



TI-*nspire*[™]

**TI-Nspire[™] CAS
Sovelluksen käsikirja**

Tämä opas koskee TI-Nspire[™]-ohjelmiston versiota 4.4. Uusin versio asiakirjoista on saatavilla Internet-sivustolta education.ti.com/guides.

Tärkeitä tietoja

Ellei muuten ilmoiteta ohjelman mukaan liitetystä 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 seurauksellisista vaurioista näiden materiaalien hankinnan tai käytön aiheuttamana, ja Texas Instruments:n yksinomainen 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.

Lisenssi

Katso täydellinen lisenssi osoitteesta **C:\Program Files\TI Education\<TI-Nspire™ Product Name>\license**.

© 2006 - 2017 Texas Instruments Incorporated

Contents

Tärkeitä tietoja	2
Lausekemallit	5
Luettelo aakkosjärjestyksessä	12
A	12
B	21
C	25
D	52
E	63
F	74
G	84
I	91
L	99
M	115
N	124
O	133
P	136
Q	145
R	149
S	164
T	190
U	206
V	207
W	208
X	210
Z	211
Symbolit	220
Tyhjät elementit	248
Matemaattisten lausekkeiden syöttäminen pikavalintojen avulla	250
EOS-järjestelmän (yhtälökäyttöjärjestelmä) hierarkia	252
Virhekoodit ja viestit	254
Varoituskoodit ja -viestit	261
Huolto ja Asiakastuki	263
TI-tuotteiden huolto- ja takuutietoa	263

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 222.

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 223.

Esimerkki:

$$2^3 = 8$$

Neliöjuurimalli

ctrl **x²** painikkeet





Huomaa: Katso myös $\sqrt{}$ (neliöjuuri), sivu 234.

Esimerkki:

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

N:s juuri -malli

  painikkeet



Huomaa: Katso myös **root()**, sivu 160.

Esimerkki:

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

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

e eksponenttimalli

 painikkeet



e-kantainen eksponenttifunktio korotettuna potenssiin



Huomaa: Katso myös **e^()**, sivu 64.

Esimerkki:

$$e^1 \quad e$$

$$e^1. \quad 2.71828182846$$

Logaritmimalli

  painike



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

Huomaa: Katso myös **log()**, sivu 111.

Esimerkki:

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

Paloittain määritellyn funktion malli (2-osainen)

Katalogi > 



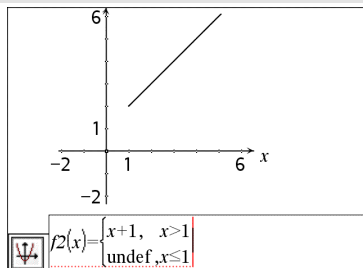
Esimerkki:

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

Huomaa: Katso myös **piecewise()**, sivu 138.

Paloittain määritellyn funktion malli (2-osainen)

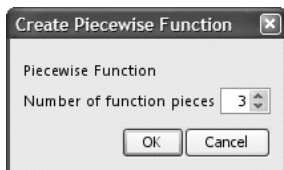
Katalogi > 



Paloittain määritellyn funktion malli (N-osainen)

Katalogi > 

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



Huomaa: Katso myös `piecewise()`, sivu 138.

Esimerkki:

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

Yhtälöparin malli

Katalogi > 



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

Huomaa: Katso myös `system()`, sivu 190.

Esimerkki:

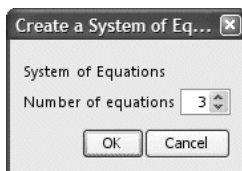
$$\text{solve} \left(\begin{cases} x+y=0 \\ x-y=5 \end{cases}, x, y \right) \quad x = \frac{5}{2} \text{ and } y = \frac{-5}{2}$$

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

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 190.

Itseisarvon malli

Katalogi > 



Huomaa: Katso myös `abs()`, sivu 12.

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:

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

Matriisimalli (2 x 1)

Katalogi > 

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

Esimerkki:

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

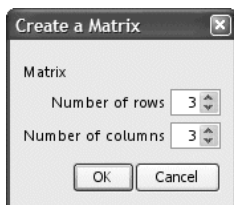
Matriisimalli (m x n)

Katalogi > 

Malli tulee näkyviin määritettyäsi rivien ja sarakkeiden lukumäärän syöttöruutuun.

Esimerkki:

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



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

Summan malli (Σ)

Katalogi > 

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

Esimerkki:

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

Huomaa: Katso myös $\Sigma()$ (**sumSeq**), sivu 235.

Tulon malli (II)

Katalogi > 

$$\frac{d}{dx} (x^n) = nx^{n-1}$$

Huomaa: Katso myös **II()** (**prodSeq**), sivu 234.

Esimerkki:

$$\frac{d}{dx} (x^5) \Big|_{x=1} = 5 \cdot 1^{4} = 5$$

Ensimmäisen derivaatan malli

Katalogi > 

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

Ensimmäisen derivaatan mallia voi käyttää myös laskettaessa ensimmäinen derivaatta pisteessä.

Huomaa: Katso myös **d()** (**derivaatta**), sivu 231.

Esimerkki:

$$\frac{d}{dx} (x^3) = 3 \cdot x^2$$
$$\frac{d}{dx} (x^3) \Big|_{x=3} = 27$$

Toisen derivaatan malli

Katalogi > 

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

Toisen derivaatan mallia voi käyttää myös laskettaessa toinen derivaatta pisteessä.

Huomaa: Katso myös **d()** (**derivaatta**), sivu 231.

Esimerkki:

$$\frac{d^2}{dx^2} (x^3) = 6 \cdot x$$
$$\frac{d^2}{dx^2} (x^3) \Big|_{x=3} = 18$$

N:nnen derivaatan malli

Katalogi > 

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

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

Huomaa: Katso myös **d()** (**derivaatta**), sivu 231.

Esimerkki:

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

Määrätyn integraalin malli

Katalogi > 

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

Huomaa: Katso myös $\int()$ integraali(), sivu 220.

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 $\int()$ integral(), sivu 220.

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 101.

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 220). Ellei toisin ole mainittu, kaikki tämän kappaleen esimerkit on suoritettu laskimen oletustilassa, eikä mitään muuttujia ole määritetty.

A

abs()

Katalogi > 

abs(Laus I)⇒*lauseke*

$\left\{ \frac{\pi}{2}, \frac{\pi}{3} \right\}$	$\left\{ \frac{\pi}{2}, \frac{\pi}{3} \right\}$
$2-3 \cdot i$	$\sqrt{13}$
$ z $	$ z $
$ x+y \cdot i $	$\sqrt{x^2+y^2}$

abs(Lista I)⇒*lista*

abs(Matriisi I)⇒*matriisi*

Laskee argumentin itseisarvon.

Huomaa: Katso myös **Itseisarvon malli**, sivu 8.

Jos argumentti on kompleksiluku, määrittää luvun moduulin.

Huomaa: Kaikkia määrittämättömiä muuttujia käsitellään reaali muuttujina.

amortTbl()

Katalogi > 

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

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

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

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 204.

- 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.

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 235, sekä $\text{bal}()$, sivu 21.

and

BooleanLaus1 and *BooleanLaus2* \Rightarrow *BooleanLausekeBooleanLista1*
and *BooleanLista2* \Rightarrow *BooleanlistaBooleanMatriisi1*
and *BooleanMatriisi2* \Rightarrow *Boolean matriisi*

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

Kokonaisluku1

and *Kokonaisluku2* \Rightarrow *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 näkyy kantelukutilan 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 kantelukutilassa:

$$0h7AC36 \text{ and } 0h3D5F \qquad \qquad \qquad 0h2C16$$

Tärkeää: Nolla, ei O-kirjain.

Binaarisessa kantelukutilassa:

$$0b100101 \text{ and } 0b100 \qquad \qquad \qquad 0b100$$

Desimaalisessa kantelukutilassa:

$$37 \text{ and } 0b100 \qquad \qquad \qquad 4$$

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

Huomaa: Binaarisessa syötteessä voi olla korkeintaan 64 numeroa (etuliitettä 0b ei lasketa). Heksadesimaalisessa syötteessä voi olla korkeintaan 16 numeroa.

angle()

angle(Laus I) ⇒ *lauseke*

Laskee argumentin kulman tulkiten argumentin kompleksiluvuksi.

Huomaa: Kaikkia määrittämättömiä muuttujia käsitellään reaali muuttujina.

angle(Lista I) ⇒ *lista*

angle(Matriisi I) ⇒ *matriisi*

Laskee listan tai matriisin *Listai*:n tai *Matriisi I*:n elementtien kulmista tulkiten jokaisen elementin kompleksiluvuksi, joka edustaa kaksiulotteista suorakulmakoordinaattipistettä.

Astekulmatilassa:

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

Graadikulmatilassa:

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

Radiaanikulmatilassa:

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

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

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

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

ANOVA

ANOVA *Listai, Listai2[, Listai3, ..., Listai20]*
[, *Lippu*]

Suorittaa yksisuuntaisen varianssianalyysin 2-20 perusjoukon keskiarvon vertailua varten. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 185.)

Lippu=0 datalle, *Lippu*=1 tilastoille

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

ANOVA2way

ANOVA2way *List1,Lista2*
 [,*List3*,...,*List10*][*TasoRivi*]

Laskee kaksisuuntaisen varianssianalyysin 2-10 perusjoukon keskiarvojen vertaamiseksi. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 185.)

TasoRivi=0 lohkolle

TasoRivi=2,3,...,*Pit*-1, kahdelle tekijälle, jossa *Pit*=pituus(*List1*)=pituus(*List2*) = ... = pituus(*List10*) ja *Pit* / *TasoRivi* ∈ {2,3,...}

Tu1okset: Lohkokuoto

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.PVallInteract	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(Laus1) ⇒ 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}

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

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

approx(Lista1) ⇒ lista

approx(Matriisi1) ⇒ matriisi

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

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

approx()

Katalogi > 

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

▶approxFraction()

Katalogi > 

Laus ▶**approxFraction**([*Tol*])⇒*lauseke*

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

Lista ▶**approxFraction**([*Tol*])⇒*lista*

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

Matriisi ▶**approxFraction**([*Tol*])⇒*matriisi*

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

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

$$\{\pi, 1.5\} \text{ ▶approxFraction}(5.E-14) \\ \left\{ \frac{5419351}{1725033}, \frac{3}{2} \right\}$$

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

approxRational()

Katalogi > 

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

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

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

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

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 37.

arccosh()

Katso $\cosh^{-1}()$, sivu 38.

arccot()

Katso $\cot^{-1}()$, sivu 39.

arccoth()**Katso $\text{coth}^{-1}()$, sivu 40.****arccsc()****Katso $\text{csc}^{-1}()$, sivu 43.****arccsch()****Katso $\text{csch}^{-1}()$, sivu 43.****arcLen()****Katalogi > ****arcLen(Laus1,Muutt,Alku,Loppu) ⇒ lauseke**Laskee *Laus1*:n kaaren pituuden alusta *Alku* loppuun *Loppu* muuttujan *Muutt* suhteen.

Kaaren pituus lasketaan kokonaislukuna käyttäen oletuksena funktioiden määrittystä.

arcLen(Lista1,Muutt,Alku,Loppu) ⇒ listaLaskee listan jokaisen *Listal*:n elementin kaaren pituuden alusta *Alku* loppuun *Loppu* muuttujan *Muutt* suhteen.

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

 $\text{arcLen}(\{\sin(x),\cos(x)\},x,0,\pi)$

 $\{3.8202,3.8202\}$ **arcsec()****Katso $\text{sec}^{-1}()$, sivu 164.****arcsech()****Katso $\text{sech}^{-1}()$, sivu 165.****arcsin()****Katso $\text{sin}^{-1}()$, sivu 176.**

arcsinh()Katso $\sinh^{-1}()$, sivu 177.**arctan()**Katso $\tan^{-1}()$, sivu 191.**arctanh()**Katso $\tanh^{-1}()$, sivu 193.**augment()**Katalogi > **augment(Lista1, Lista2)⇒lista** $\text{augment}(\{1,-3,2\},\{5,4\}) \quad \{1,-3,2,5,4\}$ Luo uuden listan, joka on *Lista2* liitettynä *Lista1*:n loppuun.**augment(Matriisi1, Matriisi2)⇒matriisi**

1 2	→ m1	1 2
3 4		3 4

Luo 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.

5	→ m2	5
6		6

augment(m1,m2)		1 2 5
		3 4 6

avgRC()Katalogi > **avgRC(Laus1, Muutt [=Arvo] [, Askel])**
⇒lauseke $\text{avgRC}(f(x),x,h) \quad \frac{f(x+h)-f(x)}{h}$ **avgRC(Laus1, Muutt [=Arvo] [, Lista1])**
⇒lista $\text{avgRC}(\sin(x),x,h)|x=2 \quad \frac{\sin(h+2)-\sin(2)}{h}$ **avgRC(Lista1, Muutt [=Arvo] [, Askel])**
⇒lista $\text{avgRC}(x^2-x+2,x) \quad 2 \cdot (x-0.4995)$ $\text{avgRC}(x^2-x+2,x,0.1) \quad 2 \cdot (x-0.45)$ **avgRC(Matriisi1, Muutt [=Arvo] [, Askel])**
⇒matriisi $\text{avgRC}(x^2-x+2,x,3) \quad 2 \cdot (x+1)$

Laskee erotusosamäärän eteenpäin (keskimääräisen muutosnopeuden).

Laus1 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*])⇒*arvo*

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

bal(*NPmt*,*amortTable*)⇒*arvo*

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

Lyhennysfunktio, joka laskee määritetyn maksuerän jälkeen jäljellä olevan velkasaldon.

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

N, *I*, *PV*, *Pmt*, *FV*, *PpY*, *CpY* ja *PmtAt* on kuvattu TVM-argumenttien taulukossa, sivu 204.

bal(4,*tbl*) 1674.27

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 204.

- 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.

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 12.

Huomaa: Katso myös $\Sigma\text{Int}()$ ja $\Sigma\text{Prn}()$, sivu 235.

►Base2 (►Kantaluku2)

Kokonaisluku1 ►Base2⇒*kokonaisluku*

256►Base2	0b100000000
0h1F►Base2	0b11111

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

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

-2⁶³ näkyy muodossa
0h8000000000000000 heksadesimaalisessa kantalukutilassa
0b100...000 (63 zeros) binaarisessa kantalukutilassa

Jos syötät desimaalilikonaisuuden, 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 kantalukutilassa
 0b100...000 (63 zeros) binaarisessa kantalukutilassa

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

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

►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 *Kokonaisluku*:ä käsitellään desimaalilukuna. Vastaus näkyy desimaalilukuna kantalukutilasta riippumatta.

Kokonaisluku ►Base16⇒*kokonaisluku*

256►Base16

0h100

Huomaa: Voit syöttää tämän operaattorin tietokoneen näppäimistöä kirjoittamalla @►Base16.

0b111100001111►Base16

0hFOF

Muuttaa *Kokonaisluku*: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 *Kokonaisluku*: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 desimaalikokonaisluvun, joka on etumerkillisen, 64 bitin binaarimuodon lukualueen ulkopuolella, laskin käyttää symmetristä modulo-operaatiota, jotta arvo saadaan oikealle alueelle. Lisätietoja, katso ►Base2, sivu 22.

binomCdf(n,p)⇒*lista*

binomCdf()Katalogi > 

binomCdf($n,p,alaraja,ylaraja$) \Rightarrow luku, jos *alaraja* ja *ylaraja* ovat lukuja, *lista*, jos *alaraja* ja *ylaraja* ovat listoja

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

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

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

binomPdf()Katalogi > 

binomPdf(n,p) \Rightarrow *lista*

binomPdf($n,p,XVal$) \Rightarrow 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()**Katalogi > 

ceiling(*Laus l*) \Rightarrow *kokonaisluku*

$\text{ceiling}(.456)$	1.
------------------------	----

Laskee lähimmän kokonaisluvun, joka on \geq argumentti.

Argumentti voi olla reaali- tai kompleksiluku.

Huomaa: Katso myös **floor()**.

ceiling(*Lista l*) \Rightarrow *lista*

ceiling(*Matriisi l*) \Rightarrow *matriisi*

$\text{ceiling}(\{-3.1, 1, 2.5\})$	$\{-3., 1, 3.\}$
------------------------------------	------------------

$\text{ceiling}\left(\begin{bmatrix} 0 & -3.2 \cdot i \\ 1.3 & 4 \end{bmatrix}\right)$	$\begin{bmatrix} 0 & -3 \cdot i \\ 2. & 4 \end{bmatrix}$
--	--

Laskee listan tai matriisin jokaisen elementin ylärajasta.

centralDiff()Katalogi > **centralDiff(LausI, Muutt [=Arvo][,Askel])**
⇒ lauseke

$$\text{centralDiff}(\cos(x), x, h)$$

$$\frac{-\cos(x-h) - \cos(x+h)}{2 \cdot h}$$

centralDiff(LausI, Muutt [,Askel])
| Muutt=Arvo ⇒ lauseke

$$\lim_{h \rightarrow 0} (\text{centralDiff}(\cos(x), x, h))$$

$$-\sin(x)$$

centralDiff(LausI, Muutt [=Arvo][,Lista])
⇒ lista

$$\text{centralDiff}(x^3, x, 0.01)$$

$$3 \cdot (x^2 + 0.000033)$$

centralDiff(ListaI, Muutt [=Arvo][,Askel])
⇒ lista

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

$$-1.$$

centralDiff(MatriisiI, Muutt [=Arvo][,Askel]) ⇒ matriisi

$$\text{centralDiff}(x^2, x, \{0.01, 0.1\})$$

$$\{2 \cdot x, 2 \cdot x\}$$

Laskee numeerisen derivaatan käyttäen keskeiserotusomääriä kaavaa.

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

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

ListaI:tä tai *MatriisiI*:tä käytettäessä operaatio mapataan listan arvojen tai matriisin elementtien suhteen.

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

cFactor()Katalogi > **cFactor(LausI[,Muutt])** ⇒ lauseke**cFactor(ListaI[,Muutt])** ⇒ lista**cFactor(MatriisiI[,Muutt])** ⇒ matriisi

$$\text{cFactor}(a^3 \cdot x^2 + a \cdot x^2 + a^3 + a \cdot x)$$

$$a \cdot (a^2 + 1) \cdot (x - i) \cdot (x + i)$$

$$\text{cFactor}\left(x^2 + \frac{4}{9}\right)$$

$$\frac{(3 \cdot x - 2 \cdot i) \cdot (3 \cdot x + 2 \cdot i)}{9}$$

$$\text{cFactor}(x^2 + 3)$$

$$x^2 + 3$$

$$\text{cFactor}(x^2 + a)$$

$$x^2 + a$$

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(*Laus I, Muutt*) jakaa *Laus I*:n tekijöihin muuttujan *Muutt* suhteen.

Laus I: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.

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.

Huomaa: Katso myös **factor()**.

$$\begin{aligned} \text{cFactor}(a^3 \cdot x^2 + a \cdot x^2 + a^3 + a \cdot x) &= a \cdot (a^2 + 1) \cdot (x - i) \cdot (x + i) \\ \text{cFactor}(x^2 + 3x) &= (x + \sqrt{3} \cdot i) \cdot (x - \sqrt{3} \cdot i) \\ \text{cFactor}(x^2 + ax) &= (x + \sqrt{a} \cdot i) \cdot (x + \sqrt{a} \cdot i) \end{aligned}$$

$$\begin{aligned} \text{cFactor}(x^5 + 4 \cdot x^4 + 5 \cdot x^3 - 6 \cdot x - 3) &= x^5 + 4 \cdot x^4 + 5 \cdot x^3 - 6 \cdot x - 3 \\ \text{cFactor}(x^5 + 4 \cdot x^4 + 5 \cdot x^3 - 6 \cdot x - 3, x) &= (x - 0.964673) \cdot (x + 0.611649) \cdot (x + 2.12543) \cdot (x + \dots) \end{aligned}$$

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

char(*Kokonaisluku*) ⇒ *merkki*

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

$$\begin{aligned} \text{char}(38) &= "&" \\ \text{char}(65) &= "A" \end{aligned}$$

charPoly(neliömatriisi,Muutt) \Rightarrow polynomilauseke**charPoly(neliömatriisi,Laus)** \Rightarrow polynomilauseke**charPoly(neliömatriisi1,Matriisi2)** \Rightarrow polynomilauseke

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

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

jossa I tarkoittaa identtistä matriisiä $n \times n$.

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

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

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

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

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

 χ^2 way **χ^2 way ObsMatriisi****chi22way ObsMatriisi**

Laskee χ^2 -testin tarkasteltavan matriisin *ObsMatriisi* sisältämän kaksisuuntaisen lukemataulukon arvojen välisestä assosiaatiosta. Tulosten yhteenvedo tallentuu *stat.results*-muuttujaan. (Katso sivu 185.)

Lisätietoja matriisissa olevien tyhjen elementtien vaikutuksesta, katso Tyhjät elementtisivulla sivu 248.

Tulosmuuttuja	Kuvaus
stat. χ^2	Khin neliö -tilasto: summa (tarkasteltava - odotettu) ² /odotettu
stat.PVal	Alin merkitsevyytaso, jolla nollassuhteeksi voidaan hylätä
stat.df	Khin neliö -tilastojen vapausasteet
stat.ExpMat	Odotetun elementtilukemataulukon matriisi, oletuksena nollassuhteeksi
stat.CompMat	Elementtien Khin neliö -tilastokontributioiden matriisi

χ^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 χ^2 -jakauman todennäköisyyden *alarajan* ja *ylärajan* väliltä määritetyille vapausasteelle *df*.

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

Lisätietoja listassa olevien tyhjen elementtien vaikutuksesta, katso Tyhjät elementitsivulla sivu 248.

χ^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 yhteenvedo tallentuu *stat.results*-muuttujaan. (Katso sivu 185.)

Lisätietoja listassa olevien tyhjen elementtien vaikutuksesta, katso Tyhjät elementitsivulla sivu 248.

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

χ^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 χ^2 -jakauman todennäköisyysfunktio (pdf) määritetyllä *XArvo* arvolla määritetylle vapausasteelle *df*.

Lisätietoja listassa olevien tyhjien elementtien vaikutuksesta, katso Tyhjät elementitsivulla sivu 248.

clearAZ

clearAZ

Poistaa kaikki yksikirjaimiset muuttujat nykyiseltä tehtäväalueelta.

Jos yksi tai useampia muuttujia on lukittu, tämä komento aiheuttaa virheilmoituksen ja poistaa vain lukitsemattomat muuttujat. Katso **unLock**, sivu 207.

$5 \rightarrow b$	5
<i>b</i>	5
ClearAZ	Done
<i>b</i>	<i>b</i>

ClrErr

ClrErr

Poistaa virhetilan ja nollaa 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 137, ja **Try**, sivu 200.

Esimerkki **ClrErr**-komennosta, katso esimerkki 2 **Try**-komennon kohdalla, sivu 200.

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

colAugment()Katalogi > 

colAugment(Matriisi1, Matriisi2)
⇒*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.

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

colDim()Katalogi > 

colDim(Matriisi)⇒*lauseke*

Laskee *Matriisin* sisältämien sarakkeiden lukumäärän.

Huomaa: Katso myös **rowDim()**.

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

colNorm()Katalogi > 

colNorm(Matriisi)⇒*lauseke*

Laskee maksimiarvon *Matriisin* sarakkeissa olevien elementtien itseisarvojen summista.

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

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

comDenom(LausI[,Muutt]) ⇒ lauseke

comDenom(ListaI[,Muutt]) ⇒ lista

comDenom(MatriisiI[,Muutt]) ⇒ matriisi

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

comDenom(LausI,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 *LausI*:ssä, **comDenom(LausI,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 nimittäjä 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 rutiinomaisesti vaihtoehtona funktioille **comDenom()** ja **factor()**.

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

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

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

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

Katalogi > **completeSquare**(ExprOrEqn, Var) \Rightarrow lauseke tai yhtälö**completeSquare**(ExprOrEqn, Var^Power) \Rightarrow lauseke tai yhtälö**completeSquare**(ExprOrEqn, Var1, Var2[...]) \Rightarrow lauseke tai yhtälö**completeSquare**(ExprOrEqn, {Var1, Var2[...]) \Rightarrow 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 vakiomuodossa 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.

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

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

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

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

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

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

conj()

Katalogi > **conj**(Lausl) \Rightarrow lauseke**conj**(Listal) \Rightarrow lista**conj**(Matriisil) \Rightarrow matriisi

Laskee argumentin liittokompleksiluvun.

Huomaa: Kaikkia määrittämättömiä muuttujia käsitellään reaali muuttujina.

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

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

$\text{conj}(z)$	\bar{z}
------------------	-----------

$\text{conj}(x+i \cdot y)$	$x-y \cdot i$
----------------------------	---------------

constructMat

(

*Laus**,Muutt1,Muutt2,numRivit,numSarakkeet)* \Rightarrow *matriisi*

Laskee matriisin 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** - *numSarakkeet*.

constructMat($\frac{1}{i+j}, i, j, 3, 4$)	$\frac{1}{2}$	$\frac{1}{3}$	$\frac{1}{4}$	$\frac{1}{5}$
	$\frac{1}{3}$	$\frac{1}{4}$	$\frac{1}{5}$	$\frac{1}{6}$
	$\frac{1}{4}$	$\frac{1}{5}$	$\frac{1}{6}$	$\frac{1}{7}$
	$\frac{1}{4}$	$\frac{1}{5}$	$\frac{1}{6}$	$\frac{1}{7}$

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













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

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.

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

<i>aa.a:=45</i>	45																
<i>aa.b:=6.78</i>	6.78																
CopyVar <i>aa.,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.mn* 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.

$\cos(Laus\ I) \Rightarrow lauseke$

$\cos(Lista\ I) \Rightarrow lista$

$\cos(Laus\ I)$ määrittää argumentin kosinin lausekkeena.

$\cos(Lista\ I)$ määrittää listan kaikkien $Lista\ I$:n sisältämien elementtien kosineista.

Huomaa: Argumentti tulkitaan aste-, graadi- tai radiaanikulmaksi käytössä olevan kulmatila-asetuksen mukaisesti. Voit ohittaa kulmatilan väliaikaisesti painikkeilla $^\circ$, G tai r .

Astekulmatilassa:

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

Graadikulmatilassa:

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

Radiaanikulmatilassa:

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

$\cos(neliomatriisi\ I) \Rightarrow neliomatriisi$

Laskee $neliomatriisi\ I$:n matriisikosinin. Tämä ei ole sama kuin kunkin elementin kosinin laskeminen.

Kun skaalarista funktiota $f(A)$ käytetään $neliomatriisi\ I$:een (A), tulos lasketaan algoritmilla:

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 (λ_i) ja ominaisvektorit (V_i).

$neliomatriisi\ I$: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⁻¹()

cos⁻¹(LausI) ⇒ lauseke

Astekulmatilassa:

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

cos⁻¹(Listal) ⇒ lista

Graadikulmatilassa:

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

cos⁻¹(LausI) määrittää lausekkeena kulman, jonka kosini on *LausI*.

cos⁻¹(Listal) laskee listan *Listal*:n jokaisen elementin käänteiskosineista.

Radiaanikulmatilassa:

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

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

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

cos⁻¹(neliömatriisiI) ⇒ neliömatriisi

Radiaanikulmatilassa ja

suorakulmakompleksimuodossa:

$$\cos^{-1} \begin{pmatrix} 1 & 5 & 3 \\ 4 & 2 & 1 \\ 6 & -2 & 1 \end{pmatrix} = \begin{bmatrix} 1.73485+0.064606 \cdot i & -1.49086+2.10514 \\ -0.725533+1.51594 \cdot i & 0.623491+0.77836 \cdot i \\ -2.08316+2.63205 \cdot i & 1.79018-1.27182 \cdot i \end{bmatrix}$$

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

$\cos^{-1}()$

trig painike

neliömatriisi1:n on oltava diagonalisoitavissa. Vastaus sisältää aina liukulukuja.

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

$\cosh()$

Katalogi >

$\cosh(Laus\ I) \Rightarrow$ lauseke

Astekulmatilassa:

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

$\cosh(Lista\ I) \Rightarrow$ lista

$\cosh(Laus\ I)$ määrittää argumentin hyperbolisen kosinin lausekkeena.

$\cosh(Lista\ I)$ määrittää listan *Listal*:n kunkin elementin hyperbolisista kosineista.

