇒ *lista*

**sec()**

trig -painike

Laskee *Laus1*:n sekantin tai määrittää listan, joka sisältää *Listal*:n kaikkien elementtien sekantit.

$$\frac{\sec(45)}{\sec(\{1, 2, 3, 4\})} \sqrt{2} \left\{ \frac{1}{\cos(1)}, 1.00081, \frac{1}{\cos(4)} \right\}$$

**Huomaa:** Argumentti tulkitaan aste-, graadi- tai radiaanikulmaksi käytössä olevan kulmatila-asetuksen mukaisesti. Voit ohittaa kulmatilan väliaikaisesti painikkeilla °, G tai r.

**sec<sup>-1</sup>()**

trig -painike

sec<sup>-1</sup>(*Laus1*) ⇒  
lauseke

Astekulmatilassa:

$$\sec^{-1}(1) \quad 0$$

sec<sup>-1</sup>(*Listal*) ⇒ lista

Määrittää kulman, jonka sekantti on *Laus1*, tai määrittää listan, joka sisältää *Listal*:n kaikkien elementtien käänteissekantit.

Graadikulmatilassa:

$$\sec^{-1}(\sqrt{2}) \quad 50$$

Radiaanikulmatilassa:

$$\sec^{-1}(\{1, 2, 5\}) \quad \left\{ 0, \frac{\pi}{3}, \cos^{-1}\left(\frac{1}{5}\right) \right\}$$

**Huomaa:** Vastaus lasketaan aste-, graadi- tai radiaanikulmana käytössä olevan kulmatila-asetuksen mukaisesti.

**Huomaa:** Voit syöttää tämän funktion näppäimistöltä kirjoittamalla **arcsec (...)**.

## sech()

Katalogi > 

**sech(Laus1)** ⇒ lauseke

**sech(Lista1)** ⇒ lista

Laskee *Laus1*:n hyperbolisen sekantin tai määrittää listan, joka sisältää *Listal*:n kaikkien elementtien hyperboliset sekantit.

$\text{sech}(3)$	$\frac{1}{\cosh(3)}$
$\text{sech}(\{1,2,3,4\})$	$\left\{ \frac{1}{\cosh(1)}, 0.198522, \frac{1}{\cosh(4)} \right\}$

## sech<sup>-1</sup>()

Katalogi > 

**sech<sup>-1</sup>(Laus1)** ⇒ lauseke

**sech<sup>-1</sup>(Lista1)** ⇒ lista

Laskee *Laus1*:n käänteisen hyperbolisen sekantin tai määrittää listan, joka sisältää *Listal*:n kaikkien elementtien käänteiset hyperboliset sekantit.

**Huomaa:** Voit syöttää tämän funktion näppäimistöä kirjoittamalla **arcsech** (...).

Radiaanikulma- ja suorakulmakompleksitilassa:

$\text{sech}^{-1}(1)$	0
$\text{sech}^{-1}(\{1, -2, 2, 1\})$	$\left\{ 0, \frac{2 \cdot \pi}{3} \cdot i, 8 \cdot \text{E-}15 + 1.07448 \cdot i \right\}$

## Send

Laite-valikko

**Send lausTaiMerkkijono1 [, lausTaiMerkkijono2] ...**

Ohjelmointikomento: Lähetää yhden tai useamman TI-Innovator™ Hub komennon liitettyyn laitteeseen.

*lausTaiMerkkijonon* täytyy olla validi TI-Innovator™ Hub Komento. Tavallisesti *lausTaiMerkkijono* sisältää "SET ..."komento laitteen tai "READ ..." hallitsemiseksi komento tiedon kysymiseksi.

Argumentit lähetetään laitteeseen peräkkäisinä.

**Huomio:** Komentoa **Send** voi käyttää käyttäjän määrittelemän ohjelman sisällä mutta ei funktion sisällä.

**Huomio:** Katso myös **Get** (sivu 84), **GetStr** (sivu 91), ja **eval()** (sivu 66).

Esimerkki: Sytytä sisäänrakennetun RGB-ledin sininen väri 0,5 sekunniksi.

```
Send "SET COLOR.BLUE ON TIME .5"
Done
```

Esimerkki: Kysy laitteen sisäänrakennetun valotasoaunturin tämänhetkinen arvo. **Get**-komento noutaa arvon ja sijoittaa sen muuttujaan *lightval*.

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

Esimerkki: Lähetä laskettu taajuus laitteen sisäänrakennettuun kaiuttimeen. Käytä erikoismuuttujaa *iostr.SendAns* näyttääksesi laitekomennon ilmaus arvioituna.

$n:=50$	50
$m:=4$	4
Send "SET SOUND eval(m · n)"	Done
<i>iostr.SendAns</i>	"SET SOUND 200"

## seq() (sekv)

Katalogi &gt;

**seq**(*Laus*, *Muutt*, *Matala*, *Korkea* [, *Askel*]) ⇒ lista

Lisää muuttujan *Var* arvoa arvosta *Low* arvoon *High* välillä *Step*, laskee *Expr*, ja antaa tulokset luettelona. Muuttujan *Var* alkuperäinen sisältö on edelleen tallessa funktion **seq()** suorittamisen jälkeen.

Oletusarvo *Step* = 1.

$\text{seq}\left(n^2, n, 1, 6\right)$	$\{1, 4, 9, 16, 25, 36\}$
$\text{seq}\left(\frac{1}{n}, n, 1, 10, 2\right)$	$\left\{1, \frac{1}{3}, \frac{1}{5}, \frac{1}{7}, \frac{1}{9}\right\}$
$\text{sum}\left(\text{seq}\left(\frac{1}{n^2}, n, 1, 10, 1\right)\right)$	$\frac{1968329}{1270080}$

**Huom:** Vastauksen pakottaminen likimääräiseksi:

**Kämmenlaite:** Paina  .

**Windows®:** Paina **Ctrl+Enter**.

**Macintosh®:** Paina **⌘+Enter**.

**iPad®:** Pidä **enter** ja valitse .

$\text{sum}\left(\text{seq}\left(\frac{1}{n^2}, n, 1, 10, 1\right)\right)$	1.54977
--	---------

## seqGen()

Katalogi &gt;

**seqGen**(*Expr*, *Var*, *depVar*, {*Var0*, *VarMax*}, [*ListOfInitTerms* [, *VarStep* [, *CeilingValue*]]) ⇒ lista

Luo termiluettelon sekvenssille *depVar* (*Var*)=*Expr* seuraavasti: Lisää riippumattoman muuttujan *Var* arvoa arvosta *Var0* arvoon *VarMax* portailla *VarStep*, laskee *depVar*(*Var*) muuttujan *Var* vastaaville arvoille käyttäen *Expr*-kaavaa ja *ListOfInitTerms* -luetteloa, ja antaa tulokset luettelona

**seqGen**(*ListOrSystemOfExpr*, *Var*, [*ListOfDepVars*, {*Var0*, *VarMax*}] [, *MatrixOfInitTerms* [, *VarStep* [, *CeilingValue*]]) ⇒ matrix

Luo sekvenssin 5 ensimmäistä termiä  $u(n) = u(n-1)^2/2$ , kun  $u(1)=2$  ja  $VarStep=1$ .

$\text{seqGen}\left(\frac{(u(n-1))^2}{n}, n, u, \{1, 5\}, \{2\}\right)$	$\left\{2, 2, \frac{4}{3}, \frac{4}{9}, \frac{16}{405}\right\}$
---	---



Luo termimatriisin sekvenssien *ListOfDepVars* (*Var*)=*ListOrSystemOfExpr* ryhmälle (tai listalle) seuraavasti: Lisää riippumattoman muuttujan *Var* arvoa arvosta *Var0* arvoon *VarMax* portailla *VarStep*, laskee *ListOfDepVars*(*Var*) muuttujan *Var* vastaaville arvoille käyttäen *Expr*-kaavaa ja *MatrixOfInitTerms* -luetteloa, ja antaa tulokset matriisina.

Muuttujan *Var* alkuperäinen sisältö on edelleen tallessa funktion **seqGen()** suorittamisen jälkeen

Oletusarvo *VarStep* = 1.

Esimerkki, jossa *Var0*=2:

$$\text{seqGen}\left(\frac{u(n-1)+1}{n}, n, u, \{2,5\}, \{3\}\right)$$


---


$$\left\{3, \frac{4}{3}, \frac{7}{12}, \frac{19}{60}\right\}$$

Esimerkki, jossa aloitustermi on symbolinen:

$$\text{seqGen}(u(n-1)+2, n, u, \{1,5\}, \{a\})$$


---


$$\{a, a+2, a+4, a+6, a+8\}$$

Kahden sekvenssin ryhmä:

$$\text{seqGen}\left(\left\{\frac{1}{n}, \frac{u2(n-1)}{2} + u1(n-1)\right\}, n, \{u1, u2\}, \{1,5\}, \left[\begin{array}{c} - \\ 2 \end{array}\right]\right)$$


---


$$\left[\begin{array}{ccccc} 1 & \frac{1}{2} & \frac{1}{3} & \frac{1}{4} & \frac{1}{5} \\ 2 & 2 & \frac{3}{2} & \frac{13}{12} & \frac{19}{24} \end{array}\right]$$

Huomaa: Muuttujaa Tyhjä ( ) yllä olevassa aloitustermimatriisissa käytetään ilmoittamaan, että *u1*(*n*):n aloitustermi on laskettu käyttämällä täsmällistä sekvenssikaavaa *u1*(*n*)=1/*n*.

## seqn()

**seqn**(*Expr*(*u*, *n* [, *ListOfInitTerms* [, *nMax* [, *CeilingValue*]]])⇒*list*

Luo termiluettelon sekvenssille *u* (*n*)=*Expr*(*u*, *n*) seuraavasti: Lisää muuttujan *n* arvoa arvosta 1 arvoon *nMax* 1 välein, laskee lausekkeen *u*(*n*) muuttujan *n* vastaaville arvoille käyttäen *Expr*(*u*, *n*) -kaavaa ja *ListOfInitTerms* -luetteloa ja antaa tulokset luettelona.

**seqn**(*Expr*(*n* [, *nMax* [, *CeilingValue*]]])⇒*list*

Luo sekvenssin 6 ensimmäistä termiä *u*(*n*) = *u* (*n*-1)/2, kun *u*(1)=2.

$$\text{seqn}\left(\frac{u(n-1)}{n}, \{2\}, 6\right)$$


---


$$\left\{2, 1, \frac{1}{3}, \frac{1}{12}, \frac{1}{60}, \frac{1}{360}\right\}$$

$$\text{seqn}\left(\frac{1}{n^2}, 6\right)$$


---


$$\left\{1, \frac{1}{4}, \frac{1}{9}, \frac{1}{16}, \frac{1}{25}, \frac{1}{36}\right\}$$

Luo termiluettelon ei-rekursiiviselle sekvenssille  $u(n)=Expr(u, n)$  seuraavasti: Lisää muuttujan  $n$  arvoa arvosta 1 arvoon  $nMax$  1 välein, laskee lausekkeen  $u(n)$  muuttujan  $n$  vastaaville arvoille käyttäen  $Expr(n)$  -kaavaa ja antaa tulokset luettelona.

Jos arvo  $nMax$  puuttuu,  $nMax$  asetetaan arvoon 2500.

Jos  $nMax=0$ ,  $nMax$  asetetaan arvoon 2500.

**Huomaa:** `seqn()` hakee funktion `seqGen( )` kun  $n0=1$  ja  $nstep = 1$

## series()

`series(Lausl, Muutt, Aste [, Piste])` ⇒ muuttuja

`series(Lausl, Muutt, Aste [, Piste]) | Muutt > Piste` ⇒ lauseke

`series(Lausl, Muutt, Aste [, Piste]) | Muutt < Piste` ⇒ lauseke

Laskee yleisen tyypistetyt potenssisarjaesityksen *Lausl*:stä, joka on lavennettu *Piste*en ympärillä asteen *Aste* kautta. *Aste* voi olla mikä tahansa rationaaliluku. Lausekkeen (*Muutt* - *Piste*) tuloksena olevilla potensseilla voi olla negatiiviset ja/tai murtolukuekspONENTIT. Näiden potenssien kertoimet voivat sisältää lausekkeen (*Muutt* - *Piste*) logaritmeja ja muita *Muutt*:n funktioita, joita hallitsevat kaikki lausekkeen (*Muutt* - *Piste*) potenssit, joilla on sama eksponentin etumerkki.

*Piste*en oletusarvo on 0. *Piste* voi olla  $\infty$  tai  $-\infty$ , jolloin lavennus tapahtuu asteen *Aste* kautta lausekkeessa  $1/(Muutt - Piste)$ .

$$\text{series}\left(\frac{1-\cos(x-1)}{(x-1)^2}, x, 4, 1\right) \quad \frac{1}{2} \frac{(x-1)^2}{24} + \frac{(x-1)^4}{720}$$

$$\text{series}\left(\frac{-1}{e^{z-}}, z, -1\right) \quad z - 1$$

$$\text{series}\left(\left(1 + \frac{1}{n}\right)^n, n, 2, \infty\right) \quad e - \frac{e}{2 \cdot n} + \frac{11 \cdot e}{24 \cdot n^2}$$

$$\text{series}\left(\tan^4\left(\frac{1}{x}\right), x, 5\right), x > 0 \quad \frac{\pi}{2} - x + \frac{x^3}{3} - \frac{x^5}{5}$$

$$\text{series}\left(\int \frac{\sin(x)}{x} dx, x, 6\right) \quad x - \frac{x^3}{18} + \frac{x^5}{600}$$

$$\text{series}\left(\int_0^x \sin(x \cdot \sin(t)) dt, x, 7\right) \quad \frac{x^3}{2} - \frac{x^5}{24} + \frac{29 \cdot x^7}{720}$$

$$\text{series}\left(\left(1 + e^x\right)^2, x, 2, 1\right) \quad (e+1)^2 + 2 \cdot e \cdot (e+1) \cdot (x-1) + e \cdot (2 \cdot e+1) \cdot (x-1)^2$$

**series(...)** antaa tuloksena "**series(...)**", ellei se pysty määrittämään tällaista esitystä, kuten olennaisille erikoispisteille, esim.  $\sin(1/z)$ , kun  $z=0$ ,  $e^{-1/z}$ , kun  $z=0$ , tai  $e^z$ , kun  $z = \infty$  tai  $-\infty$ .

Jos sarjassa tai yhdellä sen derivaatoista on hyppypäjätkuvuus kohdassa *Piste*, tulos sisältää todennäköisesti alalausekkeita, jotka ovat muotoa  $\text{sign}(\dots)$  tai  $\text{abs}(\dots)$  reaalille kehitelmän muuttujalle tai  $(-1)^{\text{floor}(\dots \text{angle}(\dots) \dots)}$  kompleksille kehitelmän muuttujalle, joka on merkkiin " $_$ " päättyvä muuttuja. Jos tarkoituksesi on käyttää sarjaa vain *Pisteen* toisella puolella oleville arvoille, siinä tapauksessa liitä sopiva lauseke, " $| \text{Muutt} > \text{Piste}$ ", " $| \text{Muutt} < \text{Piste}$ ", " $| \text{Muutt} \geq \text{Piste}$ " tai " $| \text{Muutt} \leq \text{Piste}$ ", jotta saat yksinkertaisemman vastauksen.

**series()** voi antaa symbolisia likiarvoja määrittämättömille integraaleille ja määrättyille integraaleille, joille ei muuten voida saada symbolisia ratkaisuja.

**series()** jakautuu 1. argumentin listoihin ja matriiseihin.

**series()** on **taylor()**-funktion yleistetty versio.

Kuten on havainnollistettu oikeanpuoleisessa viimeisessä esimerkissä, tuloksen alapuolella olevat **series(...)**-funktion tuottamat näyttörutiinit voivat järjestää termit uudelleen siten, että dominanttitermi ei ole vasemmanpuoleisin.

**Huomaa:** Katso myös **dominantTerm()**, sivu 60.

**setMode**(*tilaNimiKokonaisluku*,  
*asetusKokonaisluku*) ⇒ *kokonaisluku*  
**setMode**(*lista*) ⇒ *kokonaislukulista*

Toimii vain funktiossa tai ohjelmassa.

**setMode**(*tilaNimiKokonaisluku*,  
*asetusKokonaisluku*) asettaa tilan  
*tilaNimiKokonaisluku* väliaikaisesti  
uuteen asetukseen *asetusKokonaisluku*  
ja määrittää kokonaisluvun, joka vastaa  
kyseisen tilan alkuperäistä asetusta.  
Muutos on rajoitettu ohjelman/funktion  
suorittamisen ajalle.

*tilaNimiKokonaisluku* määrittää  
asetettavan tilan. Sen on oltava jokin alla  
olevan taulukon tilaa kuvaavista  
kokonaisluvuista.

*asetusKokonaisluku* määrittää tilan  
uuden asetuksen. Sen on oltava jokin  
seuraavista asetettavalle tilalle  
varatuista asetusta kuvaavista  
kokonaisluvuista.

**setMode**(*lista*)-komennolla voit muuttaa  
useita asetuksia. *lista* sisältää tilaa ja  
asetusta kuvaavat kokonaislukuparit.  
**setMode**(*lista*) luo samanlaisen listan,  
jonka kokonaislukuparit kuvaavat  
alkuperäisiä tiloja ja asetuksia.

Jos olet tallentanut kaikki tila-asetukset  
komennolla **getMode**(0) → *muutt*, voit  
palauttaa nämä asetukset komennolla  
**setMode**(*muutt*) aina funktion tai  
ohjelman sulkemiseen saakka. Katso  
**getMode**(), sivu 89.

**Huomaa:** Nykyiset tila-asetukset siirtyvät  
haettuihin alarutiineihin. Jos jokin  
alarutiini muuttaa tila-asetusta,  
tilamuutos häviää, kun ohjaus palautuu  
hakurutiiniin.

Näytä  $\pi$ :n likiarvo käyttäen Näytettävät  
numerot -tilan oletusasetusta ja näytä sen  
jälkeen  $\pi$  asetuksella Kiinteä2. Tarkista, että  
oletusarvo palautuu ohjelman suorittamisen  
jälkeen.

Define <i>progI</i> ( <i></i> )=Prgm	<i>Done</i>
Disp approx( $\pi$ )	
setMode(1,16)	
Disp approx( $\pi$ )	
EndPrgm	
<i>progI</i> ( <i></i> )	
	3.14159
	3.14
	<i>Done</i>

**Huomaa esimerkkiä syöttäessäsi:** Ohjeet monirivisten ohjelmien ja funktion määritysten syöttämisestä löytyvät tuotteen ohjekirjan Laskin-osiosta.

Tilan nimi	Tilaa vastaava kokonaisluku	Asetuksia vastaavat kokonaisluvut
Näytettävät numerot	1	1=Liukuva, 2=Liukuva1, 3=Liukuva2, 4=Liukuva3, 5=Liukuva4, 6=Liukuva5, 7=Liukuva6, 8=Liukuva7, 9=Liukuva8, 10=Liukuva9, 11=Liukuva10, 12=Liukuva11, 13=Liukuva12, 14=Kiinteä0, 15=Kiinteä1, 16=Kiinteä2, 17=Kiinteä3, 18=Kiinteä4, 19=Kiinteä5, 20=Kiinteä6, 21=Kiinteä7, 22=Kiinteä8, 23=Kiinteä9, 24=Kiinteä10, 25=Kiinteä11, 26=Kiinteä12
Kulma	2	1=Radiaani, 2=Aste, 3=Graadi
Eksponenttimuoto	3	1=Normaali, 2=Kymmenpotenssi, 3=Tekninen
Reaali- tai kompleksiluku	4	1=Reaali, 2=Suorakulma, 3=Polaarinen
Automaattinen tai likimääräinen.	5	1=Automaattinen, 2=Likimääräinen, 3=Täsmällinen
Vektorimuoto	6	1=Suorakulma, 2=Sylinteri, 3=Pallo
Kantaluku	7	1=Desimaali, 2=Heksagonaalinen, 3=Binaarinen
Yksikköjärjestelmä	8	1=SI, 2=Eng/US

## shift()

**shift**(*Kokonaisluku* [,*SiirtojenLkm*])⇒*kokonaisluku*

Binaarisessa kantalukutilassa:

```

shift(0b1111010110000110101)
                                0b111101011000011010
shift(256,1)                    0b1000000000

```

Heksadesimaalisessa kantalukutilassa:

Siirtää binaarisen kokonaisluvun bittejä. Voit syöttää *Kokonaisluku1*:n minä tahansa kantalukuna; se muunnetaan automaattisesti etumerkilliseen 64 bitin binaarimuotoon. Jos *Kokonaisluku1* on liian suuri tälle muodolle, symmetrinen modulo-operaatio sovittaa sen alueelle sopivaksi. Lisätietoja, katso **Base2**, sivu 18.

Jos *SiirtojenLkm* on positiivinen, siirto tapahtuu vasemmalle. Jos *SiirtojenLkm* on negatiivinen, siirto tapahtuu oikealle. Oletusarvo on -1 (siirrä yksi bitti oikealle).

Oikealle tapahtuvassa siirrossa oikeanpuoleisin bitti pudotetaan, ja 0 ja 1 lisätään vastaamaan vasemmanpuoleista bittiä. Vasemmalle tapahtuvassa siirrossa vasemmanpuoleisin bitti pudotetaan, ja 0 lisätään vastaamaan oikeanpuoleisinta bittiä.

Esimerkki siirrosta oikealle:

Jokainen bitti siirtyy oikealle.

0b0000000000000111101011000011010

Lisää 0:n, jos vasemmanpuoleisin bitti on 0, tai 1:n, jos vasemmanpuoleisin bitti on 1.

tuottaa:

0b00000000000000111101011000011010

Vastaus näkyy kantalukutilan mukaisesti. Alkunollia ei näytetä.

**shift**(*Listal* [,*SiirtojenLkm*])⇒*lista*

Luo *Listal*:n kopion, jota on siirretty oikealle tai vasemmalle *SiirtojenLkm*:n elementtien määrittämän määrän. Ei muuta *Listal*:ä.

Jos *SiirtojenLkm* on positiivinen, siirto tapahtuu vasemmalle. Jos *SiirtojenLkm* on negatiivinen, siirto tapahtuu oikealle. Oletusarvo on -1 (siirrä yksi elementti oikealle).

shift(0h78E)	0h3C7
shift(0h78E,-2)	0h1E3
shift(0h78E,2)	0h1E38

**Tärkeää:** Binaariluvun edelle tulee aina merkittä etumerkki 0b ja heksadesimaaliluvun edelle 0h (nolla, ei 0-kirjain).

Desimaalisessa kantalukutilassa:

shift({1,2,3,4})	{undef,1,2,3}
shift({1,2,3,4},-2)	{undef,undef,1,2}
shift({1,2,3,4},2)	{3,4,undef,undef}

Elementit, jotka siirto vie *listan* alkuun tai loppuun, asettuvat symboliksi "undef".

**shift**(*Merkkijono* [*,SiirtojenLkm*]) $\Rightarrow$ *merkkijono*

Luo *Merkkijono* :n kopion, jota on siirretty oikealle tai vasemmalle *SiirtojenLkm*:n merkkien määrittämän määrän. Ei muuta *Merkkijono*:ä.

Jos *SiirtojenLkm* on positiivinen, siirto tapahtuu vasemmalle. Jos *SiirtojenLkm* on negatiivinen, siirto tapahtuu oikealle. Oletusarvo on -1 (siirrä yksi merkki oikealle).

Merkit, jotka siirto vie *merkkijonon* alkuun tai loppuun, muuttuvat välilyönneiksi.

shift("abcd")	" abc "
shift("abcd",-2)	" ab "
shift("abcd",1)	"bcd "

## sign() (etumerkki)

**sign**(*Laus*) $\Rightarrow$ *lauseke*

**sign**(*Lista*) $\Rightarrow$ *lista*

**sign**(*Matriisi*) $\Rightarrow$ *matriisi*

Kun kyseessä on reaali- tai kompleksilukulauseke *Laus*, antaa vastauksena *Laus*/**abs**(*Laus*), kun *Laus*  $\neq$  0.

Vastaus on 1, jos *Laus* on positiivinen. Vastaus on -1, jos *Laus* on negatiivinen.

**sign(0)** edustaa kompleksialueen yksikköpiiriä.

Kun kyseessä on lista tai matriisi, määrittää kaikkien elementtien etumerkit.

sign(-3.2)	-1.
sign({2,3,4,-5})	{1,1,1,-1}
sign(1+ x )	1

Jos kompleksilukutila on Reaali:

sign([-3 0 3])	[-1 ±1 1]
----------------	-----------

**simult(kerroinMatriisi, vakioVektori[, tol])** ⇒ matriisi

Määrittää sarakevektorin, joka sisältää lineaarisen yhtälöryhmän ratkaisut.

Huomaa: Katso myös **linSolve()**, sivu 110.

*kerroinMatriisin* on oltava neliömatriisi, joka sisältää yhtälöiden kertoimet.

*vakioVektorissa* on oltava sama rivimäärä (sama koko) kuin *kerroinMatriisissa* ja sen tulee sisältää vakiot.

Valinnaisesti kaikkia matriisielementtejä käsitellään nollana, jos niiden itseisarvo on pienempi kuin *Tol*. Tätä toleranssia käytetään vain, jos matriisissa on liukulukusyötteitä eikä se sisällä symbolisia muuttujia, joille ei ole määritetty arvoa. Muussa tapauksessa *Tol*-komentoa ei huomioida.

- Jos asetat **Automaattinen tai likimääräinen** -tilan valintaan Approximate (Likimääräinen), laskut suoritetaan liukulukuaritmetiikalla.
- Jos *Tol* jätetään pois tai sitä ei käytetä, oletusarvoinen toleranssi lasketaan seuraavasti:  
 $5E-14 \cdot \max(\dim(\text{kerroinMatriisi})) \cdot \text{rowNorm}(\text{kerroinMatriisi})$

**simult(kerroinMatriisi, vakioMatriisi[, tol])** ⇒ matriisi

Ratkaisee lineaarisia yhtälöryhmiä, joissa jokaisessa ryhmässä on samat yhtälöiden kertoimet mutta eri vakiot.

Jokaisen *vakioMatriisin* sarakkeen tulee sisältää jonkin yhtälöryhmän vakiot. Jokainen tulosmatriisin sarake sisältää vastaavan ryhmän ratkaisun.

Ratkaise yhtälöstä x ja y:

$$x + 2y = 1$$

$$3x + 4y = -1$$

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

Ratkaisu on  $x=-3$  and  $y=2$ .

Ratkaise:

$$ax + by = 1$$

$$cx + dy = 2$$

$$\begin{array}{l} \begin{bmatrix} a & b \\ c & d \end{bmatrix} \rightarrow \text{matx1} \quad \begin{bmatrix} a & b \\ c & d \end{bmatrix} \\ \text{simult}\left(\text{matx1}, \begin{bmatrix} 1 \\ 2 \end{bmatrix}\right) \quad \begin{array}{l} -(2 \cdot b - d) \\ a \cdot d - b \cdot c \\ 2 \cdot a - c \\ a \cdot d - b \cdot c \end{array} \end{array}$$

Ratkaise:

$$x + 2y = 1$$

$$3x + 4y = -1$$

$$x + 2y = 2$$

$$3x + 4y = -3$$

$$\text{simult}\left(\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}, \begin{bmatrix} 1 & 2 \\ -1 & -3 \end{bmatrix}\right) \quad \begin{bmatrix} -3 & -7 \\ 2 & 9 \\ 2 & 2 \end{bmatrix}$$

Ensimmäisessä ryhmässä  $x=-3$  ja  $y=2$ . Toisessa ryhmässä  $x=-7$  ja  $y=9/2$ .



*Laus* ►sin

**Huomaa:** Voit syöttää tämän operaattorin tietokoneen näppäimistöltä kirjoittamalla @>sin.

Näyttää *Laus*:n kulman sinin. Tämä on näytön muunnosoperaattori. Sitä voidaan käyttää vain syöterivin lopussa.

►sin alentaa kaikkia lausekkeen  $\cos(\dots)$  modulo  $1 - \sin(\dots)^2$  potensseja, siten että jäljelle jäävien lausekkeen  $\sin(\dots)$  potenssien eksponentit ovat alueella (0, 2). Tulos ei täten sisällä lauseketta  $\cos(\dots)$ , jos ja vain jos  $\cos(\dots)$  esiintyy lausekkeessa korotettuna vain parillisiin potensseihin.

**Huomaa:** Tätä muunnosoperaattoria ei tueta aste- eikä graadikulmatilassa. Ennen kuin käytät sitä, varmista, että kulmatila on asetettu radiaaneiksi ja että *Laus* ei sisällä eksplisiittisiä viittauksia aste- tai graadikulmiin.

$$\frac{(\cos(x))^2 \blacktriangleright \sin}{1 - (\sin(x))^2}$$

sin()



sin(*LausI*) ⇒ lauseke

Astekulmatilassa:

sin(*Listal*) ⇒ lista

$$\frac{\sin\left(\frac{\pi}{4}r\right)}{2}$$

sin(*LausI*) määrittää argumentin sinin lausekkeena.

$$\frac{\sin(45)}{2}$$

sin(*Listal*) määrittää listan, joka sisältää *Listal*:n kaikkien elementin sinit.

$$\frac{\sin\{0,60,90\}}{\left\{0, \frac{\sqrt{3}}{2}, 1\right\}}$$

**Huomaa:** Argumentti tulkitaan aste-, graadi- tai radiaanikulmaksi käytössä olevan kulmatilan mukaisesti. Voit ohittaa kulmatila-asetuksen väliaikaisesti käyttämällä merkintää °, G tai r.

Graadikulmatilassa:

$$\frac{\sin(50)}{2}$$

Radiaanikulmatilassa:

$$\sin\left(\frac{\pi}{4}\right) = \frac{\sqrt{2}}{2}$$

$$\sin(45^\circ) = \frac{\sqrt{2}}{2}$$

$\sin(\text{neliömatrissi}) \Rightarrow \text{neliömatrissi}$

Määrittää *neliömatrissi*:n matriisin sinin. Tämä ei ole sama kuin kunkin elementin sinin laskeminen.

Laskentamenetelmä on kuvattu kohdassa **cos()**.

*neliömatrissi*:n on oltava diagonaloitavissa. Vastaus sisältää aina liukulukuja.

Radiaanikulmatilassa:

$$\sin\left(\begin{bmatrix} 1 & 5 & 3 \\ 4 & 2 & 1 \\ 6 & -2 & 1 \end{bmatrix}\right) = \begin{bmatrix} 0.9424 & -0.04542 & -0.031999 \\ -0.045492 & 0.949254 & -0.020274 \\ -0.048739 & -0.00523 & 0.961051 \end{bmatrix}$$

$\sin^{-1}(\text{LausI}) \Rightarrow \text{lauseke}$

$\sin^{-1}(\text{ListaI}) \Rightarrow \text{lista}$

$\sin^{-1}(\text{LausI})$  määrittää lausekkeena kulman, jonka sini on *LausI*.

$\sin^{-1}(\text{ListaI})$  määrittää listan, joka sisältää *ListaI*:n kaikkien elementtien käänteissinit.

**Huomaa:** Vastaus lasketaan aste-, graadi- tai radiaanikulmana käytössä olevan kulmatila-asetuksen mukaisesti.

**Huomaa:** Voit syöttää tämän funktion näppäimistöltä kirjoittamalla **arcsin** (...).

$\sin^{-1}(\text{neliömatrissiI}) \Rightarrow \text{neliömatrissi}$

Laskee *neliömatrissi*:n matriisin käänteissinin. Tämä ei ole sama kuin kunkin elementin käänteissinin laskeminen. Laskentamenetelmä on kuvattu kohdassa **cos()**.

*neliömatrissi*:n on oltava diagonaloitavissa. Vastaus sisältää aina liukulukuja.

Astekulmatilassa:

$$\sin^{-1}(1) = 90$$

Graadikulmatilassa:

$$\sin^{-1}(1) = 100$$

Radiaanikulmatilassa:

$$\sin^{-1}\{0,0,2,0,5\} = \{0,0,201358,0,523599\}$$

Radiaanikulmatilassa ja suorakulmakompleksimuodossa:

$$\sin^{-1}\left(\begin{bmatrix} 1 & 5 \\ 4 & 2 \end{bmatrix}\right) = \begin{bmatrix} -0.174533-0.12198 \cdot i & 1.74533-2.35591 \cdot i \\ 1.39626-1.88473 \cdot i & 0.174533-0.593162 \cdot i \end{bmatrix}$$

**sinh()**Katalogi > **sinh(Laus I)** ⇒ lauseke $\sinh(1.2)$  1.50946**sinh(Lista I)** ⇒ lista $\sinh(\{0,1,2,3\})$   $\{0,1.50946,10.0179\}$ **sinh(Laus I)** laskee argumentin hyperbolisen kosinin lausekkeena.**sinh(Lista I)** määrittää listan, joka sisältää *Lista I*:n jokaisen elementin hyperbolisen sinin.**sinh(neliömatriisi I)** ⇒ *neliömatriisi*Laskee *neliömatriisi I*:n matriisin hyperbolisen sinin. Tämä ei ole sama kuin kunkin elementin hyperbolisen sinin laskeminen. Laskentamenetelmä on kuvattu kohdassa **cos()**.*neliömatriisi I*:n on oltava diagonalisoitavissa. Vastaus sisältää aina liukulukuja.

Radiaanikulmatilassa:

$\sinh$	$\begin{pmatrix} 1 & 5 & 3 \\ 4 & 2 & 1 \\ 6 & -2 & 1 \end{pmatrix}$	$360.954$	$305.708$	$239.604$
		$352.912$	$233.495$	$193.564$
		$298.632$	$154.599$	$140.251$

**sinh<sup>-1</sup>()**Katalogi > **sinh<sup>-1</sup>(Laus I)** ⇒ lauseke $\sinh^{-1}(0)$  0**sinh<sup>-1</sup>(Lista I)** ⇒ lista $\sinh^{-1}(\{0,2,1,3\})$   $\{0,1.48748,\sinh^{-1}(3)\}$ **sinh<sup>-1</sup>(Laus I)** laskee argumentin käänteisen hyperbolisen sinin lausekkeena.**sinh<sup>-1</sup>(Lista I)** määrittää listan, joka sisältää *Lista I*:n jokaisen elementin käänteiset hyperboliset sinit.**Huomaa:** Voit syöttää tämän funktion näppäimistöltä kirjoittamalla **arcsinh** (...).**sinh<sup>-1</sup>(neliömatriisi I)** ⇒ *neliömatriisi*Laskee *neliömatriisi I*:n matriisin käänteisen hyperbolisen sinin. Tämä ei ole sama kuin kunkin elementin käänteisen hyperbolisen sinin laskeminen. Laskentamenetelmä on kuvattu kohdassa **cos()**.

Radiaanikulmatilassa:

$\sinh^{-1}$	$\begin{pmatrix} 1 & 5 & 3 \\ 4 & 2 & 1 \\ 6 & -2 & 1 \end{pmatrix}$	$0.041751$	$2.15557$	$1.1582$
		$1.46382$	$0.926568$	$0.112557$
		$2.75079$	$-1.5283$	$0.57268$

*neliömatriisi* *l*:n on oltava diagonalisoitavissa. Vastaus sisältää aina liukulukuja.

## SinReg

**SinReg** *X*, *Y* [, [*Iteraatiot*] , [*Jakso*] [, *Luokka*, *Sisällytä*] ]

Laskee siniregression listoista *X* ja *Y*. Tulosten yhteenvedo tallentuu *stat.results*-muuttujaan. (Katso sivu 191.)

Kaikkien listojen on oltava samankokoisia *Sisällytä*-listaa lukuunottamatta.

*X* ja *Y* ovat riippumattomien ja riippuvien muuttujien listoja.

*Iteraatiot* on arvo, joka määrittää ratkaisun yrityskertojen (1-16) enimmäismäärän. Mikäli sitä ei määritetä, oletuksena käytetään arvoa 8. Suuremmilla arvoilla saadaan tyypillisesti parempi tarkkuus, mutta suoritus aika on pitempi ja päin vastoin.

*Jakso* määrittää arvioidun jakson. Mikäli sitä ei käytetä, *X*:n arvojen välisen eron tulisi olla sama ja arvojen tulisi olla peräkkäisessä järjestyksessä. Jos määrität *Jakson*, *x*:n arvojen väliset erot voivat olla erisuuria.

*Luokka* on luokkakoodien lista vastaavalle *X*- ja *Y* -datalle..

*Sisällytä* on yhden tai usemman luokkakoodin lista. Vain ne datayksiköt, joiden luokkakoodi sisältyy tähän listaan, ovat mukana laskutoimituksessa.

**SinReg**:n tulos esitetään aina radiaaneina riippumatta kulmatilan asetuksesta.

Lisätietoja listassa olevien tyhjien elementtien vaikutuksesta, katso Tyhjät elementtsivulla sivu 272.

Tulosmuuttuja	Kuvaus
stat.RegEqn	Regressioyhtälö: $a \cdot \sin(bx+c)+d$
stat.a, stat.b, stat.c, stat.d	Regressiokertoimet
stat.Resid	Regressioyhtälön jäännökset
stat.XReg	Muokatus <i>Y Lista</i> :n sisältämä datapisteiden lista, jota käytetään regressiossa komentojen <i>Frekv</i> , <i>Luokkalista</i> ja <i>Sisällytä luokat rajoitusten mukaisesti</i>
stat.YReg	Muokatus <i>Y Lista</i> :n sisältämä datapisteiden lista, jota käytetään regressiossa komentojen <i>Frekv</i> , <i>Luokkalista</i> ja <i>Sisällytä luokat rajoitusten mukaisesti</i>
stat.FreqReg	Komentoja <i>stat.XReg</i> ja <i>stat.YReg</i> vastaava frekvenssilista

## solve()

Katalogi > 

**solve**(*Yhtälö*, *Muutt*) $\Rightarrow$ *Boolean lauseke*

**solve**(*Yhtälö*, *Muutt*=*Arvaus*) $\Rightarrow$ *Boolean lauseke*

**solve**(*Epäyhtälö*, *Muutt*) $\Rightarrow$ *Boolean lauseke*

Laskee yhtälön tai epäyhtälön mahdollisia reaaliuratkaisuja muuttujalle *Muutt*. Tavoitteena on tuottaa kaikkien ratkaisujen ehdotuksia. Sellaisia yhtälöitä tai epäyhtälöitä voi kuitenkin esiintyä, joille ratkaisujen määrä on ääretön.

Ratkaisuehdotukset eivät välttämättä ole reaalisia äärellisiä ratkaisuja joillekin arvojen yhdistelmille määrittämättömissä muuttujissa.

**Automaattinen tai likimääräinen** -tilan Auto (Automaattinen) -asetuksessa tavoitteena on tuottaa täsmällisiä ratkaisuja, jos ne ovat tiiviitä, ja näitä lisätään iteratiivisten hakujen avulla likiarvoaritmetiikalla, jos täsmälliset ratkaisut ovat epäkäytännöllisiä.

$$\text{solve}(a \cdot x^2 + b \cdot x + c = 0, x)$$

$$x = \frac{\sqrt{b^2 - 4ac} - b}{2a} \text{ or } x = \frac{-\sqrt{b^2 - 4ac} - b}{2a}$$

$$\text{Ans}|a=1 \text{ and } b=1 \text{ and } c=1$$

$$x = \frac{-1 + \sqrt{3}}{2} \cdot i \text{ or } x = \frac{-1 - \sqrt{3}}{2} \cdot i$$

$$\text{solve}((x-a) \cdot e^x = x \cdot (x-a), x)$$

$$x = a \text{ or } x = 0.567143$$

Koska suurin yhteinen tekijä peruutetaan oletusarvoisesti suhdelukujen osoittajasta ja nimittäjästä, ratkaisut voivat olla ratkaisuja, jotka ovat ainoastaan yhden tai kummankin puolen rajalla.

$$(x+1) \cdot \frac{x-1}{x-1} + x - 3 \quad 2 \cdot x - 2$$

Täsmälliset ratkaisut ovat epätodennäköisiä tyyppiä  $\geq$ ,  $\leq$ ,  $<$  tai  $>$  oleville epäyhtälöille, ellei epäyhtälö ole lineaarinen ja sisältää ainoastaan muuttujan *Muutt.*

$$\text{solve}(5 \cdot x - 2 \geq 2 \cdot x, x) \quad x \geq \frac{2}{3}$$

Exact (Täsmällinen) -tilassa osat, joita ei voida ratkaista, annetaan implisiittisenä yhtälönä tai epäyhtälönä.

$$\text{exact}\left(\text{solve}\left((x-a) \cdot e^x = x \cdot (x-a), x\right)\right) \quad e^x + x = 0 \text{ or } x = a$$

Käytä rajoittavaa ("|")-operaattoria rajoittaaksesi ratkaisuväliä ja/tai muita yhtälössä tai epäyhtälössä esiintyviä muuttujia. Kun löydät ratkaisun yhdeltä väliltä, voit poistaa tämän välin seuraavista hakutoiminnoista epäyhtälöoperaattoreiden avulla.

Radiaanikulmatilassa:

$$\text{solve}\left(\tan(x) = \frac{1}{x}, x\right), x > 0 \text{ and } x < 1 \quad x = 0.860334$$

jos reaaliukuratkaisuja ei löydy, vastauksen totuusarvoksi määritetään epätosi. vastauksen totuusarvoksi tulee tosi, jos **solve()** pystyy määrittämään, että mikä tahansa muuttujan *Muutt* äärellinen reaaliarvo sopii yhtälöön tai epäyhtälöön.

$$\begin{array}{ll} \text{solve}(x = x + 1, x) & \text{false} \\ \text{solve}(x = x, x) & \text{true} \end{array}$$

Koska **solve()** tuottaa aina Boolean vastauksen, komentojen "and", "or" ja "not" avulla voit yhdistää **solve()**-funktion tuloksia toisiinsa tai muihin Boolean lausekkeisiin.

$$2 \cdot x - 1 \leq 1 \text{ and solve}(x^2 \neq 9, x) \quad x \neq -3 \text{ and } x \leq 1$$

Ratkaisut voivat sisältää uuden yksilöllisen määrittämättömän vakion, joka on muotoa  $n_j$ , jossa  $j$  on kokonaisluku väliltä 1-255. Tällaiset muuttujat tarkoittavat mielivaltaista kokonaislukua.

Radiaanikulmatilassa:

$$\text{solve}(\sin(x) = 0, x) \quad x = n \cdot \pi$$

Reaalilukutilassa murtolukupotenssit, joilla on pariton nimittäjä, viittaavat ainoastaan reaalilukualueeseen. Muussa tapauksessa haaralausekkeet, kuten murtolukupotenssit, logaritmit ja käänteiset trigonometriset funktiot, viittaavat vain päähaaraan. Näin ollen **solve()** tuottaa vain ratkaisuja, jotka vastaavat tätä yhtä reaaliluku- tai päähaaraa.

**Huomaa:** Katso myös **cSolve()**, **cZeros()**, **nSolve()** ja **zeros()**.

**solve**(*Yht1* and *Yht2* [and... ],  
*MuuttTaiArvaus1*,  
*MuuttTaiArvaus2* [, ...  
]) ⇒ *Boolean lauseke*

**solve**(*Yhtälöryhmä*, *MuuttTaiArvaus1*,  
*MuuttTaiArvaus2* [, ...  
]) ⇒ *Boolean lauseke*

**solve**{*Yht1*, *Yht2* [...]}  
{*MuuttTaiArvaus1*, *MuuttTaiArvaus2* [, ... ]}  
} ⇒ *Boolean lauseke*

Antaa ehdotuksia reaalilukuratkaisuista samanaikaisille algebrallisille yhtälöille, joissa jokainen *MuuttTaiArvaus* määrittää ratkaistavan muuttujan.

Voit erottaa yhtälöt **and**-operaattorilla tai voit syöttää *Yhtälöryhmän* jonkin katalogin mallin avulla. *MuuttTaiArvaus*-argumenttien on oltava sama kuin yhtälöiden lukumäärä. Voit halutessasi määrittää muuttujan ensimmäisen arvauksen. Jokaisen *MuuttTaiArvaus*-komennon on oltava muodossa:

*muuttuja*  
– tai –  
*muuttuja* = *reaaliluku* tai *ei-reaaliluku*

Esimerkiksi *x* kelpaa ja samoin *x*=3.

$\text{solve}\left(\frac{1}{x^3}-1,x\right)$	$x=1$
$\text{solve}(\sqrt{x}=2,x)$	false
$\text{solve}(\sqrt{x}=-2,x)$	$x=4$

$\text{solve}(y=x^2-2 \text{ and } x+2y=1, \{x,y\})$
$x=-\frac{3}{2} \text{ and } y=\frac{1}{4} \text{ or } x=1 \text{ and } y=1$

Jos kaikki yhtälöt ovat polynomeja, ja jos ET määrittää ensimmäisiä arvauksia, **solve()** käyttää leksikaalista Gröbner/Buchbergerin eliminaatiomenetelmää yrittäessään määrittää kaikki reaaliurkatkaisut.

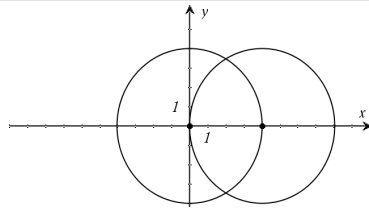
Oletetaan esimerkiksi, että origossa on ympyrän säde  $r$ , ja toinen ympyrän säde  $r$  on keskitetty kohtaan, jossa ensimmäinen ympyrä leikkaa positiivisen  $x$ -akseliin. Määritä leikkauskohdat **solve()**-funktion avulla.

Kuten on kuvattu merkinnällä  $r$  oikealla olevassa esimerkissä, samanaikaisissa polynomiyhtälöissä voi olla ylimääräisiä muuttujia, joilla ei ole arvoja, vaan ne edustavat tiettyjä numeerisia arvoja, jotka voidaan korvata myöhemmin.

Voit ottaa mukaan lisäksi ratkaisumuuttujia (tai käyttää niitä sijalla), joita ei esiinny yhtälöissä. Voit esimerkiksi ottaa mukaan muuttujan  $z$  ratkaisumuuttujaksi ulottaaksesi edellisen esimerkin näin säteen  $r$  kahteen samansuuntaiseen leikkaavaan sylinteriin.

Sylinteriratkaisut havainnollistavat, miten ratkaisujen sarjat voivat sisältää mielivaltaisia vakioita, jotka ovat muotoa  $ck$ , jossa  $k$  on kokonaisluku väliltä 1-255.

Polynomisarjoissa laskun suoritusaika tai muistin käyttö voivat riippua merkittävästi ratkaisumuuttujien järjestyksestä. Jos ensimmäinen valintasi kuluttaa muistia, tai et jaksa odottaa vastausta, yritä järjestää muuttujat uudelleen yhtälöihin ja/tai *muuttTaiArvaus*-listaan.



$$\text{solve}\left(x^2+y^2=r^2 \text{ and } (x-r)^2+y^2=r^2, \{x,y\}\right)$$

$$x=\frac{r}{2} \text{ and } y=\frac{\sqrt{3}\cdot r}{2} \text{ or } x=\frac{r}{2} \text{ and } y=-\frac{\sqrt{3}\cdot r}{2}$$

$$\text{solve}\left(x^2+y^2=r^2 \text{ and } (x-r)^2+y^2=r^2, \{x,y,z\}\right)$$

$$x=\frac{r}{2} \text{ and } y=\frac{\sqrt{3}\cdot r}{2} \text{ and } z=c1 \text{ or } x=\frac{r}{2} \text{ and } y=-\frac{\sqrt{3}\cdot r}{2} \text{ and } z=c1$$

Jos haluat nähdä koko vastauksen, paina  $\blacktriangle$  ja siirrä sen jälkeen osoitinta painikkeilla  $\blacktriangleleft$  ja  $\blacktriangleright$ .



Jos et ota mukaan arvauksia, ja jos jokin yhtälöistä on ei-polynominen missä tahansa muuttujassa, mutta kaikki yhtälöt ovat lineaarisia kaikissa ratkaisumuuttujissa, **solve()** käyttää Gaussin eliminointia yrittäessään määrittää kaikkia reaalilukuratkaisuja.

Jos yhtälöryhmä ei ole polynominen kaikilta muuttujiltaan eikä lineaarinen ratkaisumuuttujiltaan, **solve()** määrittää korkeintaan yhden ratkaisun käyttäen likimääräistä iteratiivista menetelmää. Tässä ratkaisumuuttujien lukumäärän on oltava sama kuin yhtälöiden lukumäärä, ja kaikkien muiden yhtälöiden sisältämien muuttujien on sievennyttävä luvuiksi.

Jokainen ratkaisumuuttuja alkaa arvausarvostaan, mikäli se on määritetty; muussa tapauksessa se alkaa arvosta 0.0.




Arvausten avulla voit etsiä lisäratkaisuja yksi kerrallaan. Suppenemista varten arvauksen on mahdollisesti oltava melko lähellä ratkaisua.

$$\text{solve}\left(x+e^z \cdot y=1 \text{ and } x-y=\sin(z), \{x,y\}\right)$$

$$x=\frac{e^z \cdot \sin(z)+1}{e^z+1} \text{ and } y=\frac{-(\sin(z)-1)}{e^z+1}$$

$$\text{solve}\left(e^z \cdot y=1 \text{ and } -y=\sin(z), \{y,z\}\right)$$

$$y=2.812\text{E}-10 \text{ and } z=21.9911 \text{ or } y=0.001871$$

Jos haluat nähdä koko vastauksen, paina  ja siirrä sen jälkeen osoitinta painikkeilla  ja .

$$\text{solve}\left(e^z \cdot y=1 \text{ and } -y=\sin(z), \{y,z=2 \cdot \pi\}\right)$$

$$y=0.001871 \text{ and } z=6.28131$$

## SortA

**SortA** *List1* [, *List2*] [, *List3*] ...  
**SortA** *Vektori1* [, *Vektori2*] [, *Vektori3*]  
 ...

Lajittelee ensimmäisen argumentin elementit nousevaan järjestykseen.

Jos otat mukaan lisäargumentteja, lajittelee kunkin argumentin elementit siten, että niiden uudet paikat vastaavat ensimmäisen argumentin elementtien uusia paikkoja.

Kaikkien argumenttien on oltava lista- tai vektorinimiä. Kaikkien argumenttien on oltava samankokoisia.

Ensimmäisen argumentin sisältämät tyhjät elementit siirtyvät alas. Lisätietoja tyhjästä elementeistä, katso sivu 272.

$\{2,1,4,3\} \rightarrow \text{list1}$	$\{2,1,4,3\}$
SortA list1	Done
list1	$\{1,2,3,4\}$
$\{4,3,2,1\} \rightarrow \text{list2}$	$\{4,3,2,1\}$
SortA list2,list1	Done
list2	$\{1,2,3,4\}$
list1	$\{4,3,2,1\}$

## SortD

Katalogi >

**SortD** *Lista1* [, *Lista2*] [, *List3*] ...

**SortD** *Vektori1* [, *Vektori2*] [, *Vektori3*] ...

Muuten samanlainen kuin **SortA** paitsi, että **SortD** lajittelee elementit laskevaan järjestykseen.

Ensimmäisen argumentin sisältämät tyhjät elementit siirtyvät alas. Lisätietoja tyhjästä elementeistä, katso sivu 272.

$\{2,1,4,3\} \rightarrow list1$	$\{2,1,4,3\}$
$\{1,2,3,4\} \rightarrow list2$	$\{1,2,3,4\}$
SortD <i>list1</i> , <i>list2</i>	Done
<i>list1</i>	$\{4,3,2,1\}$
<i>list2</i>	$\{3,4,1,2\}$

## ►Sphere

Katalogi >

*Vektori* ►Sphere

**Huomaa:** Voit syöttää tämän operaattorin tietokoneen näppäimistöltä kirjoittamalla @>Sphere.

Näyttää rivi- tai sarakevektorin pallonmuotoisena [ $\rho$   $\angle$   $\theta$   $\angle$   $\phi$ ].

*Vektorin* on oltava kooltaan 3, ja se voi olla rivi- tai sarakevektori.

**Huomaa:** ►Sphere on näyttömuodon ohje, ei muunnosfunktio. Voit käyttää komentoa ainoastaan syöterivin lopussa.

**Huom:** Vastauksen pakottaminen likimääräiseksi:

**Kämmenlaite:** Paina  .

**Windows®:** Paina **Ctrl+Enter**.

**Macintosh®:** Paina **⌘+Enter**.

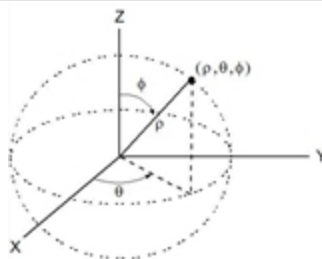
**iPad®:** Pidä **enter** ja valitse .

```
[ 1 2 3 ] ►Sphere  
[ 3.74166 1.10715 0.640522 ]
```

```
( ( 2 1/4 3 ) ) ►Sphere  
[ 3.60555 0.785398 0.588003 ]
```

Paina

```
( ( 2 1/4 3 ) ) ►Sphere  
[ sqrt(13) 1/4 sin^-1(2*sqrt(13)/13) ]
```

**sqrt()**Katalogi > **sqrt(Lausl)** ⇒ lauseke

$$\sqrt{4} \qquad 2$$

**sqrt(Listal)** ⇒ lista

$$\sqrt{\{9,a,4\}} \qquad \{3,\sqrt{a},2\}$$

Laskee argumentin neliöjuuren.

Kun kyseessä on lista, laskee kaikkien *Listal*:n elementtien neliöjuuret.**Huomaa:** Katso myös **Neliöjuurimalli**, sivu 1.**stat.results**Katalogi > 

stat.results

$$xlist:=\{1,2,3,4,5\} \qquad \{1,2,3,4,5\}$$

Näyttää tilastollisen laskutoimituksen tulokset.

$$ylist:=\{4,8,11,14,17\} \qquad \{4,8,11,14,17\}$$

Vastaukset näytetään nimiarvoparien sarjana. Näytetyt nimenomaiset nimet riippuvat viimeksi sievennetystä tilastofunktiosta tai komennosta.

LinRegMx *xlist,ylist*,1: *stat.results*

"Title"	"Linear Regression (mx+b)"
"RegEqn"	"m*x+b"
"m"	3.2
"b"	1.2
"r <sup>2</sup> "	0.996109
"t"	0.998053
"Resid"	" {... } "

Voit kopioida nimen tai arvon ja liittää sen muihin paikkoihin.

<i>stat.values</i>	"Linear Regression (mx+b)"
	"m*x+b"
	3.2
	1.2
	0.996109
	0.998053
	"{-0.4,0.4,0.2,0,-0.2}"

**Huomaa:** Älä määritä muuttujia, joilla on sama nimi kuin tilastoanalyseissä käytettävillä muuttujilla. Joissakin tapauksissa tästä voi olla seurauksena virhetilanne. Tilastoanalyseissä käytettävät muuttujanimet on esitetty alla olevassa taulukossa.

stat.a	stat.dfDenom	stat.MedianY	stat.Q3X	stat.SSBLOCK
stat.AdjR <sup>2</sup>	stat.dfBlock	stat.MEPred	stat.Q3Y	stat.SSCol
stat.b	stat.dfCol	stat.MinX	stat.r	stat.SSX
stat.b0	stat.dfError	stat.MinY	stat.r <sup>2</sup>	stat.SSY
stat.b1	stat.dfInteract	stat.MS	stat.RegEqn	stat.SSError
stat.b2	stat.dfReg	stat.MSBlock	stat.Resid	stat.SSInteract
stat.b3	stat.dfNumer	stat.MSCol	stat.ResidTrans	stat.SSReg
stat.b4	stat.dfRow	stat.MSError	stat.σ <sub>x</sub>	stat.SSRow
stat.b5	stat.DW	stat.MSInteract	stat.σ <sub>y</sub>	stat.tList
stat.b6	stat.e	stat.MSReg	stat.σ <sub>x1</sub>	stat.UpperPred
stat.b7	stat.ExpMatrix	stat.MSRow	stat.σ <sub>x2</sub>	stat.UpperVal
stat.b8	stat.F	stat.n	stat.Σ <sub>x</sub>	stat.X̄
stat.b9	stat.FBlock	Stat. $\hat{p}$	stat.Σ <sub>x</sub> <sup>2</sup>	stat.X̄ <sub>1</sub>
stat.b10	stat.Fcol	stat. $\hat{p}_1$	stat.Σ <sub>xy</sub>	stat.X̄ <sub>2</sub>
stat.bList	stat.FInteract	stat. $\hat{p}_2$	stat.Σ <sub>y</sub>	stat.X̄Diff
stat.χ <sup>2</sup>	stat.FreqReg	stat. $\hat{p}$ Diff	stat.Σ <sub>y</sub> <sup>2</sup>	stat.X̄List
stat.c	stat.Frow	stat.PList	stat.s	stat.XReg
stat.CLower	stat.Leverage	stat.PVal	stat.SE	stat.XVal
stat.CLowerList	stat.LowerPred	stat.PValBlock	stat.SEList	stat.XValList
stat.CompList	stat.LowerVal	stat.PValCol	stat.SEPred	stat.ȳ
stat.CompMatrix	stat.m	stat.PValInteract	stat.sResid	stat.ȳ
stat.CookDist	stat.MaxX	stat.PValRow	stat.SESlope	stat.ȳList
stat.CUpper	stat.MaxY	stat.Q1X	stat.sp	stat.YReg
stat.CUpperList	stat.ME	stat.Q1Y	stat.SS	
stat.d	stat.MedianX			

**Huomaa:** Aina kun Listat & Taulukot -sovellus laskee tilastolaskujen vastauksia, se kopioi "stat." -ryhmän muuttujat "stat#." -ryhmään, jossa # on luku, jota lisätään automaattisesti. Tällä tavoin voit säilyttää aikaisemmat tulokset suorittaessasi useita laskutoimituksia.

stat.values

Katso esimerkki kohdassa  
stat.results.

Näyttää matriisiin viimeksi sievennetyille tilastofunktiolle tai -komennolle lasketuista arvoista.

Toisin kuin **stat.results**, **stat.values** jättää pois arvoihin liittyvät nimet.

Voit kopioida arvon ja liittää sen muihin paikkoihin.

**stDevPop()**

**stDevPop(Lista[, frekvLista])** ⇒ lauseke

Laskee *Lista*:n sisältämien elementtien perusjoukon keskihajonnan.

Jokainen *frekvListan* elementti näyttää *Listan* vastaavien elementtien peräkkäisten esiintymien lukumäärän.

**Huomaa:** *Listassa* tulee olla vähintään kaksi elementtiä. Tyhjiä elementtejä ei huomioida. Lisätietoja tyhjästä elementeistä, katso sivu 272.

**stDevPop(MatriisiI[, frekvMatriisi])** ⇒ matriisi

Laskee rivivektorin *MatriisiI*:n sarakkeiden perusjoukon keskihajonnoista.

Jokainen *frekvMatriisin* elementti näyttää *MatriisiI*:n vastaavien elementtien peräkkäisten esiintymien lukumäärän.

**Huomaa:** *MatriisiI* :ssä on oltava vähintään kaksi riviä. Tyhjiä elementtejä ei huomioida. Lisätietoja tyhjästä elementeistä, katso sivu 272.

Radiaanikulma- ja automaattisissa tiloissa:

$$\text{stDevPop}\left\{\left\{a,b,c\right\}\right\} = \frac{\sqrt{2\cdot\left(a^2-a\cdot\left(b+c\right)+b^2-b\cdot c+c^2\right)}}{3}$$

$$\text{stDevPop}\left\{\left\{1,2,5,-6,3,-2\right\}\right\} = \frac{\sqrt{465}}{6}$$

$$\text{stDevPop}\left\{\left\{1,3,2,5,-6,4\right\},\left\{3,2,5\right\}\right\} = 4.11107$$

$$\text{stDevPop}\left(\begin{array}{ccc} 1 & 2 & 5 \\ -3 & 0 & 1 \\ 5 & 7 & 3 \end{array}\right) = \left[\begin{array}{ccc} 4\cdot\sqrt{6} & \sqrt{78} & 2\cdot\sqrt{6} \\ 3 & 3 & 3 \end{array}\right]$$

$$\text{stDevPop}\left(\begin{array}{cc} -1.2 & 5.3 \\ 2.5 & 7.3 \\ 6 & -4 \end{array}\right) = \left[\begin{array}{cc} 4 & 2 \\ 3 & 3 \\ 1 & 7 \end{array}\right]$$

$$[2.52608 \quad 5.21506]$$

**stDevSamp()**

Katalogi &gt;

**stDevSamp**(*List*[,  
*frekvList*])⇒*lauseke*Laskee *List*an sisältämien elementtien otoksen keskihajonnan.Jokainen *frekvList*an elementti näyttää *List*an vastaavien elementtien peräkkäisten esiintymien lukumäärän.**Huomaa:** *List*assa tulee olla vähintään kaksi elementtiä. Tyhjiä elementtejä ei huomioida. Lisätietoja tyhjästä elementeistä, katso sivu 272.**stDevSamp**(*Matriisi*1[,  
*frekvMatriisi*])⇒*matriisi*Laskee rivivektorin *Matriisi*1:n sarakkeiden otosten keskihajonnoista.Jokainen *frekvMatriisi*in elementti näyttää *Matriisi*1:n vastaavien elementtien peräkkäisten esiintymien lukumäärän.**Huomaa:** *Matriisi*1:ssä on oltava vähintään kaksi riviä. Tyhjiä elementtejä ei huomioida. Lisätietoja tyhjästä elementeistä, katso sivu 272.

$$\text{stDevSamp}(\{a,b,c\}) = \frac{\sqrt{3 \cdot (a^2 - a \cdot (b+c) + b^2 - b \cdot c + c^2)}}{3}$$

$$\text{stDevSamp}(\{1,2,5,-6,3,-2\}) = \frac{\sqrt{62}}{2}$$

$$\text{stDevSamp}(\{1.3,2.5,-6.4\},\{3,2,5\}) = 4.33345$$

$$\text{stDevSamp}\left(\begin{pmatrix} 1 & 2 & 5 \\ -3 & 0 & 1 \\ 5 & 7 & 3 \end{pmatrix}\right) = \begin{bmatrix} 4 & \sqrt{13} & 2 \end{bmatrix}$$

$$\text{stDevSamp}\left(\begin{pmatrix} -1.2 & 5.3 \\ 2.5 & 7.3 \\ 6 & -4 \end{pmatrix}, \begin{bmatrix} 4 & 2 \\ 3 & 3 \\ 1 & 7 \end{bmatrix}\right) = \begin{bmatrix} 2.7005 & 5.44695 \end{bmatrix}$$

**Stop**

Katalogi &gt;

**Stop**

Ohjelmointikomento: Pysäyttää ohjelman.

**Stop** ei ole sallittu funktioissa.**Huomaa esimerkkiä syöttäessäsi:** Ohjeet monirivisten ohjelmien ja funktion määrittysten syöttämisestä löytyvät tuotteen ohjekirjan Laskin-osiosta.

<i>i</i> :=0	0
Define <i>prog1</i> ()=Prgm	Done
For <i>i</i> ,1,10,1	
If <i>i</i> =5	
Stop	
EndFor	
EndPrgm	
<i>prog1</i> ()	Done
<i>i</i>	5

**Store**

Katso → (tallenna), sivu 253.

**string()** (merkkijono)Katalogi > **string**(*Laus*) $\Rightarrow$ merkkijonoSieventää lausekkeen *Laus* ja antaa vastauksen merkkijonona.

<code>string(1.2345)</code>	"1.2345"
<code>string(1+2)</code>	"3"
<code>string(cos(x)+√3)</code>	"cos(x)+√(3)"

**subMat()**Katalogi > **subMat**(*Matriisi1* [, *alkurivi*] [, *alkusarake*] [, *loppurivi*] [, *loppusarake*])  
 $\Rightarrow$ matriisiLaskee *Matriisi1*:n määritetyn alimatriisin.Oletusarvot: *alkurivi*=1, *alkusarake*=1, *loppurivi*=viimeinen rivi, *endCol*=viimeinen sarake.

$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix} \rightarrow m1$	$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$
<code>subMat(m1,2,1,3,2)</code>	$\begin{bmatrix} 4 & 5 \\ 7 & 8 \end{bmatrix}$
<code>subMat(m1,2,2)</code>	$\begin{bmatrix} 5 & 6 \\ 8 & 9 \end{bmatrix}$

**Sum (Sigma)**Katso  $\Sigma()$ , sivu 243.**sum()**Katalogi > **sum**(*Lista* [, *Alku*] [, *Loppu*]) $\Rightarrow$ lausekeLaskee *Listan* elementtien summan.*Alku* ja *Loppu* ovat valinnaisia. Ne määrittävät elementtien alueen.Mikä tahansa tyhjä argumentti tuottaa tyhjän vastauksen. *Listassa* olevia tyhjiä elementtejä ei huomioida. Lisätietoja tyhjästä elementeistä, katso sivu 272.**sum**(*Matriisi1* [, *Alku*] [, *Loppu*]) $\Rightarrow$ matriisiLaskee rivivektorin, joka sisältää *Matriisi1*:n sarakkeiden elementtien summat.*Alku* ja *Loppu* ovat valinnaisia. Ne määrittävät rivialueen.

<code>sum({1,2,3,4,5})</code>	15
<code>sum({a,2·a,3·a})</code>	6·a
<code>sum(seq(n,n,1,10))</code>	55
<code>sum({1,3,5,7,9},3)</code>	21

<code>sum</code> $\left(\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}\right)$	$\begin{bmatrix} 5 & 7 & 9 \end{bmatrix}$
<code>sum</code> $\left(\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}\right)$	$\begin{bmatrix} 12 & 15 & 18 \end{bmatrix}$
<code>sum</code> $\left(\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}, 2, 3\right)$	$\begin{bmatrix} 11 & 13 & 15 \end{bmatrix}$

Mikä tahansa tyhjä argumentti tuottaa tyhjän vastauksen. *Matriisi1*:ssä olevia tyhjiä elementtejä ei huomioida. Lisätietoja tyhjistä elementeistä, katso sivu 272.

## sumIf()

$\text{sumIf}(\text{Lista}, \text{Kriteerit}, \text{SummaLista}) \Rightarrow \text{arvo}$

Laskee kaikkien niiden *Listan* sisältämien elementtien kumuloituneen summan, jotka vastaavat määritettyjä kriteereitä *Kriteerit*. Voit halutessasi antaa kumuloivat elementit määrittämällä vaihtoehdoisen listan, *summaLista*.

*Lista* voi olla lauseke, lista tai matriisi. *SummaListalla*, mikäli se määritetään, on oltava samat mitat kuin *Listalla*.

*Kriteeri* voi olla:

- Arvo, lauseke tai merkkijono.  
Esimerkiksi **34** kumuloi vain niitä *Listan* elementtejä, jotka sieventyvät arvoon 34.
- Boolean lauseke, joka sisältää symbolin **?** kunkin elementin paikanpitäjänä.  
Esimerkiksi lauseke **?<10** kumuloi vain niitä *Listan* elementtejä, jotka ovat alle 10.

Kun jokin *Listan* elementti vastaa kriteereitä *Kriteerit*, elementti lisätään kumuloituvaan summaan. Jos sisällytät funktion *summaListan*, summaan lisätäänkin sen sijaan vastaava *summaListan* elementti.

Listat & Taulukot -sovelluksessa voit käyttää solualueita *Listan* ja *summaListan* tilalla.

Tyhjiä elementtejä ei huomioida. Lisätietoja tyhjistä elementeistä, katso sivu 272.

---

 $\text{sumIf}(\{1,2,\mathbf{e},3,\pi,4,5,6\}, 2.5 < ? < 4.5)$ 
 $\mathbf{e} + \pi + 7$ 


---

 $\text{sumIf}(\{1,2,3,4\}, 2 < ? < 5, \{10,20,30,40\})$ 


---

70



Huomaa: Katso myös **countlf()**, sivu 38.

**system**(*Yht1* [, *Yht2* [, *Yht3* [, ...]])

$$\text{solve}\left(\begin{cases} x+y=0 \\ x-y=8 \end{cases}, x, y\right) \quad x=4 \text{ and } y=-4$$

**system**(*Laus1* [, *Laus2* [, *Laus3* [, ...]])

Laskee yhtälöryhmän listaksi muotoiltuna. Voit luoda yhtälöryhmän myös mallin avulla.

Huomaa: Katso myös **Yhtälöryhmä**, sivu 4.

## T

*Matriisi*  $I^T \Rightarrow$  *matriisi*

Laskee *Matriisi*  $I$ :n transponoidun liittokompleksimatriisin.

Huomaa: Voit syöttää tämän operaattorin tietokoneen näppäimistöltä kirjoittamalla @t.

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}^T \quad \begin{bmatrix} 1 & 4 & 7 \\ 2 & 5 & 8 \\ 3 & 6 & 9 \end{bmatrix}$$

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix}^T \quad \begin{bmatrix} a & c \\ b & d \end{bmatrix}$$

$$\begin{bmatrix} 1+i & 2+i \\ 3+i & 4+i \end{bmatrix}^T \quad \begin{bmatrix} 1-i & 3-i \\ 2-i & 4-i \end{bmatrix}$$

**tan**(*Laus1*)  $\Rightarrow$  *lauseke*

Astekulmatilassa:

**tan**(*Lista1*)  $\Rightarrow$  *lista*

$$\tan\left(\frac{\pi}{4}\right) \quad 1$$

**tan**(*Laus1*) laskee argumentin tangentin lausekkeena.

$$\tan(45) \quad 1$$

**tan**(*Lista1*) määrittää *Lista1*:n kaikkien elementtien tangenttien listan.

$$\tan\{0,60,90\} \quad \{0,\sqrt{3},\text{undef}\}$$

Graadikulmatilassa:

**tan()** -painike

**Huomaa:** Argumentti tulkitaan aste-, graadi- tai radiaanikulmaksi käytössä olevan kulmatilan mukaisesti. Voit ohittaa kulmatila-asetuksen väliaikaisesti käyttämällä merkintää °, G tai  $\Gamma$ .

$\tan\left(\frac{\pi}{4}\right)$	1
$\tan(50)$	1
$\tan(\{0,50,100\})$	{0,1,undef}

Radiaanikulmatilassa:

$\tan\left(\frac{\pi}{4}\right)$	1
$\tan(45^\circ)$	1
$\tan\left(\left\{\pi, \frac{\pi}{3}, \pi, \frac{\pi}{4}\right\}\right)$	{0,√3,0,1}


**tan(neliömatriisi I)** ⇒ *neliömatriisi*

Laskee *neliömatriisi I*:n matriisin tangentin. Tämä ei ole sama kuin kunkin elementin tangentin laskeminen. Laskentamenetelmä on kuvattu kohdassa **cos()**.

*neliömatriisi I*:n on oltava diagonaloitavissa. Vastaus sisältää aina liukulukuja.

Radiaanikulmatilassa:

$\tan\left(\begin{bmatrix} 1 & 5 & 3 \\ 4 & 2 & 1 \\ 6 & -2 & 1 \end{bmatrix}\right)$	
	$\begin{bmatrix} -28.2912 & 26.0887 & 11.1142 \\ 12.1171 & -7.83536 & -5.48138 \\ 36.8181 & -32.8063 & -10.4594 \end{bmatrix}$

**tan<sup>-1</sup>()** -painike**tan<sup>-1</sup>(Laus I)** ⇒ *lauseke***tan<sup>-1</sup>(Lista I)** ⇒ *lista*

**tan<sup>-1</sup>(Laus I)** laskee kulman, jonka tangentti on *Laus I*, määrittäen vastauksen lausekkeena.

**tan<sup>-1</sup>(Lista I)** luo listan *Lista I*:n jokaisen elementin käänteistangenteista.

**Huomaa:** Vastaus lasketaan aste-, graadi- tai radiaanikulmana käytössä olevan kulmatila-asetuksen mukaisesti.

**Huomaa:** Voit syöttää tämän funktion näppäimistöä kirjoittamalla **arctan** (...).

**tan<sup>-1</sup>(neliömatriisi I)** ⇒ *neliömatriisi*

Astekulmatilassa:

$\tan^{-1}(1)$	45
----------------	----

Graadikulmatilassa:

$\tan^{-1}(1)$	50
----------------	----

Radiaanikulmatilassa:

$\tan^{-1}(\{0,0.2,0.5\})$	{0,0.197396,0.463648}
----------------------------	-----------------------

Radiaanikulmatilassa:

## $\tan^{-1}()$

 -painike

Laskee *neliömatriisi*1:n matriisin käänteistangentin. Tämä ei ole sama kuin kunkin elementin käänteisen tangentin laskeminen.

Laskentamenetelmä on kuvattu kohdassa **cos()**.

*neliömatriisi*1:n on oltava diagonalisoitavissa. Vastaus sisältää aina liukulukuja.

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

-0.083658	1.26629	0.62263
0.748539	0.630015	-0.070012
1.68608	-1.18244	0.455126

## tangentLine()

Katalogi > 

### tangentLine

*(Laus1,Muutt,Piste)*⇒lauseke

### tangentLine

*(Laus1,Muutt=Piste)*⇒lauseke

Määrittää tangenttisuoran *Laus1*:n kuvaamasta käyrästä pisteessä, joka on määritetty kohtaan *Muutt=Piste*.

Varmista, että riippumatonta muuttujaa ei ole määritetty. Esimerkiksi, jos  $f_1(x):=5$  and  $x:=3$ , tällöin **tangentLine**( $f_1(x),x,2$ ) antaa vastauksen "epätosi".

$\text{tangentLine}(x^2,x,1)$	$2 \cdot x - 1$
$\text{tangentLine}((x-3)^2-4,x=3)$	-4
$\text{tangentLine}\left(x^{\frac{1}{3}},x=0\right)$	$x=0$
$\text{tangentLine}(\sqrt{x^2-4},x=2)$	undef
$x:=3: \text{tangentLine}(x^2,x,1)$	5

## tanh()

Katalogi > 

**tanh**(*Laus1*)⇒lauseke

**tanh**(*Listal*)⇒lista

**tanh**(*Laus1*) laskee argumentin hyperbolisen tangentin lausekkeena.

**tanh**(*Listal*) luo listan *Listal*:n jokaisen elementin hyperbolisista tangenteista.

**tanh**(*neliömatriisi*1)⇒*neliömatriisi*

$\tanh(1.2)$	0.833655
$\tanh(\{0,1\})$	$\{0,\tanh(1)\}$

Radiaanikulmatilassa:

Laskee *neliömatriisi*  $I$ :n matriisin hyperbolisen tangentin. Tämä ei ole sama kuin kunkin elementin hyperbolisen tangentin laskeminen. Laskentamenetelmä on kuvattu kohdassa **cos()**.