$\cosh(neliomatriisi\ I) \Rightarrow$ neliömatriisi

Radiaanikulmatilassa:

$$\cosh\begin{pmatrix} 1 & 5 & 3 \\ 4 & 2 & 1 \\ 6 & -2 & 1 \end{pmatrix} \begin{bmatrix} 421.255 & 253.909 & 216.905 \\ 327.635 & 255.301 & 202.958 \\ 226.297 & 216.623 & 167.628 \end{bmatrix}$$

Laskee *neliömatriisi1*:n matriisin hyperbolisen kosinin. Tämä ei ole sama kuin kunkin elementin hyperbolisen kosinin laskeminen. Laskentamenetelmä on kuvattu kohdassa **cos()**.

neliömatriisi1:n on oltava diagonalisoitavissa. Vastaus sisältää aina liukulukuja.

$\cosh^{-1}()$

Katalogi >

$\cosh^{-1}(Laus\ I) \Rightarrow$ lauseke

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

$\cosh^{-1}(Lista\ I) \Rightarrow$ lista

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

$\cosh^{-1}(Laus\ I)$ määrittää argumentin käänteisen hyperbolisen kosinin lausekkeena.

$\cosh^{-1}(Lista\ I)$ määrittää listan *Listal*:n kunkin elementin käänteisistä hyperbolisista kosineista.

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

cosh⁻¹()**cosh⁻¹(*neliömatriisi*)** ⇒ *neliömatriisi*

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

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

Radiaanikulmatilassa ja suorakulmakompleksimuodossa:

$$\cosh^{-1} \begin{pmatrix} 1 & 5 & 3 \\ 4 & 2 & 1 \\ 6 & -2 & 1 \end{pmatrix} \begin{bmatrix} 2.52503+1.73485 \cdot i & -0.009241-1.4908i \\ 0.486969-0.725533 \cdot i & 1.66262+0.623491i \\ -0.322354-2.08316 \cdot i & 1.26707+1.79018i \end{bmatrix}$$

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

cot()**cot(*Laus*l)** ⇒ *lauseke*

Astekulmatilassa:

$$\cot(45) = 1$$

cot(*List*l) ⇒ *lista*

Graadikulmatilassa:

$$\cot(50) = 1$$

Laskee *Laus* *l*:n kotangentin tai määrittää listan *List* *l*:n kaikkien elementtien kotangenteista.

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

Radiaanikulmatilassa:

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

cot⁻¹()**cot⁻¹(*Laus*l)** ⇒ *lauseke*

Astekulmatilassa:

$$\cot^{-1}(1) = 45$$

cot⁻¹(*List*l) ⇒ *lista*

Graadikulmatilassa:

$$\cot^{-1}(1) = 50$$

Laskee kulman, jonka kotangentti on *Laus* *l*, tai määrittää listan, joka sisältää *List* *l*:n kunkin elementin käänteiskotangentit.

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

Radiaanikulmatilassa:

cot⁻¹()

trig painike

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

$\text{cot}^{-1}(1)$	$\frac{\pi}{4}$
----------------------	-----------------

coth()

Katalogi >

coth(Laus I)⇒*lauseke*

$\text{coth}(1.2)$	1.19954
--------------------	---------

coth(Lista1)⇒*lista*

$\text{coth}(\{1,3.2\})$	$\left\{ \frac{1}{\tanh(1)}, 1.00333 \right\}$
--------------------------	--

Laskee *Laus I*:n hyperbolisen kotangentin tai määrittää listan *Listal*:n kaikkien elementtien hyperbolisista kotangenteista.

coth⁻¹()

Katalogi >

coth⁻¹(Laus I)⇒*lauseke*

$\text{coth}^{-1}(3.5)$	0.293893
-------------------------	----------

coth⁻¹(Lista1)⇒*lista*

$\text{coth}^{-1}(\{-2.2,1.6\})$	$\left\{ \frac{-\ln(3)}{2}, 0.518046, \frac{\ln\left(\frac{7}{5}\right)}{2} \right\}$
----------------------------------	---

Laskee *Laus I*:n käänteisen hyperbolisen kotangentin tai määrittää listan, joka sisältää *Listal*:n kaikkien elementtien käänteiset hyperboliset kotangentit.

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

count()

Katalogi >

count(Arvo1taiLista1 [,Arvo2taiLista2 [,...]])⇒*arvo*

$\text{count}(2,4,6)$	3
-----------------------	---

Laskee elementtien kokonaismäärän argumenteille, jotka sievennyvät numeroarvoiksi.

$\text{count}(\{2,4,6\})$	3
---------------------------	---

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

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

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

	2
--	---

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

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 248.

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 sievenyvät 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 248.

Huomaa: Katso myös **sumlf()**, sivu 189, ja **frequency()**, sivu 82.

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⁻²,x⁻¹,1,x,x²},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.

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

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

cPolyRoots()

Katalogi >

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 142.

crossP()

Katalogi >

crossP(*List1*,*List2*)⇒*lista*

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

Määrittää listan *List1*:n ja *List2*:n ristitulosta.

$$\text{crossP}(\{0,1,2,2,-5\},\{1,-0,5,0\}) \\ \{-2,5,-5,-,2,25\}$$

List1:n ja *List2*:n on oltava samankokoiset, ja koon on oltava joko 2 tai 3.

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

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

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() **painike****csc**(*Laus1*)⇒*lauseke*

Astekulmatilassa:

csc(*List1*)⇒*lista*

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

Laskee *Laus1*:n kosekantin tai määrittää listan, joka sisältää *List1*:n kaikkien elementtien kosekantit.

Graadikulmatilassa:

csc()**trig** painike

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

Radiaanikulmatilassa:

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

csc⁻¹()**trig** painike**csc⁻¹(Lausl)** ⇒ lauseke

Astekulmatilassa:

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

csc⁻¹(Listal) ⇒ lista

Graadikulmatilassa:

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

Laskee kulman, jonka kosekanti on *Lausl*, tai määrittää listan, joka sisältää *Listal*:n kunkin elementin käänteiskosekantit.

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

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

Radiaanikulmatilassa:

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

csch()

Katalogi >

csch(Lausl) ⇒ lauseke

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

csch(Listal) ⇒ lista

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

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

csch⁻¹()

Katalogi >

csch⁻¹(Lausl) ⇒ lauseke

$$\text{csch}^{-1}(1) \quad \sinh^{-1}(1)$$

csch⁻¹(Listal) ⇒ lista

$$\text{csch}^{-1}\left(\left\{1, 2, 1, 3\right\}\right) \quad \left\{\sinh^{-1}(1), 0.459815, \sinh^{-1}\left(\frac{1}{3}\right)\right\}$$

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

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

cSolve()

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

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

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

$\text{cSolve}(x^3 = -1, x)$	
$x = \frac{1}{2} + \frac{\sqrt{3}}{2} \cdot i$ or $x = \frac{1}{2} - \frac{\sqrt{3}}{2} \cdot i$ or $x = -1$	
$\text{solve}(x^3 = -1, x)$	$x = -1$

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.

Vaikka kaikkia määrittämättömiä muuttujia, joiden lopussa ei ole alaviivaa (), käsiteltäisiin ikään kuin ne olisivat reaalisia, **cSolve()** pystyy laskemaan polynomiyhtälöiden kompleksilukuratkaisuja.

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



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

cSolve()-funktion ratkaisu aloitetaan eksakteilla symbolisilla menetelmillä.

cSolve() käyttää tarvittaessa myös iteratiivista likimääräistä kompleksipolynomin tekijöihin jakamista.

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

Huomaa: Jos *Yhtälö* on ei-polynominen funktioilla, kuten **abs()**, **angle()**, **conj()**, **real()** tai **imag()**, sijoita alaviiva (paina

 ) muuttujan *Muutt*:n loppuun.

Oletusarvoisesti muuttujaa käsitellään reaaliarvona.

Jos käytät merkintää *muutt_*, muuttujaa käsitellään kompleksilukuna.

Merkintää *muutt_* tulee käyttää myös kaikissa muissa *Yhtälön* muuttujissa, jotka voivat sisältää ei-reaaliarvoja. Muussa tapauksessa tulokset voivat olla väärin.

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 $x=3+i$.




Desimaalien näyttötilassa Kiinteä 2:

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

$$x\cdot\left(x^4+4\cdot x^3+5\cdot x^2-6\right)=3$$

cSolve(*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 .

cSolve(**conj**(*z_*)=1+i,*z_*)

$$z_ = 1 - i$$

Jos kaikki yhtälöt ovat polynomeja, ja jos ET määrittää 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.

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 kokonaislukuilite 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.

Huomaa: Seuraavissa esimerkeissä käytetään alaviivaa (paina **ctrl** ), jotta muuttujia käsitellään kompleksiarvoina.

$$\text{cSolve}\left(u_{-} \cdot v_{-} - u_{-} = v_{-} \text{ and } v_{-}^2 = u_{-}, \{u_{-}, v_{-}\}\right)$$

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

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

$$\text{cSolve}\left(u_{-} \cdot v_{-} - u_{-} = c_{-} \cdot v_{-} \text{ and } v_{-}^2 = u_{-}, \{u_{-}, v_{-}\}\right)$$

$$u_{-} = \frac{-(\sqrt{1-4 \cdot c_{-} + 1})^2}{4} \text{ and } v_{-} = \frac{\sqrt{1-4 \cdot c_{-} + 1}}{2} \text{ or } u_{-} = \frac{-(\sqrt{1-4 \cdot c_{-} + 1})^2}{4} \text{ and } v_{-} = -\frac{\sqrt{1-4 \cdot c_{-} + 1}}{2}$$

$$\text{cSolve}\left(u_{-} \cdot v_{-} - u_{-} = v_{-} \text{ and } v_{-}^2 = u_{-}, \{u_{-}, v_{-}, w_{-}\}\right)$$

$$u_{-} = \frac{1}{2} + \frac{\sqrt{3}}{2} \cdot i \text{ and } v_{-} = \frac{1}{2} - \frac{\sqrt{3}}{2} \cdot i \text{ and } w_{-} = c8 \text{ or } u_{-} = \frac{1}{2} - \frac{\sqrt{3}}{2} \cdot i \text{ and } v_{-} = \frac{1}{2} + \frac{\sqrt{3}}{2} \cdot i \text{ and } w_{-} = c8$$

$$\text{cSolve}\left(u_{-} + v_{-} = e^{w_{-}} \text{ and } u_{-} \cdot v_{-} = i, \{u_{-}, v_{-}\}\right)$$

$$u_{-} = \frac{e^{w_{-} + i}}{2} \text{ and } v_{-} = \frac{e^{w_{-} - i}}{2}$$

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.

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

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

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

$$w=0.149606+4.8919i \text{ and } z=1.58805+1i$$

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

CubicReg

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

Laskee 3. asteen polynomiregressioy = $a \cdot x^3 + b \cdot x^2 + c \cdot x + d$ listaista X ja Y frekvenssillä $Frekv$. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 185.)

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

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 .

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 elementtivuorolla sivu 248.

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 ²	Määrittelykerroin.
stat.Resid	Regressioyhtälön jäännökset
stat.XReg	Muokatus <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	Muokatus <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()

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

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

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

Tyhjä elementti listassa *Listal* tai matriisissa *Matriisi1* tuottaa tyhjän elementin tuloksena olevaan listaan tai matriisiin. Lisätietoja tyhjistä elementeistä, katso sivu 248.

Cycle

Cycle

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

Cycle

Katalogi > 

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

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.

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

$g()$ 5000

Cylind

Katalogi > 

Vektori \blacktriangleright Cylind

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

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

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

$[2 \ 2 \ 3]$ \blacktriangleright Cylind $\left[2 \cdot \sqrt{2} \ \angle \frac{\pi}{4} \ 3 \right]$

cZeros()

Katalogi > 

cZeros(Laus, Muutt) \Rightarrow lista

Määrittää listan muuttujan *Muutt* mahdollisista reaali- ja ei-reaaliarvoista, joiden tuloksena *Laus*=0. **cZeros()** suorittaa tämän seuraavasti:



exp \blacktriangleright list(**cSolve(Laus=0, Muutt), Muutt**).
Muilta osin **cZeros()** on samanlainen kuin **zeros()**.

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

Desimaalien näyttötilassa Kiinteä 3:

$cZeros(x^5+4x^4+5x^3-6x-3, x)$
 $\{-1.1138+1.07314 \cdot i, -1.1138-1.07314 \cdot i, -2. \blacktriangleright$

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

Huomaa: Jos *Laus* on ei-polynomiyhtälö funktioilla, kuten **abs()**, **angle()**, **conj()**, **real()** tai **imag()**, lisää alaviiva (paina  ) muuttujan *Muutt* loppuun. Oletusarvoisesti muuttujaa käsitellään reaaliarvona. Jos käytät merkintää *muutt_*, muuttujaa käsitellään kompleksiarvona.

Merkintää *muutt_* on käytettävä kaikille muille *Laus*:n muuttujille, jotka voivat sisältää ei-reaaliarvoja. Muussa tapauksessa tulokset voivat olla väärin.

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 nollakohtat.

Kompleksiset nollakohtat 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*].

$$cZeros(\text{conj}(z_-)-1-i, z_-) \quad \{1-i\}$$

Huomaa: Seuraavissa esimerkeissä käytetään alaviivaa *_* (paina  ), jotta muuttujia käsitellään kompleksiarvoina.

$$cZeros\left(\left\{u_- \cdot v_- - u_- \cdot v_-^2 + u_-, \{u_-, v_-\}\right\}\right)$$

0	0
$\frac{1}{2} - \frac{\sqrt{3}}{2} \cdot i$	$\frac{1}{2} + \frac{\sqrt{3}}{2} \cdot i$
$\frac{1}{2} + \frac{\sqrt{3}}{2} \cdot i$	$\frac{1}{2} - \frac{\sqrt{3}}{2} \cdot i$

Määritä rivin 2 juuri:

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

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

$$\text{cZeros}\left(\left\{u_{-}, v_{-}, u_{-}, c_{-}, v_{-}, v_{-}^2 + u_{-}\right\}, \left\{u_{-}, v_{-}\right\}\right)$$

$$\left[\begin{array}{cc} 0 & 0 \\ \frac{-\sqrt{1-4\cdot c_{-}-1}}{4} & \frac{-\sqrt{1-4\cdot c_{-}-1}}{2} \\ \frac{-\sqrt{1-4\cdot c_{-}+1}}{4} & \frac{\sqrt{1-4\cdot c_{-}+1}}{2} \end{array} \right]$$

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 kokonaisluku liite väliltä 1-255.

$$\text{cZeros}\left(\left\{u_{-}, v_{-}, u_{-}, v_{-}, v_{-}^2 + u_{-}\right\}, \left\{u_{-}, v_{-}, w_{-}\right\}\right)$$

$$\left[\begin{array}{cc} 0 & 0 & c4 \\ \frac{1-\sqrt{3}}{2}, i & \frac{1+\sqrt{3}}{2}, i & c4 \\ \frac{1+\sqrt{3}}{2}, i & \frac{1-\sqrt{3}}{2}, i & c4 \end{array} \right]$$

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.

$$\text{cZeros}\left(\left\{u_{-} + v_{-} - e^{w_{-}}, u_{-}, v_{-}, i\right\}, \left\{u_{-}, v_{-}\right\}\right)$$

$$\left[\frac{e^{w_{-}+i}}{2}, \frac{e^{w_{-}-i}}{2} \right]$$

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 nollakohdat.

$$\text{cZeros}\left(\left\{e^{z_{-}} - w_{-}, w_{-}, z_{-}^2\right\}, \left\{w_{-}, z_{-}\right\}\right)$$

$$\left[0.494866, -0.703467 \right]$$

Jos sarja ei ole polynominen kaikilta muuttujiltaan eikä lineaarinen tuntemattomilta muuttujiltaan, **cZeros()** 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.

cZeros()

Katalogi >

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

$$cZeros(\{e^z - w_-, w_- - z_-^2\}, \{w_-, z_- = 1 + i\})$$

$$[0.149606 + 4.8919 \cdot i \quad 1.58805 + 1.54022 \cdot i]$$

D**dbd()**

Katalogi >

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.

MM.DDYY (Yhdysvalloissa yleisesti käytetty esitystapa)

DDMM.YY (Euroopassa yleisesti käytetty esitystapa)

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

►DD

Katalogi >

Laus1 ►DD ⇒ *arvo* *Listal*

►DD ⇒ *lista* *Matriisi1*

►DD ⇒ *matriisi*

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

Astekulmatilassa:

{1.5°}►DD	1.5°
{45°22'14.3"}►DD	45.3706°
{ {45°22'14.3",60°0'0"} }►DD	{45.3706°,60°}

Graadikulmatilassa:

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

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

Radiaanikulmatilassa:

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

Lauseke 1►Decimal⇒*lauseke*

Listal ►Decimal⇒*lauseke*

Matriisi 1 ►Decimal⇒*lauseke*

Huomaa: Voit syöttää tämän operaattorin tietokoneen näppäimistöä kirjoittamalla @►Decimal.

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

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

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.

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

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 lausekkeitä ja ohjeita (kuten **If**, **Then**, **Else** ja **For**).

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

Huomaa: Katso myös **Define LibPriv**, sivu 54, ja **Define LibPub**, sivu 55.

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

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

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

Define LibPriv (Määritä LibPriv)

Katalogi > 

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 53, ja **Define LibPub**, sivu 55.

Define LibPub (Määritä LibPub)

Katalogi > 

```
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 53, ja **Define LibPriv**, sivu 54.

deltaList()

Katso Δ List(), sivu 107.

deltaTmpCnv()

Katso Δ tmpCnv(), sivu 198.

DelVarKatalogi > **DelVar** *Muutt1*[, *Muutt2*] [, *Muutt3*] ... $2 \rightarrow a$ 2**DelVar** *Muutt.* $(a+2)^2$ 16

Poistaa määritetyn muuttujan tai muuttujaryhmän muistista.

DelVar *a* Done $(a+2)^2$ $(a+2)^2$

Jos yksi tai useampia muuttujia on lukittu, tämä komento aiheuttaa virheilmoituksen ja poistaa vain lukitsemattomat muuttujat. Katso **unLock**, sivu 207.

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.*

aa.a:=45 45*aa.b:=5.67* 5.67*aa.c:=78.9* 78.9

getVarInfo()	<i>aa.a</i>	"NUM"	"{:}"
	<i>aa.b</i>	"NUM"	"{:}"
	<i>aa.c</i>	"NUM"	"{:}"

DelVar *aa.* Done

getVarInfo() "NONE"

delVoid()Katalogi > **delVoid**(*Listal*)⇒*lista*

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


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

Lisätietoja tyhjästä elementeistä, katso sivu 248.

derivative()Katso *d()*, sivu 231.

deSolve(1.astTai2.astODE, Muutt, riippuvaMuutt) ⇒ yleinen ratkaisu

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 muuttuajaan 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 kokonaisluku 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.

Kun vertaat 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:

deSolve($y''+2\cdot y'+y=x^2, x, y$)	
$y=(c3\cdot x+c4)\cdot e^{-x}+x^2-4\cdot x+6$	
right(Ans) → temp	$(c3\cdot x+c4)\cdot e^{-x}+x^2-4\cdot x+6$
$\frac{d^2}{dx^2}(temp)+2\cdot \frac{d}{dx}(temp)+temp-x^2$	0
DelVar temp	Done

deSolve($y'=(\cos(y))^2\cdot x, x, y$)	$\tan(y)=\frac{x^2}{2}+c4$
--	----------------------------

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

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

deSolve(ode and $y(0)=0, x, y$) → soln	
$\frac{-(2\cdot \sin(y)+y^2)}{2} = (e^x-1)\cdot e^{-x}\cdot \sin(y)$	

soln x=0 and y=0	true
ode y'=impDif(soln,x,y)	true
DelVar ode,soln	Done

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

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

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

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

$$w = \frac{x \cdot e^x}{(\ln(e))^2 + 9} + \frac{e^3 \cdot x \cdot \cos(3 \cdot x)}{(\ln(e))^2 + 9} - \frac{e^6 \cdot x \cdot \sin(3 \cdot x)}{(\ln(e))^2 + 9}$$

$$\text{deSolve}\left(y'' = y^{\frac{-1}{2}} \text{ and } y(0) = 0 \text{ and } y'(0) = 0, t, y\right)$$

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

$$\text{solve}(Ans, y)$$

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

$$\text{deSolve}(y'' = x \text{ and } y(0) = 1 \text{ and } y'(2) = 3, x, y)$$

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

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

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

det()

Katalogi >

det(neliömatriisi[, Toleranssi])⇒*lauseke*Laskee *neliömatriisin* determinantin.

Valinnaisesti kaikkia matriisielementtejä käsitellään nollassa, 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 **ctrl** **enter** tai **Automaattinen tai likimääräinen** -tilan valintaa Approximate (Likimääräinen), laskut suoritetaan liukulukuaritmetiikalla.
- Jos *Toleranssi* jätetään pois tai sitä ei käytetä, oletusarvoinen toleranssi lasketaan seuraavasti:

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

$\det\begin{pmatrix} a & b \\ c & d \end{pmatrix}$	$a \cdot d - b \cdot c$
$\det\begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}$	-2
$\det\left(\text{identity}(3) - x \cdot \begin{pmatrix} 1 & -2 & 3 \\ -2 & 4 & 1 \\ 6 & -2 & 7 \end{pmatrix}\right)$	$-(98 \cdot x^3 - 55 \cdot x^2 + 12 \cdot x - 1)$
$\begin{bmatrix} 1. \text{E}20 & 1 \\ 0 & 1 \end{bmatrix} \rightarrow \text{mat1}$	$\begin{bmatrix} 1. \text{E}20 & 1 \\ 0 & 1 \end{bmatrix}$
$\det(\text{mat1})$	0
$\det(\text{mat1}, 1)$	1. E20

diag()

Katalogi >

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

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

diag(neliömatriisi)⇒*rivimatriisi*Laskee rivimatriisin, joka sisältää elementit *neliömatriisin* päälävistäjästä.*neliömatriisi*:n on oltava neliö.

$\text{diag}([2 \ 4 \ 6])$	$\begin{bmatrix} 2 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 6 \end{bmatrix}$
$\begin{bmatrix} 4 & 6 & 8 \\ 1 & 2 & 3 \\ 5 & 7 & 9 \end{bmatrix}$	$\begin{bmatrix} 4 & 6 & 8 \\ 1 & 2 & 3 \\ 5 & 7 & 9 \end{bmatrix}$
$\text{diag}(\text{Ans})$	$\begin{bmatrix} 4 & 2 & 9 \end{bmatrix}$

dim()

Katalogi >

dim(Lista)⇒*kokonaisluku*Laskee *Listan* mitat.

$\text{dim}(\{0,1,2\})$	3
-------------------------	---

dim()

Katalogi >

dim(Matriisi)⇒*lista*

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

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

dim(Merkkijono)⇒*kokonaisluku*Laskee merkkijonon *Merkkijono* sisältämien merkkien lukumäärän.

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

Disp

Katalogi >

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
```

►DMS

Katalogi >

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

Tulkitsee argumentin kulmana ja näyttää vastaavan DMS-luvun (DDDDDD°MM'SS.ss"). DMS-muoto (asteet, minuutit, sekunnit) on kuvattu kohdissa °, ', " sivulla sivu 240 .

Astekulmatilassa:

{45.371}►DMS	45°22'15.6"
{ {45.371,60} }►DMS	{ 45°22'15.6",60° }

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

domain()

Luettelo > 

domain(Laus 1, Muut) ⇒ lauseke

Antaa vastauksena *Laus 1* 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()**, -funktiolle riippumatta siitä, ilmaantuvatko ne eksplisiitisti tai käyttäjän määrittämien muuttujien ja funktioiden puitteissa. Seuraavassa esimerkissä lauseketta ei voi sieventää, sillä $f()$ on kielletty funktio.

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

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

dominantTerm(Laus1, Muutt [, Piste])
 \Rightarrow lauseke

dominantTerm(Laus1, Muutt [, Piste]) | Muutt>Piste \Rightarrow lauseke

dominantTerm(Laus1, Muutt [, Piste]) | Muutt<Piste \Rightarrow lauseke

Laskee dominanttitermin *Laus1*: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 $-\infty$, 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 $z=0$, $e^{-1/z}$, kun $z=0$, tai e^z , kun $z = \infty$ tai $-\infty$.

$$\text{dominantTerm}(\tan(\sin(x)) - \sin(\tan(x)), x)$$

$$\frac{x^7}{30}$$

$$\text{dominantTerm}\left(\frac{1 - \cos(x-1)}{(x-1)^3}, x, 1\right) \quad \frac{1}{2 \cdot (x-1)}$$

$$\text{dominantTerm}\left(x^{-2} \cdot \tan\left(\frac{1}{x^3}\right), x\right) \quad \frac{1}{x^3}$$

$$\text{dominantTerm}(\ln(x^x - 1) \cdot x^{-2}, x) \quad \frac{\ln(x \cdot \ln(x))}{x^2}$$

$$\text{dominantTerm}\left(e^{\frac{-1}{z}}, z\right)$$

$$\text{dominantTerm}\left(e^{\frac{-1}{z}}, z, 0\right)$$

$$\text{dominantTerm}\left(\left(1 + \frac{1}{n}\right)^n, n, \infty\right) \quad e$$

$$\text{dominantTerm}\left(\tan^{-1}\left(\frac{1}{x}\right), x, 0\right) \quad \frac{\pi \cdot \text{sign}(x)}{2}$$

$$\text{dominantTerm}\left(\tan^{-1}\left(\frac{1}{x}\right), x\right), x > 0 \quad \frac{\pi}{2}$$

Jos sarjassa tai yhdellä sen derivaatoista on hyppypäjätkuvuus kohdassa *Piste*, tulos sisältää todennäköisesti alalausekkeita, jotka ovat muotoa $\text{sign}(\dots)$ tai $\text{abs}(\dots)$ reaaliselle kehitelmän muuttujalle tai $(-1)^{\text{floor}(\dots \cdot \text{angle}(\dots))}$ 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 arvailla iteratiivisesti tai interaktiivisesti ohjelmasilmukan avulla.

Huomaa: Katso myös **series()**, sivu 168.

dotP()

dotP(Lista1, Lista2) ⇒ lauseke

Laskee kahden listan “pistetulon”.

dotP(Vektori1, Vektori2) ⇒ lauseke

Laskee kahden vektorin “pistetulon”.

Kummankin on oltava rivivektoreita, tai kummankin on oltava sarakevektoreita.

$\text{dotP}(\{a,b,c\},\{d,e,f\})$	$a \cdot d + b \cdot e + c \cdot f$
------------------------------------	-------------------------------------

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

$\text{dotP}([a \ b \ c],[d \ e \ f])$	$a \cdot d + b \cdot e + c \cdot f$
--	-------------------------------------

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

E

$e^{\wedge}()$ e^x painike $e^{\wedge}(LausI) \Rightarrow lauseke$

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

Huomaa: Katso myös **e eksponenttimalli**, sivu 6.

Huomaa: Painikkeen e^x painaminen, jotta näkyviin saadaan $e^{\wedge}()$, on eri asia kuin näppäimistön merkin E painaminen.

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

 $e^{\wedge}(Listal) \Rightarrow lista$

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

 $e^{\wedge}(neliömatrissi) \Rightarrow neliömatrissi$

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

neliömatrissi:n on oltava diagonalisoitavissa. Vastaus sisältää aina liukulukuja.

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

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

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

eff()Katalogi > \int $eff(nimelliskorko,CpY) \Rightarrow arvo$

Talouseläntatoiminto, joka muuntaa nimelliskorkokannan *nimelliskorko* efektiiviseksi vuosikoroksi, kun *CpY* määritetään korkojaksojen lukumääräksi vuodessa.

nimelliskoron on oltava reaaliuku, ja *CpY*:n on oltava reaaliuku > 0 .

Huomaa: Katso myös **nom()**, sivu 128.

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

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.

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

eigVc(mI)

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

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

eigVl()

eigVl(neliömatriisi)⇒lista

Suorakulmakompleksimuodossa:

Laskee listan reaalisen tai kompleksisen *neliömatriisin* 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.

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

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 ▶.

If BooleanLaus1 Then*Lohko1***Elseif BooleanLaus2 Then***Lohko2*

⋮

Elseif BooleanLausN Then*LohkoN***EndIf**

⋮

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

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

Return 5

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

Return 3

EndIf

EndFunc

*Done***EndFor****Katso For, sivu 79.****EndFunc****Katso Func, sivu 83.****EndIf****Katso If, sivu 91.****EndLoop****Katso Loop, sivu 114.****EndPrgm****Katso Prgm, sivu 143.****EndTry****Katso Try, sivu 200.**

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 $\text{depVar}(\text{Var0}) = \text{depVar0}$ 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*).

Var on riippumaton muuttuja.

ListOfDepVars on riippuvien muuttujien luettelo.