*neliömatriisi*  $I$ :n on oltava diagonaloitavissa. Vastaus sisältää aina liukulukuja.

$$\text{tanh} \begin{pmatrix} 1 & 5 & 3 \\ 4 & 2 & 1 \\ 6 & -2 & 1 \end{pmatrix} \begin{bmatrix} -0.097966 & 0.933436 & 0.425972 \\ 0.488147 & 0.538881 & -0.129382 \\ 1.28295 & -1.03425 & 0.428817 \end{bmatrix}$$

**tanh<sup>-1</sup>(LausI)** ⇒ lauseke

**tanh<sup>-1</sup>(ListaI)** ⇒ lista

**tanh<sup>-1</sup>(LausI)** laskee argumentin käänteisen hyperbolisen tangentin lausekkeena.

**tanh<sup>-1</sup>(ListaI)** luo listan *ListaI*:n jokaisen elementin käänteisistä hyperbolisista tangenteista.

**Huomaa:** Voit syöttää tämän funktion näppäimistöltä kirjoittamalla **arctanh** (...).

**tanh<sup>-1</sup>(neliömatriisiI)** ⇒ *neliömatriisi*

Laskee *neliömatriisi*  $I$ :n matriisin käänteisen hyperbolisen tangentin. Tämä ei ole sama kuin kunkin elementin käänteisen hyperbolisen tangentin laskeminen. Laskentamenetelmä on kuvattu kohdassa **cos()**.

*neliömatriisi*  $I$ :n on oltava diagonaloitavissa. Vastaus sisältää aina liukulukuja.

Suorakulmakompleksimuodossa:




$$\text{tanh}^{-1}(0) \quad 0$$

$$\text{tanh}^{-1}(\{1,2,1,3\})$$

$$\left\{ \text{undef}, 0.518046 - 1.5708 \cdot i, \frac{\ln(2)}{2} - \frac{\pi}{2} \cdot i \right\}$$

Radiaanikulmatilassa ja suorakulmakompleksimuodossa:

$$\text{tanh}^{-1} \begin{pmatrix} 1 & 5 & 3 \\ 4 & 2 & 1 \\ 6 & -2 & 1 \end{pmatrix} \begin{bmatrix} -0.099353 + 0.164058 \cdot i & 0.267834 - 1.4908 \\ -0.087596 - 0.725533 \cdot i & 0.479679 - 0.94730 \\ 0.5111463 - 2.08316 \cdot i & -0.878563 + 1.7901 \end{bmatrix}$$

Jos haluat nähdä koko vastauksen, paina  ja siirrä sen jälkeen osoitinta painikkeilla  ja .

**taylor()**

Katalogi &gt;

**taylor(Laus1, Muutt, Aste[, Piste])** ⇒ lauseke

Laskee pyydetyn Taylorin polynomin. Polynomi sisältää kokonaislukuasteiden ei-nollatermejä nollasta arvoon *Aste* saakka komennossa (*Muutt* miinus *Piste*). **taylor()** antaa vastauksena itsensä, jos tämän asteen tyypistettyjä potenssisarjoja ei ole, tai jos tässä vaadittaisiin negatiivisia tai murtolukuekspONENTTEJA. Käytä substituutiota ja/tai väliaikaista kertomista funktion (*Muutt* miinus *Piste*) potenssilla, kun haluat määrittää yleisempiä potenssisarjoja.

*Piste* on oletusarvo on nolla, ja se on lavennuspiste.

$$\begin{array}{l} \text{taylor}(e^{\sqrt{x}}, x, 2) \qquad \text{taylor}(e^{\sqrt{x}}, x, 2, 0) \\ \text{taylor}(e^{t, 4})|_{t=\sqrt{x}} \qquad \frac{3}{24} + \frac{x^2}{6} + \frac{x}{2} + \sqrt{x} + 1 \\ \text{taylor}\left(\frac{1}{x \cdot (x-1)}, x, 3\right) \qquad \text{taylor}\left(\frac{1}{x \cdot (x-1)}, x, 3, 0\right) \\ \text{expand}\left(\frac{\text{taylor}\left(\frac{x}{x \cdot (x-1)}, x, 4\right)}{x}\right) \\ \qquad \qquad \qquad -x^3 - x^2 - x - \frac{1}{x} - 1 \end{array}$$

**tCdf()**

Katalogi &gt;

**tCdf(alaraja, yläraja, df)** ⇒ luku, jos *alaraja* ja *yläraja* ovat lukuja, *lista*, jos *alaraja* ja *yläraja* ovat listoja

Laskee Studentin *t*-todennäköisyysjakauman *alarajan* ja *ylärajan* välillä määritetyille vapausasteelle *df*.

Kun  $P(X \leq \text{yläraja})$ , aseta *alaraja* =  $-\infty$ .

**tCollect()**

Katalogi &gt;

**tCollect(Laus1)** ⇒ lauseke

Laskee lausekkeen, jossa tulot ja sinien ja kosinien kokonaislukupotenssit on muunnettu useiden kulmien sinien ja kosinien lineaariseksi kombinaatioksi, kulmien summiksi ja kulmien erotuksiksi. Transformaatio muuntaa trigonometriset polynomit niiden harmonisten lineaariseksi kombinaatioksi.

$$\begin{array}{l} \text{tCollect}(\{\cos(\alpha)\}^2) \qquad \frac{\cos(2 \cdot \alpha) + 1}{2} \\ \text{tCollect}(\sin(\alpha) \cdot \cos(\beta)) \qquad \frac{\sin(\alpha - \beta) + \sin(\alpha + \beta)}{2} \end{array}$$

Joskus **tCollect()** pystyy suorittamaan haluamasi laskutoimituksen, kun oletusarvoinen trigonometrinen sievennys ei siihen pysty. **tCollect()** pyrkii kääntämään funktion **tExpand()** aikaansaamat transformaatiot. Joskus **tExpand()**-funktion soveltaminen **tCollect()**-funktion tulokseen, tai päin vastoin, kahdessa eri vaiheessa yksinkertaistaa lauseketta.

**tExpand(Laus1)⇒lauseke**

Laskee lausekkeen, jossa kokonaislukuja olevien monikulmien sinit ja kosinit, kulmien summat ja kulmien erotukset on laajennettu. Identtisen yhtälön  $(\sin(x))^2 + (\cos(x))^2 = 1$  vuoksi mahdollisia ekvivalenttisia vastauksia on useita. Tämän vuoksi jokin vastaus voi olla erilainen kuin muissa julkaisuissa esitetty vastaus.

Joskus **tExpand()** pystyy suorittamaan haluamasi laskutoimituksen, kun oletusarvoinen trigonometrinen sievennys ei siihen pysty. **tExpand()** pyrkii kääntämään funktion **tCollect()** aikaansaamat transformaatiot. Joskus **tCollect()**-funktion soveltaminen **tExpand()**-funktion tulokseen, tai päin vastoin, kahdessa eri vaiheessa yksinkertaistaa lauseketta.

**Huomaa:** Astetilan skaalaus arvolla  $\pi/180$  häiritsee **tExpand()**-funktion kykyä tunnistaa laajennettavia muotoja. Parhaan tuloksen saavuttamiseksi **tExpand()**-funktioita tulee käyttää radiaanikulmatilassa.

$$\frac{\text{tExpand}(\sin(3 \cdot \phi))}{\text{tExpand}(\cos(\alpha - \beta))} = \frac{4 \cdot \sin(\phi) \cdot (\cos(\phi))^2 - \sin(\phi)}{\cos(\alpha) \cdot \cos(\beta) + \sin(\alpha) \cdot \sin(\beta)}$$

**Text***kehotemerkkijono[, Näytälippu]*

Ohjelmointikomento: Keskeyttää ohjelman ja näyttää merkkijonon *kehotemerkkijono* valintaruudussa.

Kun käyttäjä valitsee **OK**-näppäimen, ohjelman suoritus jatkuu.



Valinnainen *lippu*-argumentti voi olla mikä tahansa lauseke.

- Jos *NäytäLippu* jätetään pois, tai jos se sieventyy arvoksi **1**, tekstimuotoinen viesti lisätään laskimen historiaan.
- Jos *NäytäLippu* sieventyy arvoon **0**, tekstimuotoista viestiä ei lisätä historiaan.

Jos ohjelma vaatii käyttäjän kirjoittaman vastauksen, katso **Request**, sivu **161**, tai **RequestStr**, sivu **163**.

**Huomaa:** Tätä komentoa voi käyttää käyttäjän määrittämän ohjelman sisällä mutta ei funktion sisällä.

Määritä ohjelma, joka keskeytyy ja näyttää kunkin viidestä satunnaisluvusta valintaruudussa.

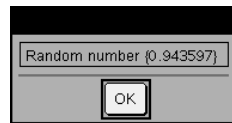
Paina mallin Prgm...EndPrgm jokaisen rivin lopussa näppäintä  näppäimen  sijaan. Tietokoneen näppäimistöllä **Alt**-näppäintä pidetään alhaalla ja painetaan **Enter**.

```
Define text_demo()=Prgm
  For i,1,5
    stringo:="Random number " &
string(rand(i))
    Text stringo
  EndFor
EndPrgm
```

Suorita ohjelma:

```
text_demo()
```

Esimerkki yhdestä valintaruudusta:

**TInterval** *ListA[,Frekv[,CTaso]]*

(Datalistan syöte)

**TInterval**  $\bar{x}$ ,*Sx,n[,CTaso]*

(Yhteenvetotilaston syöte)

Laskee  $t$ -luottamusvälin. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 191.)

Lisätietoja listassa olevien tyhjiin elementtien vaikutuksesta, katso Tyhjät elementitsivulla sivu 272.

Tulosmuuttuja	Kuvaus
stat.CLower, stat.CUpper	Tuntemattoman perusjoukon keskiarvon luottamusväli
stat. $\bar{x}$	Satunnaisesta normaalijakaumasta olevan datasekvenssin otoksen keskiarvo
stat.ME	Virhemarginaali
stat.df	Vapausasteet
stat. $\sigma_x$	Otoksen keskihajonta
stat.n	Otoksen keskiarvon sisältävän datasekvenssin pituus

## TInterval\_2Samp

**TInterval\_2Samp** *Lista1,Lista2[,Frekv1  
[,Frekv2[,CTaso[,Poolaus]]]]*

(Datalistan syöte)

**TInterval\_2Samp**  $\bar{x}1, Sx1, n1, \bar{x}2, Sx2, n2$   
[,CTaso[,Poolaus]]

(Yhteenvetotilaston syöte)

Laskee kahden otoksen  $t$ -luottamusvälin. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 191.)

*Poolaus=1* poolaa varianssit; *Poolaus=0* ei poolaa variansseja.

Lisätietoja listassa olevien tyhjiin elementtien vaikutuksesta, katso Tyhjät elementitsivulla sivu 272.



Tulosmuuttuja	Kuvaus
stat.CLower, stat.CUpper	Luottamusväli, joka sisältää jakauman luottamusvälin todennäköisyyden
stat. $\bar{x}1$ - $\bar{x}2$	Satunnaisesta normaalijakaumasta olevien datasekvenssien otosten keskiarvot
stat.ME	Virhemarginaali
stat.df	Vapausasteet
stat. $\bar{x}1$ , stat. $\bar{x}2$	Satunnaisesta normaalijakaumasta olevien datasekvenssien otosten keskiarvot
stat. $\sigma x1$ , stat. $\sigma x2$	<i>Lista 1:n</i> ja <i>Lista 2:n</i> otosten keskihajonnat
stat.n1, stat.n2	Otosten lukumäärä datasekvensseissä
stat.sp	Poolattu keskihajonta. Laskettu, kun <i>Poolaus</i> = KYLLÄ

## tmpCnv()

Katalogi > 

**tmpCnv**(*Laus* °*lämpYksikkö*, \_  
°*lämpYksikkö2*)  
⇒ *lauseke* °*lämpYksikkö2*

Muuntaa *Laus*:n määrittämän lämpötila-  
arvon yksiköstä toiseen. Kelpaavat  
lämpötilayksiköt ovat:

°CCelsius  
°FFahrenheit  
°KKelvin  
°RRankine

Asteen merkin ° saat Katalogin  
symboleista.

alaviivan \_ voi syöttää painamalla

 .

Esimerkiksi 100 °C muuntuu Fahrenheit-  
asteiksi 212 °F.

Jos haluat muuttaa lämpötila-alueen,  
käytä sen sijaan funktiota **ΔtmpCnv()**.

tmpCnv(100· °C, °F)	212· °F
tmpCnv(32· °F, °C)	0· °C
tmpCnv(0· °C, °K)	273.15· °K
tmpCnv(0· °F, °R)	459.67· °R

**Huomaa:** Lämpötilan yksiköt voi valita  
Katalogista.

## ΔtmpCnv()

Katalogi > 

**ΔtmpCnv**(*Laus* °*lämpYksikkö*, \_

Merkin Δ saat Katalogin symboleista.

## $\Delta$ tmpCnv()

Katalogi >  $^{\circ}$ lämpYksikkö2) $\Rightarrow$ lauseke  $^{\circ}$ lämpYksikkö2

**Huomaa:** Voit syöttää tämän funktion näppäimistöltä kirjoittamalla **del taTmpCnv (...)**.

Muuttaa *Laus*:n määrittämän lämpötila-alueen (kahden lämpötila-arvon välisen eron) yksiköstä toiseen. Kelpaavat lämpötilayksiköt ovat:

 $^{\circ}$ CCelsius $^{\circ}$ FFahrenheit $^{\circ}$ KKelvin $^{\circ}$ RRankine

Voit syöttää merkin  $^{\circ}$  valitsemalla sen symbolipaletista tai kirjoittamalla @d.

alaviivan \_ voi syöttää painamalla

 .

$1^{\circ}\text{C}$  ja  $1^{\circ}\text{K}$  ovat samansuuruisia, samoin kuin  $1^{\circ}\text{F}$  ja  $1^{\circ}\text{R}$ .  $1^{\circ}\text{C}$  on kuitenkin 9/5 Fahrenheit-asteen arvosta  $1^{\circ}\text{F}$ .

Esimerkiksi  $100^{\circ}\text{C}$ :n alue ( $0^{\circ}\text{C}$  -  $100^{\circ}\text{C}$ ) vastaa Fahrenheit-asteiden aluetta  $180^{\circ}\text{F}$ .

Jos haluat muuttaa jonkin lämpötila-arvon alueen sijaan, käytä funktiota **tmpCnv()**.

$\Delta$ tmpCnv( $100^{\circ}\text{C},^{\circ}\text{F}$ )	$180^{\circ}\text{F}$
$\Delta$ tmpCnv( $180^{\circ}\text{F},^{\circ}\text{C}$ )	$100^{\circ}\text{C}$
$\Delta$ tmpCnv( $100^{\circ}\text{C},^{\circ}\text{K}$ )	$100^{\circ}\text{K}$
$\Delta$ tmpCnv( $100^{\circ}\text{F},^{\circ}\text{R}$ )	$100^{\circ}\text{R}$
$\Delta$ tmpCnv( $1^{\circ}\text{C},^{\circ}\text{F}$ )	$1.8^{\circ}\text{F}$

**Huomaa:** Lämpötilan yksiköt voi valita Katalogista.

## tPdf()

Katalogi > 

**tPdf**( $XArvo,df$ ) $\Rightarrow$ luku, jos  $XArvo$  on luku,  
lista, jos  $XArvo$  on lista

Laskee todennäköisyystiheysfunktion (pdf) Studentin  $t$ -jakaumalle määritetyllä  $x$ :n arvolla ja määritetyillä vapausasteilla  $df$ .

**trace(neliömatriisi)**⇒lauseke

Laskee *neliömatriisin* jäljityksen (päälävistäjän kaikkien elementtien summan).

trace	$\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix}$	15
trace	$\begin{pmatrix} a & 0 \\ 1 & a \end{pmatrix}$	2·a

## Try (Kokeile)

**Try**

*lohko1*

**Else**

*lohko2*

**EndTry**

Suorittaa *lohko1*:n, ellei virhettä esiinny. Ohjelman suoritus siirtyy *lohko2*:een, jos *lohko1*:ssä esiintyy virhe. Järjestelmän muuttuja *errCode* sisältää virhekoodin, jotta ohjelma voi korjata virheen. Virhekoodien luettelo on esitetty kohdassa Virhekoodit ja viestit, sivu 282.

*lohko1* ja *lohko2* voivat olla joko yksi lauseke tai sarja lausekkeita, jotka on erotettu toisistaan kaksoispisteellä (:).

**Huomaa esimerkkiä syöttäessäsi:** Ohjeet monirivisten ohjelmien ja funktion määritysten syöttämisestä löytyvät tuotteen ohjekirjan Laskin-osiosta.

Esimerkki 2

Jos halaut nähdä komentojen **Try**, **ClrErr** ja **PassErr** toiminnan, syötä oikealla näkyvä *eigenvals()*-ohjelma. Suorita ohjelma suorittamalla kukin seuraavista lausekkeista.

$$\text{eigenvals}\left(\begin{pmatrix} -3 \\ -41 \\ 5 \end{pmatrix}, [-1 \ 2 \ -3.1]\right)$$

$$\text{eigenvals}\left(\begin{pmatrix} 1 & 2 & 3 \\ & & 1 \\ & & 2 \end{pmatrix}\right)$$

Define *progI()*=Prgm

Try

z:=z+1

Disp "z incremented."

Else

Disp "Sorry, z undefined."

EndTry

EndPrgm

Done

z:=1:*progI()*

z incremented.

Done

DelVar z:*progI()*

Sorry, z undefined.

Done

Define *eigenvals(a,b)*=Prgm

© Program *eigenvals(A,B)* displays eigenvalues of A·B

Try

Disp "A= ",a

Disp "B= ",b

Disp ""

Disp "Eigenvalues of A·B are:",eigVl(a\*b)

Else

If *errCode*=230 Then

**Huomaa:** Katso myös **ClrErr**, sivu 27, ja **PassErr**, sivu 142.

Disp "Error: Product of A-B must be a square matrix"

ClrErr

Else

PassErr

EndIf

EndTry

EndPrgm

## tTest

**tTest**  $\mu_0, Lista[, Frekv[, Hypot]]$

(Datalistan syöte)

**tTest**  $\mu_0, \bar{x}, sx, n, [Hypot]$

(Yhteenvetotilaston syöte)

Testaa hypoteesia yhden tuntemattoman perusjoukon keskiarvoon  $\mu$ , kun perusjoukon keskihajontaa  $\sigma$  ei tunneta. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 191.)

Testaa  $H_0: \mu = \mu_0$ , jonkin seuraavan vaihtoehdon suhteen:

Kun  $H_a: \mu < \mu_0$ , aseta *Hypot*<0

Kun  $H_a: \mu \neq \mu_0$  (oletus), aseta *Hypot*=0

Kun  $H_a: \mu > \mu_0$ , aseta *Hypot*>0

Lisätietoja listassa olevien tyhjien elementtien vaikutuksesta, katso Tyhjät elementtisivulla sivu 272.

Tulosmuuttuja	Kuvaus
stat.t	$(\bar{x} - \mu_0) / (stdev / \sqrt{n})$
stat.PVal	Alin merkitsevyystaso, jolla nollahypoteesi voidaan hylätä
stat.df	Vapausasteet

Tulosmuuttuja	Kuvaus
stat. $\bar{x}$	Listan sisältämän datasekvenssin otoksen keskiarvo
stat.sx	Datasekvenssin otoksen keskihajonta
stat.n	Otoksen koko

## tTest\_2Samp

Katalogi > 

**tTest\_2Samp** *List1,Lista2[,Frekv1[,Frekv2  
[,Hypot[,Poolaus]]]]*

(Datalistan syöte)

**tTest\_2Samp**  $\bar{x}1,sx1,n1,\bar{x}2,sx2,n2[,Hypot  
[,Poolaus]]$

(Yhteenvetotilaston syöte)

Suorittaa kahden otoksen *t*-testin. Tulosten yhteenvedo tallentuu *stat.results*-muuttujaan. (Katso sivu 191.)

Testaa  $H_0: \mu_1 = \mu_2$ , jonkin seuraavan vaihtoehdon suhteen:

Kun  $H_a: \mu_1 < \mu_2$ , aseta *Hypot*<0

Kun  $H_a: \mu_1 \neq \mu_2$  (oletus), aseta *Hypot*=0

Kun  $H_a: \mu_1 > \mu_2$ , aseta *Hypot*>0

*Poolaus*=1 poolaa varianssit

*Poolaus*=0 ei poolaa variansseja

Lisätietoja listassa olevien tyhjen elementtien vaikutuksesta, katso Tyhjät elementitsivulla sivu 272.

Tulosmuuttuja	Kuvaus
stat.t	Keskiarvojen erotukselle laskettu vakio-ohjearvo
stat.PVal	Alin merkitsevyytaso, jolla nollahypoteesi voidaan hylätä
stat.df	t-tilaston vapausasteet
stat. $\bar{x}1$ , stat. $\bar{x}2$	Listat 1:n ja Listat 2:n sisältämien datasekvenssien otosten keskiarvot
stat.sx1, stat.sx2	Listat 1:n ja Listat 2:n sisältämien datasekvenssien otosten keskihajonnat
stat.n1, stat.n2	Otosten koko
stat.sp	Poolattu keskihajonta. Laskettu, kun <i>Poolaus</i> =1

**tvmFV()**Katalogi > **tvmFV**( $N, I, PV, Pmt, [PpY], [CpY], [PmtAt]$ ) $\Rightarrow arvo$ tvmFV(120,5,0,-500,12,12) 77641.1

Talouselaskentafunktio, joka laskee rahan tulevan arvon.

**Huomaa:** TVM-funktioissa käytetyt argumentit on kuvattu TVM-argumenttien taulukossa, sivu 211. Katso myös **amortTbl()**, sivu 8.

**tvmI()**Katalogi > **tvmI**( $N, PV, Pmt, FV, [PpY], [CpY], [PmtAt]$ ) $\Rightarrow arvo$ tvmI(240,100000,-1000,0,12,12) 10.5241

Talouselaskentafunktio, joka laskee vuosikoron.

**Huomaa:** TVM-funktioissa käytetyt argumentit on kuvattu TVM-argumenttien taulukossa, sivu 211. Katso myös **amortTbl()**, sivu 8.

**tvmN()**Katalogi > **tvmN**( $I, PV, Pmt, FV, [PpY], [CpY], [PmtAt]$ ) $\Rightarrow arvo$ tvmN(5,0,-500,77641,12,12) 120.

Talouselaskentafunktio, joka laskee maksuerien lukumäärän.

**Huomaa:** TVM-funktioissa käytetyt argumentit on kuvattu TVM-argumenttien taulukossa, sivu 211. Katso myös **amortTbl()**, sivu 8.

**tvmPmt()**Katalogi > **tvmPmt**( $N, I, PV, FV, [PpY], [CpY], [PmtAt]$ ) $\Rightarrow arvo$ tvmPmt(60,4,30000,0,12,12) -552.496

Talouselaskentafunktio, joka laskee jokaisen maksuerän määrän.

**Huomaa:** TVM-funktioissa käytetyt argumentit on kuvattu TVM-argumenttien taulukossa, sivu 211. Katso myös **amortTbl()**, sivu 8.

## tvmPV()

**tvmPV**( $N, I, Pmt, FV, [PpY], [CpY], [PmtAt]$ ) $\Rightarrow$ arvo

tvmPV(48,4, 500,30000,12,12)      -3426.7

Talouselaskentafunktio, joka laskee nykyarvon.

**Huomaa:** TVM-funktioissa käytetyt argumentit on kuvattu TVM-argumenttien taulukossa, sivu 211. Katso myös **amortTbl()**, sivu 8.

TVM-argumentti*	Kuvaus	Datatyyppi
$N$	Maksuerien lukumäärä	reaaliluku
$I$	Vuosikorko	reaaliluku
$PV$	Nykyarvo	reaaliluku
$Pmt$	Maksun määrä	reaaliluku
$FV$	Tuleva arvo	reaaliluku
$PpY$	Maksuerien määrä vuodessa, oletusarvo=1	kokonaisluku > 0
$CpY$	Korkojaksoja vuodessa, oletusarvo=1	kokonaisluku > 0
$PmtAt$	Erääntyvän maksun määrä kunkin jakson lopussa tai alussa, oletusarvo=loppu	kokonaisluku (0=loppu, 1=alku)

\* Nämä rahan aika-arvon argumenttien nimet ovat samat kuin TVM-muuttujien nimet (kuten **tvm.pv** ja **tvm.pmt**), joita käytetään *Laskin*-sovelluksen talouselaskentatoiminnossa. Talouselaskentafunktioiden argumenttien arvot tai vastaukset eivät kuitenkaan tallennu TVM-muuttujiin.

## TwoVar

**TwoVar**  $X, Y, [Frekv] [, Luokka, Sisällyttä]$

Laskee kahden muuttujan tilastot. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 191.)

Kaikkien listojen on oltava samankokoisia *Sisällytä*-listaa lukuunottamatta.

$X$  ja  $Y$  ovat riippumattomien ja riippuvien muuttujien listoja.

*Frekv* on valinnainen frekvenssiarvojen lista. Jokainen *Frekv*:n elementti määrittää kunkin vastaavan datapisteen  $X$  ja  $Y$  esiintymisfrekvenssin. Oletusarvo on 1. Kaikkien elementtien on oltava kokonaislukuja 0.

*Luokka* on luokkakoodien lista vastaavalle  $X$ - ja  $Y$ -datalle.

*Sisällytä* on yhden tai usemman luokkakoodin lista. Vain ne datayksiköt, joiden luokkakoodi sisältyy tähän listaan, ovat mukana laskutoimituksessa.

Tyhjä elementti jossakin listassa  $X$ , *Frekv* tai *Luokka* saa aikaan, että kaikkien listojen vastaava elementti on tyhjä. Tyhjä elementti jossakin listassa  $X1$ - $X20$  saa aikaan, että kaikkien listojen vastaava elementti on tyhjä. Lisätietoja tyhjistä elementeistä, katso sivu 272.

Tulosmuuttuja	Kuvaus
stat. $\bar{x}$	$x$ :n arvojen keskiarvo
stat. $x$	$x$ :n arvojen summa
stat. $x^2$	$x^2$ :n arvojen summa
stat.sx	$x$ :n otoksen keskihajonta
stat. $x$	$x$ :n perusjoukon keskihajonta
stat.n	Datapisteiden lukumäärä
stat. $\bar{y}$	$y$ :n arvojen keskiarvo
stat. $\Sigma y$	$y$ :n arvojen summa
stat. $\Sigma y^2$	$y^2$ :n arvojen summa



Tulosmuuttuja	Kuvaus
stat.sy	y:n otoksen keskihajonta
stat.oy	y:n perusjoukon keskihajonta
stat.Σxy	x · y -arvojen summa
stat.r	Korrelaatiokerroin
stat.MinX	x:n arvojen minimi
stat.Q <sub>1</sub> X	x:n ensimmäinen neljännes
stat.MedianX	x:n mediaani
stat.Q <sub>3</sub> X	x:n 3. neljännes
stat.MaxX	x:n arvojen maksimi
stat.MinY	y:n arvojen minimi
stat.Q <sub>1</sub> Y	y:n ensimmäinen neljännes
stat.MedY	y:n mediaani
stat.Q <sub>3</sub> Y	x:n kolmas neljännes
stat.MaxY	y:n arvojen maksimi
stat.Σ(x- $\bar{x}$ ) <sup>2</sup>	x:n keskiarvon poikkeamien neliöiden summa
stat.Σ(y- $\bar{y}$ ) <sup>2</sup>	y:n keskiarvon poikkeamien neliöiden summa

## U

### unitV()

Katalogi > 

**yksikköV(Vektori I) ⇒ vektori**

Laskee joko rivi- tai sarakeyksikkövektorin riippuen *Vektori I*:n muodosta.

*Vektori I* :n on oltava joko yksirivinen matriisi tai yksisarakeinen matriisi.

$$\text{unitV}\left(\begin{bmatrix} a & b & c \end{bmatrix}\right) = \begin{bmatrix} \frac{a}{\sqrt{a^2+b^2+c^2}} & \frac{b}{\sqrt{a^2+b^2+c^2}} & \frac{c}{\sqrt{a^2+b^2+c^2}} \end{bmatrix}$$

$$\text{unitV}\left(\begin{bmatrix} 1 & 2 & 1 \end{bmatrix}\right) = \begin{bmatrix} \frac{\sqrt{6}}{6} & \frac{\sqrt{6}}{3} & \frac{\sqrt{6}}{6} \end{bmatrix}$$

$$\text{unitV}\left(\begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}\right) = \begin{bmatrix} \frac{\sqrt{14}}{14} \\ \frac{2\sqrt{14}}{14} \\ \frac{3\sqrt{14}}{14} \end{bmatrix}$$

Jos haluat nähdä koko vastauksen, paina ▲ ja siirrä sen jälkeen osoitinta painikkeilla ◀ ja ▶.

## unLock

**unLock***Muutt1*[, *Muutt2*] [, *Muutt3*] ...  
**unLock***Muutt*.

Vapauttaa määritetyt muuttujat tai muuttujaryhmän. Lukittuja muuttujia ei voi muokata eikä poistaa.

Katso **Lock**, sivu 114, ja **getLockInfo()**, sivu 89.

<i>a</i> :=65	65
Lock <i>a</i>	Done
getLockInfo( <i>a</i> )	1
<i>a</i> :=75	"Error: Variable is locked."
DelVar <i>a</i>	"Error: Variable is locked."
Unlock <i>a</i>	Done
<i>a</i> :=75	75
DelVar <i>a</i>	Done

## V

## varPop()

**varPop**(*List* [, *frekvList*])⇒*lauseke*

Laskee *List*an perusjoukon varianssin.

Jokainen *frekvList*an elementti näyttää *List*an vastaavien elementtien peräkkäisten esiintymien lukumäärän.

**Huomaa:** *List*assa tulee olla vähintään kaksi elementtiä.

Jos jokin elementti jommassakummassa listassa on tyhjä, kyseistä elementtiä ei huomioida, eikä toisessa listassa olevaa vastaavaa elementtiä myöskään huomioida. Lisätietoja tyhjistä elementeistä, katso sivu 272.

varPop({5,10,15,20,25,30})	875
	12
Ans·1.	72.9167

**varSamp()****varSamp(Lista[, frekvLista])** ⇒ lausekeLaskee *Listan* otosten varianssin.Jokainen *frekvLista* elementti näyttää *Listan* vastaavien elementtien peräkkäisten esiintymien lukumäärän.**Huomaa:** *Listassa* tulee olla vähintään kaksi elementtiä.

Jos jokin elementti jommassakummassa listassa on tyhjä, kyseistä elementtiä ei huomioida, eikä toisessa listassa olevaa vastaavaa elementtiä myöskään huomioida. Lisätietoja tyhjiä elementeistä, katso sivu 272.

**varSamp(MatriisiI[, frekvMatriisi])** ⇒ matriisiLaskee rivivektorin, joka sisältää *MatriisiI*:n kaikkien sarakkeiden otoksen varianssin.Jokainen *frekvMatriisin* elementti näyttää *MatriisiI*:n vastaavien elementtien peräkkäisten esiintymien lukumäärän.

Jos jokin elementti jommassakummassa matriisissa on tyhjä, kyseistä elementtiä ei huomioida, eikä toisessa matriisissa olevaa vastaavaa elementtiä myöskään huomioida. Lisätietoja tyhjiä elementeistä, katso sivu 272.

**Huomaa:** *MatriisiI*:ssä on oltava vähintään kaksi elementtiä. $\text{varSamp}(\{a,b,c\})$ 

$$\frac{a^2 - a \cdot (b+c) + b^2 - b \cdot c + c^2}{3}$$

 $\text{varSamp}(\{1,2,5,-6,3,-2\})$ 

31

2

 $\text{varSamp}(\{1,3,5\},\{4,6,2\})$ 

68

33

 $\text{varSamp}\left(\begin{pmatrix} 1 & 2 & 5 \\ -3 & 0 & 1 \\ .5 & .7 & 3 \end{pmatrix}\right)$ 

[4.75 1.03 4]

 $\text{varSamp}\left(\begin{pmatrix} -1.1 & 2.2 \\ 3.4 & 5.1 \\ -2.3 & 4.3 \end{pmatrix}, \begin{pmatrix} 6 & 3 \\ 2 & 4 \\ 5 & 1 \end{pmatrix}\right)$ 

[3.91731 2.08411]

**W****Wait****Wait aikaSekunteina**Keskeyttää toiminnon *aikaSekunteina* sekunnin ajaksi.

Odottaaksesi 4 sekuntia:

**Wait 4**



Odottaaksesi 1/2 sekuntia:

**Wait 0.5**

**Wait** on erityisen käyttökelpoinen ohjelmassa, jossa tarvitaan lyhyttä viivettä pyydettyjen tietojen saamiseksi käyttöön.

Argumentin *aikaSekunteina* on oltava lausekkeen muodossa, joka yksinkertaistaa desimaaliarvoon välillä 0–100. Komento pyöristää tämän arvon lähimpään 0,1 sekuntiin.

Peruuttaaksesi **Wait** käynnissä olevan,

- **Kämmenlaite:** Pidä -painiketta painettuna ja paina toistuvasti -painiketta.
- **Windows®:** Pidä **F12**-näppäintä pohjassa ja paina toistuvasti **Enter**.
- **Macintosh®:** Pidä **F5**-näppäintä pohjassa ja paina toistuvasti **Enter**.
- **iPad®:** Sovellus näyttää kehotuksen. Voit jatkaa odottamista tai peruuttaa.

**Huomaa:** Voit käyttää komentoa **Wait** käyttäjän määrittelemän ohjelman sisällä mutta ei funktion sisällä.

Odottaaksesi 1,3 sekuntia käyttäen muuttujaa *seklukema*:

**seklukema:=1.3**  
**Wait seklukema**

Tämä esimerkki sytyttää vihreän LED-valon päälle 0,5 sekunnin ajaksi ja sammuttaa sen sitten.

**Send "SET GREEN 1 ON"**  
**Wait 0.5**  
**Send "SET GREEN 1 OFF"**

## warnCodes ()

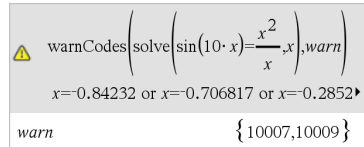
**warnCodes**(*Expr1*,  
*StatusVar*) $\Rightarrow$ *expression*

Laskee lausekkeen *Expr1*, antaa tuloksen ja varastoi mahdollisten luotujen varoitusten koodit *StatusVar* -luettelomuuttujaan. Jos varoituksia ei ole luotu, tämä funktio kohdistaa funktiolle *StatusVar* tyhjän luettelon.




*Expr1* voi olla mikä tahansa sallittu TI-Nspire™:n tai TI-Nspire™ CAS:n matemaattinen lauseke. Et voi käyttää komentoa tai tehtävää *Expr1*-lausekkeena.

*StatusVar*:n arvon on oltava sallittu muuttujan nimi.

Katso varoituskoodien ja niihin liittyvien viestien luettelo sivulla sivu 289.



```
warnCodes(solve(sin(10*x)=x^2/x), warn)
x=-0.84232 or x=-0.706817 or x=-0.2852
warn {10007,10009}
```

Jos haluat nähdä koko vastauksen, paina  ja siirrä sen jälkeen osoitinta painikkeilla  ja .

## when()

Katalogi > 

**when**(*Ehto*, *tosiTulos* [, *epätosiTulos*][, *tuntematonTulos*])  
⇒*lauseke*

Määrittää totuusarvon *tosiTulos*, *epätosiTulos* tai *tuntematonTulos* riippuen siitä, onko *Ehto* tosi, epätosi vai tuntematon. Antaa vastauksena syötteen, jos oikean vastauksen määrittämiseen on liian vähän argumentteja.

Jätä pois sekä *epätosiTulos* että *tuntematonTulos*, kun haluat, että lauseke määritetään vain alueella, jolla *Ehto* on tosi.

Käytä komentoa **undef** *epätosiTulos*, kun haluat määrittää lausekkeen, joka piirtyy vain jollekin välille.

**when()** on hyödyllinen komento rekursiivisten funktioiden määrittämisessä.

$\text{when}(x < 0, x + 3), x = 5$	undef
------------------------------------	-------

$\text{when}(n > 0, n \cdot \text{factorial}(n - 1), 1) \rightarrow \text{factorial}(n)$	Done
$\text{factorial}(3)$	6
3!	6

## While

Katalogi > 

**While** *Ehto*  
*Lohko*

**EndWhile**

Suorittaa *Lohkon* sisältämät lausekkeet, mikäli *Ehto* on tosi.

*Lohko* voi olla joko yksi lauseke tai sarja lausekkeita, jotka on erotettu toisistaan kaksoispisteellä (:).

**Huomaa esimerkkiä syöttäessäsi:** Ohjeet monirivisten ohjelmien ja funktion määrittysten syöttämisestä löytyvät tuotteen ohjekirjan Laskin-osiosta.

Define $\text{sum\_of\_recip}(n)$ = Func	
Local $i, \text{tempsum}$	
$1 \rightarrow i$	
$0 \rightarrow \text{tempsum}$	
While $i \leq n$	
$\text{tempsum} + \frac{1}{i} \rightarrow \text{tempsum}$	
$i + 1 \rightarrow i$	
EndWhile	
Return $\text{tempsum}$	
EndFunc	
	Done
$\text{sum\_of\_recip}(3)$	$\frac{11}{6}$
	6

X

*BooleanLaus1* **xor** *BooleanLaus2* antaa vastauksena *Boolean* lausekkeen *BooleanList1*

**xor** *BooleanList2* antaa vastauksena *Boolean* listan *BooleanMatriisi1*

**xor** *BooleanMatriisi2* antaa vastauksena *Boolean* matriisin

Määrittää totuusarvoksi tosi, jos *BooleanLaus1* on tosi ja *BooleanLaus2* on epätosi, tai päin vastoin.

Määrittää totuusarvoksi epätosi, jos kumpikin argumentti on tosi tai kumpikin on epätosi. Antaa vastauksena sievennetyn *Boolean* lausekkeen, jos kummarkaan argumentin totuusarvoa ei voi määrittää todeksi tai epätodeksi.

**Huomaa:** Katso **or**, sivu 139.

*Kokonaisluku1* **xor** *Kokonaisluku2*  $\Rightarrow$  *kokonaisluku*

Vertaa kahta reaalikokonaislukua bitti bitiltä **xor**-operaation avulla. Sisäisesti kumpikin kokonaisluku muunnetaan etumerkilliseksi, 64 bitin binaariluvuksi. Kun vastaavia bittejä verrataan, tulos on 1, jos jompikumpi bitti (mutta ei molemmat) on 1; tulos on 0, jos kumpikin bitti on 0 tai kumpikin bitti on 1. Laskettu arvo edustaa bittituloksia, ja se näkyy kantaluutilan mukaisesti.

Kokonaisluvut voi syöttää minkä tahansa luvun kantaluutilana. Binaarisen syötteen edelle tulee merkitä etumerkki 0b ja heksadesimaalisen syötteen edelle 0h. Jos etumerkkiä ei ole, kokonaislukuja käsitellään desimaalilukuina (kantaluuku 10).

true xor true	false
5>3 xor 3>5	true

Heksadesimaalisessa kantaluutilassa:

**Tärkeää:** Nolla, ei O-kirjain.

0h7AC36 xor 0h3D5F	0h79169
--------------------	---------

Binaarisessa kantaluutilassa:

0b100101 xor 0b100	0b100001
--------------------	----------

**Huomaa:** Binaarisessa syötteessä voi olla korkeintaan 64 numeroa (etuliitettä 0b ei lasketa). Heksadesimaalisessa syötteessä voi olla korkeintaan 16 numeroa.

Jos syötät desimaalikokonaisluvun, joka on etumerkillisen, 64 bitin binaarimuodon lukualueen ulkopuolella, laskin käyttää symmetristä modulo-operaatiota, jotta arvo saadaan oikealle alueelle. Lisätietoja, katso ▶**Base2**, sivu 18.

**Huomaa:** Katso **or**, sivu 139.

## Z

### zeros()

**zeros(Laus, Muutt)**⇒*lista*

**zeros(Laus, Muutt=Arvaus)**⇒*lista*

Määrittää listan muuttujan **Muutt** mahdollisista reaalilukuarvoista, joilla **Laus=0**. **zeros()** suorittaa tämän laskemalla yhtälön **explist(solve(Laus=0,Muutt),Muutt)**.

Joissakin tilanteissa funktion **zeros()** vastausmuoto on sopivampi kuin funktion **solve()**. Funktion **zeros()** vastausmuoto ei kuitenkaan pysty ilmaisemaan implisiittisiä ratkaisuja tai ratkaisuja, jotka vaativat epäyhtälöitä, eikä ratkaisuja, joihin ei sisälly muuttujaa **Muutt**.

**Huomaa:** Katso myös **cSolve()**, **cZeros()** ja **solve()**.

**zeros({Laus1, Laus2}, {muuttTaiArvaus1, muuttTaiArvaus2 [, ... ]})**⇒*matriisi*

Laskee mahdollisia reaalisia nollakohtia samanaikaisille algebrallisille yhtälöille, joissa jokainen **muuttTaiArvaus** määrittää ratkaistavan tuntemattoman arvon.

Voit halutessasi määrittää muuttujan ensimmäisen arvauksen. Jokaisen **muuttTaiArvaus**-komennon on oltava muodossa:

$$\text{zeros}(a \cdot x^2 + b \cdot x + c, x) \\ \left\{ \frac{\sqrt{b^2 - 4 \cdot a \cdot c} - b}{2 \cdot a}, \frac{-\left(\sqrt{b^2 - 4 \cdot a \cdot c} + b\right)}{2 \cdot a} \right\} \\ a \cdot x^2 + b \cdot x + c | x = \text{Ans}[2] \quad 0$$

$$\text{exact}\left(\text{zeros}\left(a \cdot \left(e^x + x\right) \cdot \left(\text{sign}(x) - 1\right), x\right)\right) \quad \left\{ \left[ \begin{array}{c} \square \\ \square \end{array} \right] \right\} \\ \text{exact}\left(\text{solve}\left(a \cdot \left(e^x + x\right) \cdot \left(\text{sign}(x) - 1\right) = 0, x\right)\right) \\ e^x + x = 0 \text{ or } x > 0 \text{ or } a = 0$$

muuttuja

– tai –

muuttuja = reaali tai ei-reaali luku

Esimerkiksi x kelpaa ja samoin x=3.

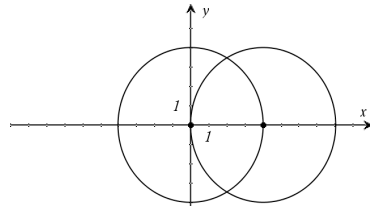
Jos kaikki lausekkeet ovat polynomeja, ja ET määrittää ensimmäisiä arvauksia, **zeros()** käyttää leksikaalista Gröbner/Buchbergerin eliminaatiomenetelmää yrittäessään määrittää kaikki reaaliset nollakohdat.

Oletetaan esimerkiksi, että origossa on ympyrän säde r, ja toinen ympyrän säde r on keskitetty kohtaan, jossa ensimmäinen ympyrä leikkaa positiivisen x-akselin. Määritä leikkauskohdat **zeros()**-funktion avulla.

Kuten on kuvattu merkinnällä r oikealla olevassa esimerkissä, samanaikaisissa polynomilausekkeissa voi olla ylimääräisiä muuttujia, joilla ei ole arvoja, vaan ne edustavat tiettyjä numeerisia arvoja, jotka voidaan korvata myöhemmin.

Jokainen tulosmatriisin rivi edustaa vaihtoehtoista nollakohtaa, jossa komponentit on järjestetty samalla tavalla kuin *MuuttTaiArvaus*-listassa. Jos haluat määrittää rivin juuren, indeksoi matriisi [*riveittäin*].

Voit ottaa mukaan (tai käyttää niitä tilalla) myös tuntemattomia muuttujia, joita ei esiinny lausekkeissa. Voit esimerkiksi ottaa mukaan z:n tuntemattomana muuttujana ulottaaksesi edellisen esimerkin säteen r kahteen samansuuntaiseen leikkaavaan sylinteriin. Sylintereiden nollakohdat havainnollistavat, miten nollakohtien sarjat voivat sisältää mielivaltaisia vakioita, jotka ovat muotoa ck, jossa k on kokonaisluku väliä 1-255.



$$\text{zeros}\left(\left\{x^2+y^2-r^2,(x-r)^2+y^2-r^2\right\},\{x,y\}\right)$$

$\frac{r}{2}$	$\frac{-\sqrt{3}\cdot r}{2}$
$\frac{r}{2}$	$\frac{\sqrt{3}\cdot r}{2}$

Määritä rivin 2 juuri:

$$\text{Ans}[2]$$

$\frac{r}{2}$	$\frac{\sqrt{3}\cdot r}{2}$
---------------	-----------------------------

$$\text{zeros}\left(\left\{x^2+y^2-r^2,(x-r)^2+y^2-r^2\right\},\{x,y,z\}\right)$$

$\frac{r}{2}$	$\frac{-\sqrt{3}\cdot r}{2}$	c1
$\frac{r}{2}$	$\frac{\sqrt{3}\cdot r}{2}$	c1



Polynomisarjoissa laskutoimituksen suoritus aika tai muistin käyttö voivat riippua merkittävästi tuntemattomien muuttujien järjestyksestä. Jos ensimmäinen valintasi kuluttaa muistia, tai et jaksaa odottaa vastausta, yritä järjestää muuttujat uudelleen lausekkeisiin ja/tai *MuuttTaiArvaus*-listaan.

Jos et ota mukaan arvauksia, ja jokin lausekkeista on ei-polynominen missä tahansa muuttujassa, mutta kaikki muuttujat ovat lineaarisia kaikissa tuntemattomissa muuttujissa, **zeros()** käyttää Gaussin eliminointia yrittäessään määrittää kaikki reaaliset nollakohdat.

Jos yhtälöryhmä ei ole polynominen kaikilta muuttujiltaan eikä lineaarinen tuntemattomilta muuttujiltaan, **zeros()** määrittää korkeintaan yhden nollakohdan käyttäen likimääräistä iteratiivista menetelmää. Tässä tuntemattomien muuttujien lukumäärän on oltava sama kuin lausekkeiden lukumäärä, ja kaikkien muiden lausekkeiden sisältämien muuttujien on sievennyttävä luvuiksi.

Jokainen tuntematon muuttuja alkaa arvausarvostaan, mikäli se on määritetty; muussa tapauksessa se alkaa arvosta 0.0.

Arvausten avulla voit etsiä lisää nollakohtia yksi kerrallaan. Suppenemista varten arvauksen on mahdollisesti oltava melko lähellä nollakohtaa.

$$\text{zeros}\left(\left\{x+e^z \cdot y-1, x-y-\sin(z)\right\},\{x,y\}\right)$$

$\frac{e^z \cdot \sin(z)+1}{e^z+1}$	$\frac{-(\sin(z)-1)}{e^z+1}$
-------------------------------------	------------------------------

$$\text{zeros}\left(\left\{e^z \cdot y-1, y-\sin(z)\right\},\{y,z\}\right)$$

0.041458	3.18306
0.001871	6.28131
4.76E-11	1796.99
2.E-13	254.469

$$\text{zeros}\left(\left\{e^z \cdot y-1, y-\sin(z)\right\},\{y,z=2 \cdot \pi\}\right)$$

0.001871	6.28131
----------	---------

## zInterval

**zInterval**  $\sigma$ ,*Lista*[,*Frekv*][,*CTaso*]

(Datalistan syöte)

**zInterval**  $\sigma$ , $\bar{x}$ ,*n* [,*CTaso*]

(Yhteenvetotilaston syöte)

Laskee  $z$ -luottamusvälin. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 191.)

Lisätietoja listassa olevien tyhjiä elementtien vaikutuksesta, katso Tyhjät elementitsivulla sivu 272.

Tulosmuuttuja	Kuvaus
stat.CLower, stat.CUpper	Tuntemattoman perusjoukon keskiarvon luottamusväli
stat. $\bar{x}$	Satunnaisesta normaalijakaumasta olevan datasekvenssin otoksen keskiarvo
stat.ME	Virhemarginaali
stat.sx	Otoksen keskihajonta
stat.n	Otoksen keskiarvon sisältävän datasekvenssin pituus
stat. $\sigma$	Datasekvenssin <i>Lista</i> tunnettu perusjoukon keskihajonta

## zInterval\_1Prop

zInterval\_1Prop  $x, n$  [,CTaso]

Laskee yhden osuuden  $z$ -luottamusvälin. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 191.)

$x$  on ei-negatiivinen kokonaisluku.

Lisätietoja listassa olevien tyhjiä elementtien vaikutuksesta, katso Tyhjät elementitsivulla sivu 272.

Tulosmuuttuja	Kuvaus
stat.CLower, stat.CUpper	Luottamusväli, joka sisältää jakauman luottamusvälin todennäköisyyden
stat. $\hat{p}$	Laskettu onnistumisten osuus
stat.ME	Virhemarginaali
stat.n	Otosten lukumäärä datasekvenssissä

**zInterval\_2Prop**  $x1, n1, x2, n2[, CTaso]$

Laskee kahden osuuden  $z$ -luottamusvälin. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 191.)

$x1$  ja  $x2$  ovat ei-negatiivisia kokonaislukuja.

Lisätietoja listassa olevien tyhjiin elementtien vaikutuksesta, katso Tyhjät elementtsivulla sivu 272.

Tulosmuuttuja	Kuvaus
stat.CLower, stat.CUpper	Luottamusväli, joka sisältää jakauman luottamusvälin todennäköisyyden
stat. $\hat{p}$ Diff	Osuuksien välinen laskettu erotus
stat.ME	Virhemarginaali
stat. $\hat{p}$ 1	Arvio ensimmäisen näytteen osuudesta
stat. $\hat{p}$ 2	Arvio toisen näytteen osuudesta
stat.n1	Otoksen koko datasekvenssissä 1
stat.n2	Otoksen koko datasekvenssissä 2

**zInterval\_2Samp**  $\sigma_1, \sigma_2, Lista1, Lista2$   
 $[, Frekv1[, Frekv2[, CTaso]]]$

(Datalistan syöte)

**zInterval\_2Samp**  $\sigma_1, \sigma_2, \bar{x}1, n1, \bar{x}2, n2[, CTaso]$

(Yhteenvetotilaston syöte)

Laskee kahden näytteen  $z$ -luottamusvälin. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 191.)

Lisätietoja listassa olevien tyhjiin elementtien vaikutuksesta, katso Tyhjät elementtsivulla sivu 272.

Tulosmuuttuja	Kuvaus
stat.CLower, stat.CUpper	Luottamusväli, joka sisältää jakauman luottamusvälin todennäköisyyden
stat. $\bar{x}1$ - $\bar{x}2$	Satunnaisesta normaalijakaumasta olevien datasekvenssien otosten keskiarvot
stat.ME	Virhemarginaali
stat. $\bar{x}1$ , stat. $\bar{x}2$	Satunnaisesta normaalijakaumasta olevien datasekvenssien otosten keskiarvot
stat. $\sigma x1$ , stat. $\sigma x2$	Listat 1:n ja Listat 2:n otosten keskihajonnat
stat.n1, stat.n2	Otosten lukumäärä datasekvensseissä
stat.r1, stat.r2	Datasekvenssien Listat 1 ja Listat 2 tunnetut perusjoukon keskihajonnat

## zTest

Katalogi > 

**zTest**  $\mu0, \sigma, Lista, [Frekv[, Hypot]]$

(Datalistan syöte)

**zTest**  $\mu0, \sigma, \bar{x}, n[, Hypot]$

(Yhteenvetotilaston syöte)

Suorittaa z-testin frekvenssillä *frekvlista*. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 191.)

Testaa  $H_0: \mu = \mu0$ , jonkin seuraavan vaihtoehdon suhteen:

Kun  $H_a: \mu < \mu0$ , aseta *Hypot*<0

Kun  $H_a: \mu \neq \mu0$  (oletus), aseta *Hypot*=0

Kun  $H_a: \mu > \mu0$ , aseta *Hypot*>0

Lisätietoja listassa olevien tyhjen elementtien vaikutuksesta, katso Tyhjät elementitsivulla sivu 272.

Tulosmuuttuja	Kuvaus
stat.z	$(\bar{x} - \mu0) / (\sigma / \text{sqrt}(n))$
stat.P Value	Pienin todennäköisyys, jolla nollahypoteesi voidaan hylätä
stat. $\bar{x}$	Listan sisältämän datasekvenssin otoksen keskiarvo
stat.sx	Datasekvenssin otoksen keskihajonta. Lasketaan vain <i>Data</i> -syötteelle.

Tulosmuuttuja	Kuvaus
stat.n	Otoksen koko

## zTest\_1Prop

Katalogi > 

### zTest\_1Prop $p_0, x, n[, Hypot]$

Laskee yhden osuuden  $z$ -testin. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 191.)

$x$  on ei-negatiivinen kokonaisluku.

Testaa  $H_0: p = p_0$  jonkin seuraavan vaihtoehdon suhteen:

Kun  $H_a: p > p_0$ , aseta *Hypot*>0

Kun  $H_a: p \neq p_0$  (oletus), aseta *Hypot*=0

Kun  $H_a: p < p_0$ , aseta *Hypot*<0

Lisätietoja listassa olevien tyhjien elementtien vaikutuksesta, katso Tyhjät elementisivulla sivu 272.

Tulosmuuttuja	Kuvaus
stat.p0	Perusjoukon osuuden hypoteesi-arvo
stat.z	Osuudelle laskettu vakio-ohjearvo
stat.PVal	Alin merkitsevyytaso, jolla nollahypoteesi voidaan hylätä
stat. $\hat{p}$	Arvioitu otoksen osuus
stat.n	Otoksen koko

## zTest\_2Prop

Katalogi > 

### zTest\_2Prop $x_1, n_1, x_2, n_2[, Hypot]$

Laskee kahden osuuden  $z$ -testin. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 191.)

$x_1$  ja  $x_2$  ovat ei-negatiivisia kokonaislukuja.

Testaa  $H_0: p_1 = p_2$  jonkin seuraavan vaihtoehdon suhteen:

Kun  $H_a: p1 > p2$ , aseta  $Hypot > 0$

Kun  $H_a: p1 \neq p2$  (oletus), aseta  $Hypot = 0$

Kun  $H_a: p < p0$ , aseta  $Hypot < 0$

Lisätietoja listassa olevien tyhjiin elementtien vaikutuksesta, katso Tyhjät elementitsivulla sivu 272.

Tulosmuuttuja	Kuvaus
stat.z	Osuuksien erotukselle laskettu vakio-ohjearvo
stat.PVal	Alin merkitsevyytaso, jolla nollahypoteesi voidaan hylätä
stat.p1	Arvio ensimmäisen näytteen osuudesta
stat.p2	Arvio toisen näytteen osuudesta
stat.p	Poolattu arvio otoksen osuudesta
stat.n1, stat.n2	Kokeissa 1 ja 2 otettujen otosten lukumäärä

**zTest\_2Samp**  $\sigma_1, \sigma_2, Lista1, Lista2[, Frekv1$   
 $[, Frekv2[, Hypot]]]$

(Datalistan syöte)

**zTest\_2Samp**  $\sigma_1, \sigma_2, \bar{x}1, n1, \bar{x}2, n2[, Hypot]$

(Yhteenvetotilaston syöte)

Laskee kahden otoksen  $z$ -testin. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 191.)

Testaa  $H_0: \mu_1 = \mu_2$ , jonkin seuraavan vaihtoehdon suhteen:

Kun  $H_a: \mu_1 < \mu_2$ , aseta  $Hypot < 0$

Kun  $H_a: \mu_1 \neq \mu_2$  (oletus), aseta  $Hypot = 0$

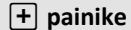
Kun  $H_a: \mu_1 > \mu_2$ ,  $Hypot > 0$

Lisätietoja listassa olevien tyhjiin elementtien vaikutuksesta, katso Tyhjät elementitsivulla sivu 272.

<b>Tulosmuuttuja</b>	<b>Kuvaus</b>
stat.z	Keskiarvojen erotukselle laskettu vakio-ohjearvo
stat.PVal	Alin merkitsevyystaso, jolla nollahypoteesi voidaan hylätä
stat. $\bar{x}1$ , stat. $\bar{x}2$	<i>Lista1</i> :n ja <i>Lista2</i> :n sisältämien datasekvenssien otosten keskiarvot
stat.sx1, stat.sx2	<i>Lista1</i> :n ja <i>Lista2</i> :n sisältämien datasekvenssien otosten keskihajonnat
stat.n1, stat.n2	Otosten koko

# Symbolit

## + (yhteenlasku)



$$Laus1 + Laus2 \Rightarrow lauseke$$

Laskee kahden argumentin summan.

56	56
56+4	60
60+4	64
64+4	68
68+4	72

$$Lista1 + Lista2 \Rightarrow lista$$

$$Matriisi2 \Rightarrow matriisi$$

Määrittää listan (tai matriisin), joka sisältää *Lista1*:n ja *Lista2*:n (tai *Matriisi1*:n ja *Matriisi2*:n) vastaavien elementtien summat.

$\left\{ 22, \pi, \frac{\pi}{2} \right\} \rightarrow I1$	$\left\{ 22, \pi, \frac{\pi}{2} \right\}$
$\left\{ 10, 5, \frac{\pi}{2} \right\} \rightarrow I2$	$\left\{ 10, 5, \frac{\pi}{2} \right\}$

$$I1 + I2 \quad \left\{ 32, \pi + 5, \pi \right\}$$

$$Ans + \left\{ \pi, -5, \pi \right\} \quad \left\{ \pi + 32, \pi, 0 \right\}$$

Argumenttien tulee olla mitoiltaan samanlaisia.

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad \begin{bmatrix} a+1 & b \\ c & d+1 \end{bmatrix}$$

$$Laus + Lista1 \Rightarrow lista$$

$$Lista1 + Laus \Rightarrow lista$$

Määrittää listan, joka sisältää *Laus*:n ja *Lista1*:n kunkin elementin summat.

$15 + \left\{ 10, 15, 20 \right\}$	$\left\{ 25, 30, 35 \right\}$
$\left\{ 10, 15, 20 \right\} + 15$	$\left\{ 25, 30, 35 \right\}$

$$Laus + Matriisi1 \Rightarrow matriisi$$

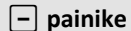
$$Matriisi1 + Laus \Rightarrow matriisi$$

Laskee matriisin, jossa *Laus* on lisätty jokaiseen elementtiin *Matriisi1*:n diagonaalimatriisissa. *Matriisi1* :n on oltava neliö.

$$20 + \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \quad \begin{bmatrix} 21 & 2 \\ 3 & 24 \end{bmatrix}$$

**Huomaa:** Käytä merkintää .+ (piste plus), kun haluat lisätä lausekkeen jokaiseen elementtiin.

## -(vähennyslasku)



$$Laus1 - Laus2 \Rightarrow lauseke$$

Laskee *Laus1* miinus *Laus2*.

6-2	4
$\pi - \frac{\pi}{6}$	$\frac{5 \cdot \pi}{6}$



**-(vähennyslasku)****[-] painike** $Listal - Lista2 \Rightarrow lista$ 

$$\left\{ 22, \pi, \frac{\pi}{2} \right\} - \left\{ 10, 5, \frac{\pi}{2} \right\} = \{ 12, \pi - 5, 0 \}$$

 $Matriisi1 - Matriisi2 \Rightarrow matriisi$ 

$$\begin{bmatrix} 3 & 4 \\ -1 & 2 \end{bmatrix} - \begin{bmatrix} 1 & 2 \\ 2 & 2 \end{bmatrix} = \begin{bmatrix} 2 & 2 \\ -3 & 0 \end{bmatrix}$$

Vähentää  $Listal$ :n (tai  $Matriisi2$ :n) jokaisen elementin  $Listal$ :n (tai  $Matriisi1$ :n) vastaavasta elementistä ja antaa tuloksena vastaukset.

Argumenttien tulee olla mitoiltaan samanlaisia.

 $Laus - Listal \Rightarrow lista$ 

$$15 - \{ 10, 15, 20 \} = \{ 5, 0, -5 \}$$

 $Listal - Laus \Rightarrow lista$ 

$$\{ 10, 15, 20 \} - 15 = \{ -5, 0, 5 \}$$

Vähentää jokaisen  $Listal$ :n elementin lausekkeesta  $Laus$  tai vähentää lausekkeen  $Laus$  jokaisesta  $Listal$ :n elementistä ja antaa vastauksena tuloslistan.

 $Laus - Matriisi1 \Rightarrow matriisi$ 

$$20 - \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} = \begin{bmatrix} 19 & -2 \\ -3 & 16 \end{bmatrix}$$

 $Matriisi1 - Laus \Rightarrow matriisi$ 

$Laus - Matriisi1$  laskee matriisin  $Laus$  kertaa identtinen matriisi miinus  $Matriisi1$ .  $Matriisi1$ :n on oltava neliö.

$Matriisi1 - Laus$  laskee lausekkeen  $Laus$  matriisin kerrottuna identtisellä matriisilla, joka on vähennetty  $Matriisi1$ :stä.  $Matriisi1$ :n on oltava neliö.

**Huomaa:** Käytä merkintää  $-$  (piste miinus), kun haluat vähentää lausekkeen jokaisesta elementistä.

**·(kertolasku)****[x] painike** $Laus1 \cdot Laus2 \Rightarrow lauseke$ 

$$2 \cdot 3,45 = 6,9$$

Laskee kahden argumentin tulon.

$$x \cdot y \cdot x = x^2 \cdot y$$

 $Listal \cdot Lista2 \Rightarrow lista$ 

$$\{ 1, 2, 3 \} \cdot \{ 4, 5, 6 \} = \{ 4, 10, 18 \}$$

Luo listan, joka sisältää  $Listal$ :n ja  $Listal$ :n vastaavien elementtien tulot.

$$\left\{ \frac{2}{a}, \frac{3}{2} \right\} \cdot \left\{ a^2, \frac{b}{3} \right\} = \left\{ 2 \cdot a, \frac{b}{2} \right\}$$

Listojen tulee olla mitoiltaan samanlaisia.

**· (kertolasku)****x** painike*Matriisi1* · *Matriisi2* ⇒ *matriisi*Laskee *Matriisi1*:n ja *Matriisi2*:n matriisitulon.*Matriisi1*:n sarakkeiden lukumäärän on oltava sama kuin *Matriisi2*:n rivien lukumäärä.*Laus* · *Lista1* ⇒ *lista**Lista1* · *Laus* ⇒ *lista*Määrittää listan, joka sisältää lausekkeen *Laus* ja kunkin *Lista1*:n elementin tulon.*Laus* · *Matriisi1* ⇒ *matriisi*  
*Laus* ⇒ *matriisi*Määrittää matriisin, joka sisältää lausekkeen *Laus* ja kunkin *Matriisi1*:n elementin tulon.**Huomaa:** Käytä merkintää  $\cdot$  (piste kerro), kun haluat kertoa lausekkeen jokaisella elementillä.

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix} \cdot \begin{bmatrix} a & d \\ b & e \\ c & f \end{bmatrix}$$


---


$$\begin{bmatrix} a+2\cdot b+3\cdot c & d+2\cdot e+3\cdot f \\ 4\cdot a+5\cdot b+6\cdot c & 4\cdot d+5\cdot e+6\cdot f \end{bmatrix}$$

$$\pi \cdot \{4,5,6\} \qquad \{4\cdot\pi, 5\cdot\pi, 6\cdot\pi\}$$

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \cdot 0.01 \qquad \begin{bmatrix} 0.01 & 0.02 \\ 0.03 & 0.04 \end{bmatrix}$$


---


$$\lambda \cdot \text{identity}(3) \qquad \begin{bmatrix} \lambda & 0 & 0 \\ 0 & \lambda & 0 \\ 0 & 0 & \lambda \end{bmatrix}$$

**/ (jakolasku)****÷** painike*Laus1* / *Laus2* ⇒ *lauseke*Laskee osamäärän *Laus1* jaettuna *Laus2*:lla.**Huomaa:** Katso myös **Murtolukumalli**, sivu 1.*Lista1* / *Lista2* ⇒ *lista*Määrittää listan, joka sisältää osamäärät laskutoimituksista *Lista1* jaettuna *Lista2*:lla.

Listojen tulee olla mitoiltaan samanlaisia.

*Laus* / *Lista1* ⇒ *lista**Lista1* / *Laus* ⇒ *lista*

$$\frac{2}{3.45} \qquad 0.57971$$


---


$$\frac{x^3}{x} \qquad x^2$$

$$\frac{\{1,2,3\}}{\{4,5,6\}} \qquad \left\{0.25, \frac{2}{5}, \frac{1}{2}\right\}$$

$$\frac{a}{\{3, a, \sqrt{a}\}} \qquad \left\{\frac{a}{3}, 1, \sqrt{a}\right\}$$


---


$$\frac{\{a, b, c\}}{a \cdot b \cdot c} \qquad \left\{\frac{1}{b \cdot c}, \frac{1}{a \cdot c}, \frac{1}{a \cdot b}\right\}$$

## / (jakolasku)

 painike

Määrittää listan, joka sisältää osamäärät laskutoimituksista *Laus* jaettuna *List1*:llä tai *List1* jaettuna lausekkeella *Laus*.

*Matriisi1* / *Laus*  $\Rightarrow$  *matriisi*

$$\frac{\begin{matrix} a & b & c \\ a \cdot b \cdot c \end{matrix}}{\begin{matrix} \frac{1}{b \cdot c} & \frac{1}{a \cdot c} & \frac{1}{a \cdot b} \end{matrix}}$$

Laskee matriisiin, joka sisältää osamäärät laskutoimituksesta *Matriisi1*/*Laus*.

**Huomaa:** Käytä merkintää . / (piste jaa), kun haluat jakaa lausekkeen jokaisella elementillä.

## ^ (potenssi)

 painike

*Laus1* ^ *Laus2*  $\Rightarrow$  *lauseke*

$$4^2 \qquad 16$$

*List1* ^ *List2*  $\Rightarrow$  *lista*

$$\{a,2,c\}^{\{1,b,3\}} \qquad \{a,2^b,c^3\}$$

Laskee ensimmäisen argumentin korotettuna toisen argumentin potenssiin.

**Huomaa:** Katso myös **Eksponenttimalli**, sivu 1.

Jos kyseessä on lista, laskee *List1*:n elementit korotettuna *List2*:n vastaavien elementtien potenssiin.

Reaalilukujen alueella murtolukupotenssit, joilla on supistetut eksponentit ja parittomat nimittäjät, käyttävät reaalista aluetta versus pääalue kompleksitilassa.

*Laus* ^ *List1*  $\Rightarrow$  *lista*

$$p^{\{a,2,-3\}} \qquad \left\{ p^a, p^2, \frac{1}{p^3} \right\}$$

Laskee lausekkeen *Laus* korotettuna *List1*:n elementtien potenssiin.

*List1* ^ *Laus*  $\Rightarrow$  *lista*

$$\{1,2,3,4\}^{-2} \qquad \left\{ \frac{1}{4}, \frac{1}{9}, \frac{1}{16} \right\}$$

Laskee *List1*:n elementit korotettuna *Laus*:n potenssiin.

**^ (potenssi)** **^ painike**

*neliomatriisi* 1 ^ kokonaisluku ⇒  
*matriisi*

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}^2 \quad \begin{bmatrix} 7 & 10 \\ 15 & 22 \end{bmatrix}$$

Laskee *neliomatriisi* 1:n korotettuna kokonaisluvun kokonaisluku potenssiin.

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}^{-1} \quad \begin{bmatrix} -2 & 1 \\ 3 & -1 \\ 2 & 2 \end{bmatrix}$$

*neliomatriisi* 1:n on oltava neliomatriisi.

Jos kokonaisluku = -1, laskee käänteismatriisin.

Jos kokonaisluku < -1, laskee käänteismatriisin korotettuna sopivaan positiiviseen potenssiin.

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}^{-2} \quad \begin{bmatrix} 11 & -5 \\ 2 & 2 \\ -15 & 7 \\ 4 & 4 \end{bmatrix}$$

**x2 (neliö)** **x<sup>2</sup> painike**

*Laus* 1<sup>2</sup> ⇒ lauseke

$$4^2 \quad 16$$

Laskee argumentin neliön.

$$\{2,4,6\}^2 \quad \{4,16,36\}$$

*Lista* 1<sup>2</sup> ⇒ lista

$$\begin{bmatrix} 2 & 4 & 6 \\ 3 & 5 & 7 \\ 4 & 6 & 8 \end{bmatrix}^2 \quad \begin{bmatrix} 40 & 64 & 88 \\ 49 & 79 & 109 \\ 58 & 94 & 130 \end{bmatrix}$$

Laskee listan, joka sisältää *Lista* 1:n elementtien neliöt.

$$\begin{bmatrix} 2 & 4 & 6 \\ 3 & 5 & 7 \\ 4 & 6 & 8 \end{bmatrix}^{\wedge 2} \quad \begin{bmatrix} 4 & 16 & 36 \\ 9 & 25 & 49 \\ 16 & 36 & 64 \end{bmatrix}$$

*neliomatriisi* 1<sup>2</sup> ⇒ matriisi

Laskee *neliomatriisi* 1:n matriisinelion. Tämä ei ole sama kuin kunkin elementin neliön laskeminen. Käytä merkintää .^2, kun haluat laskea jokaisen elementin neliön.

**.+ (piste lisää)** **.+ painikkeet**

*Matriisi* 1 .+ *Matriisi* 2 ⇒ matriisi

$$\begin{bmatrix} a & 2 \\ b & 3 \end{bmatrix} .+ \begin{bmatrix} c & 4 \\ 5 & d \end{bmatrix} \quad \begin{bmatrix} a+c & 6 \\ b+5 & d+3 \end{bmatrix}$$

*Laus* .+ *Matriisi* 1 ⇒ matriisi

*Matriisi* 1 .+ *Matriisi* 2 laskee matriisin, joka on *Matriisi* 1:n ja *Matriisi* 2:n vastaavien elementtiparien summa.

$$x .+ \begin{bmatrix} c & 4 \\ 5 & d \end{bmatrix} \quad \begin{bmatrix} x+c & x+4 \\ x+5 & x+d \end{bmatrix}$$

*Laus* .+ *Matriisi* 1 laskee matriisin, joka on *Laus*:n ja kunkin *Matriisi* 1:n elementin summa.

**.- (piste-erotus)**  painikkeet*Matriisi1* .- *Matriisi2*  $\Rightarrow$  *matriisi*

$$\begin{bmatrix} a & 2 \\ b & 3 \end{bmatrix} \cdot \begin{bmatrix} c & 4 \\ d & 5 \end{bmatrix} = \begin{bmatrix} a-c & -2 \\ b-d & -2 \end{bmatrix}$$

*Laus* .-*Matriisi1*  $\Rightarrow$  *matriisi*

$$x \cdot \begin{bmatrix} c & 4 \\ d & 5 \end{bmatrix} = \begin{bmatrix} x-c & x-4 \\ x-d & x-5 \end{bmatrix}$$

*Matriisi1* .-*Matriisi2* laskee matriisin, joka on *Matriisi1*:n ja *Matriisi2*:n vastaavien elementtiparien välinen erotus.

*Laus* .-*Matriisi1* laskee matriisin, joka on *Laus*:n ja kunkin *Matriisi1*:n elementin erotus.

**.· (pistetulo)**  painikkeet*Matriisi1* .· *Matriisi2*  $\Rightarrow$  *matriisi*

$$\begin{bmatrix} a & 2 \\ b & 3 \end{bmatrix} \cdot \begin{bmatrix} c & 4 \\ 5 & d \end{bmatrix} = \begin{bmatrix} a \cdot c & 8 \\ 5 \cdot b & 3 \cdot d \end{bmatrix}$$

*Laus* .·*Matriisi1*  $\Rightarrow$  *matriisi*

$$x \cdot \begin{bmatrix} a & b \\ c & d \end{bmatrix} = \begin{bmatrix} a \cdot x & b \cdot x \\ c \cdot x & d \cdot x \end{bmatrix}$$

*Matriisi1* .· *Matriisi2* laskee matriisin, joka on *Matriisi1*:n ja *Matriisi2*:n vastaavien elementtiparien tulo.

*Laus* .· *Matriisi1* laskee matriisin, joka sisältää *Laus*:n ja kunkin *Matriisi1*:n elementin tulot.

**./ (pisteosamäärä)**  painikkeet*Matriisi1* ./ *Matriisi2*  $\Rightarrow$  *matriisi*

$$\begin{bmatrix} a & 2 \\ b & 3 \end{bmatrix} \cdot \left( \begin{bmatrix} c & 4 \\ 5 & d \end{bmatrix} \right) = \begin{bmatrix} \frac{a}{c} & \frac{1}{2} \\ \frac{a}{c} & \frac{2}{2} \end{bmatrix}$$

*Laus* ./ *Matriisi1*  $\Rightarrow$  *matriisi*

$$\begin{bmatrix} a & 2 \\ b & 3 \end{bmatrix} \cdot \left( \begin{bmatrix} c & 4 \\ 5 & d \end{bmatrix} \right) = \begin{bmatrix} \frac{b}{5} & \frac{3}{d} \end{bmatrix}$$



*Matriisi1* ./ *Matriisi2* laskee matriisin, joka on *Matriisi1*:n ja *Matriisi2*:n vastaavien elementtiparien osamäärä.

*Laus* ./ *Matriisi1* laskee matriisin, joka on lausekkeen *Laus* ja kunkin *Matriisi1*:n elementin osamäärä.

$$x \cdot \left( \begin{bmatrix} c & 4 \\ 5 & d \end{bmatrix} \right) = \begin{bmatrix} \frac{x}{c} & \frac{x}{4} \\ \frac{x}{5} & \frac{x}{d} \end{bmatrix}$$

$$\begin{bmatrix} a & 2 \\ b & 3 \end{bmatrix} \cdot \left( \begin{bmatrix} c & 4 \\ 5 & d \end{bmatrix} \right) = \begin{bmatrix} \frac{x}{5} & \frac{x}{d} \end{bmatrix}$$

## .^ (pistepotenssi)

  painikkeet

*Matriisi1* . ^ *Matriisi2* ⇒ *matriisi*

$\begin{bmatrix} a & 2 \\ b & 3 \end{bmatrix}$	. ^	$\begin{bmatrix} c & 4 \\ 5 & d \end{bmatrix}$	$\begin{bmatrix} a^c & 16 \\ b^5 & 3^d \end{bmatrix}$
$x$	. ^	$\begin{bmatrix} c & 4 \\ 5 & d \end{bmatrix}$	$\begin{bmatrix} x^c & x^4 \\ x^5 & x^d \end{bmatrix}$

*Laus* . ^ *Matriisi1* ⇒ *matriisi*

*Matriisi1* . ^ *Matriisi2* laskee matriisin, jossa *Matriisi2*:n jokainen elementti on eksponentti *Matriisi1*:n vastaavalle elementille.

*Laus* . ^ *Matriisi1* laskee matriisin, jossa *Matriisi1*:n jokainen elementti on eksponentti lausekkeelle *Laus*.