Differentiaaliyhtälö:

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

$$\text{euler}(0.001 \cdot y \cdot (100 - y), t, y, \{0, 100\}, 10, 1) \begin{bmatrix} 0. & 1. & 2. & 3. & 4. \\ 10. & 10.9 & 11.8712 & 12.9174 & 14.042 \end{bmatrix}$$

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:

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

$$y1(0) = 2 \text{ ja } y2(0) = 5$$

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

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

$ListOfDepVars0$ on riippuvien muuttujien alkuehtoien luettelo.

$VarStep$ nolasta eroava numero niin, että $sign(VarStep) = sign(VarMax - Var0)$ ja ratkaisut lasketaan $Var0 + i \cdot VarStep$ kaikille $i=0,1,2,\dots$ niin, että $Var0 + i \cdot VarStep$ 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 $VarStep / eulerStep$.

eval()

Läitevalikko

$eval(Expr) \Rightarrow string$

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

Argumentin $Expr$ on sievennyttävä reaaliarvoiksi.

Aseta RGB-ledin sininen väri puolelle intensiteetille.

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

Palauta sininen väri OFF-tilaan.

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

$eval()$ -argumentin on sievennyttävä reaaliarvoiksi.

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.

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 85), **GetStr** (sivu 89), ja **Send** (sivu 166).

<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"

exact()

Katalogi >

exact(LausI [, Toleranssi]) ⇒ lauseke
exact(ListaI [, Toleranssi]) ⇒ lista
exact(MatriisiI [, 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)	$\frac{7·x}{2}+y$
exact({0.2,0.33,4.125})	$\left\{\frac{1}{5}, \frac{33}{100}, \frac{33}{8}\right\}$

Exit

Katalogi >

Exit

Funktio 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 $temp,i$	
$0 \rightarrow temp$	
For $i,1,100,1$	
$temp+i \rightarrow temp$	
If $temp>20$ Then	
Exit	
EndIf	
EndFor	
EndFunc	
$g()$	21

►exp**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**.

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

exp()**exp(Laus I) ⇒ lauseke**

Laskee *e*:n arvon korotettuna *Laus I*:n potenssiin.

Huomaa: Katso myös *e* eksponenttimalli, sivu 6.


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

exp(Lista I) ⇒ lista

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

e^1	<i>e</i>
$e^1.$	2.71828
e^{3^2}	<i>e</i> 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.

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

exp▶lista()

Katalogi > 

exp▶list(Laus, Muutt) ⇒ lista

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

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.

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

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

expand(Laus I, Muutt) laskee *Laus I*: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(Laus I, [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(Laus I, [Muutt]) jakaa myös itseisarvot, **sign()**, ja eksponentit muuttujasta *Muutt* riippumatta.

Huomaa: Katso myös **tExpand()**, jossa käsitellään trigonometrista kulma-summa- ja monikulmalavennusta.

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

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

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

$$\text{expand}(Ans, x) = \frac{1}{x-1} - \frac{1}{x} + \frac{1}{y \cdot (y-1)}$$

$$\text{expand}\left(\frac{x^3 + x^2 - 2}{x^2 - 2}\right) = \frac{2 \cdot x}{x^2 - 2} + x + 1$$

$$\text{expand}(Ans, x) = \frac{1}{x - \sqrt{2}} + \frac{1}{x + \sqrt{2}} + x + 1$$

$$\text{expand}(Ans) = \frac{\ln(2 \cdot x \cdot y) + \sqrt{2} \cdot x \cdot y}{\ln(x \cdot y) + \sqrt{2} \cdot \sqrt{x \cdot y} + \ln(2)}$$

$$\text{expand}(Ans) | y \geq 0 = \frac{\ln(x) + \sqrt{2} \cdot \sqrt{x} \cdot \sqrt{y} + \ln(y) + \ln(2)}{e^{2 \cdot x + y} + \text{sign}(x \cdot y) + |x \cdot y|}$$

$$\text{sign}(x \cdot y) + |x \cdot y| + e^{2 \cdot x + y} = e^{2 \cdot x + y} + \text{sign}(x \cdot y) + |x \cdot y|$$

$$\text{expand}(Ans) = \text{sign}(x) \cdot \text{sign}(y) + |x| \cdot |y| + (e^x)^2 \cdot e^y$$

expr()Katalogi > **expr**(*Merkkijono*) \Rightarrow *lauseke*

Määrittää *Merkkijono*n sisältämän merkkijono lausekkeena ja suorittaa toimenpiteen välittömästi.

<code>expr("1+2+x^2+x")</code>	x^2+x+3
<code>expr("expand((1+x)^2)")</code>	$x^2+2\cdot x+1$
<code>"Define cube(x)=x^3" \rightarrow funcstr</code>	<code>"Define cube(x)=x^3"</code>
<code>expr(funcstr)</code>	<i>Done</i>
<code>cube(2)</code>	8

ExpRegKatalogi > **ExpReg** *X*, *Y* [, [*Frekv*][, *Luokka*, *Sisällyttä*]]

Laskee eksponentiaalisen regression $y = a \cdot (b)^x$ listoista *X* ja *Y* frekvenssillä *Frekv*. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 185.)

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.

Sisällyttä 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 248.

Tulosmuuttuja	Kuvaus
stat.RegEqn	Regressioyhtälö: $a \cdot (b)^x$
stat.a, stat.b	Regressiokertoimet

Tulosmuuttuja	Kuvaus
stat.r ²	Muunnettujen tietojen lineaarimäärittelyn kerroin
stat.r	Muunnettujen tietojen korrelaatiokerroin (x, ln(y))
stat.Resid	Eksponentiaalimalliin liittyvät jäännökset
stat.ResidTrans	Muunnettujen tietojen lineaariseen sovitukseseen liittyvät jäännökset
stat.XReg	Muokatus <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	Muokatus <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(Laus I[, Muutt]) ⇒ lauseke

factor(Lista I[, Muutt]) ⇒ lista

factor(Matriisi I[, Muutt]) ⇒ matriisi

factor(Laus I) jakaa *Laus I*:n kaikki muuttujat tekijöihin yhteisen nimittäjän suhteen.

Laus I :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(Laus I, Muutt) laskee *Laus I*:n jaettuna tekijöihin muuttujan *Muutt* suhteen.

Laus I :ä jaetaan mahdollisimman paljon kohti reaalisia tekijöitä, jotka ovat lineaarisia muuttujassa *Muutt*, vaikka tästä syntyisi irrationaalisia vakioita tai alalausekkeita, jotka ovat irrationaalisia muissa muuttujissa.

$$\frac{\text{factor}(a^3 \cdot x^2 - a \cdot x^2 - a^3 + a)}{a \cdot (a-1) \cdot (a+1) \cdot (x-1) \cdot (x+1)}$$

$$\frac{\text{factor}(x^2+1)}{x^2+1}$$

$$\frac{\text{factor}(x^2-4)}{(x-2) \cdot (x+2)}$$

$$\frac{\text{factor}(x^2-3)}{x^2-3}$$

$$\frac{\text{factor}(x^2-a)}{x^2-a}$$

$$\frac{\text{factor}(a^3 \cdot x^2 - a \cdot x^2 - a^3 + a, x)}{a \cdot (a^2-1) \cdot (x-1) \cdot (x+1)}$$

$$\frac{\text{factor}(x^2-3, x)}{(x+\sqrt{3}) \cdot (x-\sqrt{3})}$$

$$\frac{\text{factor}(x^2-a, x)}{(x+\sqrt{a}) \cdot (x-\sqrt{a})}$$

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.

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.

Pysäytä laskenta käsin,

- **Kämmenlaite:** Pidä -painiketta

$$\frac{\text{factor}(x^5+4\cdot x^4+5\cdot x^3-6\cdot x-3)}{x^5+4\cdot x^4+5\cdot x^3-6\cdot x-3}$$

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

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

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 komentoa **isPrime()**. Se on paljon nopeampi, erityisesti jos *rationaaliluku* ei ole jaoton, ja jos toiseksi suurimmassa tekijässä on enemmän kuin viisi numeroa.

F Cdf()

F Cdf(alaraja,yläraja,dfOsoitt,dfNimitt)
 \Rightarrow luku, jos *alaraja* ja *yläraja* ovat lukuja,
lista, jos *alaraja* ja *yläraja* ovat listoja

F Cdf(alaraja,yläraja,dfOsoitt,dfNimitt)
 \Rightarrow 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 $P(X \leq \textit{yläraja})$:lle *alaraja* = 0.

Fill

Fill Laus, matriisiMuutt \Rightarrow *matriisi*

Korvaa muuttujan *matriisiMuutt* jokaisen elementin lausekkeella *Laus*.

matriisiMuuttujan on oltava valmiiksi olemassa.

Fill Laus, listaMuutt \Rightarrow *lista*

Korvaa muuttujan *listaMuutt* jokaisen elementin lausekkeella *Laus*.

$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$	\rightarrow <i>amatrix</i>	$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$
--	------------------------------	--

Fill 1.01,*amatrix* Done

<i>amatrix</i>	$\begin{bmatrix} 1.01 & 1.01 \\ 1.01 & 1.01 \end{bmatrix}$
----------------	--

$\{1,2,3,4,5\}$	\rightarrow <i>alist</i>	$\{1,2,3,4,5\}$
-----------------	----------------------------	-----------------

Fill 1.01,*alist* Done

<i>alist</i>	$\{1.01,1.01,1.01,1.01,1.01\}$
--------------	--------------------------------

listaMuuttujan on oltava valmiiksi olemassa.

FiveNumSummary

FiveNumSummary X [, $[Frekv]$ [, $[Luokka,Sisällytä]$]

Antaa lyhennetyn version 1 muuttujan tilastoista listalle X . Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 185.)

X edustaa datan sisältävää listaa.

$Frekv$ on valinnainen frekvenssarvojen 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 248.

Tulosmuuttuja	Kuvaus
stat.MinX	x:n arvojen minimi
stat.Q ₁ X	x:n ensimmäinen neljännes
stat.MedianX	x:n mediaani
stat.Q ₃ X	x:n kolmas neljännes
stat.MaxX	x:n arvojen maksimi

floor()Katalogi > **floor(LausI)**⇒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(ListaI)⇒lista**floor(MatriisiI)**⇒matriisi

$$\text{floor}\left(\left\{\frac{3}{2}, 0, -5.3\right\}\right) \quad \{1, 0, -6\}$$

Määrittää listan tai matriisin jokaisen elementin alarajasta.

$$\text{floor}\left(\begin{array}{cc} 1.2 & 3.4 \\ 2.5 & 4.8 \end{array}\right) \quad \begin{array}{c} 1. \quad 3. \\ 2. \quad 4. \end{array}$$

Huomaa: Katso myös **ceiling()** ja **int()**.

fMax()Katalogi > **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, yläraja)

$$\text{fMax}\left(.5 \cdot x^3 - x - 2, x\right) \quad x=\infty$$

fMax(Laus, Muutt) | alaraja ≤ Muutt ≤ 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.

$$\text{fMax}\left(0.5 \cdot x^3 - x - 2, x\right)_{x \leq 1} \quad x = -0.816497$$

Auto or Approximate (Automaattinen tai likimääräinen) -tilan Approximate (Likimääräinen) -asetuksessa **fMax()** etsii iteratiivisesti yhtä likimääräistä paikallista maksimiarvoa. Tämä on usein nopeampi menetelmä, erityisesti jos rajoitit haun operaattorilla "|" suhteellisen pienelle välille, joka sisältää täsmälleen yhden paikallisen maksimin.

Huomaa: Katso myös **fMin()** ja **max()**.

fMin(Laus, Muutt) ⇒ Boolean lauseke

fMin(Laus, Muutt, alaraja)

fMin(Laus, Muutt, alaraja, yläraja)

fMin(Laus, Muutt) | alaraja ≤ Muutt ≤ 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.

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()**.

$fMin(1-(x-a)^2-(x-b)^2, x)$	$x=-\infty$ or $x=\infty$
$fMin(0.5 \cdot x^3 - x - 2, x) x \geq 1$	$x=1.$

For Muutt, Matala, Korkea [, Askel]
Lohko

EndFor

Suorittaa *Lohkon* sisältämät lausekkeet iteratiivisesti jokaiselle muuttujan *Muutt* arvolle, *Matalasta Korkeaan* kohdassa *Askel* määritetyn 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 (:).

Define $g()$ = Func	Done
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

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

format()

format(Laus[, muotoMerkkijono])

⇒merkkijono

Määrittää lausekkeen *Laus* merkkijonona muotoilumallin perusteella.

Laus on voitava sieventää luvuksi.

muotoMerkkijono 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 säätty 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.

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"

fPart()Katalogi > **fPart**(*LausI*) \Rightarrow *lauseke***fPart**(*Listal*) \Rightarrow *lista***fPart**(*MatriisiI*) \Rightarrow *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.

FPdf()Katalogi > 

FPdf(*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()Katalogi > **freqTable►list****(Listal**,*freqvKokonaislukuLista*) \Rightarrow *lista*

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

Laskee listan, joka sisältää *Listal*:n elementit lavennettuina *freqvKokonaislukuListan* määrittämien frekvenssien mukaisesti. Tätä funktiota voidaan käyttää laadittaessa frekvenssitaulukkoa Data & Tilastot -sovelluksessa.

Listal voi olla mikä tahansa kelvollinen lista.

freqvKokonaislukuListan on oltava samankokoinen kuin *Listal* ja sen tulee sisältää ainoastaan ei-negatiivisia kokonaislukuelementtejä. Jokainen elementti määrittää kuinka monta kertaa *Listal*-elementti toistetaan tuloslistassa. Nolla-arvo sulkee pois vastaavan *Listal*-elementin.

Huomaa: Voit syöttää tämän funktion tietokoneen näppäimistöltä kirjoittamalla `freqTable@>list(...)`.

Tyhjiä elementtejä ei huomioida. Lisätietoja tyhjistä elementeistä, katso sivu 248.

frequency()

`frequency(Lista,lokerotLista)⇒lista`

Luo listan, joka sisältää `Lista`: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 `Listal`:n elementtien lukumäärää, jotka ovat kyseisen lokeron alueella. Funktion `countlf()` termeillä ilmaistuna vastaus on $\{ \text{countlf}(\text{list}, ? \leq b(1)), \text{countlf}(\text{list}, b(1) < ? \leq b(2)), \dots, \text{countlf}(\text{list}, b(n-1) < ? \leq b(n)), \text{countlf}(\text{list}, b(n) > ?) \}$.

Niitä `Listal`:n elementtejä, joita ei voi "lokeroida", ei huomioida. Tyhjiä elementtejä ei myöskään huomioida. Lisätietoja tyhjistä elementeistä, katso sivu 248.

Listat & Taulukot -sovelluksessa voit käyttää solualueita kummankin argumentin tilalla.

Huomaa: Katso myös `countlf()`, sivu 41.

<code>datalist={1,2,e,3,pi,4,5,6,"hello",7}</code>	
<code>{1,2,2.71828,3,3.14159,4,5,6,"hello",7}</code>	
<code>frequency(datalist,{2.5,4.5})</code>	<code>{2,4,3}</code>

Vastauksen selitys:

2 `Datalistan` elementtiä on ≤ 2.5

4 `Datalistan` elementtiä on > 2.5 ja ≤ 4.5

3 `Datalistan` elementtiä on > 4.5

Elementti "hei" on merkkijono, jota ei voi sijoittaa mihinkään määritetyistä lokeroista.

FTest_2Samp

`FTest_2Samp Lista,Lista2[,Freekv1[,Freekv2[,Hypot]]]`

`FTest_2Samp Lista,Lista2[,Freekv1`

[,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 yhteenvedo tallentuu *stat.results*-muuttujaan. (Katso sivu 185.)

Kun $H : \sigma_1 > \sigma_2$, aseta *Hypot*>0Kun $H^a : \sigma_1 \neq \sigma_2$ (oletus), aseta *Hypot* =0Kun $H^a : \sigma_1 < \sigma_2$, aseta *Hypot*<0

Lisätietoja listassa olevien tyhjen elementtien vaikutuksesta, katso Tyhjät elementitsivulla sivu 248.

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 keskihajonnot <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

Lohko

EndFunc

Malli käyttäjän määrittämän funktion luomista varten.

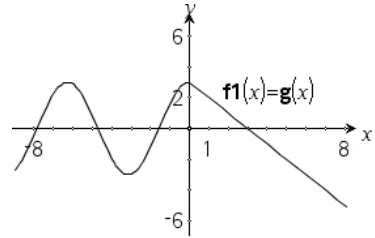
Määritä paloittain määritelty funktio:

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

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 määritysten syöttämisestä löytyvät tuotteen ohjekirjan Laskin-osiosta.

Funktion $g(x)$ kuvaajan piirtämisen tulos



G

gcd()

gcd(*Arvo1*, *Arvo2*) \Rightarrow *lauseke*

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*

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*

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

$\begin{bmatrix} 2 & 4 \\ 6 & 8 \end{bmatrix}$

Laskee *Matriisi1* :n ja *Matriisi2*:n toisiaan vastaavien elementtien suurimmat yhteiset jakajat.

geomCdf()

geomCdf(*p*,*alaraja*,*yläraja*) \Rightarrow *luku*, jos *alaraja* ja *yläraja* ovat lukuja, *lista*, jos *alaraja* ja *yläraja* ovat listoja

geomCdf(*p*,*yläraja*)kun $P(1 \leq X \leq \textit{yläraja})$
 \Rightarrow *luku*, jos *yläraja* on luku, *lista*, jos *yläraja* on lista

Laskee kumulatiivisen geometrisen todennäköisyyden *alarajalta ylärajalle* määritetyllä onnistumistodennäköisyydellä *p*.

Kun $P(X \leq \textit{yläraja})$, aseta *alaraja* = 1.

geomPdf()

geomPdf(*p*,*XArvo*) ⇒ *luku*, jos *XArvo* on luku, *lista*, jos *XArvo* on lista

Laskee diskreetin jakauman todennäköisyyden *XArvo*:n, eli ensimmäisen onnistuneen kokeen järjestysnumeron kohdalla, määritetyllä onnistumistodennäköisyydellä *p*.

Get

Laitevalikko

Get [*kehotemerkkijono*,] *var*[, *statusVar*]

Esimerkki: Kysy laitteen sisäänrakennetun valaistusanturin tämänhetkinen arvo. Käytä komentoa **Get** noutaaksesi arvon ja sijoittaaksesi se muuttujaan *lightval*.

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

Ohjelmointikomento: Noutaa arvon liitetystä TI-Innovator™ Hub ja sijoittaa arvon muuttujaan *var*.

Send "READ BRIGHTNESS"	Done
Get <i>lightval</i>	Done
<i>lightval</i>	0.347922

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.

Upota READ-kysely komentoon **Get**.

Get "READ BRIGHTNESS", <i>lightval</i>	Done
<i>lightval</i>	0.378441

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:

Määrittele *func(arg1, ...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 89 ja **Send** sivu 166.

getDenom()

Katalogi > 

getDenom(Laus l) ⇒ *lauseke*

Muuttaa argumentin lausekkeeksi, jolla on sievennetty yhteinen nimittäjä, ja laskee sen jälkeen lausekkeen nimittäjän.

$\text{getDenom}\left(\frac{x+2}{y-3}\right)$	$y-3$
$\text{getDenom}\left(\frac{2}{7}\right)$	7
$\text{getDenom}\left(\frac{1}{x} + \frac{y^2+y}{y^2}\right)$	$x \cdot y$

getLangInfo()

Katalogi > 

getLangInfo() ⇒ *merkkijono*

Antaa merkkijonon, joka vastaa parhaillaan käytössä olevan kielen lyhyttä nimeä. Voit käyttää sitä esimerkiksi ohjelmassa tai funktiossa nykyisen kielen määrittämiseen.

$\text{getLangInfo}()$	"en"
------------------------	------

getLangInfo()

Katalogi > 

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()

Katalogi > 

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 110, ja **unlock**, sivu 207.

<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()

Katalogi > 

getMode(*TilanNimiKokonaisluku*)⇒*arvo*

getMode(0)⇒*lista*

getMode(*TilanNimiKokonaisluku*) laskee arvon, joka vastaa *TilanNimiKokonaisluku*-tilan nykyistä asetusta.

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.

getMode(0)	{ 1,7,2,1,3,1,4,1,5,1,6,1,7,1,8,1 }
getMode(1)	7
getMode(8)	1

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 170.

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()

getNum(Laus1) ⇒ 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 [*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 85 ja **Send** sivu 166.

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".

$\{1,2,3\} \rightarrow temp$	$\{1,2,3\}$
<code>getType(temp)</code>	"LIST"
$3 \cdot i \rightarrow temp$	$3 \cdot i$
<code>getType(temp)</code>	"EXPR"
<code>DelVar temp</code>	<i>Done</i>
<code>getType(temp)</code>	"NONE"

getVarInfo()

Katalogi > 

getVarInfo() \Rightarrow *matriisi* tai *merkkijono*

getVarInfo(*LibNameString*) \Rightarrow *matriisi* tai *merkkijono*

getVarInfo() laskee tietomatriisin (muuttujan nimi, tyyppi, kirjaston käytettävyyss ja lukittu/lukitsematon-tila) kaikille nykyisessä tehtävässä määritetyille muuttujille ja kirjasto-objekteille.

Jos yhtään muuttujaa ei ole määritetty, **getVarInfo**() antaa vastauksena merkkijonon "NONE".

<code>getVarInfo()</code>	"NONE"												
<code>Define x=5</code>	<i>Done</i>												
<code>Lock x</code>	<i>Done</i>												
<code>Define LibPriv y={1,2,3}</code>	<i>Done</i>												
<code>Define LibPub z(x)=3*x²-x</code>	<i>Done</i>												
<code>getVarInfo()</code>	<table border="1"> <tbody> <tr> <td><i>x</i></td> <td>"NUM"</td> <td>"{"</td> <td>1</td> </tr> <tr> <td><i>y</i></td> <td>"LIST"</td> <td>"LibPriv "</td> <td>0</td> </tr> <tr> <td><i>z</i></td> <td>"FUNC"</td> <td>"LibPub "</td> <td>0</td> </tr> </tbody> </table>	<i>x</i>	"NUM"	"{"	1	<i>y</i>	"LIST"	"LibPriv "	0	<i>z</i>	"FUNC"	"LibPub "	0
<i>x</i>	"NUM"	"{"	1										
<i>y</i>	"LIST"	"LibPriv "	0										
<i>z</i>	"FUNC"	"LibPub "	0										
<code>getVarInfo(tmp3)</code>	"Error: Argument must be a string"												
<code>getVarInfo("tmp3")</code>	<i>[volcy12 "NONE" "LibPub " 0]</i>												

getVarInfo(KirjNimiMerkkijono) antaa tuloksena tietomatriisin kaikista kirjastossa *KirjNimiMerkkijono* määritetyistä kirjasto-objekteista. *KirjNimiMerkkijonon* 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.

$a:=1$	1												
$b:=\begin{bmatrix} 1 & 2 \end{bmatrix}$	$\begin{bmatrix} 1 & 2 \end{bmatrix}$												
$c:=\begin{bmatrix} 1 & 3 & 7 \end{bmatrix}$	$\begin{bmatrix} 1 & 3 & 7 \end{bmatrix}$												
$vs:=\text{getVarInfo}()$	<table border="1"> <tr> <td><i>a</i></td> <td>"NUM"</td> <td>"[]"</td> <td>0</td> </tr> <tr> <td><i>b</i></td> <td>"MAT"</td> <td>"[]"</td> <td>0</td> </tr> <tr> <td><i>c</i></td> <td>"MAT"</td> <td>"[]"</td> <td>0</td> </tr> </table>	<i>a</i>	"NUM"	"[]"	0	<i>b</i>	"MAT"	"[]"	0	<i>c</i>	"MAT"	"[]"	0
<i>a</i>	"NUM"	"[]"	0										
<i>b</i>	"MAT"	"[]"	0										
<i>c</i>	"MAT"	"[]"	0										
$vs\begin{bmatrix} 1 \end{bmatrix}$	$\begin{bmatrix} 1 & \text{"NUM"} & \text{"[]"} & 0 \end{bmatrix}$												
$vs\begin{bmatrix} 1,1 \end{bmatrix}$	1												
$vs\begin{bmatrix} 2 \end{bmatrix}$	"Error: Invalid list or matrix"												
$vs\begin{bmatrix} 2,1 \end{bmatrix}$	$\begin{bmatrix} 1 & 2 \end{bmatrix}$												

Goto

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 $g()$ =Func	<i>Done</i>
Local <i>temp,i</i>	
$0 \rightarrow temp$	
$1 \rightarrow i$	
Lbl top	
$temp+i \rightarrow temp$	
If $i < 10$ Then	
$i+1 \rightarrow i$	
Goto <i>top</i>	
EndIf	
Return <i>temp</i>	
EndFunc	
$g()$	55

Laus I ► **Grad** ⇒ *lauseke*Muuttaa *Laus I*:n graadikulmaan.

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

Astekulmatilassa:

 (1.5) ►Grad $(1.66667)^9$

Radiaanikulmatilassa:

 (1.5) ►Grad $(95.493)^9$

/

identity()

Luettelo >

identity(kokonaisluku) ⇒ *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

Luettelo >

Jos *BooleanLaus*
*Ilmaisut***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.

```
Define g(x)=Func                               Done
  If x<0 Then
  Return x2
  EndIf
  EndFunc
g(-2)                                          4
```

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.

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)=Func
  If x<0 Then
    Return -x
  Else
    Return x
  EndIf
EndFunc
```

$g(12)$	12
$g(-12)$	12

```
Define g(x)=Func
  If x<=5 Then
    Return 5
  ElseIf x>=5 and x<0 Then
    Return -x
  ElseIf x>=0 and x≠10 Then
    Return x
  ElseIf x=10 Then
    Return 3
  EndIf
EndFunc
```

	<i>Done</i>
$g(-4)$	4
$g(10)$	3

ifFn()

ifFn(*BooleanLaus*, *Arvo_Jos_tosi* [, *Arvo_Jos_epätosi* [, *Arvo_Jos_tuntematon*]]) ⇒ 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*,

```
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.

iffn()

Luettelo >

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 **jofn()** toinen, kolmas tai neljäs argumentti on yksi lauseke, Boolean testiä sovelletaan jokaiseen sijaintiin Boolean lausekkeessa *BooleanLaus*.

Huomaa: Jos sievennetty *BooleanLaus* ilmaisu sisältää listan tai matriisin, kaikkien muiden lista- tai matriisiargumenttien on oltava samansuuruisia, ja myös tuloksen on oltava samansuuruinen.

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

Arvo_Jos_tosi on yksittäinen arvo ja vastaa mitä tahansa valittua sijaintia.

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

Arvoa *Arvo_Jos_epätosi* ei ole määritelty. Käytetään merkintää EiMäär

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

Yksi elementti valittu lausekkeesta *Arvo_Jos_tosi*. Yksi elementti valittu lausekkeesta *Arvo_Jos_epätosi*.

imag()

Luettelo >

imag(Expr1) ⇒ lauseke

Laskee argumentin imaginaarisen osan.

Huomaa: Kaikkia määrittelemättömiä muuttujia käsitellään reaali muuttujina. Katso myös *real()*, page 152

imag(Lista1) ⇒ lista

Laskee listan alkutekijöiden imaginaarisista osista.

imag(Matriisi1) ⇒ matriisi

Laskee matriisin alkutekijöiden imaginaarisista osista.

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

$$\text{imag}(z) \quad 0$$

$$\text{imag}(x+i \cdot y) \quad y$$

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

$$\text{imag}\left(\begin{bmatrix} a & b \\ i \cdot c & i \cdot d \end{bmatrix}\right) \quad \begin{bmatrix} 0 & 0 \\ c & d \end{bmatrix}$$

impDif()

Luettelo >

impDif(Yhtälö, Muutt, riippuvaMuutt [,Aste]) ⇒ lauseke

jossa luokan *Aste* oletusarvo on 1

$$\text{impDif}(x^2+y^2=100,x,y) \quad \frac{-x}{y}$$

Laskee implisiittisen derivaatan yhtälöille, joissa yksi muuttuja määritellään implisiittisesti toisen suhteen

Epäsuora operaattori

Katso #(), sivu 237.

inString()

inString(*srcMerkkijono*, *alaMerkkijono*[, *Alku*]) ⇒ kokonaisluku

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

int(*Laus*) ⇒ kokonaisluku

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

int(*Listal*) ⇒ lista

int(*MatriisiI*) ⇒ 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

Laskee lausekkeen $Luku1 \div Luku2$
 etumerkillisen kokonaislukuosan.

Laskee sarjoille ja matriiseille lausekkeen
 (argumentti 1 ÷ argumentti 2)
 etumerkillisen kokonaislukuosan kullekin
 elementtiparille.

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

kokonaisluku

Katso , sivu 232.

interpoloi ()

Luettelo >

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 lukuasettelo.

Differentsiaalilyhtälö:

$$y' = -3 \cdot y + 6 \cdot t + 5 \text{ and } y(0) = 5$$

$rk := rk23(-3 \cdot y + 6 \cdot t + 5, y, \{0, 10\}, 5, 1)$				
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:

$xvalueList := seq(i, 0, 10, 0.5)$
{0, 0.5, 1., 1.5, 2., 2.5, 3., 3.5, 4., 4.5, 5., 5.5, 6., 6.5, 7., 7.5, 8., 8.5, 9., 9.5, 10.}
$xlist := mat \blacktriangleright list(rk[1])$
{0., 1., 2., 3., 4., 5., 6., 7., 8., 9., 10.}
$ylist := mat \blacktriangleright list(rk[2])$
{5., 3.19499, 5.00394, 6.99957, 9.00593, 10.9970}
$yprimeList := -3 \cdot y + 6 \cdot t + 5 y = ylist \text{ and } t = xlist$
{-10., 1.41503, 1.98819, 2.00129, 1.98221, 2.0060}
$interpolate(xvalueList, xlist, ylist, yprimeList)$
{5., 2.67062, 3.19499, 4.02782, 5.00394, 6.0001}

inv χ^2 ()

Luettelo >

inv χ^2 (Ala,df)**invChi2(Ala,df)**

Laskee käänteisen kumulatiivisen χ^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 nopparia. Maryn on arvattava, miten monta kertaa numero 6 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%?

invBinom $\left(0.77,30,\frac{1}{6}\right)$	6
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)

⇒ asteikkomuoto tai matriisi

$N:n$ 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.

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(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[, μ],[σ])

Laskee käänteisen kumulatiivisen normaalijakaumafunktion annetulle alalle *Ala*, joka on normaalijakaumakäyrän alapuolella ja jonka määräävät μ ja σ .

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) ⇒ kokonaisluku**iPart**(Lista1) ⇒ lista**iPart**(Matriisi1) ⇒ matriisi

Laskee argumentin kokonaisosan.

Laskee sarjoille ja matriiseille kunkin elementin kokonaisosan.

iPart(-1.234)	-1.
iPart($\left\{ \frac{3}{2}, -2.3, 7.003 \right\}$)	{1, -2., 7.}

Argumentti voi olla reaali- tai kompleksiluku.

irr()

$\text{irr}(CF0, CFLista [, CFFrekv]) \Rightarrow arvo$

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 120.

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

isPrime()

$\text{isPrime}(Luku) \Rightarrow \text{Boolean vakiolauseke}$

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()` sijaan. 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 määritysten syöttämisestä löytyvät tuotteen ohjekirjan Laskin-osiosta.

$\text{isPrime}(5)$	true
$\text{isPrime}(6)$	false

Funktio, jolla etsitään seuraava jaoton luku määrätyn luvun jälkeen_

Define $\text{nextprim}(n) = \text{Func}$	<i>Done</i>
Loop	
$n+1 \rightarrow n$	
If $\text{isPrime}(n)$	
Return n	
EndLoop	
EndFunc	
$\text{nextprim}(7)$	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 248.

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 <i>temp,i</i>	
$0 \rightarrow temp$	
$1 \rightarrow i$	
Lbl <i>top</i>	
$temp+i \rightarrow temp$	
If $i < 10$ Then	
$i+1 \rightarrow i$	
Goto <i>top</i>	
EndIf	
Return <i>temp</i>	
EndFunc	
$g()$	55

lcm()

Katalogi >

lcm(*Lukul*, *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()

Katalogi > 

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()

Katalogi > 

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

Muuttujaryhmän kopioiminen, katso

CopyVar sivulla sivu 34.

Muuttujaryhmän poistaminen, katso **DelVar** sivulla sivu 56.

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}
```

limit(Lausl, Muutt, Piste[, Suunta])

⇒ lauseke

raja-arvo(Listal, Muutt, Piste [, Suunta])

⇒ lista

raja-arvo(Matriisi l, Muutt, Piste [, Suunta]) ⇒ matriisi

Laskee pyydetyt raja-arvon.

Huomaa: Katso myös **Raja-arvomalli**, sivu 11.

Suunta: negatiivinen=vasemmalta, positiivinen=oikealta, muu=molemmat. (Jos ohje jätetään pois, *Suunta* on oletusarvoisesti 'molemmat'.)

Positiivisen ∞ :n ja negatiivisen ∞ :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 *Lausl* sisältää muita määrittämättömiä muuttujia kuin *Muutt*, joudut mahdollisesti rajoittamaan niitä saadaksesi suppeamman tuloksen.

Raja-arvot voivat olla hyvin herkkiä pyörästysvirheille. 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.

$\lim_{x \rightarrow 5} (2 \cdot x + 3)$	13
$\lim_{x \rightarrow 0^+} \left(\frac{1}{x} \right)$	∞
$\lim_{x \rightarrow 0} \left(\frac{\sin(x)}{x} \right)$	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
$\lim_{x \rightarrow \infty} (a^x)_{a > 1}$	∞
$\lim_{x \rightarrow \infty} (a^x)_{a > 0 \text{ and } a < 1}$	0

LinRegBx $X, Y, [Frekv], [Luokka, Sisällytä]$

Laskee lineaarisen regressioyhtälön $a + b \cdot x$ listaista X ja Y frekvenssillä $Frekv$. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 185.)

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 elementitsivulla sivu 248.

Tulosmuuttuja	Kuvaus
stat.RegEqn	Regressioyhtälö: $a + b \cdot x$
stat.a, stat.b	Regressiokertoimet
stat.r ²	Määrittäkerroin
stat.r	Korrelaatiokerroin
stat.Resid	Regressioyhtälön jäännökset
stat.XReg	Muokatuksen Y Lista:n sisältämä datapisteiden lista, jota käytetään regressiossa komentojen <i>Frekv</i> , <i>Luokkalista</i> ja <i>Sisällytä</i> luokat rajoitusten mukaisesti
stat.YReg	Muokatuksen X Lista: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

LinRegMx $X, Y, [Frekv], [Luokka, Sisällytä]$

Laskee lineaarisen regression $y = m \cdot x + b$ listoista X ja Y frekvenssillä $Frekv$. Tulosten yhteenvedo tallentuu *stat.results*-muuttujaan. (Katso sivu 185.)

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 tyhjien elementtien vaikutuksesta, katso Tyhjät elementtsivulla sivu 248.

Tulosmuuttuja	Kuvaus
stat.RegEqn	Regressioyhtälö: $m \cdot x + b$
stat.m, stat.b	Regressiokertoimet
stat.r ²	Määrittäkerroin
stat.r	Korrelaatiokerroin
stat.Resid	Regressioyhtälön jäännökset
stat.XReg	Muokatus Y Lista:n sisältämä datapisteiden lista, jota käytetään regressiossa komentojen <i>Frekv</i> , <i>Luokkalista</i> ja <i>Sisällytä</i> luokat rajoitusten mukaisesti
stat.YReg	Muokatus X Lista: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

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 185.)

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 ≥ 0 .

Lisätietoja listassa olevien tyhjiä elementtien vaikutuksesta, katso Tyhjä elementtisivulla sivu 248.

Tulosmuuttuja	Kuvaus
stat.RegEqn	Regressioyhtälö: $a+b \cdot x$
stat.a, stat.b	Regressiokertoimet
stat.df	Vapausasteet
stat.r ²	Määrittyskerroin
stat.r	Korrelaatiokerroin
stat.Resid	Regressioyhtälön jäännökset

Vain Kulmakerroin-tyyppi

Tulosmuuttuja	Kuvaus
[stat.CLower, stat.CUpper]	Kulmakertoimen luottamusväli
stat.ME	Luottamusvälin virhemarginaali

Tulosmuuttuja	Kuvaus
stat.SESlope	Kulmakertoimen keskivirhe
stat.s	Keskivirhe suoran ympärillä

Vain Vaste-tyyppi

Tulosmuuttuja	Kuvaus
[stat.CLower, stat.CUpper]	Keskivirhevasteen luottamusväli
stat.ME	Luottamusvälin virhemarginaali
stat.SE	Keskivirhevasteen keskivirhe
[stat.LowerPred, stat.UpperPred]	Yhden havainnon ennusteväli
stat.MEPred	Ennustevälin virhemarginaali
stat.SEPred	Ennusteen keskivirhe
stat. \hat{y}	$a + b \cdot X$ Arvo

LinRegtTest

Katalogi > 

LinRegtTest $X, Y, \text{Frekv}[, \text{Hypot}]$

Laskee lineaarisen regression X - ja Y -listoista ja suorittaa t -testin kulmakertoimen β ja korrelaatiokertoimen ρ 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 frekvenssiarvojen 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 $H_0: \beta \neq 0$ ja $\rho \neq 0$ (oletus), aseta $Hypot=0$

Kun $H_0^a: \beta < 0$ ja $\rho < 0$, aseta $Hypot < 0$

Kun $H_0^a: \beta > 0$ ja $\rho > 0$, aseta $Hypot > 0$

Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu sivu 185.)

Lisätietoja listassa olevien tyhjien elementtien vaikutuksesta, katso Tyhjät elementtsivulla sivu 248.

Tulosmuuttuja	Kuvaus
stat.RegEqn	Regressioyhtälö: $a + b \cdot x$
stat.t	t-tilasto merkitsevyydestille
stat.PVal	Alin merkitsevyytaso, jolla nollahypoteesi voidaan hylätä
stat.df	Vapausasteet
stat.a, stat.b	Regressiokertoimet
stat.s	Keskivirhe suoran ympärillä
stat.SESlope	Kulmakertoimen keskivirhe
stat.r ²	Määrittyskerroin
stat.r	Korrelaatiokerroin
stat.Resid	Regressioyhtälön jäännökset

linSolve()

linSolve(Lineaariyhtälöryhmä, Muutt1, Muutt2, ...)⇒lista

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

linSolve(Lineaariyht1 ja Lineaariyht2 ja ..., Muutt1, Muutt2, ...)⇒lista

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

linSolve({Lineaariyht1, Lineaariyht2, ...}, Muutt1, Muutt2, ...)⇒lista

$$\text{linSolve}\left(\left\{\begin{array}{l} \text{apple} + 4 \cdot \text{pear} = 23 \\ 5 \cdot \text{apple} - \text{pear} = 17 \end{array}\right\}, \{\text{apple}, \text{pear}\}\right)$$

$$\left\{\begin{array}{l} \frac{13}{3}, \frac{14}{3} \end{array}\right\}$$

linSolve(Lineaariyhtälöryhmä, {Muutt1, Muutt2, ...})⇒lista

$$\text{linSolve}\left(\left\{\begin{array}{l} \text{apple} \cdot 4 + \frac{\text{pear}}{3} = 14 \\ -\text{apple} + \text{pear} = 6 \end{array}\right\}, \{\text{apple}, \text{pear}\}\right)$$

$$\left\{\begin{array}{l} \frac{36}{13}, \frac{114}{13} \end{array}\right\}$$

linSolve(Lineaariyht1 ja Lineaariyht2 ja

...,
 $\{\text{Muutt1}, \text{Muutt2}, \dots\} \Rightarrow \text{lista}$

linSolve({*Lineaariyht1*, *Lineaariyht2*, ...},
 {*Muutt1*, *Muutt2*, ...})
 $\Rightarrow \text{lista}$

Laskee ratkaisulistan muuttujille *Muutt1*,
Muutt2, ...

Ensimmäisen argumentin sievennyksen tuloksena on oltava lineaariyhtälöryhmä tai yksi lineaariyhtälö. Muussa tapauksessa esiintyy argumenttivirhe.

Esimerkiksi yhtälön **linSolve**($x=1$ and $x=2,x$) sieventäminen antaa tuloksena virheilmoituksen Argumenttivistä.

ΔList()

ΔList(*Listal*) $\Rightarrow \text{lista}$

$\Delta\text{List}\{\{20,30,45,70\}\}$

$\{10,15,25\}$

Huomaa: Voit syöttää tämän funktion näppäimistöä 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*.

list▶mat()

list▶mat(*Listal* [, *elementtiäRivillä*])
 $\Rightarrow \text{matriisi}$

$\text{list}\blacktriangleright\text{mat}\{\{1,2,3\}\}$

$\begin{bmatrix} 1 & 2 & 3 \end{bmatrix}$

$\text{list}\blacktriangleright\text{mat}\{\{1,2,3,4,5\},2\}$

$\begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 0 \end{bmatrix}$

Laskee matriisin, joka on täytetty rivi riviltä *Listal* elementeillä.

elementtiäRivillä, mikäli sisällytetään, määrittää elementtien määrän rivillä. Oletusarvo on *Listal* elementtien määrä (yksi rivi).

Jos *Lista* ei täytä tulosmatriisia, siihen lisätään nolliä.

Huomaa: Voit syöttää tämän funktion näppäimistöltä kirjoittamalla `list@>mat` (...).

ln

Laus ln ⇒ *lauseke*

$$\left(\log_{10}(x)\right) \ln \quad \frac{\ln(x)}{\ln(10)}$$

Aiheuttaa sen, että lausekkeen *Laus* syöte muunnetaan luonnollisia logaritmeja (ln) sisältäväksi lausekkeeksi.

Huomaa: Voit syöttää tämän operaattorin tietokoneen näppäimistöltä kirjoittamalla `@>ln`.

ln()

ln(Laus I) ⇒ *lauseke*

$$\ln(2.) \quad 0.693147$$

ln(Lista I) ⇒ *lista*

Määrittää argumentin luonnollisen logaritmin.

Jos kompleksilukumuodon tila on Reaali:

$$\ln(\{-3, 1.2, 5\})$$

"Error: Non-real calculation"

Jos kyseessä on lista, määrittää elementtien luonnolliset logaritmit.

Jos kompleksilukumuodon tila on Suorakulma:

$$\ln(\{-3, 1.2, 5\}) \quad \{\ln(3) + \pi \cdot i, 0.182322, \ln(5)\}$$

ln(neliömatriisi I) ⇒ *neliömatriisi*

Määrittää *neliömatriisi I*:n matriisin luonnollisen logaritmin. Tämä ei ole sama kuin kunkin elementin luonnollisen logaritmin laskeminen. Laskentamenetelmä on kuvattu kohdassa `cos()`.

Radiaanikulmatilassa ja

suorakulmakompleksimuodossa:

$$\ln \begin{pmatrix} 1 & 5 & 3 \\ 4 & 2 & 1 \\ 6 & -2 & 1 \end{pmatrix} \begin{matrix} 1.83145 + 1.73485 \cdot i & 0.009193 - 1.49086 \\ 0.448761 - 0.725533 \cdot i & 1.06491 + 0.623491 \cdot i \\ -0.266891 - 2.08316 \cdot i & 1.12436 + 1.79018 \cdot i \end{matrix}$$

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

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

LnReg *X*, *Y*, [*Frekv*] [, *Luokka*, *Sisällytä*]

Laskee logaritmisen regression $y = a + b \cdot \ln(x)$ listoista *X* ja *Y* frekvenssillä *Frekv*. Tulosten yhteenveto tallentuu *stat.results*-muuttuun. (Katso sivu 185.)

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 tyhjien elementtien vaikutuksesta, katso Tyhjät elementtisivulla sivu 248.

Tulosmuuttuja	Kuvaus
stat.RegEqn	Regressioyhtälö: $a + b \cdot \ln(x)$
stat.a, stat.b	Regressiokertoimet
stat.r ²	Muunnettujen tietojen lineaarimäärittelyn kerroin
stat.r	Muunnettujen tietojen korrelaatiokerroin ($\ln(x)$, y)
stat.Resid	Logaritmimalliin liittyvät jäännökset

Tulosmuuttuja	Kuvaus
stat.ResidTrans	Muunnettujen tietojen lineaariseen sovitukseen liittyvät jäännökset
stat.XReg	Muokatusn <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	Muokatusn <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 <i>frekvenssilista</i>

Local

Katalogi > 

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 rollcount()=Func
```

```
Local i
```

```
1 → i
```

```
Loop
```

```
If randInt(1,6)=randInt(1,6)
```

```
Goto end
```

```
i+1 → i
```

```
EndLoop
```

```
Lbl end
```

```
Return i
```

```
EndFunc
```

Done

<i>rollcount</i> ()	16
---------------------	----

<i>rollcount</i> ()	3
---------------------	---

Lock

Katalogi > 

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.*

<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
-----------------	------

Huomaa: Lukitse-komento (**Lock**) tyhjentää toimintojen Kumoa/Tee uudelleen historian, kun sitä käytetään lukitsemattomiin muuttujiin.

Katso **unLock**, sivu 207, ja **getLockInfo()**, sivu 87.

log()**ctrl** **10^x** **painikkeet**

log(Laus1[,Laus2])⇒*lauseke***log(Lista1[,Laus2])**⇒*lista*

$$\log_{10}(2.) = 0.30103$$

$$\log_4(2.) = 0.5$$

$$\log_3(10) - \log_3(5) = \log_3(2)$$

Laskee ensimmäisen argumentin kantaluku-*Laus2*:n logaritmin.

Huomaa: Katso myös **Logaritmimalli**, sivu 6.

Kun kyseessä on lista, laskee elementtien kantaluku-*Laus2*:n logaritmin.

Jos kompleksilukumuodon tila on Reaali:

$$\log_{10}(\{-3,1.2,5\}) \quad \text{Error: Non-real result}$$

Jos toinen argumentti jätetään pois, kantalukuna käytetään lukua 10.

Jos kompleksilukumuodon tila on Suorakulma:

$$\log_{10}(\{-3,1.2,5\}) = \left\{ \log_{10}(3) + 1.36438 \cdot i, 0.079181, \log_{10}(5) \right\}$$

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()**.

Radiaanikulmatilassa ja

suorakulmakompleksimuodossa:

$$\log_{10} \begin{pmatrix} 1 & 5 & 3 \\ 4 & 2 & 1 \\ 6 & -2 & 1 \end{pmatrix} = \begin{bmatrix} 0.795387 + 0.753438 \cdot i & 0.003993 - 0.6474 \cdot i \\ 0.194895 - 0.315095 \cdot i & 0.462485 + 0.2707 \cdot i \\ -0.115909 - 0.904706 \cdot i & 0.488304 + 0.7774 \cdot i \end{bmatrix}$$

neliömatriisi1:n on oltava diagonalisoitavissa. Vastaus sisältää aina liukulukuja.

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

Jos kantalukuargumentti jätetään pois, kantalukuna käytetään lukua 10.

Laus ▶ **logbase**(*Laus*1) ⇒ *lauseke*

Aiheuttaa sen, että syötteenä oleva Lauseke sievennetään lausekkeeksi, joka käyttää kantaluku-*Laus*1:ä.

Huomaa: Voit syöttää tämän operaattorin tietokoneen näppäimistöltä kirjoittamalla @>**logbase** (...).

$$\log_3(10) - \log_5(5) \text{ ▶ logbase}(5) \quad \frac{\log\left(\frac{10}{3}\right)}{\log\left(\frac{3}{5}\right)}$$

Logistic *X*, *Y* [, [*Frekv*] [, *Luokka*, *Sisällyttä*]]

Laskee logistisen regressioy = (c/(1+a · e^{-bx})) listoista *X* ja *Y* frekvenssillä *Frekv*. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 185.)

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.

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

Lisätietoja listassa olevien tyhjen elementtien vaikutuksesta, katso Tyhjä elementtsivulla sivu 248.

Tulosmuuttuja	Kuvaus
stat.RegEqn	Regressioyhtälö: c/(1+a · e ^{-bx})
stat.a, stat.b, stat.c	Regressiokertoimet

Tulosmuuttuja	Kuvaus
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

LogisticD

Katalogi > 

LogisticD *X*, *Y* [, [*Iteraatiot*] , [*Frekv*] [, *Luokka*, *Sisällytä*]]

Laskee logistisen regressioon $y = (c/(1+a \cdot e^{bx})+d)$ listoista *X* ja *Y* frekvenssillä *Frekv* käyttäen tiettyä *Iteraatioiden* määrää. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 185.)

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 tyyppillisesti 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 Tyhjä elementtisivulla sivu 248.

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
stat.XReg	Muokatus <i>X Listan</i> sisältämä datapisteiden lista, jota käytetään todellisesti regressiossa komentojen <i>Frekv</i> , <i>Luokka Lista</i> ja <i>Sisällytä Luokat rajoitusten mukaisesti</i>
stat.YReg	Muokatus <i>Y Listan</i> sisältämä datapisteiden lista, jota käytetään todellisesti regressiossa komentojen <i>Frekv</i> , <i>Luokka Lista</i> ja <i>Sisällytä Luokat rajoitusten mukaisesti</i>
stat.FreqReg	Komentoja <i>stat.XReg</i> ja <i>stat.YReg</i> vastaava frekvenssilista

Loop (Silmukka)