## -(negaatio)

 painike

-*Laus1* ⇒ *lauseke*

-2.43	-2.43
$\{-1, 0.4, 1.2 \times 10^9\}$	$\{1, -0.4, -1.2 \times 10^9\}$
$-a \cdot -b$	$a \cdot b$

-*Lista1* ⇒ *lista*

-*Matriisi1* ⇒ *matriisi*

Laskee argumentin negaation.

Kun kyseessä on lista tai matriisi, määrittää kaikkien elementtien negaatiot.

Jos argumentti on binaarinen tai heksadesimaalinen kokonaisluku, negaatio antaa kakkosen komplementin.

Binaarisessa kantalukutilassa:

-0b100101	0b11111111111111111111111111111111
-----------	------------------------------------

Jos haluat nähdä koko vastauksen, paina ▲ ja siirrä sen jälkeen osoitinta painikkeilla ◀ ja ▶.

## % (prosentti)



  painikkeet

*Laus1* % ⇒ *lauseke*

**Huom:** Vastauksen pakottaminen likimääräiseksi:

*Lista1* % ⇒ *lista*

*Matriisi1* % ⇒ *matriisi*

**Kämmenlaite:** Paina  .

**Windows®:** Paina **Ctrl+Enter**.

**Macintosh®:** Paina **⌘+Enter**.

**iPad®:** Pidä **enter** ja valitse .


argument

Määrittää 100

Kun kyseessä on lista tai matriisi, määrittää listan tai matriisin, jossa jokainen elementti on jaettu luvulla 100.

13%	0.13
$\{\{1, 10, 100\}\}$ %	$\{0.01, 0.1, 1\}$

## = (on yhtä kuin)

 painike

$Laus1 = Laus2 \Rightarrow$  Boolean lauseke

$Lista1 = Lista2 \Rightarrow$  Boolean lista

$Matriisi1 = Matriisi2 \Rightarrow$  Boolean matriisi

Antaa totuusarvon tosi, jos  $Laus1$  määritetään olevan yhtä kuin  $Laus2$ .

Antaa totuusarvon epätosi, jos  $Laus1$  määritetään olevan ei yhtä kuin  $Laus2$ .

Kaikissa muissa tapauksissa vastauksena on yhtälön sievennetty muoto.

Listojen ja matriisien kohdalla vastauksena on vertailuja elementti elementiltä.

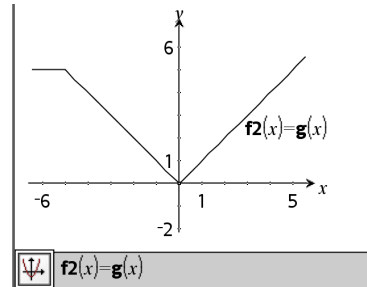
**Huomaa esimerkkiä syöttäessäsi:** Ohjeet monirivisten ohjelmien ja funktion määritysten syöttämisestä löytyvät tuotteen ohjekirjan Laskin-osiosta.

Esimerkkifunktio, jossa on käytetty matematiikkatestisympöoleita: =, ≠, <, ≤, >, ≥


```
Define g(x)=Func
  If x≤-5 Then
    Return 5
  ElseIf x>-5 and x<0 Then
    Return -x
  ElseIf x≥0 and x≠10 Then
    Return x
  ElseIf x=10 Then
    Return 3
  EndIf
EndFunc
```

Done

Funktion  $g(x)$  kuvaajan piirtämisen tulos



## ≠ (ei yhtä kuin)

  painikkeet

$Laus1 \neq Laus2 \Rightarrow$  Boolean lauseke

$Lista1 \neq Lista2 \Rightarrow$  Boolean lista

$Matriisi1 \neq Matriisi2 \Rightarrow$  Boolean matriisi



Antaa totuusarvon tosi, jos  $Laus1$  määritetään olevan ei yhtä kuin  $Laus2$ .

Antaa totuusarvon epätosi, jos  $Laus1$  määritetään olevan yhtä kuin  $Laus2$ .

Kaikissa muissa tapauksissa vastauksena on yhtälön sievennetty muoto.

Katso esimerkki kohdasta “=” (on yhtä kuin).


## ≠ (ei yhtä kuin)

  painikkeet

Listojen ja matriisien kohdalla vastauksena on vertailuja elementti elementiltä.

**Huomaa:** Voit syöttää tämän operaattorin näppäimistöltä kirjoittamalla /=

## < (pienempi kuin)

  painikkeet

$Laus1 < Laus2 \Rightarrow \text{Boolean lauseke}$

Katso esimerkki kohdasta “=” (on yhtä kuin).

$Lista1 < Lista2 \Rightarrow \text{Boolean lista}$

$Matriisi1 < Matriisi2 \Rightarrow \text{Boolean matriisi}$



Antaa totuusarvon tosi, jos *Laus1* määritetään olevan pienempi kuin *Laus2*.

Antaa totuusarvon epätosi, jos *Laus1* määritetään olevan suurempi tai yhtä suuri kuin *Laus2*.

Kaikissa muissa tapauksissa vastauksena on yhtälön sievennetty muoto.

Listojen ja matriisien kohdalla vastauksena on vertailuja elementti elementiltä.

## ≤ (pienempi tai yhtä suuri kuin)

  painikkeet

$Laus1 \leq Laus2 \Rightarrow \text{Boolean lauseke}$

Katso esimerkki kohdasta “=” (on yhtä kuin).

$Lista1 \leq Lista2 \Rightarrow \text{Boolean lista}$

$Matriisi1 \leq Matriisi2 \Rightarrow \text{Boolean matriisi}$

Antaa totuusarvon tosi, jos *Laus1* määritetään olevan pienempi tai yhtä suuri kuin *Laus2*.



Antaa totuusarvon epätosi, jos *Laus1* määritetään olevan suurempi kuin *Laus2*.

Kaikissa muissa tapauksissa vastauksena on yhtälön sievennetty muoto.

Listojen ja matriisien kohdalla vastauksena on vertailuja elementti elementiltä.





## ≤ (pienempi tai yhtä suuri kuin)

  painikkeet

**Huomaa:** Voit syöttää tämän operaattorin näppäimistöltä kirjoittamalla <=

## > (suurempi kuin)

  painikkeet

$Laus1 > Laus2 \Rightarrow \text{Boolean lauseke}$

Katso esimerkki kohdasta “=” (on yhtä kuin).

$Lista1 > Lista2 \Rightarrow \text{Boolean lista}$

$Matriisi1 > Matriisi2 \Rightarrow \text{Boolean matriisi}$


Antaa totuusarvon tosi, jos *Laus1* määritetään olevan suurempi kuin *Laus2*.

Antaa totuusarvon epätosi, jos *Laus1* määritetään olevan pienempi tai yhtä suuri kuin *Laus2*.

Kaikissa muissa tapauksissa vastauksena on yhtälön sievennetty muoto.

Listojen ja matriisien kohdalla vastauksena on vertailuja elementti elementiltä.

## ≥ (suurempi tai yhtä suuri kuin)

  painikkeet

$Laus1 \geq Laus2 \Rightarrow \text{Boolean lauseke}$

Katso esimerkki kohdasta “=” (on yhtä kuin).

$Lista1 \geq Lista2 \Rightarrow \text{Boolean lista}$

$Matriisi1 \geq Matriisi2 \Rightarrow \text{Boolean matriisi}$

Antaa totuusarvon tosi, jos *Laus1* määritetään olevan suurempi tai yhtä suuri kuin *Laus2*.

Antaa totuusarvon epätosi, jos *Laus1* määritetään olevan pienempi kuin *Laus2*.

Kaikissa muissa tapauksissa vastauksena on yhtälön sievennetty muoto.

Listojen ja matriisien kohdalla vastauksena on vertailuja elementti elementiltä.

**Huomaa:** Voit syöttää tämän operaattorin näppäimistöltä kirjoittamalla >=

**⇒ (looginen seuraus)****ctrl [= näppäimet***BooleanLaus1 ⇒ BooleanLaus2* antaa vastauksena *Boolean lausekkeen* $5 > 3 \text{ or } 3 > 5$  true*BooleanList1 ⇒ BooleanList2* antaa vastauksena *Boolean listan* $5 > 3 \Rightarrow 3 > 5$  false*BooleanMatriisi1 ⇒ BooleanMatriisi2* antaa vastauksena *Boolean matriisin* $3 \text{ or } 4$  7 $3 \Rightarrow 4$  -4 $\{1,2,3\} \text{ or } \{3,2,1\}$   $\{3,2,3\}$  $\{1,2,3\} \Rightarrow \{3,2,1\}$   $\{-1,-1,-3\}$ *Kokonaisluku1 ⇒ Kokonaisluku2* antaa vastauksena *kokonaisluvun*

Arvioi lausekkeen not <argumentti1> or <argumentti2> ja antaa vastauksena tosi, epätosi tai yhtälön sievennetyn muodon.

Listojen ja matriisien kohdalla vastauksena on vertailuja elementti elementiltä.

**Huomaa:** Voit syöttää tämän operaattorin näppäimistöltä kirjoittamalla =>

**⇔ (looginen kaksoisseuraus, XNOR)****ctrl [= näppäimet***BooleanLaus1 ⇔ BooleanLaus2* antaa vastauksena *Boolean lausekkeen* $5 > 3 \text{ xor } 3 > 5$  true*BooleanList1 ⇔ BooleanList2* antaa vastauksena *Boolean listan* $5 > 3 \Leftrightarrow 3 > 5$  false $3 \text{ xor } 4$  7*BooleanMatriisi1 ⇔ BooleanMatriisi2* antaa vastauksena *Boolean matriisin* $3 \Leftrightarrow 4$  -8 $\{1,2,3\} \text{ xor } \{3,2,1\}$   $\{2,0,2\}$  $\{1,2,3\} \Leftrightarrow \{3,2,1\}$   $\{-3,-1,-3\}$ *Kokonaisluku1 ⇔ Kokonaisluku2* antaa vastauksena *kokonaisluvun*

Antaa vastauksena XOR Boolean operaation negaation kahdesta argumentista. Antaa vastauksena totuusarvon tosi, epätosi tai yhtälön sievennetyn muodon.

Listojen ja matriisien kohdalla vastauksena on vertailuja elementti elementiltä.

**Huomaa:** Voit syöttää tämän operaattorin näppäimistöä kirjoittamalla <=>

**! (kertoma)**

*LausI!* ⇒ lauseke

$$5! \qquad 120$$

*ListaI!* ⇒ lista

$$\{\{5,4,3\}\}! \qquad \{120,24,6\}$$

*MatriisiI!* ⇒ matriisi

$$\left(\begin{matrix} 1 & 2 \\ 3 & 4 \end{matrix}\right)! \qquad \begin{matrix} 1 & 2 \\ 6 & 24 \end{matrix}$$

Määrittää argumentin kertoman.

Jos kyseessä on lista tai matriisi, määrittää listan tai matriisin elementtien kertomista.

**& (liitä)**

*Merkkijono1 & Merkkijono2* ⇒ *merkkijono*

$$\text{"Hello "&"Nick"} \qquad \text{"Hello Nick"}$$

Antaa vastauksena tekstimerkkijonon, joka on *Merkkijono2* liitettynä *Merkkijono1*:een.

**d() (derivaatta)**

*d(LausI, Muutt[, Aste])* ⇒ lauseke

$$\frac{d}{dx}(f(x) \cdot g(x)) \qquad \frac{d}{dx}(f(x)) \cdot g(x) + \frac{d}{dx}(g(x)) \cdot f(x)$$

*d(ListaI, Muutt[, Aste])* ⇒ lista

$$\frac{d}{dy} \left( \frac{d}{dx}(x^2 \cdot y^3) \right) \qquad 6 \cdot y^2 \cdot x$$

*d(MatriisiI, Muutt[, Aste])* ⇒ matriisi

$$\frac{d}{dx} \left( \left\{ x^2, x^3, x^4 \right\} \right) \qquad \left\{ 2 \cdot x, 3 \cdot x^2, 4 \cdot x^3 \right\}$$

Laskee ensimmäisen argumentin ensimmäisen derivaatan muuttujan *Muutt* suhteen.

*Asteen*, mikäli se otetaan mukaan, on oltava kokonaisluku. Jos aste on pienempi kuin nolla, vastaus on antiderivaatta.

**Huomaa:** Voit syöttää tämän funktion näppäimistöltä kirjoittamalla **derivative (...)**.

**d()** ei noudata normaalia sievennysmekanismia, jossa argumentit sievennetään täydellisesti, minkä jälkeen sovitetaan funktion määritelmää näihin täydellisesti sievennetyihin argumentteihin. Sen sijaan funktio **d()** suorittaa seuraavat vaiheet:

1. Sievennä toista argumenttia vain sen verran, että sen tuloksena ei ole eimuuttuja.
2. Sievennä ensimmäistä argumenttia vain sen verran, ettei se hae mitään tallennettua arvoa vaiheessa 1 määritetyille muuttujalle.
3. Määritä vaiheen 2 tuloksen symbolinen derivaatta vaiheessa 1 saadun muuttujan suhteen.

Jos vaiheen 1 muuttujalla on tallennettu arvo tai rajoittavalla operaattorilla ("|") määritetty arvo, korvaa tämä arvo vastaukseen vaiheesta 3.

**Huomaa:** Katso myös  
Ensimmäinen derivaatta, sivu 6,  
Toinen derivaatta, sivu 6 tai  
N:s derivaatta, sivu sivu 6.

$\int(\text{LausI}, \text{Muutt}[\text{Ala}, \text{Ylä}]) \Rightarrow \text{lauseke}$

$\int(\text{LausI}, \text{Muutt}[\text{Vakio}]) \Rightarrow \text{lauseke}$

Laskee *LausI*:n integraalin muuttujan *Muutt* suhteen välillä *Ala* - *Ylä* olevista arvoista.

**Huomaa:** Katso myös **Määrätyn tai Määrittämättömän integraalin malli**, sivu 7.

$$\int_a^b x^2 dx = \frac{b^3}{3} - \frac{a^3}{3}$$

**Huomaa:** Voit syöttää tämän funktion näppäimistöltä kirjoittamalla **integral** (...).

Laskee antiderivaatan, jos *Ala* ja *Ylä* jätetään pois. Integraalin symbolinen vakio jätetään pois, ellet anna argumenttia *Vakio*.

$\int x^2 dx$	$\frac{x^3}{3}$
$\int(a \cdot x^2, x, c)$	$\frac{a \cdot x^3}{3} + c$

Samanarvoisesti kelpaavat antiderivaatat voivat erota numeerisen vakion suhteen. Tällainen vakio on olla naamioitunut—erityisesti, kun antiderivaatta sisältää logaritmeja tai käännteisiä trigonometrisia funktioita. Lisäksi joskus lisätään paloittain määritettyjä vakiolausekkeita, jotta antiderivaatasta saadaan kelvollinen suuremmalle välille kuin tavanomainen kaava.

∫() antaa vastauksena itsensä niille *LausI*:n paloille, joita se ei pysty määrittämään sisäänrakennettujen funktioidensa ja operaattoreidensa eksplisiittisenä äärellisenä kombinaationa.

$\int b \cdot e^{-x^2} + \frac{a}{x^2+a^2} dx$	$b \cdot \int e^{-x^2} dx + \tan^{-1}\left(\frac{x}{a}\right)$
--	--

Kun annat argumentit *Ala* ja *Ylä*, laskin yrittää paikallistaa mahdolliset epäjatkuvuudet tai epäjatkuvat derivaatat välillä *Ala* < *Muutt* < *Ylä* ja jakaa välin näissä kohdissa.

**Auto or Approximate (Automaattinen tai likimääräinen)** -tilan Auto (Automaattinen) -asetuksessa käytetään numeerista integrointia, mikäli mahdollista, kun antiderivaattaa tai raja-arvoa ei voida määrittää.

Approximate (Likimääräinen) -asetuksessa yritetään ensin numeerista integrointia, mikäli mahdollista. Antiderivaattoja etsitään vain, kun tällainen numeerinen integrointi ei ole mahdollista tai epäonnistuu.

**Huom:** Vastauksen pakottaminen likimääräiseksi:

**Kämmenlaite:** Paina  .

**Windows®:** Paina **Ctrl+Enter**.

**Macintosh®:** Paina **⌘+Enter**.

**iPad®:** Pidä **enter** ja valitse .

$$\int_{-1}^1 e^{-x^2} dx \quad 1.49365$$

$\int()$ -komentoa voidaan sijoittaa sisäkkäin usean integraalin suorittamiseksi. Integroinnin raja-arvot voivat riippua niiden ulkopuolella olevista integrointimuuttujista.

**Huomaa:** Katso myös **nInt()**, sivu 131.

$$\int_0^a \int_0^x \ln(x+y) dy dx$$

$$\frac{a^2 \cdot \ln(a)}{2} + \frac{a^2 \cdot (4 \cdot \ln(2) - 3)}{4}$$

$\sqrt{}$  (neliöjuuri)

painikkeet

$\sqrt{List1} \Rightarrow$  lauseke

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

$\sqrt{List1} \Rightarrow$  lista

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

Laskee argumentin neliöjuuren.

Kun kyseessä on lista, laskee kaikkien *List1*:n elementtien neliöjuuret.

**Huomaa:** Voit syöttää tämän funktion näppäimistöltä kirjoittamalla **sqrt** (...)

**Huomaa:** Katso myös **Neliöjuurimalli**, sivu 1.

$\Pi()$  (tulo)

Katalogi >

$\Pi(List1, Muutt, Matala, Korkea) \Rightarrow$  lauseke

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

**Huomaa:** Voit syöttää tämän funktion näppäimistöltä kirjoittamalla **prodSeq** (...).

$$\prod_{k=1}^n (k^2) \quad (n!)^2$$

Sieventää *List1*:n jokaisen *Muutt*:n arvon suhteen väliltä *Matala* - *Korkea* ja laskee vastausten tulon.

$$\prod_{n=1}^5 \left\{ \left(\frac{1}{n}, n, 2\right) \right\} \quad \left\{ \frac{1}{120}, 120, 32 \right\}$$

**Huomaa:** Katso myös **Kertolaskumalli** ( $\Pi$ ), sivu 6.

**$\Pi()$  (tulo)**Katalogi >  $\Pi(Lausl, Muutt, Matala, Matala-1) \Rightarrow 1$ 

$$\prod_{k=4}^3 (k) = 1$$

$\Pi(Lausl, Muutt, Matala, Korkea) \Rightarrow 1/\Pi(Lausl, Muutt, Korkea+1, Matala-1)$ , jos  $Korkea < Matala-1$

Esimerkkien kertolaskukaavat on otettu seuraavasta viitteestä:

Ronald L. Graham, Donald E. Knuth, and Oren Patashnik. *Concrete Mathematics: A Foundation for Computer Science*. Reading, Massachusetts: Addison-Wesley, 1994.

$$\prod_{k=4}^1 \left(\frac{1}{k}\right) = 6$$

$$\prod_{k=4}^1 \left(\frac{1}{k}\right) \cdot \prod_{k=2}^4 \left(\frac{1}{k}\right) = \frac{1}{4}$$

 **$\Sigma()$  (summa)**Katalogi > 

$\Sigma(Lausl, Muutt, Matala, Korkea) \Rightarrow$  lauseke

**Huomaa:** Voit syöttää tämän funktion näppäimistöltä kirjoittamalla **sumSeq** (...).

Sieventää *Lausl*:n jokaisen muuttujan *Muutt* arvon suhteen väliltä *Matala - Korkea* ja laskee vastausten summan.

**Huomaa:** Katso myös **Summamalli**, sivu 5.

$\Sigma(Lausl, Muutt, Matala, Matala-1) \Rightarrow 0$

$$\sum_{n=1}^5 \left(\frac{1}{n}\right) = \frac{137}{60}$$

$$\sum_{k=1}^n (k^2) = \frac{n \cdot (n+1) \cdot (2 \cdot n+1)}{6}$$

$$\sum_{n=1}^{\infty} \left(\frac{1}{n^2}\right) = \frac{\pi^2}{6}$$

$$\sum_{k=4}^3 (k) = 0$$

$\Sigma(Lausl, Muutt, Matala, Korkea) \Rightarrow \Sigma(Lausl, Muutt, Korkea+1, Matala-1)$ , jos  $Korkea < Matala-1$

Esimerkkien yhteenlaskukaavat on otettu seuraavasta viitteestä:

Ronald L. Graham, Donald E. Knuth, and Oren Patashnik. *Concrete Mathematics: A Foundation for Computer Science*. Reading, Massachusetts: Addison-Wesley, 1994.

$$\sum_{k=4}^1 (k) = -5$$

$$\sum_{k=4}^1 (k) + \sum_{k=2}^4 (k) = 4$$

**ΣInt()**Katalogi > 

$\Sigma\text{Int}(NPmt1, NPmt2, N, I, PV, [Pmt], [FV], [PpY], [CpY], [PmtAt], [pyörArvo]) \Rightarrow arvo$

$\Sigma\text{Int}(1,3,12,4.75,20000,,12,12)$  -213.48

$\Sigma\text{Int}(NPmt1, NPmt2, amortTable) \Rightarrow arvo$

Lyhennystoiminto, joka laskee koron summan määritetyn maksueräjakson ajalta.

*NPmt1* ja *NPmt2* määrittävät maksujakson alku- ja loppurajat.

*N*, *I*, *PV*, *Pmt*, *FV*, *PpY*, *CpY* ja *PmtAt* on kuvattu TVM-argumenttien taulukossa, sivu 211.

- Jos jätät argumentin *Pmt* pois, sen oletusarvoksi tulee  $Pmt = \text{tvmPmt}(N, I, PV, FV, PpY, CpY, PmtAt)$ .
- Jos jätät argumentin *FV* pois, sen oletusarvoksi tulee  $FV = 0$ .
- Argumenttien *PpY*, *CpY* ja *PmtAt* oletusarvot ovat samat kuin TVM-funktioilla.

*pyörArvo* määrittää pyöristyksessä käytettävien desimaalien määrän. Oletusarvo=2.

$\Sigma\text{Int}(NPmt1, NPmt2, amortTable)$  laskee koron summan lyhennystaulukon *amortTable* mukaisesti. *amortTable*-argumentin on oltava matriisi, joka on kohdassa **amortTbl()** kuvatun muotoinen, katso sivu 8.

**Huomaa:** Katso myös  $\Sigma\text{Prn}()$ , jäljempänä, sekä **Bal()**, sivu 17.

$tbl := \text{amortTbl}(12, 12, 4.75, 20000, 12, 12)$

0	0.	0.	20000.
1	-77.49	-1632.43	18367.6
2	-71.17	-1638.75	16728.8
3	-64.82	-1645.1	15083.7
4	-58.44	-1651.48	13432.2
5	-52.05	-1657.87	11774.4
6	-45.62	-1664.3	10110.1
7	-39.17	-1670.75	8439.32
8	-32.7	-1677.22	6762.1
9	-26.2	-1683.72	5078.38
10	-19.68	-1690.24	3388.14
11	-13.13	-1696.79	1691.35
12	-6.55	-1703.37	-12.02

$\Sigma\text{Int}(1,3,tbl)$  -213.48

**ΣPrn()**Katalogi > 

$\Sigma\text{Prn}(NPmt1, NPmt2, N, I, PV, [Pmt], [FV], [PpY], [CpY], [PmtAt], [pyörArvo]) \Rightarrow arvo$

$\Sigma\text{Prn}(1,3,12,4.75,20000,,12,12)$  -4916.28

$\Sigma\text{Prn}$

$(NPmt1, NPmt2, amortTable) \Rightarrow arvo$



Lyhennystoiminto, joka laskee pääoman summan määritetyn maksujakson ajalta.

*NPmt1* ja *NPmt2* määrittävät maksujakson alku- ja loppurajat.

*N*, *I*, *PV*, *Pmt*, *FV*, *PpY*, *CpY* ja *PmtAt* on kuvattu TVM-argumenttien taulukossa, sivu 211.

- Jos jätät argumentin *Pmt* pois, sen oletusarvoksi tulee *Pmt=tvmpmt(N,I,PV,FV,PpY,CpY,PmtAt)*.
- Jos jätät argumentin *FV* pois, sen oletusarvoksi tulee *FV=0*.
- Argumenttien *PpY*, *CpY* ja *PmtAt* oletusarvot ovat samat kuin TVM-funktioilla.

*pyörArvo* määrittää pyöristyksessä käytettävien desimaalien määrän. Oletusarvo=2.

ΣPrn(*NPmt1*,*NPmt2*,*amortTable*) laskee pääoman summan lyhennystaulukon *amortTable* perusteella. *amortTable*-argumentin on oltava matriisi, joka on kohdassa **amortTbl()** kuvatun muotoinen, katso sivu 8.

**Huomaa:** Katso myös ΣInt(), edellä, sekä Bal(), sivu 17.

<i>tbl:=amortTbl(12,12,4.75,20000,,12,12)</i>			
0	0.	0.	20000.
1	-77.49	-1632.43	18367.57
2	-71.17	-1638.75	16728.82
3	-64.82	-1645.1	15083.72
4	-58.44	-1651.48	13432.24
5	-52.05	-1657.87	11774.37
6	-45.62	-1664.3	10110.07
7	-39.17	-1670.75	8439.32
8	-32.7	-1677.22	6762.1
9	-26.2	-1683.72	5078.38
10	-19.68	-1690.24	3388.14
11	-13.13	-1696.79	1691.35
12	-6.55	-1703.37	-12.02
ΣPrn(1,3, <i>tbl</i> )			-4916.28

## # (epäsuora operaattori)

  **painikkeet**

### # *muuttNimiMerkkijono*

Viittaa muuttujaan, jonka nimi on *muuttNimiMerkkijono*. Tällä operaattorilla voit luoda muuttujanimiä funktion sisältä merkkijonojen avulla.

#("x"&"y"&"z") xyz

Luo muuttujan xyz tai viittaa siihen.

10 → *r* 10

"r" → *s1* "r"

#*s1* 10

Laskee arvon muuttujalle (*r*), jonka nimi on tallennettu muuttujaan *s1*.

## E (kymmenpotenssimuoto)

EE painike

$\text{mantissa}E\text{eksponentti}$

23000. 23000.

Syöttää luvun kymmenpotenssimuodossa. Luku tulkitaan seuraavasti:  $\text{mantissa} \times 10^{\text{eksponentti}}$ .

2300000000.+4.1E15 4.1E15

$3 \cdot 10^4$  30000

Vinkki: Jos haluat syöttää 10-potenssin ilman, että vastauksena on desimaaliluku, käytä komentoa  $10^{\text{kokonaisluku}}$ .

**Huomaa:** Voit syöttää tämän operaattorin tietokoneen näppäimistöltä kirjoittamalla @E. Kirjoita esimerkiksi  $2.3@E4$ , kun haluat syöttää  $2.3E4$ .

## g (graadi)

1 painike

$\text{Laus}I\text{g} \Rightarrow \text{lauseke}$

Aste-, graadi- tai radiaanikulmatilassa.

$\text{Lista}I\text{g} \Rightarrow \text{lista}$

$\cos(50^{\circ})$   $\frac{\sqrt{2}}{2}$

$\text{Matriisi}I\text{g} \Rightarrow \text{matriisi}$

$\cos\{\{0,100^{\circ},200^{\circ}\}\}$   $\{1,0,-1\}$

Tämän funktion avulla voit määrittää graadikulman ollessasi aste- tai radiaanikulmatilassa.

Kun laskin on radiaanikulmatilassa, kertoo  $\text{Laus}I$ :n arvolla  $\pi/200$ .

Kun laskin on astekulmatilassa, kertoo  $\text{Laus}I$ :n arvolla  $g/100$ .

Graadikulmatilassa antaa vastauksena lausekkeen  $\text{Laus}I$  muuttumattomana.

**Huomaa:** Voit syöttää tämän symbolin tietokoneen näppäimistöltä kirjoittamalla @g.

## r (radiaani)

1 painike

$\text{Laus}I^r \Rightarrow \text{lauseke}$

Aste-, graadi- tai radiaanikulmatilassa:

## r (radiaani)

1 painike

$Listar \Rightarrow lista$

$Matriisir \Rightarrow matriisi$

Tämän funktion avulla voit määrittää radiaanikulman ollessasi aste- tai graadikulmatilassa.

Kun laskin on astekulmatilassa, kertoo argumentin arvolla  $180/\pi$ .

Radiaanikulmatilassa antaa vastauksena argumentin muuttumattomana.

Kun laskin on graadikulmatilassa, kertoo argumentin arvolla  $200/\pi$ .

Vinkki: Käytä komentoa  $r$ , jos haluat pakottaa funktion määritelmän yksiköksi radiaanit riippumatta tilasta, joka on käytössä funktion käytön aikana.

**Huomaa:** Voit syöttää tämän symbolin tietokoneen näppäimistöltä kirjoittamalla  $@r$ .

$$\cos\left(\frac{\pi}{4^r}\right) \quad \frac{\sqrt{2}}{2}$$
$$\cos\left\{0^r, \frac{\pi}{12}r, (\pi)r\right\} \quad \left\{1, \frac{(\sqrt{3+1})\sqrt{2}}{4}, -1\right\}$$

## ° (aste)

1 painike

$Lausr \Rightarrow lauseke$

$Listar \Rightarrow lista$

$Matriisir \Rightarrow matriisi$

Tämän funktion avulla voit määrittää astekulman ollessasi graadi- tai radiaanikulmatilassa.

Kun laskin on radiaanikulmatilassa, kertoo argumentin arvolla  $\pi/180$ .

Astekulmatilassa antaa vastauksena argumentin muuttumattomana.

Kun laskin on graadikulmatilassa, kertoo argumentin arvolla  $10/9$ .

**Huomaa:** Voit syöttää tämän symbolin tietokoneen näppäimistöltä kirjoittamalla  $@d$ .

Aste-, graadi- tai radiaanikulmatilassa:

$$\cos(45^\circ) \quad \frac{\sqrt{2}}{2}$$

Radiaanikulmatilassa:

**Huom:** Vastauksen pakottaminen likimääräiseksi:


**Kämmenlaite:** Paina  $\text{ctrl}$   $\text{enter}$ .

**Windows®:** Paina  $\text{Ctrl}+\text{Enter}$ .

**Macintosh®:** Paina  $\text{⌘}+\text{Enter}$ .

**iPad®:** Pidä  $\text{enter}$  ja valitse  $\approx$ .

$$\cos\left\{0, \frac{\pi}{4}, 90^\circ, 30.12^\circ\right\}$$
$$\{1, 0.707107, 0., 0.864976\}$$

**° , ' , " (astetta/minuuttia/sekuntia)****ctrl**  **painikkeet** $dd^{\circ}mm'ss.ss'' \Rightarrow lauseke$  $dd$ Positiivinen tai negatiivinen luku $mm$ Ei-negatiivinen luku $ss.ss$ Ei-negatiivinen lukuLaskee  $dd+(mm/60)+(ss.ss/3600)$ .


Tässä kantaluku-60:n syötemuodossa voit:

- Syöttää kulman asteina/minuutteina/sekunteina nykyisestä kulmatilasta riippumatta.
- Syöttää kellonajan tunteina/minuutteina/sekunteina.

**Huomaa:** Merkitse sekuntien  $ss.ss$  perään kaksi heittomerkkiä ("), ei lainausmerkkiä (").

Astekulmatilassa:

$25^{\circ}13'17.5''$	25.2215
$25^{\circ}30'$	$\frac{51}{2}$

**∠ (kulma)****ctrl**  **painikkeet** $[Säde, \angle \_ Kulma] \Rightarrow vektori$   
(polaarinen syöte) $[Säde, \angle \_ Kulma, Z$   
 $Koordinaatti] \Rightarrow vektori$   
(lieriömäinen syöte) $[Säde, \angle \_ Kulma, \angle \_ Kulma] \Rightarrow vektori$   
(pallonmuotoinen syöte)

Laskee koordinaatit vektorina riippuen vektorimuotoilan asetuksesta: suorakulma, sylinteri tai pallo.

**Huomaa:** Voit syöttää tämän symbolin tietokoneen näppäimistöltä kirjoittamalla @<. $(Magnitudi \angle \_ Kulma) \Rightarrow kompleksiarvo$   
(polaarinen syöte)Syöttää kompleksilukuarvon ( $r\angle\theta$ ) polaarissa muodossa.  $Kulma$  tulkitaan nykyisen kulmatila-asetuksen mukaisesti.

Radiaanikulmatilassa ja kun vektorimuoto on asetettu valintaan: suorakulma

$[5 \angle 60^{\circ} \angle 45^{\circ}]$	$\frac{5 \cdot \sqrt{2}}{4}$	$\frac{5 \cdot \sqrt{6}}{4}$	$\frac{5 \cdot \sqrt{2}}{2}$
---	------------------------------	------------------------------	------------------------------

sylinteri

$[5 \angle 60^{\circ} \angle 45^{\circ}]$	$\frac{5 \cdot \sqrt{2}}{2}$	$\angle \frac{\pi}{3}$	$\frac{5 \cdot \sqrt{2}}{2}$
---	------------------------------	------------------------	------------------------------

pallo

$[5 \angle 60^{\circ} \angle 45^{\circ}]$	$5 \angle \frac{\pi}{3} \angle \frac{\pi}{4}$
---	---


Radiaanikulmatilassa ja suorakulmakompleksimuodossa:

$5+3 \cdot i \left( 10 \angle \frac{\pi}{4} \right)$	$5-5 \cdot \sqrt{2} + (3-5 \cdot \sqrt{2}) \cdot i$
--	---

## ∠ (kulma)

  painikkeet

**Huom:** Vastauksen pakottaminen likimääräiseksi:

**Kämmenlaite:** Paina  .

**Windows®:** Paina **Ctrl+Enter**.

**Macintosh®:** Paina **⌘+Enter**.

**iPad®:** Pidä **enter** ja valitse .

$$5+3 \cdot i - \left( 10 \angle \frac{\pi}{4} \right) \quad -2.07107 - 4.07107 \cdot i$$

## ' (jaoton)

 painike

*muuttuja'*  
*muuttuja''*

Syöttää jaottoman symbolin differentiaaliyhtälöön. Yksi jaottoman symboli tarkoittaa 1. asteen differentiaaliyhtälöä, kaksi jaottoman symbolia tarkoittaa 2. asteen differentiaaliyhtälöä jne.

$$\text{deSolve} \left( y'' = y \frac{-1}{2} \text{ and } y(0) = 0 \text{ and } y'(0) = 0, t, y \right)$$
$$\frac{2 \cdot y^4}{3} = t$$

## \_ (alaviiva edustaa tyhjää elementtiä)

**Katso Tyhjät elementit, sivu 272.**

## \_ (alaviiva edustaa yksikön nimeä)

  painikkeet


*Laus\_Yksikkö*

3·\_m▶\_ft

9.84252·\_ft

Määrittää lausekkeen *Laus* yksiköt. Kaikkien yksiköiden nimien alussa on oltava alaviiva.

**Huomaa:** Muunnoksen symboli ▶ löytyy

Katalogista. Napsauta ensin  ja sen jälkeen **Matemaattiset operaattorit**.

Voit käyttää valmiiksi määritettyjä yksiköitä tai luoda omia yksiköitä. Valmiiksi määritetyt yksiköt on esitetty vakioiden ja mittausyksiköiden moduulissa. Voit valita yksiköiden nimet Katalogista tai kirjoittaa ne suoraan.

*Muuttuja\_*

Olettaen, että z on määrittämätön:

## \_ (alaviiva edustaa yksikön nimeä)

  painikkeet



Kun *Muuttujalla* ei ole arvoa, sitä käsitellään ikään kuin se edustaisi kompleksilukua. Oletusarvoisesti, ilman merkkiä `_`, muuttujaa käsitellään reaaliilukuna.

$\text{real}(z)$	$z$
$\text{real}(z\_)$	$\text{real}(z\_)$
$\text{imag}(z)$	0
$\text{imag}(z\_)$	$\text{imag}(z\_)$

Jos *Muuttujalla* on arvo, merkkiä `_` ei huomioida, ja *Muuttuja* säilyttää alkuperäisen datatyypinsä.

**Huomaa:** Voit tallentaa kompleksiluvun muuttujaan käyttämättä alaviivaa `_`. Parhaan tuloksen aikaansaamiseksi alaviivaa `_` on kuitenkin suositeltavaa käyttää laskutoimituksissa, kuten `cSolve()` ja `cZeros()`.

## ► (muunna)

  painikkeet

*Laus\_Unit1* ► *\_Yksikkö2* ⇒ *Laus\_*  
*Yksikkö2*

3·\_m ►\_ft 9.84252·\_ft

Muuntaa lausekkeen yksiköstä toiseen.

Alaviivan merkki `_` määrittää yksiköt. Yksiköiden tulee kuulua samaan luokkaan, esimerkiksi Pituus tai Pinta-ala.

Saat valmiiksi määritettyjen yksiköiden luettelon näkyviin avaamalla Katalogin ja näyttämällä Yksikkömuunnokset-välilehden:

- Voit valita yksikön nimen luettelosta.
- Voit valita muunnosoperaattorin, ►, luettelon yläosasta.

Voit myös kirjoittaa yksiköiden nimet käsin. Voit syöttää merkin “`_`” kirjoittaessasi yksiköiden nimiä kämmenlaitteesta painamalla painiketta

 .

**Huomaa:** Kun muunnat lämpötilan yksiköitä, käytä funktioita `tmpCnv()` ja `ΔtmpCnv()`. Muunnosoperaattori ► ei käsittele lämpötilan yksiköitä.