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 rollcount()=Func
```

```
Local i
```

```
1 → i
```

```
Loop
```

```
If randInt(1,6)=randInt(1,6)
```

```
Goto end
```

```
i+1 → i
```

```
EndLoop
```

```
Lbl end
```

```
Return i
```

```
EndFunc
```

Done

```
rollcount()
```

16

```
rollcount()
```



3

LU *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   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.

6 12 18	$\rightarrow m1$	6 12 18
5 14 31		5 14 31
3 8 18		3 8 18

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

<i>lower</i>	1 0 0
	$\frac{5}{6}$ 1 0
	$\frac{1}{2}$ $\frac{1}{2}$ 1

<i>upper</i>	6 12 18
	0 4 16
	0 0 1

<i>perm</i>	1 0 0
	0 1 0
	0 0 1

$\begin{bmatrix} m & n \\ o & p \end{bmatrix} \rightarrow m1$	$\begin{bmatrix} m & n \\ o & p \end{bmatrix}$
---	--

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

<i>lower</i>	1 0
	$\frac{m}{o}$ 1
	o

<i>upper</i>	$\begin{bmatrix} o & p \\ 0 & n - \frac{m \cdot p}{o} \end{bmatrix}$
--------------	--

<i>perm</i>	0 1
	1 0

mat▶list([1 2 3]) {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*) {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 248.

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(Matriisi1[, frekvMatriisi])⇒matriisi`

Suorakulmavektori muodossa:

Luo rivivektorin kaikkien *Matriisi1*:n sarakkeiden keskiarvoista.

mean()

Katalogi > 

Jokainen *frekvMatriisin* elementi näyttää *MatriisiI*:n vastaavien elementtien peräkkäisten esiintymien lukumäärän.

$$\text{mean} \left(\begin{array}{cc} 0.2 & 0 \\ -1 & 3 \\ 0.4 & -0.5 \end{array} \right) \quad \left[-0.133333 \quad 0.833333 \right]$$

Tyhjiä elementtejä ei huomioida. Lisätietoja tyhjiistä elementeistä, katso sivu 248.

$$\text{mean} \left(\begin{array}{cc} \frac{1}{5} & 0 \\ -1 & 3 \\ \frac{2}{5} & \frac{-1}{2} \end{array} \right) \quad \left[\frac{-2}{15} \quad \frac{5}{6} \right]$$

$$\text{mean} \left(\begin{array}{cc} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{array} \right) \left(\begin{array}{cc} 5 & 3 \\ 4 & 1 \\ 6 & 2 \end{array} \right) \quad \left[\frac{47}{15} \quad \frac{11}{3} \right]$$

median()

Katalogi > 

median(Lista[, frekvLista]) ⇒ lauseke

$$\text{median}(\{0.2, 0.1, -0.3, 0.4\}) \quad 0.2$$

Laskee *Listan* elementtien mediaanin.

Jokainen *frekvListan* elementi näyttää *Listan* vastaavien elementtien peräkkäisten esiintymien lukumäärän.

median(MatriisiI[, frekvMatriisi])

⇒ matriisi

Luo rivivektorin, joka sisältää *MatriisiI*:n sarakkeiden mediaanit.

$$\text{median} \left(\begin{array}{cc} 0.2 & 0 \\ 1 & -0.3 \\ 0.4 & -0.5 \end{array} \right) \quad \left[0.4 \quad -0.3 \right]$$

Jokainen *frekvMatriisin* elementi näyttää *MatriisiI*:n vastaavien elementtien peräkkäisten esiintymien lukumäärän.

Huomaa:

- Kaikkien listan tai matriisin syötteiden tulee sieventyä luvuiksi.
- Listassa tai matriisissa olevia tyhjiä elementtejä ei huomioida. Lisätietoja tyhjiistä elementeistä, katso sivu 248.

MedMed

Katalogi > 

MedMed X, Y [, Frekv] [, Luokka, Sisällyttä]

Laskee mediaani-mediaani-suoran = $(m \cdot x + b)$ listoista X ja Y frekvenssillä *Frekv*. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 185.)

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 elementitsivulla sivu 248.

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	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 <i>frekvenssilista</i>

mid()Katalogi > 

mid(lähdemerkkijono, Alku[, Count])
 \Rightarrow 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 ≥ 0 . Jos *Count* = 0, antaa vastauksena tyhjän merkkijonon.

mid(lähdelistä, Alku [, Count]) \Rightarrow lista

Laskee *Count*:n elementit *lähdelistä* aloittaen elementistä numero *Alku*.

Jos *Count* jätetään pois, tai jos se on suurempi kuin *lähdelistä*, laskee kaikki elementit *lähdelistä* aloittaen elementistä numero *Alku*.

Count:n on oltava ≥ 0 . Jos *Count* = 0, antaa vastauksena tyhjän listan.

mid(lähdeMerkkijonoLista, Alku[, Count])
 \Rightarrow 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) \Rightarrow lauseke

min(Lista1, Lista2) \Rightarrow lista
min(Matriisi1, Matriisi2) \Rightarrow 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) \Rightarrow 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(*Matriisi I*) ⇒ *matriisi*

$$\min \begin{pmatrix} 1 & -3 & 7 \\ -4 & 0 & 0.3 \end{pmatrix} \quad [-4 \quad -3 \quad 0.3]$$

Luo rivivektorin, joka sisältää *Matriisi I*:n jokaisen sarakkeen minimielementin.

Huomaa: Katso myös **fMin()** ja **max()**.

mirr()

Katalogi >

mirr(*tal.arvo*,*uud.invest.arvo*,*CF0*,*CFLista* [,*CFFrekv*])

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

Talouseläskentätoiminto, joka laskee investoinnin modifioidun sisäisen korkokannan.

$$\text{list2} := \{2, 2, 2, 1\} \quad \{2, 2, 2, 1\}$$

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

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 98.

mod()

Katalogi >

mod(*Laus1*, *Laus2*) ⇒ *lauseke*

$$\text{mod}(7, 0) \quad 7$$

mod(*Listal1*, *Listal2*) ⇒ *lista*

$$\text{mod}(7, 3) \quad 1$$

mod(*Matriisi1*, *Matriisi2*) ⇒ *matriisi*

$$\text{mod}(-7, 3) \quad 2$$

Laskee ensimmäisen argumentin modulo toinen argumentti identtisten yhtälöiden määrittelyn mukaisesti:

$$\text{mod}(7, -3) \quad -2$$

$$\text{mod}(-7, -3) \quad -1$$

$$\text{mod}(\{12, -14, 16\}, \{9, 7, -5\}) \quad \{3, 0, -4\}$$

mod()

Katalogi >

$$\text{mod}(x,0) = x$$

$$\text{mod}(x,y) = x - y \text{ floor}(x/y)$$

Kun toinen argumentti on ei-nolla, vastaus on periodinen tässä argumentissa. Vastaus on joko nolla tai samanmerkkinen kuin toinen argumentti.

Jos argumentteina on kaksi listaa tai matriisia, laskee listan tai matriisin, joka sisältää vastaavien elementtien kunkin parin modulon (jakojäännöksen).

Huomaa: Katso myös **remain()**, sivu 156

mRow()

Katalogi >

mRow(Laus, Matriisi1, Indeksi) ⇒ matriisi

Luo kopion *Matriisi1*:stä, jossa jokainen rivin *Indeksi* elementti *Matriisi1*:ssä on kerrottu lausekkeella *Laus*.

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

mRowAdd()

Katalogi >

mRowAdd(Laus, Matriisi1, Indeksi1, Indeksi2) ⇒ matriisi

Luo kopion *Matriisi1*:stä, jossa jokainen rivin *Indeksi2* elementti *Matriisi1*:ssä on korvattu seuraavasti:

$mRowAdd\left(-3, \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}, 1, 2\right)$	$\begin{bmatrix} 1 & 2 \\ 0 & -2 \end{bmatrix}$
$mRowAdd\left(n, \begin{bmatrix} a & b \\ c & d \end{bmatrix}, 1, 2\right)$	$\begin{bmatrix} a & b \\ a \cdot n + c & b \cdot n + d \end{bmatrix}$

Laus · rivi *Indeksi1* + rivi *Indeksi2*

MultReg

Katalogi >

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

Laskee listan *Y* moninkertaisen lineaarisen regression listojen *X1, X2, ..., X10* suhteen. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 185.)

Kaikkien listojen on oltava samankokoisia.

Lisätietoja listassa olevien tyhjiin elementtien vaikutuksesta, katso Tyhjät elementitsivulla sivu 248.

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 ²	Moninkertaisen määrityksen kerroin
stat.ŷ Lista	\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 185.)

Kaikkien listojen on oltava samankokoisia.

Lisätietoja listassa olevien tyhjiin elementtien vaikutuksesta, katso Tyhjät elementitsivulla sivu 248.

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
stat.ME	Luottamusvälin virhemarginaali
stat.SE	Keskiarvovasteen keskivirhe
stat.LowerPred, stat.UpperrPred	Yhden havainnon ennusteväli

Tulosmuuttuja	Kuvaus
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 185.)

Lisätietoja listassa olevien tyhjiin elementtien vaikutuksesta, katso Tyhjät elementitsivulla sivu 248.



Tulokset

Tulosmuuttuja	Kuvaus
stat.RegEqn	Regressioyhtälö: $b_0+b_1 \cdot x_1+b_2 \cdot x_2+ \dots$
stat.F	Globaalin F -testin tilasto
stat.PVal	Globaaliin F -tilastoon liittyvä P-arvo
stat.R ²	Moninkertaisen määrittelyn kerroin
stat.AdjR ²	Moninkertaisen määrittelyn säädetty kerroin
stat.s	Virheen keskihajonta
stat.DW	Durbin-Watsonin tilasto; käytetään määrittäessä, esiintyykö mallissa ensimmäisen asteen automaattista korrelaatiota
stat.dfReg	Regression vapausasteet
stat.SSReg	Regression neliöiden summa
stat.MSReg	Regression keskineliö
stat.dfError	Virheen vapausasteet
stat.SSError	Virheen neliöiden summa

Tulosmuuttuja	Kuvaus
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.ŷLista	ŷLista = b0+b1 · x1+ . . .
stat.Resid	Regressioyhtälön jäännökset
stat.sResid	Standardoidut jäännökset; saadaan jakamalla jäännös keskihajonnalla
stat.CookDist	Cookin etäisyys; jäännökseen ja tuotto-suhteeseen 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*

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*

$x \geq 3$ and $x \geq 4$	$x \geq 4$
$x \geq 3$ nand $x \geq 4$	$x < 4$

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}

Vertaa kahta reaalikokonaislukua bitti bitiltä **nand**-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 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).

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

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

Laskee listan kombinaatioista kahden listan sisältämiin vastaaviin elementtipareihin perustuen. Argumenttien on oltava samankokoisia listoja.

nCr()Katalogi > **nCr**(*Matriisi1*, *Matriisi2*) \Rightarrow *matriisi*

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

Laskee matriisin kombinaatioista kahden matriisin sisältämiin vastaaviin elementtipareihin perustuen. Argumenttien on oltava samankokoisia matriiseja.

nDerivative()Katalogi > **nDerivative**(*Laus1*, *Muutt=Arvo* [, *Aste*])
 \Rightarrow *arvo*

$nDerivative(x , x=1)$	1
-------------------------	---

$nDerivative(x , x) _{x=0}$	undef
------------------------------	-------

nDerivative(*Laus1*, *Muutt* [, *Aste*]) |
Muutt=Arvo \Rightarrow *arvo*

$nDerivative(\sqrt{x-1}, x) _{x=1}$	undef
-------------------------------------	-------

Laskee numeerisen derivaatan käyttäen automaattisia derivointimenetelmiä.

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

Derivaatan Asteen on oltava **1** tai **2**.

newList()Katalogi > **newList**(*numElementit*) \Rightarrow *lista*

$newList(4)$	{0,0,0,0}
--------------	-----------

Antaa tuloksena listan, jonka koko on *numElementit*. Jokainen elementti on nolla.

newMat()Katalogi > **newMat**(*numRivit*, *numSarakeet*)
 \Rightarrow *matriisi*

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

Antaa tuloksena nollamatriisin, jonka koko on *numRivit* ja *numSarakeet*.

nfMax()

Katalogi >

nfMax(*Laus*, *Muutt*) \Rightarrow *arvo***nfMax**(*Laus*, *Muutt*, *alaraja*) \Rightarrow *arvo***nfMax**(*Laus*, *Muutt*, *alaraja*, *yläraja*)
 \Rightarrow *arvo***nfMax**(*Laus*, *Muutt*) *alaraja* \leq *Muutt*
 \leq *yläraja* \Rightarrow *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*) \Rightarrow *arvo***nfMin**(*Laus*, *Muutt*, *alaraja*) \Rightarrow *arvo***nfMin**(*Laus*, *Muutt*, *alaraja*, *yläraja*)
 \Rightarrow *arvo***nfMin**(*Laus*, *Muutt*) *alaraja* \leq *Muutt*
 \leq *yläraja* \Rightarrow *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()**.

nInt()

Katalogi >

nInt(*Laus I*, *Muutt*, *Ala*, *Ylä*) \Rightarrow *lauseke*

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

Jos integroitava funktio *Laus I* ei sisällä muita muuttujia kuin *Muutt*, ja jos *Ala* ja *Ylä* ovat vakioita, positiivinen ∞ tai negatiivinen ∞ , tällöin **nInt()** laskee likiarvon lausekkeesta $\int(\text{Laus I}, \text{Var}, \text{Ala}, \text{Ylä})$. Tämä likiarvo on integrandin välillä *Ala* $<$ *Muutt* $<$ *Ylä* olevien joidenkin otosarvojen painotettu keskiarvo.

nInt()

Katalogi >

Tavoitteena on kuusi merkitsevää numeroa. Adaptiivinen algoritmi päättyy, kun näyttää todennäköiseltä, että tavoite on saavutettu, tai kun näyttää epätodennäköiseltä, että lisäotokset tuottaisivat merkittävää parannusta.

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

$$\int_{-\pi}^{\pi+10^{-12}} \cos(x) dx \quad -\sin\left(\frac{1}{1000000000000}\right)$$

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.

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

Huomaa: Katso myös **J()**, sivu 220.

nom()

Katalogi >

nom(*efektiivinenKorko*, *CpY*) ⇒ arvo

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

Talouseläintalvotointo, 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 64.

nor (eikä) **näppäimet**

BooleanLaus1 **nor** *BooleanLaus2* antaa vastauksena *Boolean lausekkeen BooleanList1*

$$x \geq 3 \text{ or } x \geq 4 \quad x \geq 3$$


nor *BooleanList2* antaa vastauksena *Boolean listan BooleanMatriisi1*

$$x \geq 3 \text{ nor } x \geq 4 \quad x < 3$$

nor *BooleanMatriisi2* antaa vastauksena *Boolean matriisin*

Antaa vastauksena loogisen **or** operaation negaation kahdesta argumentista. Antaa vastauksena totuusarvon tosi, epätosi tai yhtälön sievennetyn muodon.

nor (eikä)

ctrl  näppäimet

Listojen ja matriisien kohdalla vastauksena on vertailuja elementti elementiltä.

Kokonaisluku1

nor*Kokonaisluku2*⇒*kokonaisluku*

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

norm()

Katalogi > 

norm(*Matriisi*)⇒*lauseke*

norm(*Vektori*)⇒*lauseke*

Laskee Frobeniusin normin.

$\text{norm}\begin{pmatrix} a & b \\ c & d \end{pmatrix}$	$\sqrt{a^2+b^2+c^2+d^2}$
$\text{norm}\begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}$	$\sqrt{30}$
$\text{norm}\begin{pmatrix} 1 & 2 \end{pmatrix}$	$\sqrt{5}$
$\text{norm}\begin{pmatrix} 1 \\ 2 \end{pmatrix}$	$\sqrt{5}$

normalLine()

Katalogi > 

normalLine(*Lausl*,*Muutt*,*Piste*)⇒*lauseke*

normalLine(*Lausl*,*Muutt*=*Piste*)⇒*lauseke*

Määrittää normaalisuoran *Lausl*:n kuvaamasta käyrästä pisteessä, joka on määritetty kohtaan *Muutt*=*Piste*.

$\text{normalLine}(x^2,x,1)$	$\frac{3}{2} \frac{x}{2}$
$\text{normalLine}((x-3)^2-4,x,3)$	$x=3$
$\text{normalLine}\left(\frac{1}{x^3},x=0\right)$	0
$\text{normalLine}(\sqrt{ x },x=0)$	undef

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 desimaalilikokonaisluvun, joka on etumerkillisen, 64 bitin binaarimuodon lukualaueen ulkopuolella, laskin käyttää symmetristä modulo-operaatiota, jotta arvo saadaan oikealle alueelle. Lisätietoja, katso **Base2**, sivu 22.

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

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

nPr()

$nPr(Laus1, Laus2) \Rightarrow \text{lauseke}$

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.

$nPr(Laus, 0) \Rightarrow 1$

$nPr(Laus, \text{negKokonaisluku}) \Rightarrow 1 / ((Laus+1) \cdot (Laus+2) \dots (\text{lauseke} - \text{negKokonaisluku}))$

$nPr(Laus, \text{posKokonaisluku}) \Rightarrow Laus \cdot (Laus-1) \dots (Laus - \text{posKokonaisluku} + 1)$

$nPr(Laus, \text{eiKokonaisluku}) \Rightarrow Laus! / (Laus - \text{eiKokonaisluku})!$

$nPr(Lista1, Lista2) \Rightarrow \text{lista}$

Luo listan permutaatioista kahden listan sisältämiin vastaaviin elementtipareihin perustuen. Argumenttien on oltava samankokoisia listoja.

$nPr(Matriisi1, Matriisi2) \Rightarrow \text{matriisi}$

$nPr(z,3)$	$z \cdot (z-2) \cdot (z-1)$
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

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

$nPr\left(\begin{bmatrix} 6 & 5 \\ 4 & 3 \end{bmatrix}, \begin{bmatrix} 2 & 2 \\ 2 & 2 \end{bmatrix}\right)$	$\begin{bmatrix} 30 & 20 \\ 12 & 6 \end{bmatrix}$
--	---

Luo matriisin permutaatioista kahden matriisin sisältämiin vastaaviin elementtipareihin perustuen. Argumenttien on oltava samankokoisia matriiseja.

npv()

npv(*Korkoprosentti*,*CFO*,*CFLista* [*,CFFrekv*])

Talouseläintaloudellisuus, joka laskee nettopykkyarvon; tulevien ja poistuvien kassavirtojen nykyisten arvojen summan. Jos npv:n tulos on positiivinen, investointi on kannattava.

Korkoprosentti on prosentti, jolla kassavirtoja (rahan kustannusta) vähennetään yhden jakson aikana.

CFO on alkuperäinen kassavirta aikana 0; arvon on oltava kokonaisluku.

CFLista on lista kassavirtamääristä ensimmäisen kassavirran *CFO* jälkeen.

CFFrekv on 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.

$list1 := \{6000, -8000, 2000, -3000\}$	$\{6000, -8000, 2000, -3000\}$
$list2 := \{2, 2, 2, 1\}$	$\{2, 2, 2, 1\}$
$npv(10, 5000, list1, list2)$	4769.91

nSolve()

nSolve(*Yhtälö*,*Muutt*[=*Arvaus*]) \Rightarrow luku tai virhe_merkijono

$nSolve(x^2 + 5x - 25 = 9, x)$	3.84429
--------------------------------	---------

nSolve(*Yhtälö*,*Muutt*[=*Arvaus*],*alaraja*) \Rightarrow luku tai virhe_merkijono

$nSolve(x^2 = 4, x = -1)$	-2.
---------------------------	-----

$nSolve(x^2 = 4, x = 1)$	2.
--------------------------	----

nSolve(*Yhtälö*,*Muutt*[=*Arvaus*],*alaraja*,*yläraja*) \Rightarrow luku tai virhe_merkijono

Huomaa: Jos ratkaisuja on useita, voit yrittää löytää tietyn ratkaisun käyttämällä apuna arvausta.

nSolve(*Yhtälö*,*Muutt*[=*Arvaus*]) | *alaraja* \leq *Muutt* \leq *yläraja*

⇒ luku tai virhe_merkkijono

Etsii iteratiivisesti yhtä likimääräistä numeerista ratkaisua *Yhtälön* yhdelle muuttujalle. Määritä muuttuja seuraavasti:

muuttuja

– tai –

muuttuja = reaaliluku

Esimerkiksi x kelpaa ja samoin $x=3$.

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

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 185.)

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

X-argumentit ovat datalistoja.

Frekv on valinnainen frekvenssarvojen 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.

Sisällytät 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 248.

Tulosmuuttuja	Kuvaus
stat. \bar{x}	x :n arvojen keskiarvo
stat. Σx	x :n arvojen summa
stat. Σx^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 ₁ X	x :n ensimmäinen neljännes
stat.MedianX	x :n mediaani
stat.Q ₃ X	x :n kolmas neljännes
stat.MaxX	x :n arvojen maksimi
stat.SSX	x :n keskiarvon poikkeamien neliöiden summa

BooleanLaus1 **or** *BooleanLaus2* antaa vastauksena *Boolean* lausekkeen *BooleanList1* **or** *BooleanList2* antaa vastauksena *Boolean* listan *BooleanMatriisi1* **or** *BooleanMatriisi2* antaa vastauksena *Boolean* matriisin

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 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).

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

$x \geq 3$ or $x \geq 4$	$x \geq 3$
--------------------------	------------

Define $g(x) = \text{Func}$	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 kantaluutilassa:

0h7AC36 or 0h3D5F	0h7BD7F
-------------------	---------

Tärkeää: Nolla, ei O-kirjain.

Binaarisessa kantaluutilassa:

0b100101 or 0b100	0b100101
-------------------	----------

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

Jos syötät desimaalikoona luvun, joka on etumerkillisen, 64 bitin binaarimuodon lukualueen ulkopuolella, laskin käyttää symmetristä modulo-operaatiota, jotta arvo saadaan oikealle alueelle. Lisätietoja, katso **►Base2**, sivu 22.

Huomaa: Katso **xor**.

ord()

ord(Merkkijono) ⇒ kokonaisluku
ord(Lista1) ⇒ lista

Määrittää merkkijonon *Merkkijono* ensimmäisen merkin numerokoodin tai luo listan jokaisen listaelementin ensimmäisistä merkeistä.

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

P

►Rx()

►Rx(rLaus, θLaus) ⇒ lauseke
►Rx(rLista, θLista) ⇒ lista
►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:

$\text{►Rx}(r, \theta)$	$\cos(\theta) \cdot r$
$\text{►Rx}(4, 60^\circ)$	2
$\text{►Rx}\left(\left\{-3, 10, 1.3\right\}, \left\{\frac{\pi}{3}, \frac{\pi}{4}, 0\right\}\right)$	$\left\{\frac{-3}{2}, 5\sqrt{2}, 1.3\right\}$

►Ry()

►Ry(rLaus, θLaus) ⇒ lauseke

Radiaanikulmatilassa:

P►Ry()

Katalogi > 

P►Ry(*rLista*, *θLista*) ⇒ *lista*

P►Ry(*rMatriisi*, *θMatriisi*) ⇒ *matriisi*

Määrittää paria (*r*, *θ*) vastaavan *y*-koordinaatin.

Huomaa: Argumentti *θ* 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 (...).

$$P►Ry(r, \theta) \quad \sin(\theta) \cdot r$$

$$P►Ry(4, 60^\circ) \quad 2 \cdot \sqrt{3}$$

$$P►Ry\left(\{-3, 10, 1.3\}, \left\{\frac{\pi}{3}, \frac{\pi}{4}, 0\right\}\right) \\ \left\{\frac{-3 \cdot \sqrt{3}}{2}, -5 \cdot \sqrt{2}, 0\right\}$$

PassErr

Katalogi > 

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 30, ja **Try**, sivu 200.

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

Esimerkki **PassErr**-komentosta, katso esimerkki 2 **Try**-komenton kohdalla, sivu 200.

piecewise()

Katalogi >

piecewise(*Laus1* [, *Ehto1* [, *Laus2* [, *Ehto2* [, ...]]])

Laatii määritelmät paloittain määritellylle funktiolle listan muodossa. Voit luoda paloittain määriteltyjen funktioiden määrittämiä myös mallin avulla.

Huomaa: Katso myös **Paloittain määritellyn funktion malli**, sivu 7.

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(λ , *alaraja*, *yläraja*) \Rightarrow luku, jos *alaraja* ja *yläraja* ovat lukuja, *lista*, jos *alaraja* ja *yläraja* ovat listoja

poissCdf(λ , *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 λ .

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

poissPdf()

Katalogi >

poissPdf(λ , *XArvo*) \Rightarrow luku, jos *XArvo* on luku, *lista*, jos *XArvo* on lista

Laskee todennäköisyyden diskreetille Poissonin jakaumalle, jolla on määritetty keskiarvo λ .

►Polar (►Polaarinen)

Katalogi >

Vektori ►Polar

Huomaa: Voit syöttää tämän operaattorin tietokoneen näppäimistöltä kirjoittamalla @►Polar.

Näyttää *vektorin* polaarisisä muodossa [*r* \angle θ]. Vektorin on oltava kooltaan 2, ja se voi olla rivi tai sarake.

[1 3]►Polar	[3.16228 \angle 1.24905]
[x y]►Polar	$\left[\sqrt{x^2+y^2} \angle \frac{\pi \cdot \text{sign}(y)}{2} - \tan^{-1}\left(\frac{x}{y}\right) \right]$

Huomaa: ►Polar on näyttömuodon ohje, ei muunnosfunktio. Voit käyttää komentoa ainoastaan syöterivin lopussa, eikä se päivitä *ans*:n arvoa.

Huomaa: Katso myös ►Rect, sivu 153.

kompleksiArvo ►Polaarinen

Näyttää *kompleksiVektorin* polaarisisästä 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: Polaarisisästä syötessä $(r \angle \theta)$ on käytettävä sulkeita.

Radiaanikulmatilassa:

$$(3+4 \cdot i) \text{►Polar} \quad e^{i \left(\frac{\pi}{2} - \tan^{-1} \left(\frac{3}{4} \right) \right)} \cdot 5$$

$$\left(4 \angle \frac{\pi}{3} \right) \text{►Polar} \quad e^{i \frac{\pi}{3}} \cdot 4$$

Graadikulmatilassa:

$$(4 \cdot i) \text{►Polar} \quad (4 \angle 100)$$

Astekulmatilassa:

$$(3+4 \cdot i) \text{►Polar} \quad \left(5 \angle 90 - \tan^{-1} \left(\frac{3}{4} \right) \right)$$

polyCoeffs()

polyCoeffs(*Poly* [, *Muutt*]) ⇒ lista

Laatii listan polynomien *Poly* kertoimista muuttujan *Muutt* suhteen.

Poly on oltava polynomilauseke muuttujassa *Muutt*. Suositeltavaa on, ettet jätä muuttujaa *Muutt* pois, ellei *Poly* ole lauseke yhdessä muuttujassa.

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

$$\text{polyCoeffs}((x-1)^2 \cdot (x+2)^3) \quad \{1, 4, 1, -10, -4, 8\}$$

Laventaa polynomien ja valitsee *x*:n poisjätetyille muuttujalle *Muutt*.

$\text{polyCoeffs}((x+y+z)^2, x)$	$\{1, 2 \cdot (y+z), (y+z)^2\}$
$\text{polyCoeffs}((x+y+z)^2, y)$	$\{1, 2 \cdot (x+z), (x+z)^2\}$
$\text{polyCoeffs}((x+y+z)^2, z)$	$\{1, 2 \cdot (x+y), (x+y)^2\}$

polyDegree()

polyDegree(Poly [,Muutt]) ⇒ 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
$\text{polyDegree}((x+y^2+z^3)^2, x)$	2
$\text{polyDegree}((x+y^2+z^3)^2, y)$	4
$\text{polyDegree}((x-1)^{10000}, x)$	10000

Asteesta voidaan ottaa juuri, vaikka kertoimista ei voida. Tämä johtuu siitä, että asteesta voidaan ottaa juuri laventamatta polynomia.

polyEval()

polyEval(Lista1, Laus1) ⇒ lauseke
polyEval(Lista1, Lista2) ⇒ 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.

polyGcd(100,30)	10
polyGcd($x^2-1,x-1$)	$x-1$
polyGcd($x^3-6x^2+11x-6,x^2-6x+8$)	$x-2$

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.

polyQuotient($x-1,x-3$)	1
polyQuotient($x-1,x^2-1$)	0
polyQuotient($x^2-1,x-1$)	$x+1$
polyQuotient($x^3-6x^2+11x-6,x^2-6x+8$)	x
polyQuotient($(x-y) \cdot (y-z), x+y+z, x$)	$y-z$
polyQuotient($(x-y) \cdot (y-z), x+y+z, y$)	$2 \cdot x - y + 2 \cdot z$
polyQuotient($(x-y) \cdot (y-z), x+y+z, z$)	$-(x-y)$

polyRemainder()Katalogi > 

polyRemainder(Poly1,Poly2 [,Muutt])
⇒ lauseke

polyRemainder($x-1,x-3$)	2
polyRemainder($x-1,x^2-1$)	$x-1$
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) ⇒ lista

polyRoots(Kertoinlista) ⇒ lista

Ensimmäinen syntaksi, **polyRoots(Poly, Muutt)**, laskee polynomien *Poly* reaalityyppisten juurten listan muuttujan *Muutt* suhteen. Jos reaalityyppistä juurta ei ole, tuloksena on tyhjä lista: { }.

Poly on oltava polynomi yhdessä muuttujassa.

Toinen syntaksi, **polyRoots(Kertoinlista)**, askee reaalityyppisten juurten listan kertoimille, jotka sisältyvät *Kertoinlistaan*.

Huomaa: Katso myös **cPolyRoots()**, sivu 42.

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

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 185.)

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 Tyhjät elementitsivulla sivu 248.

Tulosmuuttuja	Kuvaus
stat.RegEqn	Regressioyhtälö: $a \cdot (x)^b$
stat.a, stat.b	Regressiokertoimet
stat.r ²	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

Prgm

Laske GCD ja näytä välitulokset.

Lohko

EndPrgm

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 määrittysten 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 234.**Product (PI)**Katso $\Pi()$, sivu 234.**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 248.

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 polynomien 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.

$$\text{propFrac}\left(\frac{4}{3}\right) \quad 1 + \frac{1}{3}$$

$$\text{propFrac}\left(\frac{-4}{3}\right) \quad -1 - \frac{1}{3}$$

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

$$\frac{1}{x+1} + x + \frac{y^2+y+1}{y+1}$$

$$\text{propFrac(Ans)} \quad \frac{1}{x+1} + x + \frac{1}{y+1} + y$$

$$\text{propFrac}\left(\frac{11}{7}\right) \quad 1 + \frac{4}{7}$$

$$\text{propFrac}\left(3 + \frac{1}{11} + 5 + \frac{3}{4}\right) \quad 8 + \frac{37}{44}$$



$$\text{propFrac}\left(3 + \frac{1}{11} - \left(5 + \frac{3}{4}\right)\right) \quad -2 - \frac{29}{44}$$

Q

QR *Matriisi*, *qMatNimi*, *rMatNimi* [, *Tol*]

Suorittaa Householderin QR-dekomponoinnin reaali- tai kompleksilukumatriisista. Tuloksena ovat Q- ja R-matriisit tallentuvat määritettyihin *MatNimi*-muuttujiin. Q-matriisi on jakamaton. R-matriisi on yläkolmiomatriisi.

Valinnaisesti kaikkia matriisielementtejä käsitellään nollassa, 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.

$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix} \rightarrow m1$	$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$
--	---

QR <i>m1,qm,rm</i>	Done		
<i>qm</i>	0.123091	0.904534	0.408248
	0.492366	0.301511	-0.816497
	0.86164	-0.301511	0.408248
<i>rm</i>	8.12404	9.60114	11.0782
	0.	0.904534	1.80907
	0.	0.	0.

$\begin{bmatrix} m & n \\ o & p \end{bmatrix} \rightarrow m1$	$\begin{bmatrix} m & n \\ o & p \end{bmatrix}$
---	--

QR <i>m1,qm,rm</i>	Done	
<i>qm</i>	$\frac{m}{\sqrt{m^2+o^2}}$	$\frac{-\text{sign}(m \cdot p - n \cdot o) \cdot o}{\sqrt{m^2+o^2}}$
	$\frac{o}{\sqrt{m^2+o^2}}$	$\frac{m \cdot \text{sign}(m \cdot p - n \cdot o)}{\sqrt{m^2+o^2}}$
<i>rm</i>	$\sqrt{m^2+o^2}$	$\frac{m \cdot n + o \cdot p}{\sqrt{m^2+o^2}}$
	0	$\frac{m \cdot p - n \cdot o}{\sqrt{m^2+o^2}}$

QuadReg *X,Y* [, *Frekv*] [, *Luokka*, *Sisällytät*]

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 185.)

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

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.

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 248.

Tulosmuuttuja	Kuvaus
stat.RegEqn	Regressioyhtälö: $a \cdot x^2 + b \cdot x + c$
stat.a, stat.b, stat.c	Regressiokertoimet
stat.R ²	Määrittäyskerroin
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
 $y = a \cdot x^4 + b \cdot x^3 + c \cdot x^2 + d \cdot x + e$ elistoista X ja Y
frekvenssillä *Frekv*. Tulosten yhteenveto
tallentuu *stat.results*-muuttujaan.
(Katso sivu sivu 185.)

Kaikkien listojen on oltava samankokoisia
Sisällytä-listaa lukuunottamatta.

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.

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 248.

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 ²	Määrittäkerroin
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

R

R►Pθ()

Luettelo > 

R►Pθ (*xLaus*, *yLaus*) ⇒ *lauseke*

Astekulmatilassa:

R►Pθ (*xLista*, *yLista*) ⇒ *lista*

$$\text{R►P}\theta(x,y) \quad 90 \cdot \text{sign}(y) - \tan^{-1}\left(\frac{x}{y}\right)$$

R►Pθ (*xMatriisi*, *yMatriisi*) ⇒ *matriisi*

Laskee yhtäpitävän θ-koordinaatin argumenttiparille (x,y).

Graadikulmatilassa:

$$\text{R►P}\theta(x,y) \quad 100 \cdot \text{sign}(y) - \tan^{-1}\left(\frac{x}{y}\right)$$

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

Radiaanikulmatilassa:

Huomaa: Voit syöttää tämän funktion tietokoneen näppäimistöltä kirjoittamalla R@>Ptheta (...).

$$\text{R►P}\theta(3,2) \quad \tan^{-1}\left(\frac{2}{3}\right)$$

$$\text{R►P}\theta\left(\left[3 \ -4 \ 2\right], \left[0 \ \frac{\pi}{4} \ 1.5\right]\right) \quad \left[0 \ \tan^{-1}\left(\frac{16}{\pi}\right) + \frac{\pi}{2} \ 0.643501\right]$$

R►Pr()

Luettelo > 

R►Pr (*xLaus*, *yLaus*) ⇒ *lauseke*

Radiaanikulmatilassa:

R►Pr (*xLista*, *yLista*) ⇒ *lista*

$$\text{R►Pr}(3,2) \quad \sqrt{13}$$

R►Pr (*xMatriisi*, *yMatriisi*) ⇒ *matriisi*

$$\text{R►Pr}(x,y) \quad \sqrt{x^2+y^2}$$

Laskee yhtäpitävän r-koordinaatin argumenttiparille (x,y).

$$\text{R►Pr}\left(\left[3 \ -4 \ 2\right], \left[0 \ \frac{\pi}{4} \ 1.5\right]\right)$$

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.

$$\left(1.5\right)\text{►Rad} \quad \left(0.02618\right)^r$$

Graadikulmatilassa:

► Rad

Luettelo > 

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

{1.5}►Rad

{0.023562}^r

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 reaali lukua määrätystä binomijakaumasta.

randInt()

Luettelo > 

randInt

(
Muutt.alaraja
,Muutt.yläaraja) ⇒
lauseke

randInt

(
Muutt.alaraja
,Muutt.yläaraja
,Kokeiden lkm) ⇒
lista

randInt(3,10)

5

randInt(3,10,4)

{9,7,5,8}

randInt()

Luettelo > 

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
määrätyllä välillä.

randMat()

Luettelo > 

randMat(*numRivit*, *numSarakkeet*) ⇒
matriisi

Laskee välillä -9 ja 9 määrätylle mitalle
kokonaislukumatriisin.

Kummankin argumentin on sievennyttyvä
kokonaisluvuksi.

RandSeed 1147	Done									
randMat(3,3)	<table><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()

Luettelo > 

randNorm(μ , σ) ⇒ *lauseke*
randNorm(μ , σ , *Kokeiden lkm*) ⇒ *lista*

randNorm(μ , σ) laskee desimaaliluvun
määrätystä normaalijakaumasta. Luku voi
olla mikä tahansa reaaliuku, mutta se
keskittyy voimakkaasti välille [$\mu-3\cdot\sigma$,
 $\mu+3\cdot\sigma$].

randNorm(μ , σ , *Kokeiden lkm*) listan (list),
joka sisältää *Kokeiden lkm* desimaalilukua
määrätystä normaalijakaumasta.

RandSeed 1147	Done
randNorm(0,1)	0.492541
randNorm(3,4.5)	-3.54356

randPoly()

Luettelo >

randPoly(*Muutt*, *Aste*) ⇒ *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*]) ⇒ *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 <i>list3</i> ={1,2,3,4,5}	Done
Define <i>list4</i> =randSamp(<i>list3</i> ,6)	Done
<i>list4</i>	{2,3,4,3,1,2}

RandSeed

Luettelo >

RandSeed *Luku*

Jos *Luku* = 0 , tämä asettaa siemenluvut satunnaislukugeneraattorin tehdasasetuksiin. Jos *Luku* ≠ 0 , sitä käytetään luomaan kaksi siemenlukua, jotka tallennetaan järjestelmän muuttujiin *siemen1* ja *siemen2*

RandSeed 1147	Done
rand()	0.158206

real()

Luettelo >

real(*Lausl*) ⇒ *lauseke*

Laskee argumentin reaalilukuosan.

Huomaa: Kaikkia määrittelemättömiä muuttujia käsitellään reaalimuuttujina. Katso myös **imag()**, page 93.

real(*Listl*) ⇒ *lista*

Laskee kaikkien alkutekijöiden reaaliosat

real(*Matriisi l*) ⇒ *matriisi*

Laskee kaikkien alkutekijöiden reaaliosat.

real($2+3 \cdot i$)	2
real(z)	z
real($x+i \cdot y$)	x

real($\{a+i \cdot b, 3, i\}$)	$\{a, 3, 0\}$
---------------------------------	---------------

real($\begin{bmatrix} a+i \cdot b & 3 \\ c & i \end{bmatrix}$)	$\begin{bmatrix} a & 3 \\ c & 0 \end{bmatrix}$
--	--

Vectr ► Rect

Huomaa: Voit syöttää tämän operaattorin tietokoneen näppäimistöltä kirjoittamalla $\text{e}>\text{Rect}$.

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 komentoa ainoastaan syöterivin lopussa, eikä se päivitä arvoa *ans*.

Huomaa: Katso myös ► Polar, sivu 138.

kompleksiArvo ► Rect

Näyttää kompleksisen arvon *complexValues* suorakulmaisessa muodossa $a+bi$. Arvolla *complexValue* voi olla mikä tahansa kompleksinen muoto. Syöte re^{θ} aiheuttaa kuitenkin virheen astekulmatilassa.

Huomaa: Polaarisisä syötteessä $(r \angle \theta)$ on käytettävä sulkeita.

$$\left(\left(3 \angle \frac{\pi}{4} \quad \angle \frac{\pi}{6} \right) \right) \text{►Rect}$$

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

$$\left[a \quad \angle b \quad \angle c \right]$$

$$\left[a \cdot \cos(b) \cdot \sin(c) \quad a \cdot \sin(b) \cdot \sin(c) \quad a \cdot \cos(c) \right]$$

Radiaanikulmatilassa:

$$\left(4 \cdot e^{\frac{\pi}{3}} \right) \text{►Rect}$$

$$\frac{\pi}{4 \cdot e^3}$$

$$\left(\left(4 \angle \frac{\pi}{3} \right) \right) \text{►Rect}$$

$$2 + 2 \cdot \sqrt{3} \cdot i$$

Graadikulmatilassa:

$$\left((1 \angle 100) \right) \text{►Rect}$$

 i

Astekulmatilassa:

$$\left((4 \angle 60) \right) \text{►Rect}$$

$$2 + 2 \cdot \sqrt{3} \cdot i$$

Huomaa: Näppäilläksesi merkin \angle valitse se Luettelon symboliluettelosta.

ref()

$\text{ref}(\text{Matriisi}I[, \text{Tol}]) \Rightarrow \text{matriisi}$

Laskee matriisin *Matriisi I* rivi-echelonmuodossa.

$$\text{ref} \left(\begin{pmatrix} -2 & -2 & 0 & -6 \\ 1 & -1 & 9 & -9 \\ -5 & 2 & 4 & -4 \end{pmatrix} \right)$$

$$\begin{pmatrix} 1 & -\frac{2}{5} & -\frac{4}{5} & \frac{4}{5} \\ 0 & 1 & \frac{4}{7} & \frac{11}{7} \\ 0 & 0 & 1 & -\frac{62}{71} \end{pmatrix}$$

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.

$\begin{bmatrix} a & b \\ c & d \end{bmatrix} \rightarrow m1$	$\begin{bmatrix} a & b \\ c & d \end{bmatrix}$
ref(m1)	$\begin{bmatrix} 1 & \frac{d}{c} \\ 0 & 1 \end{bmatrix}$

- Jos käytät painikkeita **ctrl** **enter** tai asetat tilan **Auto or Approximate**, laskenta suoritetaan liukulukuaritmetiikalla.
- Jos *Tol* jätetään pois tai sitä ei käytetä, oletusarvoinen toleranssi lasketaan seuraavasti:
 $5E-14 \cdot \max(\dim(\text{Matriisi } I)) \cdot \text{rowNorm}(\text{Matriisi } I)$

Vältä määrittelemättömiä alkutekijöitä matriisissa *Matriisi I*. Tuloksena saattaa olla odottamattomia vastauksia.

Jos esimerkiksi *a* on määrittelemätön seuraavassa lausekkeessa, näkyviin tulee varoitusviesti, ja vastaus näyttää seuraavalta:

$$\text{ref}\left(\begin{bmatrix} a & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}\right) = \begin{bmatrix} 1 & \frac{1}{a} & 0 \\ & a & \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

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.

$$\text{ref}\left(\begin{bmatrix} a & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}\right) | a=0 = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}$$

Huomaa: Katso myös **rref()**, page 163.

RefreshProbeVars

Päaset mittausarvoon kaikista TI-Basic-ohjelmaan kytketyistä antureista.

StatusVar Value	Tila
<i>statusVar</i> =0	Normaali (jatka ohjelmalla)
<i>statusVar</i> =1	Vernier DataQuest™ on tiedonkeruutilassa. Huomaa: 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™i-laitteen keskiössä.

remain()

Luettelo > 

remain(Laus1, Laus2) ⇒ lauseke

remain(Lista1, Lista2) ⇒ lista

remain(Matriisi1, Matriisi2) ⇒ matriisi

Laskee ensimmäisen argumentin jäännöksen toisen argumentin suhteen seuraavien identtisten yhtälöiden määritelmien mukaisesti:

$\text{remain}(x,0) \quad x$

$\text{remain}(x,y) \quad x-y \cdot \text{Part}(x/y)$

Tämän seurauksena huomaa, että **remain(-x,y) = remain(x,y)**. Vastaus on joko nolla tai samanmerkkinen kuin ensimmäinen argumentti.

Huomaa: Katso myös **mod()**, sivu 120.

$\text{remain}(7,0)$	7
$\text{remain}(7,3)$	1
$\text{remain}(-7,3)$	-1
$\text{remain}(7,-3)$	1
$\text{remain}(-7,-3)$	-1
$\text{remain}(\{12,-14,16\},\{9,7,-5\})$	$\{3,0,1\}$

$\text{remain}\left(\begin{pmatrix} 9 & -7 \\ 6 & 4 \end{pmatrix}, \begin{pmatrix} 4 & 3 \\ 4 & -3 \end{pmatrix}\right)$	$\begin{pmatrix} 1 & -1 \\ 2 & 1 \end{pmatrix}$
--	---

Pyydä

Luettelo > 

Pyydä *promptString*, *muutt* [, *NäytäLippu* [, *statusVar*]]

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

Ohjelmointikomento: Keskeyttää ohjelman ja näyttää valintaruudun, jossa on viesti *kehote*merkkijono sekä syöttöruutu käyttäjän antamaa vastausta varten.

Kun käyttäjä kirjoittaa vastauksen ja klikkaa **OK**-painiketta, syöttöruudun sisältö määrätään muuttujalle *var*.

Jos käyttäjä klikkaa **Cancel**, ohjelma etenee hyväksymättä mitään syötteitä. Ohjelma käyttää muuttujan *muutt* aikaisempaa arvoa, jos *var* on jo määritelty.

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.

Määrittele ohjelma:

Määrittele `pyydä_demo()`=Ohjelma

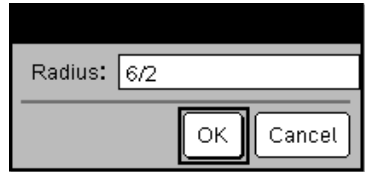
 Pyydä "Säde: ",r

 Disp "Ala = ",pi*r^2

EndPrgm

Aja ohjelma ja kirjoita vastaus:

`Pyydä_demo()`



Vastaus **OK**-näppäimen painamisen jälkeen:

Säde: 6/2

Ala = 28.2743

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 muuttujas*statusVar* 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:

- **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.

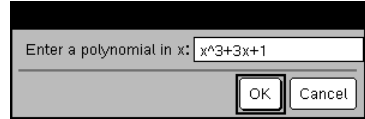
Huomaa: Katso myös **RequestStr**, page 157.

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: $\{-0.322185\}$

RequestStr *kehotemerkkijono, muutt[, NäytäLippu]*

Määrittele ohjelma:

RequestStr



Luettelo > 

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 156.