**10^()**Katalogi > **10^ (Lausl) ⇒ lauseke**

$10^{1.5}$	31.6228
------------	---------

**10^ (Listal) ⇒ lista**

$10^{\{0, -2.2, a\}}$	$\left\{1, \frac{1}{100}, 100, 10^a\right\}$
-----------------------	--

Laskee luvun 10 korotettuna argumentin potenssiin.

Jos kyseessä on lista, laskee luvun 10 korotettuna *Listal*:n elementtien potenssiin.

**10^ (neliomatriisi l) ⇒ neliomatriisi**

Laskee luvun 10 korotettuna *neliomatriisi l*:n potenssiin. Tämä ei ole sama kuin kunkin elementin 10-potenssiin korottamisen laskeminen. Laskentamenetelmä on kuvattu kohdassa **cos()**.

$10^{\begin{bmatrix} 1 & 5 & 3 \\ 4 & 2 & 1 \\ 6 & -2 & 1 \end{bmatrix}}$	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="text-align: center;"><math>1.14336\text{E}7</math></td> <td style="text-align: center;"><math>8.17155\text{E}6</math></td> <td style="text-align: center;"><math>6.67589\text{E}6</math></td> </tr> <tr> <td style="text-align: center;"><math>9.95651\text{E}6</math></td> <td style="text-align: center;"><math>7.11587\text{E}6</math></td> <td style="text-align: center;"><math>5.81342\text{E}6</math></td> </tr> <tr> <td style="text-align: center;"><math>7.65298\text{E}6</math></td> <td style="text-align: center;"><math>5.46952\text{E}6</math></td> <td style="text-align: center;"><math>4.46845\text{E}6</math></td> </tr> </table>	$1.14336\text{E}7$	$8.17155\text{E}6$	$6.67589\text{E}6$	$9.95651\text{E}6$	$7.11587\text{E}6$	$5.81342\text{E}6$	$7.65298\text{E}6$	$5.46952\text{E}6$	$4.46845\text{E}6$
$1.14336\text{E}7$	$8.17155\text{E}6$	$6.67589\text{E}6$								
$9.95651\text{E}6$	$7.11587\text{E}6$	$5.81342\text{E}6$								
$7.65298\text{E}6$	$5.46952\text{E}6$	$4.46845\text{E}6$								

*neliomatriisi l*:n on oltava diagonaloitavissa. Vastaus sisältää aina liukulukuja.

**^-1 (käänteisluku)**Katalogi > **Lausl ^-1 ⇒ lauseke**

$(3.1)^{-1}$	0.322581
--------------	----------

**Listal ^-1 ⇒ lista**

$\{a, 4, -0.1, x, -2\}^{-1}$	$\left\{\frac{1}{a}, \frac{1}{4}, -10, \frac{1}{x}, \frac{-1}{2}\right\}$
------------------------------	---

Laskee argumentin käänteisluvun.

Jos kyseessä on lista, laskee *Listal*:n elementtien käänteisluvut.

**neliomatriisi l ^-1 ⇒ neliomatriisi**

Laskee *neliomatriisi l*:n käänteisluvun.

*neliomatriisi l*:n oltava ei-singulaarinen neliomatriisi.

$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}^{-1}$	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="text-align: center;"><math>\frac{-2}{3}</math></td> <td style="text-align: center;"><math>\frac{1}{2}</math></td> </tr> <tr> <td style="text-align: center;"><math>\frac{3}{2}</math></td> <td style="text-align: center;"><math>\frac{-1}{2}</math></td> </tr> </table>	$\frac{-2}{3}$	$\frac{1}{2}$	$\frac{3}{2}$	$\frac{-1}{2}$
$\frac{-2}{3}$	$\frac{1}{2}$				
$\frac{3}{2}$	$\frac{-1}{2}$				
$\begin{bmatrix} 1 & 2 \\ a & 4 \end{bmatrix}^{-1}$	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="text-align: center;"><math>\frac{-2}{a-2}</math></td> <td style="text-align: center;"><math>\frac{1}{a-2}</math></td> </tr> <tr> <td style="text-align: center;"><math>\frac{a}{2 \cdot (a-2)}</math></td> <td style="text-align: center;"><math>\frac{-1}{2 \cdot (a-2)}</math></td> </tr> </table>	$\frac{-2}{a-2}$	$\frac{1}{a-2}$	$\frac{a}{2 \cdot (a-2)}$	$\frac{-1}{2 \cdot (a-2)}$
$\frac{-2}{a-2}$	$\frac{1}{a-2}$				
$\frac{a}{2 \cdot (a-2)}$	$\frac{-1}{2 \cdot (a-2)}$				

*Laus* | *Booleanlaus1* [**and***Booleanlaus2*]...

*Laus* | *Booleanlaus1* [**or***Booleanlaus2*]...

Rajoittava ("|")-symboli toimii binaarisena operaattorina. Operaattorin | vasemmalla puolella oleva operandi on lauseke. Operaattorin | oikealla puolella oleva operandi määrittää yhden tai useampia suhteita, joiden tarkoitus on vaikuttaa lausekkeen sieventämiseen. Operaattorin | jäljessä olevat useat suhteet on yhdistettävä loogisilla operaattoreilla "and" tai "or".

Rajoittava operaattori tarjoaa kolme perustoimintoa:

- Sijoitukset
- Välin rajoitusehdot
- Pois rajaaminen

Sijoitukset ovat yhtälön muodossa, kuten  $x=3$  tai  $y=\sin(x)$ . Tehokkainta on, kun vasen puoli on yksinkertainen muuttuja.

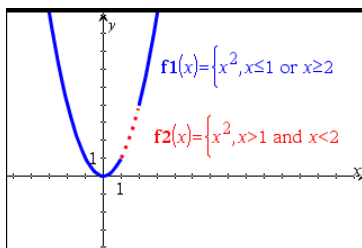
*Laus* | *Muuttuja* = *arvo* korvaa *arvon* jokaisessa *Muuttujan* esiintymiskohdassa lausekkeessa *Laus*.

Välien rajoitusten muoto on yksi tai useampia epäyhtälöitä, jotka on yhdistetty loogisilla operaattoreilla "and" or "or". Välien rajoitusehdot sallivat myös sievennyksen, joka saattaisi muuten olla kelpaamaton tai ei laskettavissa.

$x+1 x=3$	4
$x+y x=\sin(y)$	$\sin(y)+y$
$x+y \sin(y)=x$	$x+y$



$x^3-2\cdot x+7\rightarrow f(x)$	Done
$f(x) x=\sqrt{3}$	$\sqrt{3}+7$
$(\sin(x))^2+2\cdot\sin(x)-6 \sin(x)=d$	$d^2+2\cdot d-6$

$\text{solve}(x^2-1=0,x) x>0 \text{ and } x<2$	$x=1$
$\sqrt{x}\cdot\frac{1}{x} x>0$	1
$\sqrt{x}\cdot\frac{1}{x}$	$\frac{1}{x}\cdot\sqrt{x}$







## | (rajoittava operaattori)

  -näppäimet

Pois sulkemisessa käytetään suhdeoperaattoria "ei ole yhtä kuin" ( $\neq$  tai  $\neq$ ), jonka avulla jokin tietty arvo suljetaan pois. Näitä käytetään pääasiallisesti täsmällisen ratkaisun poissulkemiseen käytettäessä funktioita **cSolve()**, **cZeros()**, **fMax()**, **fMin()**, **solve()**, **zeros()**, jne.

$\text{solve}(x^2-1=0,x) _{x\neq 1}$	$x=-1$
--------------------------------------	--------

## → (tallenna)

  **painike**

*Laus* → *Muutt*

*Lista* → *Muutt*

*Matriisi* → *Muutt*

*Laus* → *Funktio(Param1,...)*

*Lista* → *Funktio(Param1,...)* *Matriisi* → *Funktio(Param1,...)*

$\frac{\pi}{4} \rightarrow \text{myvar}$	$\frac{\pi}{4}$
$2 \cdot \cos(x) \rightarrow y1(x)$	<i>Done</i>
$\{1,2,3,4\} \rightarrow \text{lst5}$	$\{1,2,3,4\}$
$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix} \rightarrow \text{matg}$	$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}$
"Hello" → <i>str1</i>	"Hello"

Jos muuttujaa *Muutt* ei ole, laskin luo sen ja alustaa sen muotoon *Laus*, *Lista* tai *Matriisi*.

Jos muuttuja *Muutt* on jo olemassa eikä se ole lukittu tai suojattu, laskin korvaa sen sisällön lausekkeella *Laus*, listalla *Lista* tai matriisilla *Matriisi*.

Vinkki: Jos aiot suorittaa symbolisia laskutoimituksia käyttäen määrittämättömiä muuttujia, vältä tallentamasta mitään yleisesti käytettyihin, yksikirjaimisiin muuttujiin, kuten a, b, c, x, y, z ja niin edelleen.

**Huomaa:** Voit syöttää tämän operaattorin näppäimistöltä kirjoittamalla **=:** pikavalintana. Kirjoita esimerkiksi **pi/4 =: myvar**.

## := (määrittä)

ctrl  painikkeet

*Muutt := Laus*

*Muutt := Lista*

*Muutt := Matriisi*

*Funktio(Param1,...) := Laus*

*Funktio(Param1,...) := Lista*  
*Funktio(Param1,...) := Matriisi*


Jos muuttujaa *Muutt* ei ole, laskin luo muuttujan *Muutt* ja alustaa sen muotoon *Laus*, *Lista* tai *Matriisi*.

Jos *Muutt* on jo olemassa eikä se ole lukittu tai suojattu, laskin korvaa sen sisällön lausekkeella *Laus*, listalla *List* tai matriisilla *Matriisi*.

Vinkki: Jos aiot suorittaa symbolisia laskutoimituksia käyttäen määrittämättömiä muuttujia, vältä tallentamasta mitään yleisesti käytettyihin, yksikirjaimisiin muuttujiin, kuten a, b, c, x, y, z ja niin edelleen.

$myvar := \frac{\pi}{4}$	$\frac{\pi}{4}$
$y1(x) := 2 \cdot \cos(x)$	Done
$lst5 := \{1, 2, 3, 4\}$	$\{1, 2, 3, 4\}$
$matg := \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}$	$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}$
$str1 := "Hello"$	"Hello"

## © (kommentti)

ctrl  painikkeet

© [*teksti*]

© käsittelee *tekstin* kommenttirivinä, minkä avulla voit lisätä merkintöjä luomisi funktioihin ja ohjelmiin.

© voi olla rivin alussa tai missä tahansa rivin kohdassa. Kaikki merkin © jäljessä oleva, aina rivin loppuun saakka, on kommenttia.

**Huomaa esimerkkiä syöttäessäsi:** Ohjeet monirivisten ohjelmien ja funktion määritysten syöttämisestä löytyvät tuotteen ohjekirjan Laskin-osiosta.

Define $g(n) = \text{Func}$	
© Declare variables	
Local $i, result$	
$result := 0$	
For $i, 1, n, 1$ © Loop $n$ times	
$result := result + i^2$	
EndFor	
Return $result$	
EndFunc	
	Done
$g(3)$	14

**Ob, Oh****0 B painikkeet, 0 H painikkeet****Ob** binaariluku**Oh** heksadesimaaliluku

Määrittää binaari- (Ob) tai heksadesimaaliluvun (Oh). Syöttääksesi binaari- tai heksadesimaaliluvun sinun on syötettävä etuliite Ob tai Oh riippumatta kantalukutilasta. Ilman etuliitettä lukua käsitellään desimaalilukuna (kantalu 10).

Tulokset näytetään kantalukutilan mukaisesti.

Desimaalisessa kantalukutilassa:

$0b10+0hF+10$	27
---------------	----

Binaarisessa kantalukutilassa:

$0b10+0hF+10$	0b11011
---------------	---------

Heksadesimaalisessa kantalukutilassa:

$0b10+0hF+10$	0h1B
---------------	------

# TI-Nspire™ CX II – Piirtokomennot

Tämä on täydentävä asiakirja TI-Nspire™-viiteoppaalle ja TI-Nspire™ CAS -viiteoppaalle. Kaikki TI-Nspire™ CX II -komennot sisällytetään ja julkaistaan TI-Nspire™-viiteoppaassa ja TI-Nspire™ CAS -viiteoppaassa.

## ***Kuvaajien ohjelmointi***

Uusia komentoja on lisätty TI-Nspire™ CX II -kämmenlaitteisiin ja TI-Nspire™-työpöytäsovellukseen kuvaajaohjelmointia varten.

TI-Nspire™ CX II -kämmenlaitteet siirtyvät tähän kuvaajatilaan, kun ne suorittavat kuvaajakomentoja, ja siirtyvät ohjelman suorittamisen jälkeen takaisin kontekstiin, jossa ohjelma suoritettiin.

Näytön yläreunassa näkyy ”Running...”, kun ohjelmaa suoritetaan. Näkyviin tulee ”Finished”, kun ohjelma on valmis. Minkä tahansa näppäimen painaminen siirtää järjestelmän pois kuvaajatilasta.

- Siirtyminen kuvaajatilaan käynnistyy automaattisesti, kun jokin Draw (kuvaajat) -komennoista kohdataan TI-Basic -ohjelman suorituksen aikana.
- Tämä siirtyminen tapahtuu vain, kun ohjelma suoritetaan laskimella; asiakirjassa tai muistiolehtiön laskimessa.
- Siirtyminen pois kuvaajatilasta tapahtuu ohjelman sulkemisen jälkeen.
- Kuvaajatila on käytettävissä vain TI-Nspire™ CX II- kämmenlaitteissa ja TI-Nspire™ CX II -kämmenlaitteiden työpöytäversion näkymässä. Tämä tarkoittaa, että se ei ole käytettävissä tietokoneen asiakirjanäkymässä työpöytäversiossa tai iOS-käyttöjärjestelmässä.
  - Jos kuvaajakomento kohdataan, kun TI-Basic -ohjelmaa suoritetaan virheellisestä kontekstista, näkyviin tulee virheviesti ja TI-Basic -ohjelma suljetaan.

## ***Kuvaajanäyttö***

Kuvaajanäytön yläosassa on otsikko, jota ei voi kirjoittaa kuvaajakomennoilla.

Kuvaajanäytön piirustusalue tyhjennetään (color = 255,255,255), kun kuvaajanäyttö alustetaan.

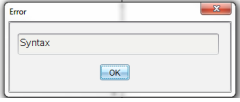
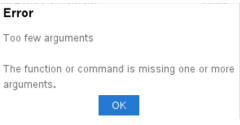
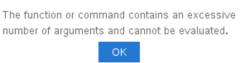
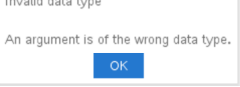
<b>Kuvaajanäyttö</b>	<b>Oletus</b>
Korkeus	212
Leveys	318
Väri	valkoinen: 255,255,255

## Oletusnäkyvä ja asetukset

- Yläpalkin tilakuvakkeet (akun tila, press-to-start-tila, verkon ilmainen jne.) eivät näy, kun kuvaajaohjelma on käynnissä.
- Piirtämisen oletusväri: Musta (0,0,0)
- Kynän oletustyyli – normaali, tasainen
  - Paksuus: 1 (tasainen), 2 (normaali), 3 (paksuin)
  - Tyyli 1 (tasainen), 2 (pisteet), 3 (viivat)
- Kaikki piirtokomennot käyttävät nykyisiä väri- ja kynäasetuksia eli joko oletusarvoja tai TI-Basic-komentojen avulla asetettuja arvoja.
- Tekstin fontti on pysyvä eikä sitä voi muuttaa.
- Kaikki kuvaajanäytön piirrokset piirretään leikkausikkunassa, joka on kuvaajanäytön piirustusalueen kokoinen. Mitään piirroksia, jotka ulottuvat leikatun kuvaajanäytön piirustusalueen ulkopuolelle, ei piirretä. Virheviestiä ei näytetä.
- Kaikki piirtokomennoille määritetyt x- ja y-koordinaatit on määritelty siten, että 0,0 on kuvaajanäytön piirustusalueen vasemmassa yläkulmassa.
  - **Poikkeukset:**
    - **DrawText** käyttää koordinaatteja tekstin rajaavan ruudun vasemmassa alakulmassa.
    - **SetWindow** käyttää näytön vasenta alakulmaa.
- Kaikki komentojen parametrit voidaan antaa lausekkeina, jotka arvioidaan numeroksi, joka pyöristetään sitten lähimpään kokonaislukuun.

## Kuvaajanäytön virheviestit

Jos validointi epäonnistuu, näyttöön ilmestyy virheviesti.

Virheviestit	Kuvaus	Tarkastele
Virhe Syntaksi	Jos syntaksin tarkastuksessa löytyy syntaksivirheitä, näkyviin tulee virheilmoitus, ja editori yrittää asettaa kohdistimen ensimmäisen virheen kohdalle, jotta voit korjata sen.	
Virhe Liian vähän argumentteja	Funktiosta tai komennosta puuttuu yksi tai useampia argumentteja	
Virhe Liian monta argumenttia	Funktio tai komento sisältää liian monta argumenttia eikä sitä voida arvioida.	
Virhe Virheellinen tietotyyppi	Argumentin tietotyyppi on väärä.	

## Virheelliset komennot kuvaajatilassa

Jotkin komennot eivät ole sallittuja, kun ohjelma siirtyy kuvaajatilaan. Jos nämä komennot kohdataan kuvaajatilassa, näyttöön tulee virheviesti ja ohjelma suljetaan.

Komentoa ei sallita	Virheviestit
Pyyntö	Pyyntöä ei voida toteuttaa kuvaajatilassa
PyydäMerkkij	RequestStr-toimintoa ei voida toteuttaa kuvaajatilassa
Teksti	Tekstiä ei voida toteuttaa kuvaajatilassa

Komennot, jotka kirjoittavat tekstin laskimeen – **disp** ja **dispAt** – ovat tuettuja komentoja kuvaajakontekstissa. Näiden komentojen teksti lähetetään Laskin-näyttöön (ei Kuvaaja-näyttöön) ja se näkyy sen jälkeen, kun ohjelma on suljettu ja järjestelmä siirtyy takaisin Laskin-sovellukseen.



## Tyhjennä

**Clear** *x, y, width, height*

Tyhjentää koko näytön, jos parametreja ei ole.

Jos *x, y, width* ja *height* on määritetty, parametrien määrittämä suorakulmio tyhjennetään.

Tyhjennä


Tyhjentää koko näytön

Clear 10,10,100,50

Tyhjentää suorakulmioalueen, jonka vasen yläkulma on (10, 10), leveys on 100 ja korkeus on 50.



## DrawArc

Katalogi >   
CXII**DrawArc**  $x, y, width, height, startAngle, arcAngle$ 

Piirrä kaari rajaavan suorakulmion sisällä annetulla aloitus- ja kaarikulmalla.

 $x, y$ : rajaavan suorakulmion vasen yläkoordinaatti $width, height$ : rajaavan suorakulmion mitat

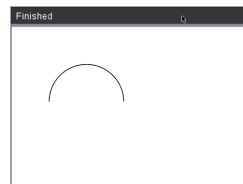
"Arc Angle" määrittää kaaren kaarevuuden.

Nämä parametrit voidaan antaa lausekkeina, jotka arvioidaan numeroksi, joka pyöristetään sitten lähimpään kokonaislukuun.

DrawArc 20,20,100,100,0,90



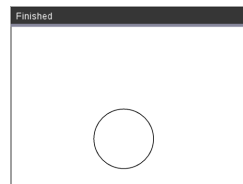
DrawArc 50,50,100,100,0,180

Katso myös: [FillArc](#)

## DrawCircle

Katalogi >   
CXII**DrawCircle**  $x, y, radius$  $x, y$ : keskikohdan koordinaatti $radius$ : ympyrän säde

DrawCircle 150,150,40

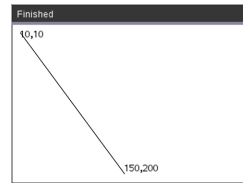
Katso myös: [FillCircle](#)

**DrawLine**  $x1, y1, x2, y2$ Piirrä viiva pisteillä  $x1, y1, x2, y2$ .

Lausekkeet, jotka arvioidaan numeroksi, joka pyöristetään sitten lähimpään kokonaislukuun.

**Näytön rajat:** Jos määritetyt koordinaatit aiheuttavat viivan osan piirtämisen kuvaajanäytön ulkopuolelle, kyseinen viivan osa leikataan eikä virheilmoitusta näytetä.

DrawLine 10,10,150,200

**DrawPoly**

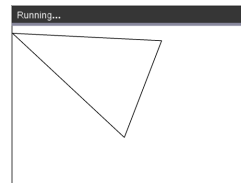
Komennoilla on kaksi varianttia:

**DrawPoly**  $xlist, ylist$ 

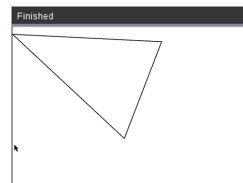
tai

**DrawPoly**  $x1, y1, x2, y2, x3, y3...xn, yn$ **Huomaa:** DrawPoly  $xlist, ylist$   
Muoto yhdistää kohteet  $x1, y1$  to  $x2, y2, x2, y2$  to  $x3, y3$  ja niin edelleen.**Huomaa:** DrawPoly  $x1, y1, x2, y2, x3, y3...xn, yn$   
Kohteita  $xn, yn$  EI yhdistetä automaattisesti kohteisiin  $x1, y1$ .Lausekkeet, jotka arvioidaan todellisten liukulukujen luetteloon  
 $xlist, ylist$ Lausekkeet, jotka arvioidaan yksittäiseksi liukuluvuksi  
 $x1, y1...xn, yn$  = monikulmion kärkien koordinaatit

```
xlist:={0,200,150,0}
ylist:={10,20,150,10}
DrawPoly xlist,ylist
```



DrawPoly 0,10,200,20,150,150,0,10



**Huomaa: DrawPoly:** Lisää kokomitat (leveys/korkeus), jotka vastaavat piirrettyjä viivoja.

Viivat piirretään rajaavan ruudun määrittelyn koordinaatin ja mittojen ympärille siten, että piirretyn monikulmion todellinen koko on suurempi kuin leveys ja korkeus.

Katso myös: [FillPoly](#)

## DrawRect

**DrawRect** *x, y, width, height*

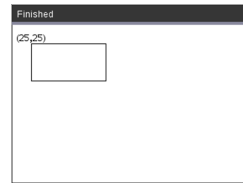
*x, y*: suorakulmion vasen yläkoordinaatti

*width, height*: suorakulmion leveys ja korkeus (suorakulmio, joka on piirretty alaspäin ja oikealle lähtökoordinaatista)

**Huomaa:** Viivat piirretään rajaavan ruudun määrittelyn koordinaatin ja mittojen ympärille siten, että piirretyn suorakulmion todellinen koko on suurempi kuin leveys ja korkeus osoittavat.

Katso myös: [FillRect](#)

DrawRect 25,25,100,50



## DrawText

**DrawText** *x, y, exprOrString1*  
*[,exprOrString2]...*

*x, y*: tekstisyötön koordinaatti

Piirtää tekstin kohteeseen *exprOrString* määritetyssä *x, y* -koordinaattien sijainnissa.

Säännöt kohteelle *exprOrString* ovat samat kuin kohteelle **Disp** – **DrawText** voi käsitellä useita argumentteja.

DrawText 50,50,"Hello World"



## FillArc

Katalogi >   
CXII

**FillArc** *x, y, width, height, startAngle, arcAngle*

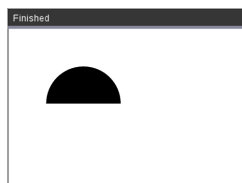
*x, y*: rajaavan suorakulmion vasen yläkoordinaatti

Piirrä ja täytä kaari rajaavan suorakulmion sisällä annetulla aloitus- ja kaarikulmalla.

Täytön oletusväri on musta. Täyttöväri voidaan asettaa komennolla [SetColor](#).

”Arc Angle” määrittää kaaren kaarevuuden.

FillArc 50,50,100,100,0,180



## FillCircle

Katalogi >   
CXII

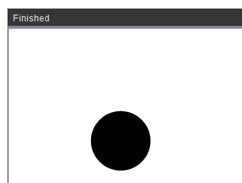
**FillCircle** *x, y, radius*

*x, y*: keskikohdan koordinaatti

Piirrä ja täytä ympyrä määritellyllä keskikohdalla ja määritetyllä säteellä.

Täytön oletusväri on musta. Täyttöväri voidaan asettaa komennolla [SetColor](#).

FillCircle 150,150,40



Täällä!

## FillPoly

Katalogi >   
CXII

**FillPoly** *xlist, ylist*

tai

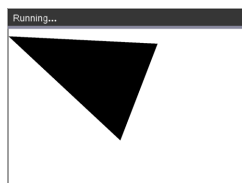
**FillPoly** *x1, y1, x2, y2, x3, y3...xn, yn*

**Huomaa:** Viivan ja värin määrittää komennot [SetColor](#) ja [SetPen](#)

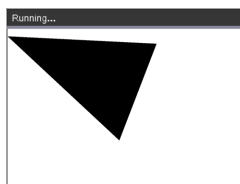
*xlist*:= {0,200,150,0}

*ylist*:= {10,20,150,10}

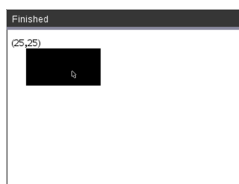
FillPoly *xlist, ylist*



FillPoly 0,10,200,20,150,150,0,10

**FillRect****FillRect**  $x, y, width, height$  $x, y$ : suorakulmion vasen yläkoordinaatti $width, height$ : suorakulmion leveys ja korkeusPiirrä ja täytä suorakulmio, jonka vasen yläkulma koordinaatissa, jonka määrittää  $(x,y)$ .Täytön oletusväri on musta. Täyttöväri voidaan asettaa komennolla [SetColor](#).**Huomaa:** Viivan ja värin määrittää komennot [SetColor](#) ja [SetPen](#)

FillRect 25,25,100,50



## G

### getPlatform()

Katalogi >   
CXII

### getPlatform()

getPlatform()

"dt"

Palauttaa:

"dt" työpöytäohjelmistosovelluksissa

"hh" TI-Nspire™ CX -kämmenlaitteissa

"ios" TI-Nspire™ CX iPad® -sovelluksessa

---

**PaintBuffer**

Maalaa kuvaajan puskurin näytölle

Tätä komentoa käytetään yhdessä UseBuffer-komennon kanssa näyttönopeuden lisäämiseksi, kun ohjelma luo useita kuvaajaobjekteja.

**UseBuffer**

Luvulle  $n, 1, 10$

```
x:=randInt(0,300)
```

```
y:=randInt(0,200)
```

```
radius:=randInt(10,50)
```

```
Wait 0,5
```

```
DrawCircle x,y,radius
```

```
EndFor
```

**PaintBuffer**

Tämä ohjelma näyttää 10 ympyrää samanaikaisesti.

Jos UseBuffer-komento poistetaan, jokainen ympyrä näytetään sellaisena kuin se on piirretty.

Katso myös: [UseBuffer](#)

**PlotXY**  $x, y, shape$ 

$x, y$ : koordinaatit muodon muodostamiseen

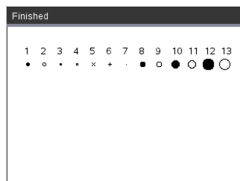
$shape$  : numero väliltä 1–13, joka määrittää muodon

- 1 – Täytetty ympyrä
- 2 – Tyhjä ympyrä
- 3 – Täytetty neliö
- 4 – Tyhjä neliö
- 5 – Risti
- 6 – Plusmerkkikuvio
- 7 – Ohut
- 8 – keskikokoinen piste, täytetty
- 9 – keskikokoinen piste, tyhjä
- 10 – suurempi piste, täytetty
- 11 – suurempi piste, tyhjä
- 12 – suurin piste, täytetty
- 13 – suurin piste, tyhjä

PlotXY 100,100,1

Luvulle  $n, 1, 13$ DrawText  $1+22*n, 40, n$ PlotXY  $5+22*n, 50, n$ 

EndFor





**SetColor**Katalogi >   
CXII**SetColor**

Red-arvo, Green-arvo, Blue-arvo

Kelvolliset arvot punaiselle, vihreälle ja siniselle ovat välillä 0–255

Asettaa tulevien Draw-komentojen väriä

SetColor 255,0,0

DrawCircle 150,150,100

**SetPen**Katalogi >   
CXII**SetPen**

paksuus, tyyli

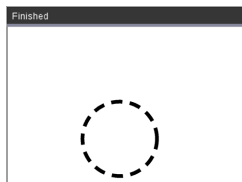
Paksuus:  $1 \leq \text{paksuus} \leq 3$  | 1 on ohuin, 3 on paksuin

style: 1 = tasainen, 2 = pisteet, 3 = viivat

Asettaa tulevien Draw-komentojen kynän tyylin

SetPen 3,3

DrawCircle 150,150,50

**SetWindow**Katalogi >   
CXII**SetWindow**

xMin, xMax, yMin, yMax

Luo loogisen ikkunan, joka kartoittaa kuvaajan piirustusalueen. Kaikki parametrit vaaditaan.

Jos piirretyn objektin osa on ikkunan ulkopuolella, syöttö leikataan (ei näytetä) eikä virheilmoitusta näytetä.

SetWindow 0,160,0,120

Asettaa syöttöikkunan vasemman alakulman kohtaan 0,0. Leveys on 160 ja korkeus on 120.

DrawLine 0,0,100,100

SetWindow 0,160,0,120

SetPen 3,3

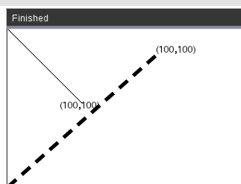
DrawLine 0,0,100,100

Jo  $x_{min}$  on suurempi tai yhtä suuri kuin  $x_{max}$  tai  $y_{min}$  on suurempi tai yhtä suuri kuin  $y_{max}$ , näyttöön tulee virheviesti.

Mitään objekteja, jotka on piirretty ennen SetWindow-komentoa, ei piirretä uudelleen uudessa kokoonpanossa.

Käytä seuraavia ikkunan parametrien nollaamiseksi oletusarvoihin:

SetWindow 0,0,0,0



**UseBuffer****UseBuffer**

Piirrä kuvaaja puskuriin näytön sijaan (suorituskyvyn parantamiseksi)

Tätä komentoa käytetään yhdessä PaintBuffer-komennon kanssa näyttönopeuden lisäämiseksi, kun ohjelma luo useita kuvaajaobjekteja.

UseBuffer-komennolla kaikki kuvaajat näytetään vasta seuraavan PaintBuffer-komennon suorittamisen jälkeen.

UseBufferia täytyy käyttää vain kerran ohjelmassa, eli jokainen käytetty PaintBuffer-komento ei tarvitse vastaavaa UseBuffer-komentoa.

**Katso myös:** [PaintBuffer](#)

**UseBuffer**

```
Luvulle n,1,10
x:=randInt(0,300)
y:=randInt(0,200)
radius:=randInt(10,50)
Wait 0,5
DrawCircle x,y,radius
EndFor
PaintBuffer
```

Tämä ohjelma näyttää 10 ympyrää samanaikaisesti.



Jos UseBuffer-komento poistetaan, jokainen ympyrä näytetään sellaisena kuin se on piirretty.

# Tyhjät elementit

Reaalimaailman dataa analysoitaessa käytössäsi ei aina välttämättä ole täydellistä datasarjaa. TI-Nspire™ CAS sallii käyttää tyhjiä dataelementtejä, jolloin voit jatkaa toimintaa lähes täydellisellä datalla tarvitsematta aloittaa alusta tai hylätä epätäydellisiä tapauksia.

Esimerkki tyhjiä elementtejä sisältävästä datasta löytyy luvusta Listat & Taulukot, kohdasta Kuvaajien piirtäminen taulukko-datasta.

Funktion **delVoid()** avulla voit poistaa tyhjät elementit listasta. Funktion **isVoid()** avulla voit testata tyhjää elementtiä. Lisätietoja, katso **delVoid()**, sivu 53, ja **isVoid()**, sivu 102.

**Huomaa:** Voit syöttää tyhjän elementin manuaalisesti matemaattiseen lausekkeeseen syöttämällä merkin “\_” tai avainsanan **void**. Avainsana **void** muuntuu automaattisesti merkiksi “\_”, kun lauseke sievennetään. Voit syöttää merkin “\_” kämmenlaitteesta painamalla painiketta  .

## Tyhjiä elementtejä sisältävät laskutoimitukset

Suurin osa tyhjän syötteen sisältävistä laskutoimituksista tuottaa tyhjän tuloksen. Katso erikoistapaukset jäljempänä.

$ \_$	–
$\text{gcd}(100,\_)$	–
$3+\_$	–
$\{5,\_,10\}-\{3,6,9\}$	$\{2,\_,1\}$

## Tyhjiä elementtejä sisältävät lista-argumentit

Seuraavat funktiot ja komennot eivät huomioi (ohittavat ne) lista-argumenteista löytyviä tyhjiä elementtejä.

**count**, **countIf**, **cumulativeSum**, **freqTable** ▶ lista, **frekvenssi**, **max**, **mean**, **median**, **tulo**, **stDevPop**, **stDevSamp**, **sum**, **sumIf**, **varPop** ja **varSamp** sekä regressiolaskutoimitukset yhden muuttujan (**OneVar**), kahden muuttujan (**TwoVar**) ja **FiveNumSummary**-tilastot, luottamusvälit ja tilastotestit

$\text{sum}(\{2,\_,3,5,6,6\})$	16.6
$\text{median}(\{1,2,\_,\_,3\})$	2
$\text{cumulativeSum}(\{1,2,\_,4,5\})$	$\{1,3,\_,7,12\}$
$\text{cumulativeSum}\left(\begin{bmatrix} 1 & 2 \\ 3 & - \\ 5 & 6 \end{bmatrix}\right)$	$\begin{bmatrix} 1 & 2 \\ 4 & - \\ 9 & 8 \end{bmatrix}$

## Tyhjiä elementtejä sisältävät lista-argumentit

**SortA** ja **SortD** siirtävät kaikki ensimmäisen argumentin sisältämät tyhjät elementit viimeiseksi.

$\{5,4,3,_,1\} \rightarrow list1$	$\{5,4,3,_,1\}$
$\{5,4,3,2,1\} \rightarrow list2$	$\{5,4,3,2,1\}$
SortA list1,list2	Done
list1	$\{1,3,4,5,_\}$
list2	$\{1,3,4,5,2\}$

$\{1,2,3,_,5\} \rightarrow list1$	$\{1,2,3,_,5\}$
$\{1,2,3,4,5\} \rightarrow list2$	$\{1,2,3,4,5\}$
SortD list1,list2	Done
list1	$\{5,3,2,1,_\}$
list2	$\{5,3,2,1,4\}$

Regressioissa X- tai Y-listassa oleva tyhjä elementti aiheuttaa tyhjän elementin jäännöksen vastaavalle elementille.

$ll:=\{1,2,3,4,5\}; l2:=\{2,_,3,5,6,6\}$	$\{2,_,3,5,6,6\}$
LinRegMx ll,l2	Done
stat.Resid	$\{0.434286,_, -0.862857, -0.011429, 0.44\}$
stat.XReg	$\{1,_,3,4,5\}$
stat.YReg	$\{2,_,3,5,6,6\}$
stat.FreqReg	$\{1,_,1,1,1,1\}$

Poisjätetty luokka regressioissa aiheuttaa tyhjän elementin jäännöksen vastaavalle elementille.

$ll:=\{1,3,4,5\}; l2:=\{2,3,5,6,6\}$	$\{2,3,5,6,6\}$
cat:={"M","M","F","F"}; incl:={"F"}	$\{ "F" \}$
LinRegMx ll,l2,1,cat,incl	Done
stat.Resid	$\{_,_,0,0\}$
stat.XReg	$\{_,_,4,5\}$
stat.YReg	$\{_,_,5,6,6\}$
stat.FreqReg	$\{_,_,1,1,1\}$

Frekvenssi 0 regressioissa aiheuttaa tyhjän elementin jäännöksen vastaavalle elementille.

$ll:=\{1,3,4,5\}; l2:=\{2,3,5,6,6\}$	$\{2,3,5,6,6\}$
LinRegMx ll,l2,{1,0,1,1}	Done
stat.Resid	$\{0.069231,_, -0.276923, 0.207692\}$
stat.XReg	$\{1,_,4,5\}$
stat.YReg	$\{2,_,5,6,6\}$
stat.FreqReg	$\{1,_,1,1,1\}$

# Matemaattisten lausekkeiden syöttäminen pikavalintojen avulla

Pikavalintojen avulla voit syöttää matemaattisten lausekkeiden elementtejä kirjoittamalla ne näppäimistöltä sen sijaan, että käyttäisit katalogia tai symbolipalettia. Kun haluat esimerkiksi syöttää lausekkeen  $\sqrt{6}$ , voit kirjoittaa syöteriville `sqrt (6)`. Kun painat painiketta `enter`, lauseke `sqrt (6)` muuttuu muotoon  $\sqrt{6}$ . Jotkin pikavalinnat ovat näppäriä sekä kämmenlaitteesta että tietokoneen näppäimistöltä syötettyinä. Toiset taas ovat käteviä pääasiassa tietokoneen näppäimistöltä syötettyinä.

## Kämmenlaitteesta tai tietokoneen näppäimistöltä

---

Syötettävä kohde:	Pikavalinta:
$\pi$	<code>pi</code>
$\theta$	<code>theta</code>
$\infty$	<code>infinity</code>
$\leq$	<code>&lt;=</code>
$\geq$	<code>&gt;=</code>
$\neq$	<code>/=</code>
$\Rightarrow$ (looginen seuraus)	<code>=&gt;</code>
$\Leftrightarrow$ (looginen kaksoisseuraus, XNOR)	<code>&lt;=&gt;</code>
$\rightarrow$ (tallennusoperaattori)	<code>=:</code>
<code>   </code> (itseisarvo)	<code>abs (...)</code>
$\sqrt{()}$	<code>sqrt (...)</code>
$d()$	<code>derivative (...)</code>
$\int()$	<code>integral (...)</code>
$\Sigma()$ (yhteenlaskumalli)	<code>sumSeq (...)</code>
$\Pi()$ (tulon malli)	<code>prodSeq (...)</code>
<code>sin<sup>-1</sup>()</code> , <code>cos<sup>-1</sup>()</code> , ...	<code>arcsin (...)</code> , <code>arccos (...)</code> , ...
<code><math>\Delta</math>List()</code>	<code>deltaList (...)</code>
<code><math>\Delta</math>tmpCnv()</code>	<code>deltaTmpCnv (...)</code>

---

## Tietokoneen näppäimistöltä

---

Syötettävä kohde:	Pikavalinta:
<code>c1, c2, ...</code> (vakiot)	<code>@c1, @c2, ...</code>
<code>n1, n2, ...</code> (kokonaislukuvakiot)	<code>@n1, @n2, ...</code>
<code>i</code> (imaginaarinen vakio)	<code>@i</code>
<code>e</code> (luonnollisen logaritmin kantaluku $e$ )	<code>@e</code>
<code>E</code> (kymmenpotenssimuoto)	<code>@E</code>
<code>T</code> (transponointi)	<code>@t</code>
<code>r</code> (radiaanit)	<code>@r</code>
<code>°</code> (asteet)	<code>@d</code>

---

---

Syötettävä kohde:	Pikavalinta:
g (graadit)	@g
∠(kulma)	@<
► (muunnos)	@>
►Decimal, ►approxFraction ( ), jne.	@>Decimal, @>approxFraction ( ), jne.

---

# EOS-järjestelmän (yhtälökäyttöjärjestelmä) hierarkia

Tässä kappaleessa kuvataan yhtälökäyttöjärjestelmä (EOS™), jota sovelletaan matematiikan ja luonnontieteiden TI-Nspire™ CAS -oppimisteknologiassa. Luvut, muuttujat ja funktiot syötetään yksinkertaisena ja suoraviivaisena sekvenssinä. EOS™-ohjelmisto sieventää lausekkeet ja yhtälöt käyttäen sulkuryhmyksiä sekä alla kuvattuja prioriteetteja.

## Laskemisjärjestys

---

### Taso Operaattori

- 1 Kaarisulkeet ( ), hakasulkeet [ ], aaltosulkeet { }
  - 2 Epäsuora operaattori (#)
  - 3 Funktioiden komennot
  - 4 Jälkioperaattorit: astetta-minuuttia-sekuntia (°, ', " ), kertoma (!), prosentti (%), radiaani (x<sup>r</sup>), alaindeksi ([ ]), transponointi (T)
  - 5 Potenssiin korotus, potenssioperaattori (^)
  - 6 Negaatio (-)
  - 7 Merkkijonojen ketjutus (&)
  - 8 Kertolasku (•), jakolasku (/)
  - 9 Yhteenlasku (+), vähennyslasku (-)
  - 10 Yhtäläisyyssuhteet: on yhtä kuin (=), ei ole yhtä kuin (≠ tai /=), pienempi kuin (<), pienempi tai yhtä suuri kuin (≤ tai <=), suurempi tai yhtä suuri kuin (>), suurempi tai yhtä suuri kuin (≥ tai >=)
  - 11 Looginen **not**
  - 12 Looginen **and**
  - 13 Looginen **or**
  - 14 **xor, nor, ei-ja**
  - 15 Looginen seuraus (⇒)
  - 16 Looginen kaksoisseuraus, XNOR (⇔)
  - 17 Rajoittava operaattori ("|")
  - 18 Tallenna (→)
- 

### Kaarisulkeet, hakasulkeet ja aaltosulkeet

Kaikki kaarisulkeiden, hakasulkeiden tai aaltosulkeiden sisällä olevat laskutoimitukset sievennetään ensin. Esimerkiksi lausekkeesta 4(1+2) EOS sieventää ensin kaarisulkeiden sisällä olevan osan, 1+2, ja kertoo tuloksen, 3, sen jälkeen luvulla 4.

Kaikki kaarisulkeiden, hakasulkeiden tai aaltosulkeiden sisällä olevat laskutoimitukset sievennetään ensin. Esimerkiksi lausekkeesta 4(1+2) EOS™-ohjelmisto sieventää ensin kaarisulkeiden sisällä olevan osan, 1+2, ja kertoo sen jälkeen tuloksen, 3, luvulla 4.

**Huomaa:** Koska käyttäjä voi määrittää omia funktioita TI-Nspire™ CAS -ohjelmistossa, ohjelmisto tulkitsee muuttujan nimen, jonka perässä on sulkeissa oleva lauseke, funktion komennoiksi eikä implisiittiseksi kertolaskuksi. Esimerkiksi  $a(b+c)$  on funktio  $a$ , jota sievennetään lausekkeella  $b+c$ . Kun haluat kertoa lausekkeen  $b+c$  muuttujalla  $a$ , käytä eksplisiittistä kertolaskua:  $a*(b+c)$ .



## Epäsuora operaattori

Epäsuora operaattori (#) muuttaa merkkijonon muuttujaksi tai funktion nimeksi. Esimerkiksi lausekkeesta #("x"&"y"&"z") luodaan muuttujanimi xyz. Epäsuora operaattori sallii myös muuttujien luomisen ja muokkaamisen ohjelman sisällä. Esimerkiksi, jos  $10 \rightarrow r$  ja " $r$ "  $\rightarrow s1$ , niin  $\#s1=10$ .

## Jälkioperaattorit

Jälkioperaattorit ovat operaattoreita, jotka tulevat suoraan argumentin perään, kuten 5!, 25% tai  $60^{\circ}15'45''$ . Argumentit, joiden perässä on jälkioperaattori, sievennetään neljännellä prioriteettitasolla. Esimerkiksi lausekkeesta  $4^3!$  sievennetään ensimmäiseksi 3! Vastauksesta, 6, tulee luvun 4 eksponentti, josta saadaan 4096.

## Eksponenttilausekkeet

Eksponenttilausekkeet (^) ja potenssiin korotus elementti elementiltä (.^) sievennetään oikealta vasemmalle. Esimerkiksi lauseke  $2^3^2$  sievennetään samalla tavalla kuin  $2^{\wedge}(3^2)$ , josta saadaan 512. Tämä on eri asia kuin  $(2^{\wedge}3)^2$ , josta saadaan 64.

## Negaatio

Negatiivinen luku syötetään painamalla ensin näppäintä  $\boxed{-}$  ja syöttämällä luku sen perään. Jälkioperaattorit ja eksponenttilausekkeet suoritetaan ennen negatiota. Esimerkiksi lausekkeen  $-x^2$  tulos on negatiivinen luku, ja  $-9^2 = -81$ . Käytä sulkeita, kun haluat määrittää neliön negatiivisesta luvusta, kuten  $(-9)^2$ , josta saadaan 81.

Rajoittavan operaattorin ("|") jälkeen seuraava argumentti tuottaa sarjan ehtoja, jotka vaikuttavat operaattoria edeltävän argumentin laskemiseen.

# TI-Nspire CX II – TI-Basic-ohjelmointiominaisuudet

## Automaattinen sisennys Ohjelmointieditorissa

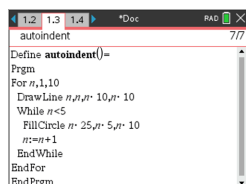
TI-Nspire™-ohjelmaeditori sientää nyt automaattisesti lauseita lohkokomennon sisällä.

Lohkokomennot ovat If/EndIf, For/EndFor, While/EndWhile, Loop/EndLoop, Try/EndTry

Editori lisää automaattisesti välilyönnit ohjelmakäskyihin lohkokomennon sisällä.

Lohkon sulkemiskomento kohdistetaan avauskomennon kanssa.

Alla oleva esimerkki näyttää automaattisen sisennyksen sisäkkäisissä lohkokomennossa.



```
autoindent 77
Define autoindent()=
Prgm
For n,1,10
DrawLine n,n,n-10,n-10
While n<5
FillCircle n-25,n-5,n-10
n:=n+1
EndWhile
EndFor
EndPrgm
```

Kopioidut ja liitetyt koodifragmentit säilyttävät alkuperäisen sisennyksen.

Ohjelmiston aikaisemmassa versiossa luodun ohjelman avaaminen säilyttää alkuperäisen sisennyksen.

---

## TI-Basicin virheviestien parannukset

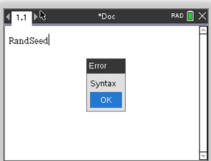
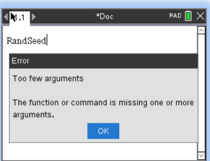
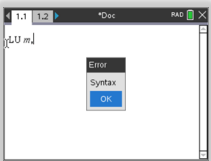
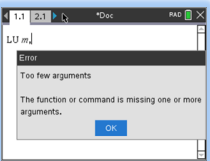
### Virheet

Virhetila	Uusi viesti
Ehtokohteen virhe (If/While)	Ehdollista lausuntoa ei voitu ratkaista arvolla <b>TRUE</b> tai <b>FALSE</b> <b>HUOMAA:</b> Koska muutos on sijoitettu riville virheen kanssa, ei ole enää tarpeen tarkentaa, onko virhe "If"- vai "While"-kohde.
Puuttuva <b>EndIf</b>	Odotettiin kohdetta <b>EndIf</b> , mutta löydettiin toinen End-kohde
Puuttuva <b>EndFor</b>	Odotettiin kohdetta <b>EndFor</b> , mutta löydettiin toinen End-kohde
Puuttuva <b>EndWhile</b>	Odotettiin kohdetta <b>EndWhile</b> , mutta löydettiin toinen End-kohde
Puuttuva <b>EndLoop</b>	Odotettiin kohdetta <b>EndLoop</b> , mutta löydettiin toinen End-kohde
Puuttuva <b>EndTry</b>	Odotettiin kohdetta <b>EndTry</b> , mutta löydettiin

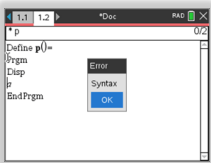
Virhetila	Uusi viesti
	toinen End-kohde
"Then" jätetty pois kohteen If <condition> jälkeen	Puuttuva If..Then
"Then" jätetty pois kohteen Elseif <condition> jälkeen	Then puuttuu kohdasta: Elseif.
Kun "Then", "Else", ja "Elseif" kohdattiin ohjauskohtien ulkopuolella	Then virheellinen kohtien ulkopuolella: If..Then..EndIf tai Try..EndTry
"Elseif" esiintyy kohdan "If..Then..EndIf" ulkopuolella	Elseif virheellinen kohdan ulkopuolella: If..Then..EndIf
"Then" esiintyy kohdan "If...EndIf" ulkopuolella	Then virheellinen kohdan ulkopuolella: If..EndIf

## Syntaksivirheet

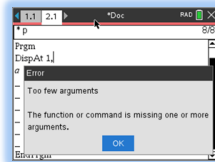
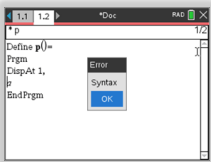
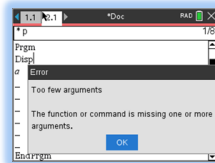
Jos tapauskomentoja, jotka odottavat yhtä tai useampaa argumenttia, kutsutaan epätäydellisellä argumenttiluettelolla, "Too few argument error" -virhe näytetään "syntax" -virheen sijaan.

Nykyinen käyttäytyminen	Uusi CX II -käyttäytyminen
 <p>A screenshot of a code editor window titled "1.1" with the text "RasdSeed" entered. A dialog box is displayed with the title "Error" and the message "Syntax". There are "Enter", "Syntax", and "OK" buttons.</p>	 <p>A screenshot of a code editor window titled "1.1" with the text "RasdSeed" entered. A dialog box is displayed with the title "Error" and the message "Too few arguments". Below the message, it says "The function or command is missing one or more arguments." There is an "OK" button.</p>
 <p>A screenshot of a code editor window titled "1.1" with the text "LU m" entered. A dialog box is displayed with the title "Error" and the message "Syntax". There are "Enter", "Syntax", and "OK" buttons.</p>	 <p>A screenshot of a code editor window titled "1.1" with the text "LU m" entered. A dialog box is displayed with the title "Error" and the message "Too few arguments". Below the message, it says "The function or command is missing one or more arguments." There is an "OK" button.</p>

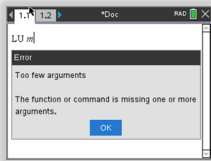
## Nykyinen käyttäytyminen



## Uusi CX II -käyttäytyminen



**Huomaa:** Kun epätäydellisen argumenttiluettelon jäljessä ei ole pilkkua, virheviesti on: "too few arguments". Tämä on sama kuin aiemmissa julkaisuissa.



# Vakiot ja arvot

Seuraavassa taulukossa on lueteltu vakiot ja niiden arvot, jotka ovat käytettävissä suoritettaessa yksikköjen muunnoksia. Ne voidaan kirjoittaa manuaalisesti tai valita

**Vakiot**-luettelosta **Utilities> Yksikkömuunnokset** (Kannettava laite: Paina  3 ).

Vakio	Nimi	Arvo
_c	valon nopeus	299792458 _m/_s
_Cc	Coulombin vakio	8987551787.3682 _m/_F
_Fc	Faradayn vakio	96485.33289 _coul/_mol
_g	Painovoiman kiihtyvyyys	9.80665 _m/_s <sup>2</sup>
_Gc	Gravitaatiovakio	6.67408E-11 _m <sup>3</sup> /_kg/_s <sup>2</sup>
_h	Planckin vakio	6.626070040E-34 _J _s
_k	Boltzmannin vakio	1.38064852E-23 _J/_°K
_μ0	Tyhjiön permeabiliteetti	1.2566370614359E-6 _N/_A <sup>2</sup>
_μb	Bohr magnetoni	9.274009994E-24 _J _m <sup>2</sup> /_Wb
_Me	Elektronin lepomassa	9.10938356E-31 _kg
_Mμ	Myonin massa	1.883531594E-28 _kg
_Mn	Neutronin lepomassa	1.674927471E-27 _kg
_Mp	Protonin lepomassa	1.672621898E-27 _kg
_Na	Avogadron vakio	6.022140857E23 / _mol
_q	Elektronin varaus	1.6021766208E-19 _coul
_Rb	Bohrin säde	5.2917721067E-11 _m
_Rc	Moolinen kaasuvakio	8.3144598 _J/_mol/_°K
_Rdb	Rydbergin vakio	10973731.568508/_m
_Re	Elektronin säde	2.8179403227E-15 _m
_u	Atomimassa	1.660539040E-27 _kg
_Vm	moolinen tilavuus	2.2413962E-2 _m <sup>3</sup> /_mol
_ε0	Tyhjiön permittiivisyys (vakio)	8.8541878176204E-12 _F/_m
_σ	Stefan-Boltzmannin vakio	5.670367E-8 _W/_m <sup>2</sup> /_°K <sup>4</sup>
_φ0	Magneettivuon kvantti	2.067833831E-15 _Wb

# Virhekoodit ja viestit

Kun tapahtuu virhe, virhekoodi määritetään muuttujaan *errCode*. Käyttäjän määrittämällä ohjelmilla ja funktioilla voidaan tutkia *errCode*-muuttujaa virheen syyä määrittämiseksi. Esimerkki muuttujan *errCode* käytöstä on esitetty Esimerkissä 2 **Try**-komennon kohdalla, sivu 207.

**Huomaa:** Jotkin virhetilanteet koskevat vain *TI-Nspire™ CAS -tuotteita ja toiset taas vain TI-Nspire™-tuotteita*.

## Virhekoodi Kuvaus

10 Funktio ei laskenut arvoa.

20 Kokeen tulos ei ollut TOSI eikä EPÄTOSI.

Määrittämättömiä muuttujia ei yleensä voi verrata. Esimerkiksi testi  $if\ a < b$  aiheuttaa virheen, jos joko  $a$  tai  $b$  on määrittämätön suoritettaessa  $if$ -lauseketta.

30 Argumentti ei voi olla kansion nimi.

40 Argumenttinvirhe

50 Argumentin yhteensopivuusvirhe

Kahden tai useamman argumentin on oltava samaa tyyppiä.

60 Argumentin on oltava Boolean lauseke tai kokonaisluku.

70 Argumentin on oltava desimaaliluku.

90 Argumentin on oltava lista.

100 Argumentin on oltava matriisi.

130 Argumentin on oltava merkkijono.

140 Argumentin on oltava muuttujan nimi.

Tarkista, että nimi:

- ei ala numerolla
- ei sisällä välilyöntejä tai erikoismerkkejä
- ei käytä alaviivaa tai pistettä väärällä tavalla
- ei ylitä pituusrajoituksia

Lisätietoja löytyy ohjekirjan Laskin-sovellusta käsittelevästä osasta.

160 Argumentin on oltava lauseke.



165 Paristovirta ei riitä lähettämiseen tai vastaanottamiseen.

Asenna uudet paristot ennen lähettämistä tai vastaanottoa.

170 Raja

Alarajan on oltava ylärajaa pienempi hakuväliä määritettäessä.

180 Keskeytys

Näppäintä  tai  on painettu pitkän laskutoimituksen tai ohjelman suorituksen aikana.

<b>Virhekoodi</b>	<b>Kuvaus</b>
190	Kehämäärittäminen  Tämä viesti tulee näkyviin muistin loppumisen välttämiseksi muuttujien arvojen äärettömän korvauksen aikana sievennettäessä lausekkeita. Virheen aiheuttaa esimerkiksi lauseke $a+1 \rightarrow a$ , jossa $a$ on määrittämätön muuttuja.
200	Määrittelyalueen lauseke ei kelpaa.  Tämän virheilmoituksen voi aiheuttaa esimerkiksi lauseke $\text{solve}(3x^2-4=0,x) \mid x < 0$ tai $x > 5$ , koska ehto on erotettu operaattorilla "or" eikä operaattorilla "and".
210	Datatyypin ei kelpaa.  Argumentin datatyyppi on väärä.
220	Riippuvainen raja-arvo
230	Dimensio  Lista tai matriisi-indeksi ei kelpaa. Jos esimerkiksi lista $\{1,2,3,4\}$ on tallennettu L1:een, tällöin $L1[5]$ aiheuttaa dimensiovirheen, koska L1 sisältää vain neljä elementtiä.
235	Dimensiovirhe Listoissa ei ole riittävästi elementtejä.
240	Dimensioiden yhteensopivuusvirhe  Kahdella tai useammalla argumentilla on oltava sama dimensio. Esimerkiksi lauseke $[1,2]+[1,2,3]$ aiheuttaa dimensioiden yhteensopivuusvirheen, koska matriisit sisältävät eri määrän elementtejä.
250	Jako nolalla
260	Määrittelyjoukkovirhe  Argumentin on kuuluttava tiettyyn määrittelyjoukkoon. Esimerkiksi <b>rand(0)</b> ei kelpaa.
270	Muuttujan nimi jo käytössä
280	Else ja Elseif eivät kelpaa lohkon If..EndIf ulkopuolella
290	EndTry-komennosta puuttuu sopiva Else-lauseke
295	Liian monta iteraatiota
300	Odotettavissa 2- tai 3-elementtinen lista tai matriisi
310	Funktion <b>nSolve</b> ensimmäisen argumentin on oltava yhden muuttujan yhtälö. Se ei voi sisältää muuta arvoitonta muuttujaa kuin käsittelyn kohteena olevan muuttujan.
320	$\text{solve}$ - tai $\text{cSolve}$ -funktion ensimmäisen argumentin on oltava yhtälö tai epäyhtälö.  Esimerkiksi $\text{solve}(3x^2-4,x)$ ei kelpaa, koska ensimmäinen argumentti ei ole yhtälö.
345	Ristiriitaiset yksiköt
350	Indeksi alueen ulkopuolella
360	Epäsuora merkkijono ei kelpaa muuttujan nimeksi.
380	Määrittämätön Ans  Edellinen laskutoimitus ei luonut Ans:n arvoa, tai aikaisempaa laskutoimitusta ei ole syötetty.

Virhekoodi	Kuvaus
390	Epäkelpo tehtävä
400	Epäkelpo tehtävän arvo
410	Virheellinen komento
430	Virheellinen nykyisille tila-asetuksille
435	Virheellinen arvaus
440	Virheellinen kertomerkitön kertolasku
	Esimerkiksi lauseke $x(x+1)$ ei kelpaa, mutta lausekkeen $x*(x+1)$ syntaksi on oikein. Tarkoituksena on välttää sekaannusta kertomerkitömiin kertolaskujen ja funktioiden komentojen kanssa.
450	Virhe funktiossa tai nykyisessä lausekkeessa
	Vain tietyt komennot kelpaavat käyttäjän määrittämässä funktiossa.
490	Virhe Try..EndTry-lohkossa
510	Virheellinen lista tai matriisi
550	Virheellinen ulkopuolinen funktio tai ohjelma
	Jotkin komennot eivät kelpaa funktion tai ohjelman ulkopuolella. Esimerkiksi <b>Local</b> -komentoa ei voi käyttää, ellei se ole funktion tai ohjelman sisällä.
560	Virhe lohkojen Loop..EndLoop, For..EndFor tai While..EndWhile ulkopuolella
	Esimerkiksi Exit-komento kelpaa vain näiden silmukkalohkojen sisällä.
565	Virheellinen ulkopuolinen ohjelma
570	Polkunimi ei kelpaa
	Esimerkiksi \muutt ei kelpaa.
575	Virheellinen polaarin kompleksiluku
580	Virheellinen ohjelmaviittaus
	Ohjelmiin ei voi viitata funktioiden tai lausekkeiden sisällä, kuten $1+p(x)$ , jossa p on ohjelma.
600	Kelpaamaton taulukko
605	Yksiköitä käytetty väärin
610	Virheellinen muuttujan nimi Local-lausekkeessa
620	Virheellinen muuttujan tai funktion nimi
630	Virheellinen muuttujan viittaus
640	Virheellinen vektorin syntaksi
650	Linkkisiirto
	Kahden laitteen välistä siirtoa ei ole suoritettu loppuun. Tarkista, että kaapeli on kytketty kunnolla kumpaankin laitteeseen.
665	Matriisia ei voi diagonalisoida
670	Muisti vähissä



**Virhekoodi Kuvaus**

1. Poista tietoja tästä asiakirjasta

2. Tallenna ja sulje tämä asiakirja

Jos toimenpiteet 1 ja 2 eivät auta, poista ja asenna paristot uudelleen

672 Resource exhaustion (Resurssit lopussa)

673 Resource exhaustion (Resurssit lopussa)

680 Puuttuva (

690 Puuttuva )

700 Puuttuva "

710 Puuttuva ]

720 Puuttuva }

730 Syntaksista puuttuu lohkon alku tai loppu

740 Then puuttuu lohokosta If..EndIf

750 Nimi ei ole funktio eikä ohjelma

765 Yhtään funktiota ei ole valittu

780 Ratkaisua ei löydy

800 Ei-reaalinen tulos

Esimerkiksi, jos ohjelmisto on Real-asetuksessa,  $\sqrt{-1}$  ei kelpaa.

Jotta voit saada kompleksilukuvastauksia, muuta reaali- tai kompleksitilan asetus valintaan RECTANGULAR (SUORAKULMA) tai POLAR (POLAARINEN).

830 Laskualueen ylitys

850 Ohjelmaa ei löydy

Toisen ohjelman sisällä olevaa ohjelman viittausta ei löytynyt määritetystä polusta ohjelman suorittamisen aikana.

855 Rand-tyyppiset funktiot eivät ole sallittuja kuvaajan piirtämisen aikana

860 Liian syvä rekursio

870 Varattu nimi tai järjestelmän muuttuja

900 Argumenttivrhe

Mediaani-mediaani-mallia ei voitu käyttää datasarjaan.

910 Syntaksivirhe

920 Tekstiä ei löydy

930 Liian vähän argumentteja

Funktioista tai komennosta puuttuu yksi tai useampia argumentteja.

940 Liian monta argumenttia

<b>Virhekoodi</b>	<b>Kuvaus</b>
	Lauseke tai yhtälö sisältää liian monta argumenttia, eikä sitä voi ratkaista.
950	Liian monta alaindeksiä
955	Liian monta määrittämätöntä muuttujaa
960	Muuttujaa ei ole määritetty
	Muuttujalle ei ole määritetty arvoa. Käytä jotakin seuraavista komennoista: <ul style="list-style-type: none"> <li>• <b>sto</b> →</li> <li>• <b>:=</b></li> <li>• <b>Define</b></li> </ul> määrittääksesi muuttujille arvot.
965	Käyttöjärjestelmälle ei ole lisenssiä
970	Muuttuja on käytössä, joten viittaukset tai muutokset eivät ole sallittuja
980	Muuttuja on suojattu
990	Virheellinen muuttujan nimi
	Varmista, että nimi ei ylitä pituusrajoituksia
1000	Ikkunamuuttujien määrittelyalue
1010	Zoomaus
1020	Sisäinen virhe
1030	Suojatun muistin rikkomus
1040	Toimintoa ei tueta Tämä toiminto vaatii tietokonealgebrajärjestelmän Kokeile TI-Nspire™ CAS -järjestelmää.
1045	Operaattoria ei tueta. Tämä operaattori vaatii tietokonealgebrajärjestelmän. Kokeile TI-Nspire™ CAS -järjestelmää.
1050	Toimintoa ei tueta. Tämä operaattori vaatii tietokonealgebrajärjestelmän. Kokeile TI-Nspire™ CAS -järjestelmää.
1060	Syötetyn argumentin on oltava numeerinen. Vain numeerisia arvoja sisältävät syötteet sallitaan.
1070	Trig-funktion argumentti on liian suuri täsmälliseen sievennykseen
1080	Ans-muuttujan käyttöä ei tueta. Tämä sovellus ei tue Ans-muuttujan käyttöä.
1090	Funktiota ei ole määritetty. Käytä jotakin seuraavista komennoista: <ul style="list-style-type: none"> <li>• <b>Define</b></li> <li>• <b>:=</b></li> <li>• <b>sto</b> →</li> </ul> määrittääksesi funktion.
1100	Ei-reaalinen laskutoimitus
	Esimerkiksi, jos ohjelmisto on Real-asetuksessa, $\sqrt{-1}$ ei kelpaa.

<b>Virhekoodi</b>	<b>Kuvaus</b>
	Jotta voit saada kompleksilukuvastauksia, muuta reaali- tai kompleksitilan asetus valintaan RECTANGULAR (SUORAKULMA) tai POLAR (POLAARINEN).
1110	Virheelliset rajat
1120	Ei etumerkin muutosta
1130	Argumentti ei voi olla lista tai matriisi
1140	Argumenttivorhe
	Ensimmäisen argumentin on oltava toisen argumentin sisällä oleva polynomilauseke. Jos toinen argumentti jätetään pois, ohjelmisto yrittää valita oletusarvon.
1150	Argumenttivorhe
	Ensimmäisten kahden argumentin on oltava kolmannen argumentin sisällä olevia polynomilausekkeita. Jos kolmas argumentti jätetään pois, ohjelmisto yrittää valita oletusarvon.
1160	Virheellinen kirjaston polkunimi
	Polkunimen on oltava muodossa <code>xxx\yyy</code> , jossa:
	<ul style="list-style-type: none"> <li>• <code>xxx</code>-osassa voi olla 1-16 merkkiä.</li> <li>• <code>yyy</code>-osassa voi olla 1–15 merkkiä.</li> </ul>
	Lisätietoja löydät ohjekirjan osasta Kirjasto.
1170	Kirjaston polkunimeä on käytetty väärin
	<ul style="list-style-type: none"> <li>• Polkunimelle ei voi määrittää arvoa komennon <b>Define</b>, := tai sto → avulla.</li> <li>• Polkunimeä ei voi määrittää paikalliseksi muuttujaksi tai käyttää funktion tai ohjelman parametrina.</li> </ul>
1180	Virheellinen kirjastomuuttujan nimi.
	Tarkista, että nimi:
	<ul style="list-style-type: none"> <li>• ei sisällä pistettä</li> <li>• ei ala alaviivalla</li> <li>• ei ylitä 15 merkkiä</li> </ul>
	Lisätietoja löydät ohjekirjan osasta Kirjasto.
1190	Kirjastodokumenttia ei löydy:
	<ul style="list-style-type: none"> <li>• Tarkista, että kirjasto on MyLib-kansiossa.</li> <li>• Päivitä kirjastot.</li> </ul>
	Lisätietoja löydät ohjekirjan osasta Kirjasto.
1200	Kirjastomuuttujaa ei löydy:
	<ul style="list-style-type: none"> <li>• Tarkista, että kirjaston ensimmäisessä tehtävässä on kirjastomuuttuja.</li> <li>• Tarkista, että kirjastomuuttujaksi on määritetty LibPub tai LibPriv.</li> <li>• Päivitä kirjastot.</li> </ul>
	Lisätietoja löydät ohjekirjan osasta Kirjasto.
1210	Virheellinen kirjaston pikavalintanimi.

**Virhekoodi Kuvaus**

Tarkista, että nimi:

- ei sisällä pistettä
- ei ala alaviivalla
- ei ylitä 16 merkkiä
- ei ole varattu nimi

Lisätietoja löydät ohjekirjan osasta Kirjasto.

1220

Määrittelyjoukkovirhe:

Funktiot `tangentLine` ja `normalLine` tukevat vain reaaliarvoisia funktioita.

1230

Määrittelyjoukkovirhe.

Trigonometrysten muunnosten operaattoreita ei tueta aste- eikä graadikulmatilassa.

1250

Argumenttievirhe

Käytä lineeariyhtälöryhmää.

Esimerkki kahden lineeariyhtälön ryhmästä, jossa on muuttujat  $x$  ja  $y$ :

$$3x+7y=5$$

$$2y-5x=-1$$

1260

Argumenttievirhe:

**nfMin:n** tai **nfMax:n** ensimmäisen argumentin on oltava yhden muuttujan sisältämä lauseke. Se ei voi sisältää muuta arvotonta muuttujaa kuin käsittelyn kohteena olevan muuttujan.

1270

Argumenttievirhe

Derivaatan asteen on oltava 1 tai 2.

1280

Argumenttievirhe

Käytä yhden muuttujan sisältämää laajennetussa muodossa olevaa polynomia.

1290

Argumenttievirhe

Käytä yhden muuttujan sisältämää polynomia.

1300

Argumenttievirhe

Polynomin kertoimien tuloksena on oltava numeerinen arvo.

1310

Argumenttievirhe:

Yhtä tai useampaa funktion argumenttia ei voi sieventää.

1380

Argumenttievirhe:

Sisäkkäisiä kutsuja `domain()` funktioon ei sallita.

## Varoituskoodit ja -viestit

Voit käyttää funktiota `warnCodes()` tallentaaksesi lausekkeen laskennan luomat varoituskoodit. Tämä taulukko luettelee jokaisen numeerisen varoituskoodin ja siihen liittyvän viestin. Esimerkki varoituskoodien tallentamisesta, katso `warnCodes()`, sivu 216.

Varoituskoodi	Viesti
10000	Laskutoimitus voi antaa väärä ratkaisuja. Kokeile tarvittaessa graafisia menetelmiä tulosten varmistamiseksi.
10001	Yhtälön derivointi voi antaa väärän yhtälön.
10002	Kyseenalainen ratkaisu Kokeile tarvittaessa graafisia menetelmiä tulosten varmistamiseksi.
10003	Kyseenalainen tarkkuus Kokeile tarvittaessa graafisia menetelmiä tulosten varmistamiseksi.
10004	Laskutoimituksessa voi hävitä ratkaisuja. Kokeile tarvittaessa graafisia menetelmiä tulosten varmistamiseksi.
10005	<code>cSolve</code> voi määrittää lisää nollakohtia.
10006	<code>Solve</code> voi määrittää lisää nollia. Kokeile tarvittaessa graafisia menetelmiä tulosten varmistamiseksi.
10007	Lisää ratkaisuja voi olla olemassa. Yritä määrittämällä oikeat ala- ja ylärajat ja/tai arvaamalla.  Esimerkkejä <code>solve()</code> käytöstä: <ul style="list-style-type: none"><li>• <code>solve(Yhtälö, Muut=Arvaus)   alaraja&lt;Muut&lt;yläraja</code></li><li>• <code>solve(Yhtälö, Muut)   alaraja&lt;Muut&lt;yläraja</code></li><li>• <code>solve(Yhtälö, Muut=Arvaus)</code></li></ul> Kokeile tarvittaessa graafisia menetelmiä tulosten varmistamiseksi.
10008	Tuloksen määrittelyjoukko voi olla pienempi kuin syötteen määrittelyjoukko.
10009	Tuloksen määrittelyjoukko voi olla suurempi kuin syötteen määrittelyjoukko.
10012	Ei-reaalinen laskutoimitus
10013	$\infty^0$ tai $\text{undef}^0$ korvattu 1:llä
10014	$\text{undef}^0$ korvattu 1:llä
10015	$1^\infty$ tai $1^\text{undef}$ korvattu 1:llä

Varoituskoodi	Viesti
10016	1^undef korvattu 1:ll
10017	Laskualueen ylitys korvattu $\infty$ :lla tai $-\infty$ :lla
10018	Laskutoimitus vaatii ja laskee 64-bittisen arvon.
10019	Muisti täynnä, sievennystä ei mahdollisesti suoriteta loppuun.
10020	Trig-funktion argumentti on liian suuri täsmälliseen sievennykseen.
10021	Syöte sisältää määrittämättömän parametrin. Tulos ei ehkä ole voimassa kaikille mahdollisille parametriarvoille.
10022	Ratkaisu on mahdollinen, kun määritetään oikeat ala- ja ylärajat.
10023	Skalaari on kerrottu tunnistematriisilla.
10024	Tulos saatu käyttäen likiarvo-aritmetiikkaa.
10025	Ekvivalenssia ei voida varmistaa EXACT-tilassa.
10026	Ehto saatetaan sivuuttaa. Määritä ehto muodossa "\" 'Muuttujan MathTestSymbol-vakio' tai näiden yhteismuoto, esim. 'x<3 and x>-12'

# Yleistä

## **Online-tuki**

[education.ti.com/eguide](http://education.ti.com/eguide)

Valitse maasi, niin näet lisää tuotetietoja.

## **Ota yhteyttä TI-tukeen**

[education.ti.com/ti-cares](http://education.ti.com/ti-cares)

Valitse maasi, niin näet teknisiä tietoja ja muita tukiresursseja.

## **Huolto- ja takuutiedot**

[education.ti.com/warranty](http://education.ti.com/warranty)

Valitse maasi, niin saat tietoa takuun kestosta ja ehdoista tai tuotepalvelusta.

Rajoitettu takuu. Tämä takuu ei vaikuta lainmukaisiin oikeuksiisi.