```
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.

Laske

Luettelo > 

Return [Laus]

Laskee lausekkeen *Laus* funktion tuloksena. Käytetään lohkon **Func...EndFunc** sisällä.

Huomaa: Käytä **Return** ilman argumenttia lohossa **Prgm...EndPrgm** 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()

Luettelo > 

oikea(ListI[, Num]) ⇒ lista

right({1,3,-2,4},3) {3,-2,4}

Laskee oikeanpuoleisimmat *Num*-alkutekijät, jotka sisältyvät listaan *List1*.

Jos jätät pois komennon *Num*, laskee kaiken listasta *List1*.

right(lädemerkkijono[, Num]) ⇒
merkkijono

right("Hello",2) "lo"

Laskee oikeanpuoleisimmat *Num*-merkit, jotka sisältyvät merkkijonoon *lädemerkkijono*

Jos jätät pois komennon *Num*, laskee kaiken *lädemerkkijonosta*.

right(Vertailu) ⇒ *lauseke*

right(x<3) 3

Laskee yhtälön tai epäyhtälön oikean puolen.

rk23 ()

rk23(Laus, Var, depVar, {Var0, VarMax}, depVar0, VarStep [, diftol]) ⇒ *matriisi*

Differentiaaliyhtälö:

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

rk23(SystemOfExpr, Var, ListOfDepVars, {Var0, VarMax}, ListOfDepVars0, VarStep[, diftol]) ⇒ *matriisic*

rk23{0.001·y·(100-y),t,y,{0,100},10,1}

0.	1.	2.	3.	4.
10.	10.9367	11.9493	13.042	14.2

rk23(ListOfExpr, Var, ListOfDepVars, {Var0, VarMax}, ListOfDepVars0, VarStep[, diftol]) ⇒ *matriisi*

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

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 matriisiin, jonka ensimmäinen rivi määrittelee *Var* tulosarvot muuttujan *VarStep* määrittelemällä tavalla. Toinen rivi määrää ensimmäisen ratkaisukomponentin arvon vastaavissa *Var*-arvoissa, jne.

Sama yhtälö, jossa *diftol* on asetettu 1.E-6

rk23{0.001·y·(100-y),t,y,{0,100},10,1,1.E-6}

0.	1.	2.	3.	4.
10.	10.9367	11.9495	13.0423	14.2189

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

deSolve{y'=0.001·y·(100-y) and y(0)=10,t,y}

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

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 *VarStep* on arvoltaan nolasta 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 nolaksi, ratkaisut lasketaan "Runge-Kutta" -muuttujan *Muutt* arvoilla.

diftol on virhetoleranssi (oletuksena 0.001).

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

$$\{10., 10.9367, 11.9494, 13.0423, 14.2189, 15.4\}$$

Yhtälöryhmä:

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

kun $y1(\theta) = 2$ and $y2(\theta) = 5$

$$\text{rk23}\left(\begin{cases} y1 + 0.1 \cdot y1 \cdot y2 \\ 3 \cdot y2 - y1 \cdot y2 \end{cases}, t, \{y1, y2\}, \{0, 5\}, \{2, 5\}, 1\right)$$

0.	1.	2.	3.	4.
2.	1.94103	4.78694	3.25253	1.82848
5.	16.8311	12.3133	3.51112	6.27245

root(Laus) ⇒ juuri

root(Laus1, Laus2) ⇒ juuri

root(Laus) laskee neliöjuuren lausekkeelle *Laus*.

$\sqrt[3]{8}$	2
$\sqrt[3]{3}$	$\frac{1}{3^3}$
$\sqrt[3]{3}$	1.44225

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 6.

rotate()

rotate(*Kokonaisluku1*[,*KiertojenLkm*]) ⇒ *kokonaisluku*

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 22.

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.

0b000000000000111101011000110101

Oikeanpuoleisin bitti siirtyy vasemmaksi.

tuottaa:

0b100000000000011110101100011010

Vastaus näkyy kantaluokituksen mukaisesti.

rotate(*Listal*[,*KiertojenLkm*]) ⇒ *lista*

Laskee *Listal*:nkopion, jota on kierretty oikealle tai vasemmalle *KiertojenLkm*:n elementtien määrittämän määrän. Ei muuta *Listal*.

Binaarisessa kantaluokituksessa:

rotate(0b11111111111111111111111111111111)	
0b10000000000000000000000000000001	►
rotate(256,1)	0b1000000000

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

Heksadesimaalisessa kantaluokituksessa:

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 O-kirjain).

Desimaalisessa kantaluokituksessa:

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()

Luettelo > 

Jos *KiertojenLkm* on positiivinen, kierto on vasemmalle. Jos *KiertojenLkm* on negatiivinen, kierto on oikealle. Oletusarvo on -1 (kierrä oikealle yksi elementti).

rotate(MerkkijonoI[,KiertojenLkm]) \Rightarrow 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).

rotate("abcd")	"dabc"
rotate("abcd",-2)	"cdab"
rotate("abcd",1)	"bcda"

round()

Luettelo > 

round(LausI[, numeroa]) \Rightarrow lauseke

Laskee argumentin pyöristettynä jättäen desimaalipisteen jälkeen määrätyn määrän numeroita.

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(ListaI[, numerot]) \Rightarrow lista

Pyöristää alkutekijöiden listan jättäen desimaalipisteen jälkeen määrätyn määrän numeroita.

round(MatriisiI[, numerot]) \Rightarrow matriisi

Pyöristää alkutekijöiden matriisin jättäen desimaalipisteen jälkeen määrätyn määrän numeroita.

round(1.234567,3)	1.235
-------------------	-------

round($\{\pi, \sqrt{2}, \ln(2)\}, 4$)	{ 3.1416, 1.4142, 0.6931 }
---	----------------------------

round($\begin{bmatrix} \ln(5) & \ln(3) \\ \pi & e^1 \end{bmatrix}, 1$)	$\begin{bmatrix} 1.6 & 1.1 \\ 3.1 & 2.7 \end{bmatrix}$
--	--

rowAdd()

Luettelo >

rowAdd(Matriisi1, rIndeksi1, rIndeksi2)
 \Rightarrow matriisi

Laskee matriisin *Matriisi1* kopion rivirIndeksi2 korvattuna rivien summalla rIndeksi1 ja rIndeksi2.

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

$$\text{rowAdd}\left(\begin{bmatrix} a & b \\ c & d \end{bmatrix}, 1, 2\right) \quad \begin{bmatrix} a & b \\ a+c & b+d \end{bmatrix}$$

rowDim()

Luettelo >

rowDim(Matriisi) \Rightarrow lauseke

Laskee *Matriisin* sisältämien rivien lukumäärän.

Huomaa: Katso myös **colDim()**, sivu 31.

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

$$\text{rowDim}(m1) \quad 3$$

rowNorm()

Luettelo >

rowNorm(Matriisi) \Rightarrow lauseke

Laskee *Matriisin* riveillä olevien elementtien itseisarvojen summien maksimin.

Huomaa: Kaikkien matriisien alkutekijöiden on sievennyttävä luvuiksi. Katso myös **colNorm()**, sivu 31.

$$\text{rowNorm}\left(\begin{bmatrix} -5 & 6 & -7 \\ 3 & 4 & 9 \\ 9 & -9 & -7 \end{bmatrix}\right) \quad 25$$

rowSwap()

Luettelo >

rowSwap(Matriisi1, rIndeksi1, rIndeksi2)
 \Rightarrow matriisi

Laskee matriisin *Matriisi1*, jossa rivien rIndeksi1 ja rIndeksi2 paikkoja on vaihdettu.

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

$$\text{rowSwap}(mat, 1, 3) \quad \begin{bmatrix} 5 & 6 \\ 3 & 4 \\ 1 & 2 \end{bmatrix}$$

rref()


Luettelo >

rref(Matriisi1[, Tol]) \Rightarrow matriisi



Laskee matriisin *Matriisi1* pelkistetyn rivi-echelon-muodon.

$$\text{rref}\left(\begin{bmatrix} -2 & -2 & 0 & -6 \\ 1 & -1 & 9 & -9 \\ -5 & 2 & 4 & -4 \end{bmatrix}\right) \quad \begin{bmatrix} 1 & 0 & 0 & \frac{66}{71} \\ 0 & 1 & 0 & \frac{147}{71} \\ 0 & 0 & 1 & \frac{147}{71} \\ 0 & 0 & 1 & \frac{-62}{71} \end{bmatrix}$$

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 komentoa *Tol* ei huomioida.



$$\text{rref}\left(\begin{pmatrix} a & b \\ c & d \end{pmatrix}\right) \quad \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

- 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:
 $5E-14 \cdot \max(\dim(\text{Matriisi } I)) \cdot \text{rowNorm}(\text{Matriisi } I)$

Huomaa: Katso myös **ref()**, page 153.

S

sec()

 -painike

sec(LausI) ⇒ lauseke

Astekulmatilassa:

sec(ListaI) ⇒ lista

Laskee *LausI*:n sekantin tai määrittää listan, joka sisältää *Listal*:n kaikkien elementtien sekantit.

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

$$\text{sec}(\{1,2,3,4\}) \quad \left\{ \frac{1}{\cos(1)}, 1.00081, \frac{1}{\cos(4)} \right\}$$

Huomaa: Argumentti tulkitaan aste-, graadi- tai radiaanikulmaksi käytössä olevan kulmatila-asetuksen mukaisesti. Voit ohittaa kulmatilan väliaikaisesti painikkeilla $^{\circ}$, $^{\text{G}}$ tai $^{\text{r}}$.

sec⁻¹()

 -painike

sec⁻¹(LausI) ⇒ lauseke

Astekulmatilassa:

sec⁻¹(ListaI) ⇒ lista

$$\text{sec}^{-1}(1) \quad 0$$

sec⁻¹()
 -painike

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 > 
 $\text{sech}(Laus1) \Rightarrow \text{lauseke}$

$$\text{sech}(3) \quad \frac{1}{\cosh(3)}$$

 $\text{sech}(Listal) \Rightarrow \text{lista}$

Laskee *Laus1*:n hyperbolisen sekantin tai määrittää listan, joka sisältää *Listal*:n kaikkien elementtien hyperboliset sekantit.

$$\text{sech}(\{1,2,3,4\}) \quad \left\{\frac{1}{\cosh(1)}, 0.198522, \frac{1}{\cosh(4)}\right\}$$

sech⁻¹()
Katalogi > 
 $\text{sech}^{-1}(Laus1) \Rightarrow \text{lauseke}$

Radiaanikulma- ja

suorakulmakompleksitulassa:

 $\text{sech}^{-1}(Listal) \Rightarrow \text{lista}$

Laskee *Laus1*:n käänteisen hyperbolisen sekantin tai määrittää listan, joka sisältää *Listal*:n kaikkien elementtien käänteiset hyperboliset sekantit.

$$\text{sech}^{-1}(1) \quad 0$$

$$\text{sech}^{-1}(\{1,2,2.1\}) \quad \left\{0, \frac{2 \cdot \pi}{3} \cdot i, 8. \text{E}^{-15} + 1.07448 \cdot i\right\}$$

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

Send *lausTaiMerkkijono1* [, *lausTaiMerkkijono2*] ...

Ohjelmointikomento: Lähettää 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 85), **GetStr** (sivu 89), ja **eval()** (sivu 68).

Esimerkki: Sytytä sisäänrakennetun RGB-ledin sininen väri 0,5 sekunniksi.

```
Send "SET COLOR.BLUE ON TIME .5"
Done
```

Esimerkki: Kysy laitteen sisäänrakennetun valotasoanturin tämänhetkinen arvo. **Get**-komento noutaa arvon ja sijoittaa sen muuttujaan *lightval*.