Texas Instruments Incorporated

12500 TI Blvd.

Dallas, TX 75243

# Indeksi

	–		–
–, vähennyslasku[*]	228	_, yksikön merkki	249
!		_, yksikön nimi	249
!, kertoma	239		
"		, (rajoittava operaattori)	252
", sekuntimuoto	248	+	
#		+, yhteenlasku	228
#, epäsuora operaattori	245, 277	/	
%		/, jakolasku[*]	230
%, prosentti	234	=	
&		≠, ei yhtä kuin[*]	235
&, liitä	239	=, on yhtä kuin	235
*		>	
*, kertolasku	229	>, suurempi kuin	237
,		∏	
, jaoton	249	∏, tulo[*]	242
, minuuttimuoto	248	∑	
.		∑( ), summa[*]	243
.-, piste-erotus	233	∑Int( )	244
.*, pistetulo	233	∑Prn( )	244
./, pisteosamäärä	233	√	
.^, pistepotenssi	234	√, neliöjuuri[*]	242
+. pistesumma	232	∫	
:		∫, integraali[*]	240
:=, määritä	254	≤	
∧		≤, pienempi tai yhtä suuri kuin	236
∧ <sup>-1</sup> , käänteisluku	251	≥	
∧, potenssi	231	≥, suurempi tai yhtä suuri kuin	237



►, muunna graadikulmaksi[Grad] ...	93	°, astekulmamuuoto[*] .....	247
►, muunna yksiköt[*] .....	250	°, astetta/minuuttia/sekuntia[*] .....	248
►, asteen regressio, QuartReg .....	153		
►approxFraction( ) .....	14	<b>0</b>	
►cos, näytä kulman kosini[cos] .....	32	Ob, binaarinen indikaattori .....	255
►Cylind, näytä vektorina sylinderikoordinaatistossa [Cylind] .....	46	Oh, heksadesimaalinen indikaattori .....	255
►DD, näytä desimaalikulmana[DD] ..	49	<b>1</b>	
►Desimaali, näytä tulos desimaalilukuna[Desimaali]	49	10^( ), kymmenen potenssi .....	251
►DMS, näytä asteina/minuutteina/sekun teina[DMS] .....	58	<b>2</b>	
►exp, näytä e:n arvolla[exp] .....	68	2 otoksen F-testi .....	81
►Kantaluku10, näytä desimaalikonaislukuna [Kantaluku10] .....	19	2. asteen regressio, QuadReg .....	151
►Kantaluku16, näytä heksadesimaalilukuna [Kantaluku16] .....	20	<b>3</b>	
►Kantaluku2, näytä binaarisena [Kantaluku2] .....	18	3. asteen regressio, CubicReg .....	44
►Polaarinen, näytä polaarisen vektorina[Polaarinen] .....	143	<b>A</b>	
►Rad, muunna radiaanikulmaan .....	155	abs( ), itseisarvo .....	8
►Rect, näytä suorakulmavektorina ..	158	alaraja, floor( ) .....	76
►sin, näytä kulman sini[sin] .....	181	alaviiva, _ .....	249
►Sphere, näytä pallonmuotoisena vektorina[Sphere] .....	190	alimatriisi, subMat( ) .....	195, 197
		amortTbl( ), lyhennystaulukko .....	8, 17
<b>⇒</b>		and, Boolean operator .....	9
⇒, looginen seuraus[*] .....	238, 274	angle( ), kulma .....	10
		ANOVA, yksisuuntainen varianssianalyysi .....	11
<b>→</b>		ANOVA2way, kaksisuuntainen varianssianalyysi .....	11
→, tallenna .....	253	ans, viimeinen vastaus .....	13
		approx( ), likimääräinen .....	13
<b>↔</b>		approxRational( ) .....	14
↔, looginen kaksoisseuraus[*] .....	238	arccos( ) .....	14
		arccosh( ) .....	14
<b>©</b>		arccot( ) .....	15
©, kommentti .....	254	arcoth( ) .....	15
		arccsc( ) .....	15
		arccsch( ) .....	15
		arcLen( ), kaaren pituus .....	15
		arcsec( ) .....	15
		arcsech( ) .....	15
		arcsin( ) .....	16
		arcsinh( ) .....	16
		arctan( ) .....	16

arctanh()	16	cos( ), kosini	33
arkuskosini, cos <sup>-1</sup> ( )	34	cosh <sup>-1</sup> ( ), hyperbolinen arkuskosini	35
arkussini, sin <sup>-1</sup> ( )	182	cosh( ), hyperbolinen kosini	35
arkustangentti, tan <sup>-1</sup> ( )	198	cot <sup>-1</sup> ( ), arkuskotangentti	36
aseta		cot( ), kotangentti	36
tila, setMode( )	176	coth <sup>-1</sup> ( ), hyperbolinen	
asetukset, hae nykyiset	89	arkuskotangentti	37
aste/minuutti/sekunti-muoto	248	coth( ), hyperbolinen kotangentti	37
astekulmamuoto, -	247	count( ), laske listan kohtien	
augment( ), lisää/ketjuta	16	lukumäärä	37
avgRC( ), keskimääräinen		countif( ), laske listan yksiköiden	
muutosnopeus	16	lukumäärä ehdollisesti	38
		cPolyRoots()	39
<b>B</b>		crossP( ), ristitulo	39
binaarinen		csc <sup>-1</sup> ( ), käänteiskosekantti	40
indikaattori, 0b	255	csc( ), kosekantti	40
näytä, 4Kantaluku2	18	csch <sup>-1</sup> ( ), käänteinen hyperbolinen	
binomCdf( )	21, 99	kosekantti	41
binomPdf( )	21	csch( ), hyperbolinen kosekantti	40
Boolean operators		cSolve( ), ratkaise kompleksiyhtälö	41
and	9	CubicReg, 3. asteen regressio	44
Booleen operaattorit		cumulativeSum( ), kumulatiivinen	
⇒	238, 274	summa	45
⇔	238	cycle, Cycle	45
ei	135	Cycle, cycle	45
ei-ja	128	cZeros( ), kompleksiset nollakohtat	46
eikä	133		
tai	139		
xor	218		
		<b>D</b>	
<b>C</b>		d( ), ensimmäinen derivaatta	239
Cdf( )	74	dbd( ), päivämäärien väliset päivät	48
ceiling( ), yläraja	21	Define (Määritä)	49
centralDiff( )	22	Define LibPriv (Määritä LibPriv)	51
cFactor( ), kompleksilukutekijä	23	Define LibPub (Määritä LibPub)	51
char( ), merkkijono	24	Define, määritä	49
charPoly( )	24	deltaList()	52
clearAZ	27	deltaTmpCnv()	52
ClrErr, poista virhe	27	DelVar, poista muuttuja	52
colAugment	28	delVoid( ), poista tyhjät elementit	53
colDim( ), matriisin sarakemitat	28	derivaatat	
colNorm( ), matriisin sarakenormi	28	ensimmäinen derivaatta, d( )	239
comDenom( ), yhteinen nimittäjä	28	numeerinen derivaatta, nDeriv( )	131
completeSquare( ), complete square	29	numeerinen derivaatta,	
conj( ), liittokompleksiluku	30	nDerivative( )	130
constructMat( ), luo matriisi	30	derivaatta tai n:s derivaatta	
corrMat( ), korrelaatiomatriisi	32	malli	6
cos <sup>-1</sup> , arkuskosini	34	desimaali	
		kokonaislukunäyttö,	19

4Kantaluku10 .....		epäsuora operaattori (#) .....	277
kulmanäyttö, ►DD .....	49	epäsuora operaattori, # .....	245
deSolve( ), ratkaisu .....	53	etumerkki, sign( ) .....	179
det( ), matriisin determinantti .....	55	euler( ), Euler function .....	65
diag( ), diagonaalimatriisi .....	56	exact( ), täsmällinen .....	67
dim( ), mitta .....	56	Exit, lopeta .....	67
Disp, näytä tiedot .....	56, 171	exp( ), e korotettuna potenssiin ....	68
DispAt .....	57	exp►hist( ), lauseke listaksi .....	69
dominantTerm( ), dominanttitermi ..	60	expand( ), lavenna .....	69
dominanttitermi, dominantTerm( ) ..	60	expr( ), merkkijono lausekkeeksi ....	71, 115
dotP( ), pistetulo .....	61	ExpReg, eksponentiaalinen regressio	71

## E

e eksponentti	
malli .....	2
e korotettuna potenssiin, e^( ) .....	62, 68
E, eksponentti .....	246
e, näytä lauseke e:n arvolla .....	68
e^( ), e korotettuna potenssiin .....	62
efektiivinen korko, eff( ) .....	62
eff( ), muunna nimelliskorko	
efektiiviseksi koroksi .....	62
ei-ja, Boolean operaattori .....	128
ei yhtä kuin, ≠ .....	235
ei, Boolean operaattori .....	135
eigVc( ), ominaisvektori .....	63
eigVl( ), ominaisarvo .....	63
eikä, Boolean operaattori .....	133
eksklusioio "   " -operaattorilla .....	252
eksponentiaalinen regressio, ExpReg	71
eksponentti	
malli .....	1
eksponentti, E .....	246
else if, Elseif .....	64
Elseif, else if .....	64
end	
for, EndFor .....	78
funktio, EndFunc .....	82
ohjelma, EndPrgm .....	148
silmukka, EndLoop .....	118
try, EndTry .....	207
while, EndWhile .....	217
end while, EndWhile .....	217
EndTry, kokeilun loppu .....	207
EndWhile, end while .....	217
ensimmäinen derivaatta	
malli .....	6
EOS (yhtälökäyttöjärjestelmä) .....	276

## F

factor( ), tekijä .....	72
Fill, matriisin täyttö .....	75
FiveNumSummary .....	75
floor( ), alaraja .....	76
fMax( ), funktion maksimi .....	76
fMin( ), funktion minimi .....	77
For .....	78
for, For .....	78
For, for .....	78
format( ), merkkijonon muoto .....	78
fpart( ), funktion osa .....	79
freqTable( ) .....	80
frequency( ) .....	80
Frobeniusin normi, norm( ) .....	134
Func, funktio .....	82
Func, ohjelmafunktio .....	82
funktion loppu, EndFunc .....	82
funktiot	
käyttäjän määrittämät .....	49
maksimi, fMax( ) .....	76
minimi, fMin( ) .....	77
ohjelmafunktio, Func .....	82
osa, fpart( ) .....	79
funktiot ja muuttujat	
kopiointi .....	31

## G

g, graadit .....	246
gcd( ), suurin yhteinen jakaja .....	83
geomCdf( ) .....	83
geomPdf( ) .....	83
Get .....	84, 266
getDenom( ), hae/laske nimittäjä ...	85
getKey( ) .....	85
getLangInfo( ), hae/anna kielitiedot	88

getLockInfo( ), testaa muuttujan tai muuttujaryhmän lukitustilan .....	89	normaalijakauma) .....	
getMode( ), hae tila-asetukset .....	89	invnt( ) .....	100
getNum( ), hae/laske luku .....	91	Inv $\chi^2$ ( ) .....	98
GetStr .....	91	iPart(), kokonaisosa .....	100
getType( ), get type of variable .....	91	irr( ), sisäinen palautuksen määrä sisäinen palautuksen määrä , irr( ) .....	101
getVarInfo( ), hae/laske muuttujien tiedot .....	92	isPrime(), jaottoman testi .....	101
Goto, siirry .....	93	isVoid( ), tyhjän testi .....	102
graadimuoto, g .....	246	itseisarvo .....	
		malli .....	4

## H

hae/laske .....	
luku, getNum( ) .....	91
muuttujien tiedot, getVarInfo( ) .....	88, 92
nimitäjä, getDenom( ) .....	85
heksadesimaali .....	
näytä, 4Kantaluku16 .....	20
heksadesimaalinen .....	
indikaattori, Oh .....	255
hyperbolinen .....	
areatangenti, tanh <sup>-1</sup> ( ) .....	200
arkuskosini, cosh <sup>-1</sup> ( ) .....	35
arkussini, sinh <sup>-1</sup> ( ) .....	183
kosini, cosh( ) .....	35
sini, sinh( ) .....	183
tangenti, tanh( ) .....	199

## I

identity(), identtinen matriisi .....	93
identtinen matriisi, identity() .....	93
If, jos .....	94
ifFn( ) .....	95
imag(), imaginaarinen osa .....	96
imaginaarinen osa, imag() .....	96
ImpDif( ), implisiittinen derivaatta ..	96
implisiittinen derivaatta, Impdif( ) ..	96
Input, syöte .....	96
inString(), merkkijonon sisällä .....	97
int(), kokonaisluku .....	97
intDiv(), kokonaisluvun jakaminen ..	97
integraali, %o .....	240
interpolate( ), interpoloi .....	98
invF( ) .....	99
invNorm(), käänteinen kumulatiivinen .....	100

## J

jäännös, remain() .....	161
jakakaumafunktiot .....	
binomCdf( ) .....	99
invNorm( ) .....	100
invnt( ) .....	100
jakaumafunktiot .....	
binomCdf( ) .....	21
binomPdf( ) .....	21
Inv $\chi^2$ ( ) .....	98
normCdf( ) .....	134
normPdf( ) .....	134
poissCdf( ) .....	143
poissPdf( ) .....	143
tCdf( ) .....	201
tPdf( ) .....	206
$\chi^2$ 2way( ) .....	25
$\chi^2$ Cdf( ) .....	25
$\chi^2$ GOF( ) .....	26
$\chi^2$ Pdf( ) .....	26
jakolasku, P .....	230
jaoton, .....	249
jaottoman luvun testi, isPrime() .....	101
jos käytettävissäsi on   .....	252
jos, EndIf .....	94
jos, If .....	94

## K

käänteinen kumulatiivinen normaalijakauma (invNorm( ) .....	100
käänteinen, $\wedge^{-1}$ .....	251
käänteisluku, $\wedge^{-1}$ .....	251
kaaren pituus, arcLen( ) .....	15
kahden muuttujan tulokset, TwoVar .....	211
käyttäjän määrittämät funktiot .....	49

käyttäjän määrittämät funktiot ja ohjelmat .....	51	laske polynomi, polyEval( ) .....	145
kertolasku, * .....	229	lausekkeet	
kertoma, ! .....	239	lauseke listaksi, exp►list( ) .....	69
keskiarvo, mean( ) .....	120	merkkijono lausekkeeksi, expr( ) .....	71, 115
keskihajonta, stdDev( ) .....	193-194, 214	lavenna, expand( ) .....	69
keskimääräinen muutosnopeus, avgRC( ) .....	16	Lbl, tunnus .....	102
keskimäinen merkkijono, mid( ) ..	123	lcm, pienin yhteinen jaettava .....	102
kieli		left( ), vasen .....	103
hae kielitiedot .....	88	LibPriv .....	51
kierrä, rotate( ) .....	166	LibPub .....	51
kirjasto		libShortcut( ), luo pikavalinnat kirjasto-objekteihin .....	103
luo pikavalinnat objekteihin .....	103	liitä, & .....	239
kokonaisluku, int( ) .....	97	likimääräinen, approx( ) .....	13
kokonaisluvun jakaminen, intDiv( ) ..	97	limit( ) tai lim( ), raja-arvo .....	104
kokonaisuosa, iPart( ) .....	100	lineaarinen regressio, LinRegAx .....	106
kombinaatiot, nCr( ) .....	129	lineaarinen regressio, LinRegBx .....	105, 107
kommentti, © .....	254	LinRegBx, lineaarinen regressio .....	105
kompleksi		LinRegMx, lineaarinen regressio .....	106
liitto, conj( ) .....	30	LinRegtIntervals, lineaarinen regressio .....	107
nollakohdat, cZeros( ) .....	46	LinRegtTest .....	109
ratkaise, cSolve( ) .....	41	linSolve( ) .....	110
tekijä, cFactor( ) .....	23	lisää/ketjuta, augment( ) .....	16
kopioi muuttuja tai funktio, CopyVar	31	list►mat( ), listasta matriisiksi .....	111
korkeoerien summa .....	244	lista, laske kohtien lukumäärä .....	37
korrelaatiomatriisi, corrMat( ) .....	32	lista, laske yksiköiden lukumäärä ehdollisesti .....	38
korvaaminen " "-operaattorilla .....	252	listasta matriisiksi, list►mat( ) .....	111
kosini		listat	
näytä kulman .....	32	erotukset listassa, @list( ) .....	111
kosini, cos( ) .....	33	erotus, @list( ) .....	111
kotangentti, cot( ) .....	36	keskimäinen merkkijono, mid( ) .....	123
kulma, angle( ) .....	10	kumulatiivinen summa, cumulativeSum( ) .....	45
kumulatiivinen summa, cumulativeSum( ) .....	45	lajittelle laskevaan järjestykseen, SortD .....	190
kun, when( ) .....	217	lajittelle nousevaan järjestykseen, SortA .....	189
kymmenen potenssi, 10^( ) .....	251	lauseke listaksi, exp►list( ) .....	69
<b>L</b>			
lajitteleminen		lisää/ketjuta, augment( ) .....	16
laskeva järjestys, SortD .....	190	listasta matriisiksi, list►mat( ) ..	111
nouseva järjestys, SortA .....	189	maksimi, max( ) .....	120
laske listan kohtien lukumäärä, count( ) .....	37	matriisi listaksi, mat►list( ) .....	119
laske listan yksiköiden lukumäärä ehdollisesti, countif( ) .....	38	minimi, min( ) .....	123
laske päivämäärien väliset päivät, dbd( ) .....	48	pistetulo, dotP( ) .....	61
		ristitulo, crossP( ) .....	39
		tulo, product( ) .....	149

tyhjät elementit .....	272	Logaritmi .....	2
uusi, newList() .....	130	määrätty integraali .....	7
yhteenlasku, sum() .....	195-196	määrittämätön integraali .....	7
ln(), luonnollinen logaritmi .....	112	matriisi (1 × 2) .....	4
LnReg, logaritminen regressio .....	112	matriisi (2 × 1) .....	5
Local, paikallinen muuttuja .....	114	matriisi (2 × 2) .....	4
Lock, lukitse muuttuja tai		matriisi (m × n) .....	5
muuttujaryhmä .....	114	murtoluku .....	1
Logaritmi		n:s juuri .....	2
malli .....	2	neliöjuuri .....	1
logaritminen regressio, LnReg .....	112	paloittain määritelty funktio (2-	
logaritmit .....	112	osainen) .....	2
Logistic, logistinen regressio .....	116	paloittain määritelty funktio (N-	
LogisticD, logistinen regressio .....	117	osainen) .....	3
logistinen regressio, Logistic .....	116	raja-arvo .....	7
logistinen regressio, LogisticD .....	117	sum (G) .....	5
looginen kaksoisseuraus, ⇔ .....	238	toinen derivaatta .....	6
looginen seuraus, ⇒ .....	238, 274	tulo (P) .....	6
Loop, silmukka .....	118	yhtälöpari (2 yhtälöä) .....	3
lopeta, Exit .....	67	yhtälöryhmä (N yhtälöä) .....	4
loppu .....	94	matristi(), matriisi listaksi .....	119
loppu jos, EndIf .....	94	matriisi (1 × 2)	
LU, matriisin ala-ylä-dekomponointi	119	malli .....	4
lukuja		matriisi (2 × 1)	
kierrä, rotate() .....	166	malli .....	5
oikea, right() .....	164	matriisi (2 × 2)	
lukuja		malli .....	4
oikea, right() .....	98	matriisi (m × n)	
luo matriisi, constructMat() .....	30	malli .....	5
luonnollinen logaritmi, ln() .....	112	matriisi listaksi, matristi() .....	119
lyhennystaulukko, amortTbl() .....	8, 17	matriisit	
		ala-ylä-dekomponointi, LU .....	119
		alimatriisi, subMat() .....	195, 197
		determinantti, det() .....	55
		diagonaali, diag() .....	56
		identtinen, identity() .....	93
		kumulatiivinen summa,	
		cumulativeSum() .....	45
		lisää/ketjuta, augment() .....	16
		listasta matriisiksi, listmat() ..	111
		maksimi, max() .....	120
		matriisi listaksi, matristi() .....	119
		minimi, min() .....	123
		mitta, dim() .....	56
		ominaisarvo, eigVl() .....	63
		ominaisvektori, eigVc() .....	63
		piste-erotus, - .....	233
		pisteosamäärä, ./ .....	233
		pistepotenssi, .^ .....	234
<b>M</b>			
määrätty integraali			
malli .....	7		
määritä, Define .....	49		
määrittämätön integraali			
malli .....	7		
määrittäminen			
julkinen funktio tai ohjelma .....	51		
yksityinen funktio tai ohjelma ..	51		
maksimi, max() .....	120		
mallit			
derivaatta tai n:s derivaatta .....	6		
eksponentti .....	2		
eksponentti .....	1		
ensimmäinen derivaatta .....	6		
itseisarvo .....	4		

pistesumma, .+	232	muoto, format( )	78
pistetulo, *	233	muotoilu	78
QR-hajottaminen, QR	151	siirrä, shift( )	177
rivi-echelon-muoto, ref( )	159	sisällä, inString()	97
rivien kertominen ja yhteenlasku, mRowAdd ( )	125	vasen, left( )	103
rivien yhteenlasku, rowAdd( )	168	mid( ), keskimäinen merkkijono	123
rivikoko, rowDim( )	168	min( ), minimi	123
rivinormi, rowNorm( )	168	minimi, min( )	123
rivinvaihto, rowSwap( )	169	minuuttimuoto,	248
rivioperaatio, mRow( )	125	mirr( ), modifioitu sisäinen korkokanta	124
sarakemitat, colDim( )	28	mitta, dim( )	56
sarakenormi, colNorm( )	28	mod( ), modulo	125
satunnainen, randMat( )	156	modifioitu sisäinen korkokanta, mirr ( )	124
täyttäminen, Fill	75	modulo, mod( )	125
transponoi, T	197	Moninkertaisen lineaarisen regression t-testi	127
tulo, product( )	149	mRow( ), matriisin rivioperaatio	125
uusi, newMat( )	130	mRowAdd( ), matriisin rivien kertominen ja yhteenlasku	125
yhteenlasku, sum( )	195-196	MultReg	125
max( ), maksimi	120	MultRegIntervals( )	126
mean( ), keskiarvo	120	MultRegTests( )	127
mediaani-mediaani-suoran regressio, MedMed	121	murtoluvut	
mediaani, median( )	121	malli	1
median( ), mediaani	121	propFrac	150
MedMed, mediaani-mediaani- suoran regressio	121	muunna	
merkit		4Grad	93
merkkijono, char( )	24	yksiköt	250
numeerinen koodi, ord( )	141	muuntaminen	
merkkijono		►Rad	155
mitta, dim( )	56	muutoin, Else	94
pituus	56	muuttuja	
merkkijono, char( )	24	nimen luominen merkkijonosta	277
merkkijonon muoto, format( )	78	muuttujat	
merkkijonon pituus	56	paikallinen, Local	114
merkkijonon sisällä, inString()	97	poista kaikki yksikirjaimiset	27
merkkijonot		poista, DelVar	52
epäsuora operaattori, #	245	muuttujat ja funktiot	
käyttö muuttujanimien luomisessa	277	kopioiminen	31
keskimäinen merkkijono, mid( )	123	muuttujat, lukitseminen ja vapauttaminen	89, 114, 214
lauseke merkkijonoksi, string( )	195	muuttujien ja muuttujaryhmien lukitseminen	114
liitä, &	239	muuttujien ja muuttujaryhmien vapauttaminen	214
merkkijono lausekkeeksi, expr( )	71, 115		
merkkijono, char( )	24		
merkkikoodi, ord( )	141		

## N

n:s juuri	
malli .....	2
näppäimistön pikavalinnat .....	274
näytä	
asteina/minuutteina/sekunteina,	
►DMS .....	58
binaarisena, 4Kantaluku2 .....	18
desimaalikokonaislukuna,	
4Kantaluku10 .....	19
desimaalikulmana, ►DD .....	49
heksadesimaalilukuna,	
4Kantaluku16 .....	20
pallonmuotoinen vektori,	
►Sphere .....	190
polaarinen vektori, 4Polaarinen .....	143
suorakulmavektorina, ►Rect .....	158
vektorina	
sylinterikoordinaatistos	
sa, 4Cylind .....	46
näytä tiedot, Disp .....	56, 171
näyttö	
asteina/minuutteina/sekun	
teina, ►DMS .....	58
näyttö pallonmuotoisena vektorina,	
►Sphere .....	190
näyttö vektorina	
sylinterikoordinaatistossa,	
►Cylind .....	46
nCr( ), kombinaatiot .....	129
nDerivative( ), numeerinen	
derivaatta .....	130
negaatio, negatiivisten lukujen	
syöttäminen .....	277
neliöjuuri	
malli .....	1
neliöjuuri, √( ) .....	191, 242
nettonykyarvo, npv( ) .....	136
newList( ), uusi lista .....	130
newMat( ), uusi matriisi .....	130
nfMax( ), numeerisen funktion	
maksimi .....	131
nfMin( ), numeerisen funktion	
minimi .....	131
nimelliskorko, nom( ) .....	132
nimittäjä .....	28
nInt( ), numeerinen integraali .....	131

nollakohdat, zeroes( ) .....	219
nom ), muuta efektiivinen korko	
nimelliskoroksi .....	132
norm( ), Frobeniusin normi .....	134
normaalijakuman todennäköisyys,	
normCdf( ) .....	134
normaalisuora, normalLine( ) .....	134
normalLine( ) .....	134
normCdf( ) .....	134
normPdf( ) .....	134
nPr( ), permutaatiot .....	136
npv( ), nettonykyarvo .....	136
nSolve( ), numeerinen ratkaisu .....	137
numeerinen	
derivaatta, nDeriv( ) .....	131
derivaatta, nDerivative( ) .....	130
integraali, nInt( ) .....	131
ratkaisu, nSolve( ) .....	137

## O

objektit	
luo pikavalinnat kirjastoon .....	103
Odota-komento .....	215
ohita virhe, PassErr .....	142
ohjelmat	
julkisen kirjaston määrittäminen .....	51
yksityisen kirjaston	
määrittäminen .....	51
ohjelmat ja ohjelmointi	
kokeilun loppu, EndTry .....	207
lopeta ohjelma, EndPrgm .....	148
näytä I/O-näyttö, Disp .....	56, 171
poista virhe, ClrErr .....	27
try, Try .....	207
ohjelmointi	
määritä ohjelma, Prgm .....	148
näytä tiedot, Disp .....	56, 171
ohita virhe, PassErr .....	142
oikea, right( ) .....	98, 164
ominaisarvo, eigVl( ) .....	63
ominaisvektori, eigVc( ) .....	63
on yhtä kuin, = .....	235
OneVar, yhden muuttujan tilastot ..	138
operaattorit	
sievennysjärjestys .....	276
ord( ), numeerinen merkkikoodi .....	141



<b>P</b>		
$P \rightarrow R x()$ , x:n suorakulmakoordinaatti	141	
$P \rightarrow R y()$ , y:n suorakulmakoordinaatti	141	
pääoman maksuerien summa	244	
paikallinen muuttuja, Local	114	
paikallinen, Local	114	
päivämäärien väliset päivät, dbd()	48	
Palaa, return	164	
paloittain määritelty funktio (2-osainen)		
malli	2	
paloittain määritelty funktio (N-osainen)		
malli	3	
PassErr, ohita virhe	142	
Pdf()	80	
permutaatiot, nPr()	136	
piecewise()	143	
pienempi tai yhtä suuri kuin, {	236	
pienin yhteinen jaettava, lcm	102	
piirrä	261-263	
pikavalinnat, näppäimistö	274	
piste		
erotus, -	233	
osamäärä, ./	233	
potenssi, ^	234	
summa, .+	232	
tulo, *	233	
tulo, dotP()	61	
poissCdf()	143	
poissPdf()	143	
poista		
poista, ClrErr	27	
tyhjät elementit listasta	53	
poistaminen		
muuttuja, DelVar	52	
polaarinen		
koordinaatti, $R \rightarrow Pr()$	154	
koordinaatti, $R \rightarrow P\theta()$	154	
vektorinäyttö, 4Polaarinen	143	
polyCoef()	144	
polyDegree()	145	
polyEval(), laske polynomi	145	
polyGcd()	146	
polynomit		
laske, polyEval()	145	
satunnainen, randPoly()	157	
PolyRoots()	147	
potenssi, ^	231	
potenssiregressio,		
PowerReg	147, 161, 163, 203	
PowerReg, potenssiregressio	147	
Prgm, määritysohjelma	148	
prodSeq()	149	
product(), tulo	149	
propFrac, varsinainen murtoluku	150	
prosentti, %	234	
pyöristä, round()	167	
PyydäMerkkij	163	
Pyyntö	161	
<b>Q</b>		
QR-hajottaminen, QR	151	
QR, QR-hajottaminen (tekijöihin jako)	151	
QuadReg, 2. asteen regressio	151	
QuartReg, 4. asteen regressio	153	
<b>R</b>		
R, radiaani	246	
$R \rightarrow Pr()$ , polaarinen koordinaatti	154	
$R \rightarrow P\theta()$ , polaarinen koordinaatti	154	
radiaani, R	246	
rahan aika-arvo, Korko	210	
rahan aika-arvo, maksuerä	210	
rahan aika-arvo, maksuerien lukumäärä	210	
rahan aika-arvo, nykyarvo	211	
rahan aika-arvo, Tuleva arvo	210	
raja-arvo		
lim()	104	
limit()	104	
malli	7	
rajoittava operaattori " "	252	
rajoittava operaattori,		
laskemisjärjestys	276	
rand(), satunnaisluku	155	
randBin, satunnaisluku	155	
randInt(), satunnainen kokonaisluku	155	
randMat(), satunnaismatriisi	156	
randNorm(), satunnainen normaalijakauma	156	
randPoly(), satunnaispolynomi	157	
randSamp()	157	
RandSeed, satunnaisluvun	157	

siemenluku .....		normaalijakauma, randNorm() .....	156
ratkaise, solve() .....	185	polynomi, randPoly() .....	157
ratkaisu, deSolve() .....	53	siemenluku, RandSeed .....	157
reaali, real() .....	157	satunnaisotos .....	157
real(), reaali .....	157	sec <sup>-1</sup> ( ), käänteissekantti .....	170
ref(), rivi-echelon-muoto .....	159	sec( ), sekantti .....	169
RefreshProbeVars .....	160	sech <sup>-1</sup> ( ), käänteinen hyperbolinen sekantti .....	171
regressiot		sech( ), hyperbolinen sekantti .....	171
2. asteen, QuadReg .....	151	sekaluvut, käyttäen funktiota propFrac(>) .....	150
3. asteen, CubicReg .....	44	sekuntimuoto, " .....	248
4. asteen, QuartReg .....	153	sekvenssi, seq( ) .....	172
eksponentiaalinen, ExpReg .....	71	seq( ), sekvenssi .....	172
lineaarinen regressio, LinRegAx .....	106	seqGen( ) .....	172
lineaarinen regressio, LinRegBx .....	105, 107	seqn( ) .....	173
logaritminen, LnReg .....	112	sequence, seq( ) .....	172-173
Logistinen .....	116	series( ), sarja .....	174
logistinen, Logistic .....	117	setMode( ), aseta tila .....	176
mediaani-mediaani-suora, MedMed .....	121	shift( ), siirrä .....	177
MultReg .....	125	sievennetty rivi-echelon-muoto, rref ( ) .....	169
potenssiregressio, PowerReg .....	147, 161, 163, 203	sievennysjärjestys .....	276
sinimuotoinen, SinReg .....	184	sign( ), etumerkki .....	179
remain(), jäännös .....	161	siirry, Goto .....	93
return, Palaa .....	164	silmukan loppu, EndLoop .....	118
right( ), oikea .....	164	silmukka, Loop .....	118
right, right( ) .....	29, 65, 216	simult( ), samanaikaiset yhtälöt .....	180
ristitulo, crossP( ) .....	39	sin <sup>-1</sup> ( ), arkussini .....	182
rivi-echelon-muoto, ref() .....	159	sin( ), sini .....	181
rk23( ), Runge Kutta -funktio .....	164	sinh <sup>-1</sup> ( ), hyperbolinen arkussini .....	183
rotate(), kierrä .....	166	sinh( ), hyperbolinen sini .....	183
round(), pyöristä .....	167	sini näytä kulman .....	181
rowAdd(), matriisin rivien yhteenlasku .....	168	sini, sin( ) .....	181
rowDim(), matriisin rivikoko .....	168	siniregressio, SinReg .....	184
rowNorm(), matriisin rivinormi .....	168	SinReg, siniregressio .....	184
rowSwap(), matriisin rivinvaihto .....	169	solve( ), ratkaise .....	185
rref( ), sievennetty rivi-echelon- muoto .....	169	SortA, lajittele nousevaan järjestykseen .....	189
ryhmät, lukitseminen ja vapauttaminen .....	114, 214	SortD, lajittele laskevaan järjestykseen .....	190
ryhmät, lukitustilan testaaminen .....	89	sqrt( ), neliöjuuri .....	191
<b>S</b>		stat.results .....	191
samanaikaiset yhtälöt, simult( ) .....	180	stat.values .....	193
sarja, series( ) .....	174	stdDevPop( ), perusjoukon keskihajonta .....	193
satunnainen matriisi, randMat( ) .....	156	stdDevSamp( ), otoksen keskihajonta Stop-komento .....	194

string( ), lauseke merkkijonoksi	195	Test_2S, 2 otoksen F-testi	81
strings		tExpand( ), trigonometrinen	
right, right( )	29, 65, 216	laajennus	202
studentint-todennäköisyysjakauma,		tila-asetukset, getMode( )	89
tCdf( )	201	tilastot	
studentint-todennäköisyystiheys,		kahden muuttujan tulokset,	
tPdf( )	206	TwoVar	211
subMat( ), alimatriisi	195, 197	kertoma, !	239
sum( ), yhteenlasku	195	keskiarvo, mean( )	120
sumlf( )	196	keskihajonta, stdDev( )	193-194, 214
summa (G)		kombinaatiot, nCr( )	129
malli	5	mediaani, median( )	121
summa, S( )	243	permutaatiot, nPr( )	136
sumSeq()	197	satunnainen normaalijakauma,	
suorakulmavektorinäyttö, ►Rect	158	randNorm( )	156
suurempi kuin, >	237	satunnainen siemenluku,	
suurempi tai yhtä suuri kuin,	237	RandSeed	157
suurin yhteinen jakaja, gcd( )	83	varianssi, variance( )	215
syöte, Input	96	yhden muuttujan tilastot,	
		OneVar	138
		tilat	
		asetus, setMode( )	176
		TInterval, t-luottamusväli	203
		TInterval_2Samp, kahdenotoksen t-	
		luottamusvälin	204
		tmpCnv()	205
		todennäköisyystiheys, normPdf( )	134
		toinen derivaatta	
		malli	6
		tPdf( ), studentint-	
		todennäköisyystiheys	206
		trace( )	207
		transponoi, T	197
		trigonometrinen keruu, tCollect( )	201
		trigonometrinen laajennus, tExpand( )	202
		Try, virheenkäsitteilykomento	207
		tTest, t-testi	208
		tTest_2Samp, kahden otoksen t-testi	209
		tulo (P)	
		malli	6
		tulo, P( )	242
		tulo, product( )	149
		tulokset, tilastot	191
		tulosarvot, tilastot	193
		tunnus, Lbl	102
		TVM-argumentit	211
		TVM-funktioiden argumentit	211
		tvmFV( )	210
<b>T</b>			
t-testi, tTest	208		
T, transponoi	197		
tai (Boolean), tai	139		
tai, Boolean operaattori	139		
tallentaminen			
symboli, &	253-254		
talouslaskentafunktiot, tvnFV( )	210		
talouslaskentafunktiot, tvnI( )	210		
talouslaskentafunktiot, tvnN( )	210		
talouslaskentafunktiot, tvnPmt( )	210		
talouslaskentafunktiot, tvnPv( )	211		
tan <sup>-1</sup> ( ), arkustangentti	198		
tan( ), tangentti	197		
tangentLine( )	199		
tangentti, tan( )	197		
tangenttisuora, tangentLine( )	199		
tanh <sup>-1</sup> ( ), hyperbolinen areatangentti	200		
tanh( ), hyperbolinen tangentti	199		
täsmällinen, exact( )	67		
taylor( ), Taylorin polynomi	201		
Taylorin polynomi, taylor( )	201		
täyttö	264-265		
tCdf( ), studentint-			
todennäköisyysjakauma	201		
tCollect( ), trigonometrinen keruu	201		
tekijä, factor( )	72		
Teksti-komento	203		

tvml( ) .....	210	virheet ja vianmääritys	
tvMn( ) .....	210	ohita virhe, PassErr .....	142
tvMPmt( ) .....	210	poista virhe, ClrErr .....	27
tvMPV( ) .....	211	virhekoodit ja -viestit .....	289
TwoVar, kahden muuttujan tulokset	211		
tyhjä, testi .....	102		
tyhjän testi, isVoid( ) .....	102		
tyhjät elementit .....	272		
tyhjät elementit, poista .....	53		
Tyhjennä .....	260		
		<b>W</b>	
		warnCodes( ), Warning codes .....	216
		when( ), kun .....	217
		while, While .....	217
		While, while .....	217
		<b>X</b>	
		x:n suorakulmakoordinaatti, P►Rx( )	141
		x <sup>2</sup> , neliö .....	232
		XNOR .....	238
		xor, Boolean eksklusiivinen tai (or) ...	218
		<b>Y</b>	
		y:n suorakulmakoordinaatti, P►Ry( )	141
		yhden muuttujan tilastot, OneVar ..	138
		Yhtälökäyttöjärjestelmä (EOS) .....	276
		yhtälöpari (2 yhtälöä)	
		malli .....	3
		yhtälöryhmä (N yhtälöä)	
		malli .....	4
		yhteenlasku, + .....	228
		yhteenlasku, sum( ) .....	195
		yhteinen nimittäjä, comDenom( ) ...	28
		yksikkövektori, unitV( ) .....	213
		yksiköt	
		muunna .....	250
		yläraja, ceiling( ) .....	21-22, 39
		<b>Z</b>	
		zeros( ), nollikohdat .....	219
		zInterval, z-luottamusväli .....	221
		zInterval_1Prop, yhden osuuden z-	
		luottamusväli .....	222
		zInterval_2Prop, kahden osuuden z-	
		luottamusväli .....	223
		zInterval_2Samp, kahden näytteen	
		z-luottamusväli .....	223
		zTest .....	224
		zTest_1Prop, yhden osuuden z-testi	225
		zTest_2Prop, kahden osuuden z-testi	225
		zTest_2Samp, kahden otoksen z-	226
<b>U</b>			
unitV( ), yksikkövektori .....	213		
unLock, vapautta muuttuja tai			
muuttujaryhmä .....	214		
uusi			
lista, newList( ) .....	130		
matriisi, newMat( ) .....	130		
		<b>V</b>	
vähennyslasku, N .....	228		
vaihto, shift( ) .....	177		
vakio			
funktiossa solve( ) .....	186		
vakiot			
funktiossa cSolve( ) .....	43		
funktiossa cZeros( ) .....	47		
funktiossa deSolve( ) .....	53		
funktiossa solve( ) .....	188		
pikavalinnat .....	274		
varianssi, variance( ) .....	215		
varoituskoodit ja -viestit .....	289		
varPop( ) .....	214		
varSamp( ), otoksen varianssi .....	215		
varsinainen murtoluku, propFrac ...	150		
vasen, left( ) .....	103		
vastaus			
näytä e:n arvolla .....	68		
näytä kulman kosini .....	32		
näytä kulman sini .....	181		
vastaus (viimeinen), ans .....	13		
vektorit			
näyttö vektorina			
sylinderikoordinaatistos			
sa, ►Cylind .....	46		
pistetulo, dotP( ) .....	61		
ristitulo, crossP( ) .....	39		
yksikkö, unitV( ) .....	213		

testi .....

**Δ**

$\Delta$ list( ), listaerotus .....	111
$\Delta$ tmpCnv() [tmpCnv] .....	205

**X**

$\chi^2$ 2way .....	25
$\chi^2$ Cdf( ) .....	25
$\chi^2$ GOF .....	26
$\chi^2$ Pdf( ) .....	26