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

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
iostr.SendAns "SET SOUND 200"
```

seq() (sekv)

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.

Katalogi >

$\text{seq}\left(n^2, n, 1, 6\right)$	$\{1, 4, 9, 16, 25, 36\}$
$\text{seq}\left(\frac{1}{n}, n, 1, 10, 2\right)$	$\left\{1, \frac{1}{3}, \frac{1}{5}, \frac{1}{7}, \frac{1}{9}\right\}$
$\text{sum}\left(\text{seq}\left(\frac{1}{n^2}, n, 1, 10, 1\right)\right)$	$\frac{1968329}{1270080}$

Huom: Vastauksen pakottaminen likimääräiseksi:

Kämmenlaite: Paina  .

Windows®: Paina **Ctrl+Enter**.

Macintosh®: Paina **⌘+Enter**.

iPad®: Pidä **enter** ja valitse .

$$\text{sum}\left(\text{seq}\left(\frac{1}{n^2}, n, 1, 10, 1\right)\right) \quad 1.54977$$

seqGen()

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 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 tallella funktion **seqGen()** suorittamisen jälkeen

Oletusarvo *VarStep* = 1.

Luo sekvenssin 5 ensimmäistä termiä $u(n) = u(n-1)^2/2$, kun $u(1)=2$ ja $VarStep=1$.

$$\text{seqGen}\left(\frac{\{u(n-1)\}^2}{n}, n, u, \{1,5\}, \{2\}\right) \\ \left\{2,2, \frac{4}{3}, \frac{4}{9}, \frac{16}{405}\right\}$$

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:

$$\text{seqGen}\left\{\left\{\frac{1}{n}, \frac{u2(n-1)}{2} + u1(n-1)\right\}, n, \{u1, u2\}, \{1,5\}, \left[-\right]\right\} \\ \begin{bmatrix} 1 & \frac{1}{2} & \frac{1}{3} & \frac{1}{4} & \frac{1}{5} \\ 2 & 2 & \frac{3}{2} & \frac{13}{12} & \frac{19}{24} \end{bmatrix}$$

Huomaa: Muuttujaa Tyhjä () yllä olevassa aloitustermimatriisissa käytetään ilmoittamaan, että $u1(n)$:n aloitustermi on laskettu käyttämällä täsmällistä sekvenssikaavaa $u1(n)=1/n$.

seqn()

Katalogi >

seqn(*Expr*(*u*, *n* [, *ListOfInitTerms* [, *nMax* [, *CeilingValue*]]]) \Rightarrow *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*]]) \Rightarrow *list*

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 $nstep=1$

Luo sekvenssin 6 ensimmäistä termiä $u(n) = u(n-1)/2$, kun $u(1)=2$.

$$\text{seqn}\left(\frac{u(n-1)}{n}, \{2\}, 6\right) \quad \left\{2, 1, \frac{1}{3}, \frac{1}{12}, \frac{1}{60}, \frac{1}{360}\right\}$$

$$\text{seqn}\left(\frac{1}{n^2}, 6\right) \quad \left\{1, \frac{1}{4}, \frac{1}{9}, \frac{1}{16}, \frac{1}{25}, \frac{1}{36}\right\}$$

series()

Katalogi >

series(*LausI*, *Muutt*, *Aste* [, *Piste*])
 \Rightarrow *muuttuja*

series(*LausI*, *Muutt*, *Aste* [, *Piste*]) |
Muutt>*Piste* \Rightarrow *lauseke*

series(*LausI*, *Muutt*, *Aste* [, *Piste*]) |
Muutt<*Piste* \Rightarrow *lauseke*

$$\text{series}\left(\frac{1-\cos(x-1)}{(x-1)^2}, x, 4, 1\right) \quad \frac{1}{2} \frac{(x-1)^2}{24} + \frac{(x-1)^4}{720}$$

$$\text{series}\left(\frac{-1}{e^z}, z, 1\right) \quad z, -1$$

$$\text{series}\left(\left(1 + \frac{1}{n}\right)^n, n, 2, \infty\right) \quad e - \frac{e}{2 \cdot n} + \frac{11 \cdot e}{24 \cdot n^2}$$

Laskee yleisen tyypistetyn 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 ∞ tai $-\infty$, jolloin lavennus tapahtuu asteen *Aste* kautta lausekkeessa $1/(\text{Muutt} - \text{Piste})$.

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 hyppyepäjatkuvuus kohdassa *Piste*, tulos sisältää todennäköisesti alalausekkeita, jotka ovat muotoa $\text{sign}(\dots)$ tai $\text{abs}(\dots)$ reaalille kehittelmän muuttujalle tai $(-1)^{\text{floor}(\dots \cdot \text{angle}(\dots))}$ kompleksille kehittelmän muuttujalle, joka on merkkiin " $_$ " päättyvä muuttuja. Jos tarkoituksesi on käyttää sarjaa vain *Piste*en 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.

series($\tan^{-1}\left(\frac{1}{x}\right), x, 5$), $x > 0$	$\frac{\pi}{2} - x + \frac{x^3}{3} - \frac{x^5}{5}$
series($\int \frac{\sin(x)}{x} dx, x, 6$)	$x - \frac{x^3}{18} + \frac{x^5}{600}$
series($\int_0^x \sin(x \cdot \sin(t)) dt, x, 7$)	$\frac{x^3}{2} - \frac{x^5}{24} + \frac{29 \cdot x^7}{720}$
series($(1 + e^x)^2, x, 2, 1$)	$(e+1)^2 + 2 \cdot e \cdot (e+1) \cdot (x-1) + e \cdot (2 \cdot e+1) \cdot (x-1)^2$

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 62.

setMode()

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.

Näytä π :n likiarvo käyttäen Näytettävät numerot-tilan oletusasetusta ja näytä sen jälkeen π asetuksella Kiinteä2. Tarkista, että oletusarvo palautuu ohjelman suorittamisen jälkeen.

Define <i>prog1()</i> =Prgm	Done
Disp approx(π)	
setMode(1,16)	
Disp approx(π)	
EndPrgm	
<i>prog1()</i>	3.14159
	3.14
	Done

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 87.

Huomaa: Nykyiset tila-asetukset siirtyvät haettuihin alarutiineihin. Jos jokin alarutiini muuttaa tila-asetusta, tilamuutos häviää, kun ohjaus palautuu hakurutiiniin.

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(KokonaislukuI[,SiirtojenLkm])
 \Rightarrow kokonaisluku

Siirtää binaarisen kokonaisluvun bittejä. Voit syöttää *KokonaislukuI*:n minä tahansa kantalukena; se muunnetaan automaattisesti etumerkilliseen 64 bitin binaarimuotoon. Jos *KokonaislukuI* on liian suuri tälle muodolle, symmetrinen modulo-operaatio sovittaa sen alueelle sopivaksi. Lisätietoja, katso ▶**Base2**, sivu 22.

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 bittiiä. Vasemmalle tapahtuvassa siirrossa vasemmanpuoleisin bitti pudotetaan, ja 0 lisätään vastaamaan oikeanpuoleisinta bittiiä.

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(ListaI [,SiirtojenLkm]) \Rightarrow lista

Luo *Listal*:n kopion, jota on siirretty oikealle tai vasemmalle *SiirtojenLkm*:n elementtien määrittämän määrän. Ei muuta *Listal*:ä.

Binaarisessa kantalukutilassa:

shift(0b1111010110000110101)	0b111101011000011010
shift(256,1)	0b1000000000

Heksadesimaalisessa kantalukutilassa:

shift(0h78E)	0h3C7
shift(0h78E,-2)	0h1E3
shift(0h78E,2)	0h1E38

Tärkeää: Binaariluvun edelle tulee aina merkitä etumerkki 0b ja heksadesimaaliluvun edelle 0h (nolla, ei O-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}

Jos *SiirtojenLkm* on positiivinen, siirto tapahtuu vasemmalle. Jos *SiirtojenLkm* on negatiivinen, siirto tapahtuu oikealle. Oletusarvo on -1 (siirrä yksi elementti oikealle).

Elementit, jotka siirto vie *listan* alkuun tai loppuun, asettuvat symboliksi "undef".

shift(Merkkijono1 [,SiirtojenLkm])
 \Rightarrow merkkijono

Luo *Merkkijono1*:n kopion, jota on siirretty oikealle tai vasemmalle *SiirtojenLkm*:n merkkien määrittämän määrän. Ei muuta *Merkkijono1*: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(Laus1) \Rightarrow lauseke

sign(-3.2)	-1.
sign({2,3,4,-5})	{1,1,1,-1}
sign(1+ x)	1

sign(Lista1) \Rightarrow lista

sign(Matriisi1) \Rightarrow matriisi

Kun kyseessä on reaali- tai kompleksilukulauseke *Laus1*, antaa vastauksena *Laus1/abs(Laus1)*, kun *Laus1* \neq 0.

Jos kompleksilukutila on Reaali:

sign([-3 0 3])	[-1 ±1 1]
----------------	-----------

Vastaus on 1, jos *Laus1* on positiivinen. Vastaus on -1, jos *Laus1* on negatiivinen.

sign(0) edustaa kompleksialueen yksikköpiiriä.

Kun kyseessä on lista tai matriisi, määrittää kaikkien elementtien etumerkit.

simult()

simult(kerroinMatriisi, vakioVektori[, tol]) ⇒ matriisi

Määrittää sarakevektorin, joka sisältää lineaarisen yhtälöryhmän ratkaisut.

Huomaa: Katso myös **linSolve()**, sivu 106.

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:
 $5E-14 \cdot \max(\dim(\text{kerroinMatriisi}), \text{rowNorm}(\text{kerroinMatriisi}))$

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:

$$x + 2y = 1$$

$$3x + 4y = -1$$

$$\text{simult}\left(\begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}, \begin{pmatrix} 1 \\ -1 \end{pmatrix}\right) \quad \begin{matrix} -3 \\ 2 \end{matrix}$$

Ratkaisu on $x = -3$ and $y = 2$.

Ratkaise:

$$ax + by = 1$$

$$cx + dy = 2$$

$$\begin{matrix} \begin{matrix} a & b \\ c & d \end{matrix} \rightarrow \text{matx1} & \begin{matrix} a & b \\ c & d \end{matrix} \\ \text{simult}\left(\text{matx1}, \begin{pmatrix} 1 \\ 2 \end{pmatrix}\right) & \begin{matrix} -(2 \cdot b - d) \\ a \cdot d - b \cdot c \\ 2 \cdot a - c \\ a \cdot d - b \cdot c \end{matrix} \end{matrix}$$

Ratkaise:

$$x + 2y = 1$$

$$3x + 4y = -1$$

$$x + 2y = 2$$

$$3x + 4y = -3$$

$$\text{simult}\left(\begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}, \begin{pmatrix} 1 & 2 \\ -1 & -3 \end{pmatrix}\right) \quad \begin{matrix} -3 & -7 \\ 2 & 9 \\ 2 & 2 \end{matrix}$$

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.

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

$\sin(LausI) \Rightarrow lauseke$

Astekulmatilassa:

$\sin(ListaI) \Rightarrow lista$

$$\frac{\sin\left(\frac{\pi}{4}\right)}{\sin(45)} = \frac{\sqrt{2}}{2}$$

$\sin(LausI)$ määrittää argumentin sinin lausekkeena.

$$\frac{\sin(\{0,60,90\})}{2} = \frac{\sqrt{2}}{2}$$

$\sin(ListaI)$ 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 $^{\circ}$, $^{\text{G}}$ tai $^{\text{r}}$.

Graadikulmatilassa:

sin() -painike

$$\sin(50) \quad \frac{\sqrt{2}}{2}$$

Radiaanikulmatilassa:

$$\sin\left(\frac{\pi}{4}\right) \quad \frac{\sqrt{2}}{2}$$

$$\sin(45^\circ) \quad \frac{\sqrt{2}}{2}$$

sin(neliömatriisi I) ⇒ *neliömatriisi*

Määrittää *neliömatriisi I*:n matriisin sinin. Tämä ei ole sama kuin kunkin elementin sinin laskeminen. Laskentamenetelmä on kuvattu kohdassa **cos()**.

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

Radiaanikulmatilassa:

$$\sin\left(\begin{pmatrix} 1 & 5 & 3 \\ 4 & 2 & 1 \\ 6 & -2 & 1 \end{pmatrix}\right) \quad \begin{bmatrix} 0.9424 & -0.04542 & -0.031999 \\ -0.045492 & 0.949254 & -0.020274 \\ -0.048739 & -0.00523 & 0.961051 \end{bmatrix}$$

sin⁻¹() -painike**sin⁻¹(Laus I)** ⇒ *lauseke***sin⁻¹(Lista I)** ⇒ *lista*

sin⁻¹(Laus I) määrittää lausekkeena kulman, jonka sini on *Laus I*.

sin⁻¹(Lista I) määrittää listan, joka sisältää *Lista I*: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(neliömatriisi I) ⇒ *neliömatriisi*

Laskee *neliömatriisi I*:n matriisin käänteissinin. Tämä ei ole sama kuin kunkin elementin käänteissinin laskeminen. Laskentamenetelmä on kuvattu kohdassa **cos()**.

Astekulmatilassa:

$$\sin^{-1}(1) \quad 90$$

Graadikulmatilassa:

$$\sin^{-1}(1) \quad 100$$

Radiaanikulmatilassa:

$$\sin^{-1}(\{0,0,2,0,5\}) \quad \{0,0,201358,0,523599\}$$

Radiaanikulmatilassa ja

suorakulmakompleksimuodossa:

$$\sin^{-1}\left(\begin{pmatrix} 1 & 5 \\ 4 & 2 \end{pmatrix}\right) \quad \begin{bmatrix} -0.174533-0.12198 \cdot i & 1.74533-2.35591 \cdot i \\ 1.39626-1.88473 \cdot i & 0.174533-0.593162 \cdot i \end{bmatrix}$$

sin⁻¹()

trig -painike

neliömatriisi l:n on oltava diagonalisoitavissa. Vastaus sisältää aina liukulukuja.

sinh()

Katalogi >

sinh(Laus l) ⇒ lauseke

sinh(1.2) 1.50946

sinh(Lista l) ⇒ lista

sinh({0,1.2,3}) {0,1.50946,10.0179}

sinh (Laus l) laskee argumentin hyperbolisen kosinin lausekkeena.

sinh (Lista l) määrittää listan, joka sisältää *Listal*:n jokaisen elementin hyperbolisen sinin.

sinh(neliömatriisi l) ⇒ neliömatriisi

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

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

neliömatriisi l:n on oltava diagonalisoitavissa. Vastaus sisältää aina liukulukuja.

sinh⁻¹()

Katalogi >

sinh⁻¹(Laus l) ⇒ lausekesinh⁻¹(0) 0**sinh⁻¹(Lista l)** ⇒ listasinh⁻¹({0,2,1,3}) {0,1.48748,sinh⁻¹(3)}

sinh⁻¹(Laus l) laskee argumentin käänteisen hyperbolisen sinin lausekkeena.

sinh⁻¹(Lista l) määrittää listan, joka sisältää *Listal*:n jokaisen elementin käänteiset hyperboliset sinin.

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

sinh⁻¹(neliömatriisi l) ⇒ neliömatriisi

Radiaanikulmatilassa:

Lisätietoja listassa olevien tyhjien elementtien vaikutuksesta, katso Tyhjät elementtisivulla sivu 248.

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	Muokatusn <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	Muokatusn <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()

solve(*Yhtälö*, *Muutt*) \Rightarrow *Boolean lauseke*

solve(*Yhtälö*, *Muutt*=*Arvaus*) \Rightarrow *Boolean lauseke*

solve(*Epäyhtälö*, *Muutt*) \Rightarrow *Boolean lauseke*

Laskee yhtälön tai epäyhtälön mahdollisia reaali- ja kompleksilukuratkaisuja 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.

$$\text{solve}(a \cdot x^2 + b \cdot x + c = 0, x)$$

$$x = \frac{-b \pm \sqrt{b^2 - 4 \cdot a \cdot c}}{2 \cdot a} \text{ or } x = \frac{-b \pm \sqrt{b^2 - 4 \cdot a \cdot c} \cdot i}{2 \cdot a}$$

$$\text{Ans}|a=1 \text{ and } b=1 \text{ and } c=1$$

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

$$\text{solve}((x-a) \cdot e^x = x \cdot (x-a), x)$$

$$x = a \text{ or } x = -0.567143$$

Koska suurin yhteinen tekijä peruutetaan oletusarvoisesti suhdelukujen osoittajasta ja nimittäjästä, ratkaisut voivat olla ratkaisuja, jotka ovat ainoastaan yhden tai kummankin puolen rajalla.

Täsmälliset ratkaisut ovat epätodennäköisiä tyyppiä \geq , \leq , $<$ tai $>$ oleville epäyhtälöille, ellei epäyhtälö ole lineaarinen ja sisältää ainoastaan muuttujan *Muutt.*

Exact (Täsmällinen) -tilassa osat, joita ei voida ratkaista, annetaan implisiittisenä yhtälönä tai epäyhtälönä.

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.

jos reaaliurakuratkaisuja 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.

Koska **solve()** tuottaa aina Boolean vastauksen, komentojen "and", "or" ja "not" avulla voit yhdistää **solve()**-funktion tuloksia toisiinsa tai muihin Boolean lausekkeisiin.

Ratkaisut voivat sisältää uuden yksilöllisen määrittämättömän vakion, joka on muotoa *nj*, jossa *j* on kokonaisluku väliltä 1-255. Tällaiset muuttujat tarkoittavat mielivaltaista kokonaislukua.

Reaaliurakutilassa murtolukupotenssit, joilla on pariton nimittäjä, viittaavat ainoastaan reaaliurakalueeseen. Muussa tapauksessa haaralausekkeet, kuten murtolukupotenssit, logaritmit ja käänteiset trigonometriset funktiot, viittaavat vain päähaaraan. Näin ollen **solve()** tuottaa vain ratkaisuja, jotka vastaavat tätä yhtä reaaliuraku- tai päähaaraa.

$$(x+1) \cdot \frac{x-1}{x-1} + x - 3 \quad 2 \cdot x - 2$$

$$\text{solve}(5 \cdot x - 2 \geq 2 \cdot x, x) \quad x \geq \frac{2}{3}$$

$$\text{exact}(\text{solve}((x-a) \cdot e^x = x \cdot (x-a), x)) \quad e^x + x = 0 \text{ or } x = a$$

Radiaanikulmatilassa:

$$\text{solve}\left(\tan(x) = \frac{1}{x}, x\right) | x > 0 \text{ and } x < 1 \quad x = 0.860334$$

$$\text{solve}(x = x + 1, x) \quad \text{false}$$

$$\text{solve}(x = x, x) \quad \text{true}$$

$$2 \cdot x - 1 \leq 1 \text{ and solve}(x^2 \neq 9, x) \quad x \neq -3 \text{ and } x \leq 1$$

Radiaanikulmatilassa:

$$\text{solve}(\sin(x) = 0, x) \quad x = n \cdot \pi$$

$$\text{solve}\left(x^{\frac{1}{3}} = -1, x\right) \quad x = -1$$

$$\text{solve}(\sqrt{x} = 2, x) \quad \text{false}$$

$$\text{solve}(-\sqrt{x} = 2, x) \quad x = 4$$

Huomaa: Katso myös `cSolve()`, `cZeros()`, `nSolve()` ja `zeros()`.

`solve(Yht1 and Yht2 [and...], MuuttTaiArvaus1, MuuttTaiArvaus2 [, ...])` ⇒ Boolean lauseke

`solve(Yht1öryhmä, MuuttTaiArvaus1, MuuttTaiArvaus2 [, ...])` ⇒ Boolean lauseke

`solve({Yht1, Yht2 [,...]} {MuuttTaiArvaus1, MuuttTaiArvaus2 [, ...]})`
⇒ Boolean lauseke

Antaa ehdotuksia reaalityyppisistä 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 = reaalityyppi tai ei-reaalityyppi

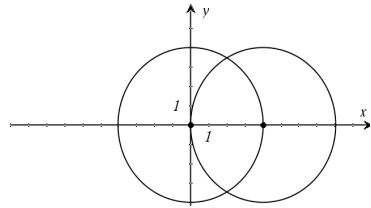
Esimerkiksi x kelpaa ja samoin $x=3$.

Jos kaikki yhtälöt ovat polynomeja, ja jos ET määrittää ensimmäisiä arvauksia, `solve()` käyttää leksikaalista Gröbner/Buchbergerin eliminaatiomenetelmää yrittäessään määrittää kaikki reaalityyppiset ratkaisut.

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 `solve()`-funktion avulla.

$$\text{solve}(y=x^2-2 \text{ and } x+2y=1, \{x,y\})$$

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



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 kokonaislukualue 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.

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 reaaliuratkaisuja.

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.

$$\text{solve}\left(x^2+y^2=r^2 \text{ and } (x-r)^2+y^2=r^2, \{x,y\}\right)$$

$$x=\frac{r}{2} \text{ and } y=\frac{\sqrt{3}\cdot r}{2} \text{ or } x=\frac{r}{2} \text{ and } y=\frac{-\sqrt{3}\cdot r}{2}$$

$$\text{solve}\left(x^2+y^2=r^2 \text{ and } (x-r)^2+y^2=r^2, \{x,y,z\}\right)$$

$$x=\frac{r}{2} \text{ and } y=\frac{\sqrt{3}\cdot r}{2} \text{ and } z=c1 \text{ or } x=\frac{r}{2} \text{ and } y\rightarrow$$

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

$$\text{solve}\left(x+e^z\cdot y=1 \text{ and } x-y=\sin(z), \{x,y\}\right)$$

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

$$\text{solve}\left(e^z\cdot y=1 \text{ and } -y=\sin(z), \{y,z\}\right)$$

$$y=2.812\text{E}-10 \text{ and } z=21.9911 \text{ or } y=0.00187\text{I}$$

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

$$\text{solve}\left(e^z\cdot y=1 \text{ and } -y=\sin(z), \{y,z=2\cdot\pi\}\right)$$

$$y=0.001871 \text{ and } z=6.28131$$

Arvausten avulla voit etsiä lisäratkaisuja yksi kerrallaan. Suppenemista varten arvauksen on mahdollisesti oltava melko lähellä ratkaisua.

SortA

SortA *Listal* [, *Lista2*] [, *Lista3*] ...

SortA *Vektori1* [, *Vektori2*] [, *Vektori3*] ...

Lajittelee ensimmäisen argumentin elementit nousevaan järjestykseen.

Jos otat mukaan lisääargumentteja, 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 elementistä, katso sivu 248.

$\{2,1,4,3\} \rightarrow list1$	$\{2,1,4,3\}$
SortA <i>list1</i>	Done
<i>list1</i>	$\{1,2,3,4\}$
$\{4,3,2,1\} \rightarrow list2$	$\{4,3,2,1\}$
SortA <i>list2,list1</i>	Done
<i>list2</i>	$\{1,2,3,4\}$
<i>list1</i>	$\{4,3,2,1\}$

SortD

SortD *Listal* [, *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 elementistä, katso sivu 248.

$\{2,1,4,3\} \rightarrow list1$	$\{2,1,4,3\}$
$\{1,2,3,4\} \rightarrow list2$	$\{1,2,3,4\}$
SortD <i>list1,list2</i>	Done
<i>list1</i>	$\{4,3,2,1\}$
<i>list2</i>	$\{3,4,1,2\}$

►Sphere

Vektori ►**Sphere**

Huom: Vastauksen pako ttaminen likimääräiseksi:

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.

Kämmenlaite: Paina .

Windows®: Paina **Ctrl+Enter**.

Macintosh®: Paina **⌘+Enter**.

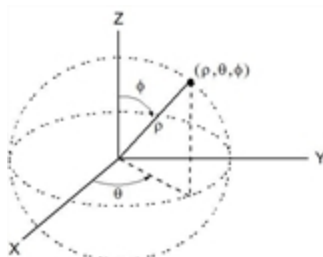
iPad®: Pidä **enter** ja valitse .

$$\begin{bmatrix} 1 & 2 & 3 \end{bmatrix} \text{►Sphere} \\ \left[3.74166 \quad \angle 1.10715 \quad \angle 0.640522 \right]$$

$$\begin{pmatrix} 2 & \angle \frac{\pi}{4} & 3 \end{pmatrix} \text{►Sphere} \\ \left[3.60555 \quad \angle 0.785398 \quad \angle 0.588003 \right]$$

Paina

$$\begin{pmatrix} 2 & \angle \frac{\pi}{4} & 3 \end{pmatrix} \text{►Sphere} \\ \left[\sqrt{13} \quad \angle \frac{\pi}{4} \quad \angle \sin^{-1} \left(\frac{2 \cdot \sqrt{13}}{13} \right) \right]$$



sqrt()

sqrt(Lausl) ⇒ lauseke

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

sqrt(Listal) ⇒ lista

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

Laskee argumentin neliöjuuren.

Kun kyseessä on lista, laskee kaikkien Listal:n elementtien neliöjuuret.

Huomaa: Katso myös Neliöjuurimalli, sivu 5.

stat.results

 $xlist = \{1, 2, 3, 4, 5\}$ $\{1, 2, 3, 4, 5\}$

Näyttää tilastollisen laskutoimituksen tulokset.

 $ylist = \{4, 8, 11, 14, 17\}$ $\{4, 8, 11, 14, 17\}$

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 ² "	0.996109
"r"	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 tilastoanalyysissä käytettävillä muuttujilla. Joissakin tapauksissa tästä voi olla seurauksena virhetilanne. Tilastoanalyysissä käytettävät muuttujanimet on esitetty alla olevassa taulukossa.

stat.a	stat.dfDenom	stat.MedianY	stat.Q3X	stat.SSBlock
stat.AdjR ²	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 ²	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 _x	stat.SSRow
stat.b5	stat.DW	stat.MSInteract	stat.σ _y	stat.tList
stat.b6	stat.e	stat.MSReg	stat.σ _{x1}	stat.UpperPred
stat.b7	stat.ExpMatrix	stat.MSRow	stat.σ _{x2}	stat.UpperVal
stat.b8	stat.F	stat.n	stat.Σ _x	stat.X̄
stat.b9	stat.FBlock	Stat. \hat{p}	stat.Σ _{x²}	stat.X̄ ₁
stat.b10	stat.Fcol	stat. \hat{p}_1	stat.Σ _{xy}	stat.X̄ ₂
stat.bList	stat.FInteract	stat. \hat{p}_2	stat.Σ _y	stat.X̄Diff
stat.χ ²	stat.FreqReg	stat. \hat{p} Diff	stat.Σ _{y²}	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.ŷ
stat.CookDist	stat.MaxX	stat.PValRow	stat.SESlope	stat.ŷList

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

Katalogi >

stat.values

Katso esimerkki kohdassa
stat.results.

Näyttää matriisin 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()

Katalogi >

stDevPop(Lista[, frekvLista]) ⇒ *lauseke*

Radiaanikulma- ja automaattisissa tiloissa:

Laskee *Lista*:n sisältämien elementtien perusjoukon keskihajonnan.

$$\text{stDevPop}\{\{a,b,c\}\}$$

$$\frac{\sqrt{2 \cdot (a^2 - a \cdot (b+c) + b^2 - b \cdot c + c^2)}}{3}$$

$$\text{stDevPop}\{\{1,2,5,-6,3,-2\}\} \quad \frac{\sqrt{465}}{6}$$

$$\text{stDevPop}\{\{1.3,2.5,-6.4\},\{3,2,5\}\} \quad 4.11107$$

Jokainen *frekvLista*n elementti näyttää *Listan* vastaavien elementtien peräkkäisten esiintymien lukumäärän.

Huomaa: *Listassa* tulee olla vähintään kaksi elementtiä. Tyhjiä elementtejä ei huomioida. Lisätietoja tyhjästä elementeistä, katso sivu 248.

stDevPop(MatriisiI[, frekvMatriisi])
⇒ *matriisi*

$$\text{stDevPop}\left(\begin{bmatrix} 1 & 2 & 5 \\ -3 & 0 & 1 \\ 5 & 7 & 3 \end{bmatrix}\right) \quad \begin{bmatrix} 4 \cdot \sqrt{6} & \sqrt{78} & 2 \cdot \sqrt{6} \\ 3 & 3 & 3 \end{bmatrix}$$

$$\text{stDevPop}\left(\begin{bmatrix} -1.2 & 5.3 \\ 2.5 & 7.3 \\ 6 & -4 \end{bmatrix}\right) \quad \begin{bmatrix} 4 & 2 \\ 3 & 3 \\ 1 & 7 \end{bmatrix}$$

$$\quad \quad \quad [2.52608 \quad 5.21506]$$

Laskee rivivektorin *MatriisiI*:n sarakkeiden perusjoukon keskihajonnoista.

Jokainen *frekvMatriisin* elementi näyttää *Matriisi1*:n vastaavien elementtien peräkkäisten esiintymien lukumäärän.

Huomaa: *Matriisi1* :ssä on oltava vähintään kaksi riviä. Tyhjiä elementtejä ei huomioida. Lisätietoja tyhjistä elementeistä, katso sivu 248.

stDevSamp()

stDevSamp(Lista[,frekvLista]) ⇒ lauseke

Laskee *Listan* sisältämien elementtien otoksen keskihajonnan.

Jokainen *frekvListan* elementi näyttää *Listan* vastaavien elementtien peräkkäisten esiintymien lukumäärän.

Huomaa: *Listassa* tulee olla vähintään kaksi elementtiä. Tyhjiä elementtejä ei huomioida. Lisätietoja tyhjistä elementeistä, katso sivu 248.

stDevSamp(Matriisi1[,frekvMatriisi])
⇒ *matriisi*

Laskee rivivektorin *Matriisi1*:n sarakkeiden otosten keskihajonnoista.

Jokainen *frekvMatriisin* elementi näyttää *Matriisi1*:n vastaavien elementtien peräkkäisten esiintymien lukumäärän.

Huomaa: *Matriisi1* :ssä on oltava vähintään kaksi riviä. Tyhjiä elementtejä ei huomioida. Lisätietoja tyhjistä elementeistä, katso sivu 248.

$$\begin{aligned} \text{stDevSamp}(\{a,b,c\}) &= \frac{\sqrt{3 \cdot (a^2 - a \cdot (b+c) + b^2 - b \cdot c + c^2)}}{3} \\ \text{stDevSamp}(\{1,2,5,-6,3,-2\}) &= \frac{\sqrt{62}}{2} \\ \text{stDevSamp}(\{1.3,2.5,-6.4\},\{3,2,5\}) &= 4.33345 \end{aligned}$$

$$\begin{aligned} \text{stDevSamp}\left(\begin{pmatrix} 1 & 2 & 5 \\ -3 & 0 & 1 \\ 5 & 7 & 3 \end{pmatrix}\right) &= \begin{bmatrix} 4 & \sqrt{13} & 2 \end{bmatrix} \\ \text{stDevSamp}\left(\begin{pmatrix} -1.2 & 5.3 \\ 2.5 & 7.3 \\ 6 & -4 \end{pmatrix}, \begin{bmatrix} 4 & 2 \\ 3 & 3 \\ 1 & 7 \end{bmatrix}\right) &= \begin{bmatrix} 2.7005 & 5.44695 \end{bmatrix} \end{aligned}$$

Stop

Katalogi >

Stop

Ohjelmointikomento: Pysäyttää ohjelman.

Stop ei ole sallittu funktioissa.

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

$i:=0$	0
Define $prog1()$ =Prgm	Done
For $i,1,10,1$	
$i=5$	
Stop	
EndFor	
EndPrgm	
$prog1()$	Done
i	5

Store

Katso → (tallenna), sivu 245.

string() (merkkijono)

Katalogi >

 $string(Laus) \Rightarrow merkkijono$ Sieventää lausekkeen *Laus* ja antaa vastauksen merkkijonona.

$string(1.2345)$	"1.2345"
$string(1+2)$	"3"
$string(\cos(x)+\sqrt{3})$	"cos(x)+√(3)"

subMat()

Katalogi >

$subMat(Matriisi1[, alkurivi] [, alkusarake] [, loppurivi] [, loppusarake]) \Rightarrow matriisi$

Laskee *Matriisi1*:n määritetyn alimatriisin.

Oletusarvot: *alkurivi*=1, *alkusarake*=1, *loppurivi*=viimeinen rivi, *endCol*=viimeinen sarake.

$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix} \rightarrow m1$	$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$
$subMat(m1,2,1,3,2)$	$\begin{bmatrix} 4 & 5 \\ 7 & 8 \end{bmatrix}$
$subMat(m1,2,2)$	$\begin{bmatrix} 5 & 6 \\ 8 & 9 \end{bmatrix}$

Sum (Sigma)Katso $\Sigma()$, sivu 235.

sum()Katalogi > **sum(Lista[, Alku[, Loppu]])** ⇒ 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 tyhjistä elementeistä, katso sivu 248.**sum(MatriisiI[, Alku[, Loppu]])** ⇒ matriisiLaskee rivivektorin, joka sisältää *MatriisiI*:n sarakkeiden elementtien summat.*Alku* ja *Loppu* ovat valinnaisia. Ne määrittävät rivialueen.Mikä tahansa tyhjä argumentti tuottaa tyhjän vastauksen. *MatriisiI*:ssä olevia tyhjiä elementtejä ei huomioida. Lisätietoja tyhjistä elementeistä, katso sivu 248.

$\text{sum}\{1,2,3,4,5\}$	15
$\text{sum}\{a,2\cdot a,3\cdot a\}$	$6\cdot a$
$\text{sum}(\text{seq}(n,n,1,10))$	55
$\text{sum}\{1,3,5,7,9\},3\}$	21

$\text{sum}\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{pmatrix}$	$[5 \ 7 \ 9]$
$\text{sum}\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix}$	$[12 \ 15 \ 18]$
$\text{sum}\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix},2,3$	$[11 \ 13 \ 15]$

sumIf()Katalogi > **sumIf(Lista,Kriteerit[, SummaLista])**
⇒ arvoLaskee kaikkien niiden *Listan* sisältämien elementtien kumuloituneen summan, jotka vastaavat määritettyjä kriteereitä *Kriteerit*. Voit halutessasi antaa kumuloivat elementit määrittämällä vaihtoehdoisen listan, *summaLista*.*List*a voi olla lauseke, lista tai matriisi. *SummaListalla*, mikäli se määritetään, on oltava samat mitat kuin *Listalla*.*Kriteeri* voi olla:

- Arvo, lauseke tai merkijono. Esimerkiksi **34** kumuloi vain niitä *Listan* elementtejä, jotka sieventyvät arvoon 34.
- Boolean lauseke, joka sisältää symbolin ? kunkin elementin paikanpitäjänä. Esimerkiksi lauseke **?<10** kumuloi vain

$\text{sumIf}\{1,2,e,3,\pi,4,5,6\},2.5<?<4.5\}$	$e+\pi+7$
$\text{sumIf}\{1,2,3,4\},2<?<5,\{10,20,30,40\}\}$	70

niitä *Listan* elementtejä, jotka ovat alle 10.

Kun jokin *Listan* elementti vastaa kriteereitä *Kriteerit*, elementti lisätään kumuloituvaan summaan. Jos sisällyttät funktioon *summaListan*, summaan lisätäänkin sen sijaan vastaava *summaListan* elementti.

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

Tyhjiä elementtejä ei huomioida. Lisätietoja tyhjihistä elementeistä, katso sivu 248.

Huomaa: Katso myös `countIf()`, sivu 41.

`system(Yht1 [, Yht2 [, Yht3 [, ...]])`

$$\text{solve} \left\{ \begin{array}{l} x+y=0 \\ x-y=8 \end{array} \right. \quad x=4 \text{ and } y=-4$$

`system(Laus1 [, Laus2 [, Laus3 [, ...]])`

Laskee yhtälöryhmän listaksi muotoiltuna. Voit luoda yhtälöryhmän myös mallin avulla.

Huomaa: Katso myös **Yhtälöryhmä**, sivu 8.

T

Matriisi $I^T \Rightarrow$ *matriisi*

Laskee *Matriisi* I :n transponoidun liittokompleksimatriisin.

Huomaa: Voit syöttää tämän operaattorin tietokoneen näppäimistöltä kirjoittamalla @t.

$$\begin{array}{|c|c|c|} \hline 1 & 2 & 3 \\ \hline 4 & 5 & 6 \\ \hline 7 & 8 & 9 \\ \hline \end{array}^T \quad \begin{array}{|c|c|c|} \hline 1 & 4 & 7 \\ \hline 2 & 5 & 8 \\ \hline 3 & 6 & 9 \\ \hline \end{array}$$

$$\begin{array}{|c|c|} \hline a & b \\ \hline c & d \\ \hline \end{array}^T \quad \begin{array}{|c|c|} \hline a & c \\ \hline b & d \\ \hline \end{array}$$

$$\begin{array}{|c|c|c|} \hline 1+i & 2+i \\ \hline 3+i & 4+i \\ \hline \end{array}^T \quad \begin{array}{|c|c|c|} \hline 1-i & 3-i \\ \hline 2-i & 4-i \\ \hline \end{array}$$

tan() -painike**tan(Laus I)** ⇒ lauseke

Astekulmatilassa:

tan(Lista I) ⇒ lista

$\tan\left(\frac{\pi}{4}\right)$	1
----------------------------------	---

tan(Laus I) laskee argumentin tangentin lausekkeena.

$\tan(45)$	1
------------	---

tan(Lista I) määrittää *Listal*:n kaikkien elementtien tangenttien listan.

$\tan(\{0,60,90\})$	$\{0,\sqrt{3},\text{undef}\}$
---------------------	-------------------------------

Huomaa: Argumentti tulkitaan aste-, graadi- tai radiaanikulmaksi käytössä olevan kulmatilan mukaisesti. Voit ohittaa kulmatila-asetuksen väliaikaisesti käyttämällä merkintää °, G tai R.

Graadikulmatilassa:

$\tan\left(\frac{\pi}{4}\right)$	1
----------------------------------	---

$\tan(50)$	1
------------	---

$\tan(\{0,50,100\})$	$\{0,1,\text{undef}\}$
----------------------	------------------------

Radiaanikulmatilassa:

$\tan\left(\frac{\pi}{4}\right)$	1
----------------------------------	---

$\tan(45^\circ)$	1
------------------	---

$\tan\left(\left\{\pi,\frac{\pi}{3},\pi,\frac{\pi}{4}\right\}\right)$	$\{0,\sqrt{3},0,1\}$
---	----------------------

tan(neliömatriisi I) ⇒ *neliömatriisi*

Radiaanikulmatilassa:

Laskee *neliömatriisi I*:n matriisin tangentin. Tämä ei ole sama kuin kunkin elementin tangentin laskeminen. Laskentamenetelmä on kuvattu kohdassa **cos()**.

$\tan\left(\begin{bmatrix} 1 & 5 & 3 \\ 4 & 2 & 1 \\ 6 & -2 & 1 \end{bmatrix}\right)$	$\begin{bmatrix} -28.2912 & 26.0887 & 11.1142 \\ 12.1171 & -7.83536 & -5.48138 \\ 36.8181 & -32.8063 & -10.4594 \end{bmatrix}$
---	--

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

tan⁻¹() -painike**tan⁻¹(Laus I)** ⇒ lauseke

Astekulmatilassa:

tan⁻¹(Lista I) ⇒ lista

$\tan^{-1}(1)$	45
----------------	----

tan⁻¹(Laus I) laskee kulman, jonka tangenti on *Laus I*, määrittäen vastauksen lausekkeena.

Graadikulmatilassa:

$\tan^{-1}()$

trig -painike

$\tan^{-1}(\text{Listal})$ luo listan *Listal*:n jokaisen elementin käänteistangenteista.

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

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

$\tan^{-1}(\text{neliomatriisi1}) \Rightarrow \text{neliomatriisi}$

Laskee *neliomatriisi1*:n matriisin käänteistangentin. Tämä ei ole sama kuin kunkin elementin käänteisen tangentin laskeminen. Laskentamenetelmä on kuvattu kohdassa **cos()**.

neliomatriisi1:n on oltava diagonalisoitavissa. Vastaus sisältää aina liukulukuja.

$$\tan^{-1}(1) \quad 50$$

Radiaanikulmatilassa:

$$\tan^{-1}(\{0,0,2,0,5\}) \quad \{0,0,197396,0,463648\}$$

Radiaanikulmatilassa:

$$\tan^{-1} \begin{pmatrix} 1 & 5 & 3 \\ 4 & 2 & 1 \\ 6 & -2 & 1 \end{pmatrix} \begin{bmatrix} -0.083658 & 1.26629 & 0.62263 \\ 0.748539 & 0.630015 & -0.070012 \\ 1.68608 & -1.18244 & 0.455126 \end{bmatrix}$$

tangentLine()

Katalogi >

$\text{tangentLine}(\text{Laus1}, \text{Muutt}, \text{Piste}) \Rightarrow \text{lauseke}$

$\text{tangentLine}(\text{Laus1}, \text{Muutt}=\text{Piste}) \Rightarrow \text{lauseke}$

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 $\text{tangentLine}(f_1(x), x, 2)$ antaa vastauksen "epätosi".

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

$$\text{tangentLine}((x-3)^2 - 4, x=3) \quad -4$$

$$\text{tangentLine}\left(\frac{1}{x^3}, x=0\right) \quad x=0$$

$$\text{tangentLine}(\sqrt{x^2 - 4}, x=2) \quad \text{undef}$$

$$x:=3: \text{tangentLine}(x^2, x, 1) \quad 5$$

tanh()

Katalogi >

$\text{tanh}(\text{Laus1}) \Rightarrow \text{lauseke}$

$\text{tanh}(\text{Listal}) \Rightarrow \text{lista}$

$\text{tanh}(\text{Laus1})$ laskee argumentin hyperbolisen tangentin lausekkeena.

$\text{tanh}(\text{Listal})$ luo listan *Listal*:n jokaisen elementin hyperbolisista tangenteista.

$$\text{tanh}(1.2) \quad 0.833655$$

$$\text{tanh}(\{0,1\}) \quad \{0, \text{tanh}(1)\}$$

tanh(neliömatriisi I) ⇒ *neliömatriisi*

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 diagonalisoitavissa. Vastaus sisältää aina liukulukuja.

Radiaanikulmatilassa:

$$\tanh \begin{pmatrix} 1 & 5 & 3 \\ 4 & 2 & 1 \\ 6 & -2 & 1 \end{pmatrix} = \begin{bmatrix} -0.097966 & 0.933436 & 0.425972 \\ 0.488147 & 0.538881 & -0.129382 \\ 1.28295 & -1.03425 & 0.428817 \end{bmatrix}$$

tanh⁻¹()

tanh⁻¹(Laus I) ⇒ *lauseke*

tanh⁻¹(Lista I) ⇒ *lista*

tanh⁻¹(Laus I) laskee argumentin käänteisen hyperbolisen tangentin lausekkeena.

tanh⁻¹(Lista I) luo listan *Lista I*:n jokaisen elementin käänteisistä hyperbolisista tangenteista.

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

tanh⁻¹(neliömatriisi I) ⇒ *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 diagonalisoitavissa. Vastaus sisältää aina liukulukuja.

Suorakulmakompleksimuodossa:

$$\begin{aligned} \tanh^{-1}(0) &= 0 \\ \tanh^{-1}(\{1, 2, 1, 3\}) &= \left\{ \text{undef}, 0.518046 - 1.5708 \cdot i, \frac{\ln(2)}{2} - \frac{\pi}{2} \cdot i \right\} \end{aligned}$$

Radiaanikulmatilassa ja suorakulmakompleksimuodossa:

$$\tanh^{-1} \begin{pmatrix} 1 & 5 & 3 \\ 4 & 2 & 1 \\ 6 & -2 & 1 \end{pmatrix} = \begin{bmatrix} -0.099353 + 0.164058 \cdot i & 0.267834 - 1.4908 \\ -0.087596 - 0.725533 \cdot i & 0.479679 - 0.94730 \\ 0.511463 - 2.08316 \cdot i & -0.878563 + 1.7901 \end{bmatrix}$$

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

taylor()

Katalogi > 

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 tyipistettyjä 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 en oletusarvo on nolla, ja se on lavennuspiste.

$\text{taylor}(e^{\sqrt{x}}, x, 2)$	$\text{taylor}(e^{\sqrt{x}}, x, 2, 0)$
$\text{taylor}(e^{t, t, 4}) _{t=\sqrt{x}}$	$\frac{3}{24} + \frac{x^2}{6} + \frac{x}{2} + \sqrt{x} + 1$
$\text{taylor}\left(\frac{1}{x \cdot (x-1)}, x, 3\right)$	$\text{taylor}\left(\frac{1}{x \cdot (x-1)}, x, 3, 0\right)$
$\text{expand}\left(\frac{\text{taylor}\left(\frac{x}{x \cdot (x-1)}, x, 4\right)}{x}, x\right)$	$-x^3 - x^2 - x - \frac{1}{x} - 1$

tCdf()

Katalogi > 

tCdf(alaraja, yläaraja, df) ⇒ luku, jos *alaraja* ja *yläaraja* ovat lukuja, *lista*, jos *alaraja* ja *yläaraja* ovat listoja

Laskee Studentin *t*-todennäköisyysjakauman *alarajan* ja *yläarajan* välillä määritetyille vapausasteelle *df*.

Kun $P(X \leq \textit{yläaraja})$, aseta *alaraja* = $-\infty$.

tCollect()

Katalogi > 

tCollect(Laus1) ⇒ lauseke

Laskee lausekkeen, jossa tulot ja sinien ja kosinien kokonaislukupotenssit on muunnettu useiden kulmien sinien ja kosinien lineaarisiksi kombinaatioksi, kulmien summiksi ja kulmien erotuksiksi. Transformaatio muuntaa trigonometriset polynomit niiden harmonisten lineaarisiksi kombinaatioksi.

$\text{tCollect}((\cos(\alpha))^2)$	$\frac{\cos(2 \cdot \alpha) + 1}{2}$
$\text{tCollect}(\sin(\alpha) \cdot \cos(\beta))$	$\frac{\sin(\alpha - \beta) + \sin(\alpha + \beta)}{2}$

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()

tExpand(Laus I) ⇒ 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()**-funktiota tulee käyttää radiaanikulmatilassa.

$$\begin{aligned} \text{tExpand}(\sin(3 \cdot \phi)) &= 4 \cdot \sin(\phi) \cdot (\cos(\phi))^2 - \sin(\phi) \\ \text{tExpand}(\cos(\alpha - \beta)) &= \cos(\alpha) \cdot \cos(\beta) + \sin(\alpha) \cdot \sin(\beta) \end{aligned}$$

Text

Text*kehotemerkkijono*[, *Näytälippu*]

Ohjelmointikomento: Keskeyttää ohjelman ja näyttää merkkijonon *kehotemerkkijono* valintaruudussa.

Määritä ohjelma, joka keskeytyy ja näyttää kunkin viidestä satunnaisluvusta 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 **156**, tai **RequestStr**, sivu **157**.

Huomaa: Tätä komentoa voi käyttää käyttäjän määrittämän ohjelman sisällä mutta ei funktion sisällä.

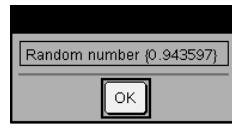
Paina mallin Prgm...EndPrgm jokaisen rivin lopussa näppäintä  näppäimen  sijaan. Tietokoneen näppäimistöllä **Alt**-näppäintä pidetään alhaalla ja painetaan **Enter**.

```
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

TInterval *Lista[,Frekv[,CTaso]]*

(Datalistan syöte)

TInterval $\bar{x}, S_x, n[,CTaso]$

(Yhteenvetotilaston syöte)

Laskee *t*-luottamusvälin. Tulosten yhteenveko tallentuu *stat.results*-muuttujaan. (Katso sivu 185.)

Lisätietoja listassa olevien tyhjiin elementtien vaikutuksesta, katso Tyhjät elementitsivulla sivu 248.

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. σ_x	Otoksen keskihajonta
stat.n	Otoksen keskiarvon sisältävän datasekvenssin pituus

TInterval_2Samp

TInterval_2Samp *List1,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 yhteenvedo tallentuu *stat.results*-muuttujaan. (Katso sivu 185.)

Poolaus=1 poolaa varianssit; *Poolaus=0* ei poolaa variansseja.

Lisätietoja listassa olevien tyhjiin elementtien vaikutuksesta, katso Tyhjät elementitsivulla sivu 248.

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

Tulosmuuttuja	Kuvaus
stat.ME	Virhemarginaali
stat.df	Vapausasteet
stat. $\bar{x}1$, stat. $\bar{x}2$	Satunnaisesta normaalijakaumasta olevien datasekvenssien otosten keskiarvot
stat. $\sigma x1$, stat. $\sigma x2$	Listat 1:n ja Listat 2:n otosten keskihajonnat
stat.n1, stat.n2	Otosten lukumäärä datasekvensseissä
stat.sp	Poolattu keskihajonta. Laskettu, kun Poolaus = 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*, °*lämpYksikkö2*)
 ⇒*lauseke* °*lämpYksikkö2*

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

Merkin Δ saat Katalogin symboleista.

ΔtmpCnv(100 °C, °F)	180 °F
ΔtmpCnv(180 °F, °C)	100 °C
ΔtmpCnv(100 °C, °K)	100 °K
ΔtmpCnv(100 °F, °R)	100 °R
ΔtmpCnv(1 °C, °F)	1.8 °F



Δ tmpCnv()

Katalogi > 

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_°C ja 1_°K ovat samansuuruisia, samoin kuin 1_°F ja 1_°R. 1_°C on kuitenkin 9/5 Fahrenheit-asteen arvosta 1_°F.

Esimerkiksi 100_°C:n alue (0_°C - 100_°C) vastaa Fahrenheit-asteiden aluetta 180_°F.

Jos haluat muuttaa jonkin lämpötila-arvon alueen sijaan, käytä funktiota **tmpCnv()**.

tPdf()

Katalogi > 

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

Laskee todennäköisyysfunktio (pdf) Studentin *t*-jakaumalle määritetyllä *x*:n arvolla ja määritetyillä vapausasteilla *df*.

trace()

Katalogi > 

trace(*neliömatriisi*) \Rightarrow *lauseke*

Laskee *neliömatriisin* jäljityksen (päälävistäjän kaikkien elementtien summan).

$\text{trace}\left(\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix}\right)$	15
$\text{trace}\left(\begin{pmatrix} a & 0 \\ 1 & a \end{pmatrix}\right)$	$2 \cdot a$

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 254.

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 kommentojen **Try**, **ClrErr** ja **PassErr** toiminnan, syötä oikealla näkyvä *eigenvals()*-ohjelma. Suorita ohjelma suorittamalla kukin seuraavista lausekkeista.

$$\text{eigenvals}\left(\begin{bmatrix} -3 \\ -41 \\ 5 \end{bmatrix}, [-1 \ 2 \ -3.1]\right)$$

$$\text{eigenvals}\left([1 \ 2 \ 3], \begin{bmatrix} 1 \\ 2 \end{bmatrix}\right)$$

Huomaa: Katso myös **ClrErr**, sivu 30, ja **PassErr**, sivu 137.

```
Define prog1()=Prgm
  Try
    z:=z+1
    Disp "z incremented."
  Else
    Disp "Sorry, z undefined."
  EndTry
EndPrgm
```

Done

```
z:=1:prog1()
```

```
z incremented.
```

Done

```
DelVar z:prog1()
```

```
Sorry, z undefined.
```

Done

```
Define eigenvals(a,b)=Prgm
```

© Program *eigenvals(A,B)* displays eigenvalues of A·B

Try

```
Disp "A= ",a
```

```
Disp "B= ",b
```

```
Disp ""
```

```
Disp "Eigenvalues of A·B are:",eigVl(a*b)
```

Else

```
If errCode=230 Then
```

```
Disp "Error: Product of A·B must be a square matrix"
```

```
ClrErr
```

```
Else
```

```
PassErr
```

```
EndIf
```

```
EndTry
```

tTest**tTest** $\mu_0, \text{Lista}, [\text{Frekv}], [\text{Hypot}]$

(Datalistan syöte)

tTest $\mu_0, \bar{x}, s_x, n, [\text{Hypot}]$

(Yhteenvetotilaston syöte)

Testaa hypoteesia yhden tuntemattoman perusjoukon keskiarvoon μ , kun perusjoukon keskihajontaa σ ei tunneta. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 185.)

Testaa $H_0: \mu = \mu_0$, jonkin seuraavan vaihtoehdon suhteen:

Kun $H_1: \mu < \mu_0$, aseta *Hypot*<0

Kun $H_1^a: \mu \neq \mu_0$ (oletus), aseta *Hypot*=0

Kun $H_1^a: \mu > \mu_0$, aseta *Hypot*>0

Lisätietoja listassa olevien tyhjen elementtien vaikutuksesta, katso Tyhjät elementtsivulla sivu 248.

Tulosmuuttuja	Kuvaus
stat.t	$(\bar{x} - \mu_0) / (\text{stdev} / \text{sqrt}(n))$
stat.PVal	Alin merkitsevyytaso, jolla nollahypoteesi voidaan hylätä
stat.df	Vapausasteet
stat. \bar{x}	<i>Lista</i> n sisältämän datasekvenssin otoksen keskiarvo
stat.sx	Datasekvenssin otoksen keskihajonta
stat.n	Otoksen koko

tTest_2Samp

tTest_2Samp *Lista1, Lista2, Frekv1*
 $[\text{Frekv2}, [\text{Hypot}, \text{Poolaus}]]]$

(Datalistan syöte)

tTest_2Samp $\bar{x}1, sx1, n1, \bar{x}2, sx2, n2[, Hypot$
 $[, Poolaus]$

(Yhteenvetotilaston syöte)

Suorittaa kahden otoksen t -testin. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 185.)

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 Tyhjä elementtisivulla sivu 248.

Tulosmuuttuja	Kuvaus
stat.t	Keskiarvojen erotukselle laskettu vakio-ohjearvo
stat.PVal	Alin merkitsevyytaso, jolla nollihypoteesi voidaan hylätä
stat.df	t-tilaston vapausasteet
stat. $\bar{x}1$, stat. $\bar{x}2$	<i>List</i> 1:n ja <i>List</i> 2:n sisältämien datasekvenssien otosten keskiarvot
stat.sx1, stat.sx2	<i>List</i> 1:n ja <i>List</i> 2:n sisältämien datasekvenssien otosten keskihajonnat
stat.n1, stat.n2	Otosten koko
stat.sp	Poolattu keskihajonta. Laskettu, kun <i>Poolaus</i> =1

tvmFV()

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.

tvmFV()

Katalogi > 

Huomaa: TVM-funktioissa käytetyt argumentit on kuvattu TVM-argumenttien taulukossa, sivu 204. Katso myös **amortTbl()**, sivu 12.

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 204. Katso myös **amortTbl()**, sivu 12.

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 204. Katso myös **amortTbl()**, sivu 12.

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 204. Katso myös **amortTbl()**, sivu 12.

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 204. Katso myös **amortTbl** (), sivu 12.

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 $X, Y, [Frekv] [, Luokka, Sisällyttä]$

Laskee kahden muuttujan tilastot. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 185.)

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

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.

Luokka on luokkakoodien lista vastaavalle X - ja Y -datalle.

Sisällytät 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 tyhjästä elementeistä, katso sivu 248.

Tulosmuuttuja	Kuvaus
stat. \bar{x}	x :n arvojen keskiarvo
stat. x	x :n arvojen summa
stat. x^2	x^2 :n arvojen summa
stat. s_x	x :n otoksen keskihajonta
stat. x	x :n perusjoukon keskihajonta
stat. n	Datapisteiden lukumäärä
stat. \bar{y}	y :n arvojen keskiarvo
stat. Σy	y :n arvojen summa
stat. Σy^2	y^2 :n arvojen summa
stat. s_y	y :n otoksen keskihajonta
stat. σ_y	y :n perusjoukon keskihajonta
stat. Σxy	$x \cdot y$ -arvojen summa
stat. r	Korrelaatiokerroin
stat. MinX	x :n arvojen minimi

Tulosmuuttuja	Kuvaus
stat.Q ₁ X	x:n ensimmäinen neljännes
stat.MedianX	x:n mediaani
stat.Q ₃ X	x:n 3. neljännes
stat.MaxX	x:n arvojen maksimi
stat.MinY	y:n arvojen minimi
stat.Q ₁ Y	y:n ensimmäinen neljännes
stat.MedY	y:n mediaani
stat.Q ₃ Y	x:n kolmas neljännes
stat.MaxY	y:n arvojen maksimi
stat.Σ(x- \bar{x}) ²	x:n keskiarvon poikkeamien neliöiden summa
stat.Σ(y- \bar{y}) ²	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.

$$\text{unitV}\left(\begin{bmatrix} a & b & c \end{bmatrix}\right) = \left[\frac{a}{\sqrt{a^2+b^2+c^2}} \quad \frac{b}{\sqrt{a^2+b^2+c^2}} \quad \frac{c}{\sqrt{a^2+b^2+c^2}} \right]$$

$$\text{unitV}\left(\begin{bmatrix} 1 & 2 & 1 \end{bmatrix}\right) = \left[\frac{\sqrt{6}}{6} \quad \frac{\sqrt{6}}{3} \quad \frac{\sqrt{6}}{6} \right]$$

$$\text{unitV}\left(\begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}\right) = \begin{bmatrix} \frac{\sqrt{14}}{14} \\ \frac{14}{\sqrt{14}} \\ 7 \\ 3 \cdot \frac{\sqrt{14}}{14} \\ 14 \end{bmatrix}$$

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

unLock

Katalogi > 

unLock*Muutt1* [, *Muutt2*] [, *Muutt3*] ...
unLock*Muutt*.

Vapauttaa määritetyt muuttujat tai muuttujaryhmän. Lukittuja muuttujia ei voi muokata eikä poistaa.

Katso **Lock**, sivu 110, ja **getLockInfo()**, sivu 87.

<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()

Katalogi > 

varPop(*Listan* [, *frekvListan*]) \Rightarrow *lauseke*

Laskee *Listan* perusjoukon varianssin.

Jokainen *frekvListan* elementi näyttää *Listan* vastaavien elementtien peräkkäisten esiintymien lukumäärän.

Huomaa: *Listassa* tulee olla vähintään kaksi elementtiä.

Jos jokin elementi jommassakummassa listassa on tyhjä, kyseistä elementtiä ei huomioida, eikä toisessa listassa olevaa vastaavaa elementtiä myöskään huomioida. Lisätietoja tyhjistä elementeistä, katso sivu 248.

varPop({5,10,15,20,25,30})	875
	12
Ans: 1.	72.9167

varSamp()

Katalogi > 

varSamp(*Listan* [, *frekvListan*]) \Rightarrow *lauseke*

Laskee *Listan* otosten varianssin.

Jokainen *frekvListan* elementi näyttää *Listan* vastaavien elementtien peräkkäisten esiintymien lukumäärän.

Huomaa: *Listassa* tulee olla vähintään kaksi elementtiä.

varSamp({ <i>a</i> , <i>b</i> , <i>c</i> })	
	$\frac{a^2 - a \cdot (b+c) + b^2 - b \cdot c + c^2}{3}$
varSamp({1,2,5,6,3,2})	31
	2
varSamp({1,3,5},{4,6,2})	68
	33

Jos jokin elementti jommassakummassa listassa on tyhjä, kyseistä elementtiä ei huomioida, eikä toisessa listassa olevaa vastaavaa elementtiä myöskään huomioida. Lisätietoja tyhjistä elementeistä, katso sivu 248.

varSamp(MatriisiI[,frekvMatriisi])
⇒*matriisi*

Laskee rivivektorin, joka sisältää *MatriisiI*:n kaikkien sarakkeiden otoksen varianssin.

Jokainen *frekvMatriisi*n 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 tyhjistä elementeistä, katso sivu 248.

Huomaa: *MatriisiI*:ssä on oltava vähintään kaksi elementtiä.

varSamp	$\begin{pmatrix} 1 & 2 & 5 \\ -3 & 0 & 1 \\ .5 & .7 & 3 \end{pmatrix}$	[4.75 1.03 4]
varSamp	$\begin{pmatrix} -1.1 & 2.2 & 6 & 3 \\ 3.4 & 5.1 & 2 & 4 \\ -2.3 & 4.3 & 5 & 1 \end{pmatrix}$	[3.91731 2.08411]

W

Wait

Luettelo >



Wait *aikaSekunteina*

Keskeyttää toiminnon *aikaSekunteina* sekunnin ajaksi.

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 .

Odottaaksesi 4 sekuntia:

Wait 4

Odottaaksesi 1/2 sekuntia:

Wait 0.5

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.

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.

Send "SET GREEN 1 ON"
Wait 0.5
Send "SET GREEN 1 OFF"

warnCodes ()

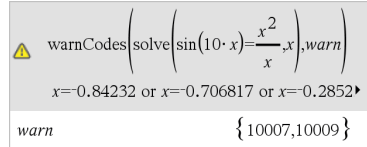
warnCodes(*Expr1*, *StatusVar*) \Rightarrow *expression*

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 261.



The screenshot shows the command `warnCodes(solve(sin(10*x)=x^2,x),warn)` in a TI-Nspire CAS window. Below the command, the output is displayed as `x=-0.84232 or x=-0.706817 or x=-0.2852`. At the bottom, the `warn` variable is shown with a list of values: `{10007,10009}`.

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

when()

when(*Ehto*, *tosiTulos* [, *epätosiTulos*][, *tuntematonTulos*])
 \Rightarrow *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.

when()

Katalogi > 

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 $\text{sum_of_recip}(n) = \text{Func}$	
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 tempsum	
EndFunc	
	Done
$\text{sum_of_recip}(3)$	$\frac{11}{6}$
	6

X

xor

Katalogi > 

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.

true xor true	false
5 > 3 xor 3 > 5	true

Määrittää totuusarvoksi epätosi, jos kumpikin argumentti on tosi tai kumpikin on epätosi. Antaa vastauksena sievennetyn Boolean lausekkeen, jos kummankaan argumentin totuusarvoa ei voi määrittää todeksi tai epätodeksi.

Huomaa: Katso **or**, sivu 135.

Kokonaisluku1 xor Kokonaisluku2 ⇒ *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 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 desimaalikonaisluvun, joka on etumerkillisen, 64 bitin binaarimuodon lukualueen ulkopuolella, laskin käyttää symmetristä modulo-operaatiota, jotta arvo saadaan oikealle alueelle. Lisätietoja, katso **►Base2**, sivu 22.

Huomaa: Katso **or**, sivu 135.

Z

zeros()

zeros(Laus, Muutt) ⇒ lista

zeros(Laus, Muutt=Arvaus) ⇒ lista

$$\text{zeros}\left(a \cdot x^2 + b \cdot x + c, x\right)$$

$$\left\{ \frac{\sqrt{b^2 - 4 \cdot a \cdot c} - b}{2 \cdot a}, \frac{-\left(\sqrt{b^2 - 4 \cdot a \cdot c} + b\right)}{2 \cdot a} \right\}$$

$$a \cdot x^2 + b \cdot x + c | x = \text{Ans}[2] \quad 0$$

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:

muuttuja
– tai –
muuttuja = reaali tai ei-reaali luku

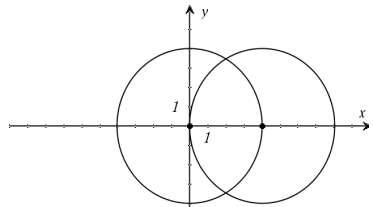
Esimerkiksi x kelpaa ja samoin $x=3$.

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.

$$\frac{\text{exact}\left(\text{zeros}\left(a \cdot \left(e^x + x\right) \cdot \left(\text{sign}(x) - 1\right), x\right)\right)}{\text{exact}\left(\text{solve}\left(a \cdot \left(e^x + x\right) \cdot \left(\text{sign}(x) - 1\right) = 0, x\right)\right)}$$

$$e^x + x = 0 \text{ or } x > 0 \text{ or } a = 0$$



zeros()

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 kokonaislukuliite väliltä 1-255.

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-polynomien missä tahansa muuttujassa, mutta kaikki muuttujat ovat lineaarisia kaikissa tuntemattomissa muuttujissa, **zeros()** käyttää Gaussin eliminointia yrittäessään määrittää kaikki reaaliset nollakohdat.

$$\text{zeros}\left(\left\{x^2+y^2-r^2,(x-r)^2+y^2-r^2\right\},\{x,y\}\right)$$
$$\begin{bmatrix} r & -\sqrt{3}\cdot r \\ 2 & 2 \\ r & \sqrt{3}\cdot r \\ 2 & 2 \end{bmatrix}$$

Määritä rivin 2 juuri:

$$\text{Ans}[2] \quad \begin{bmatrix} r & \sqrt{3}\cdot r \\ 2 & 2 \end{bmatrix}$$

$$\text{zeros}\left(\left\{x^2+y^2-r^2,(x-r)^2+y^2-r^2\right\},\{x,y,z\}\right)$$
$$\begin{bmatrix} r & -\sqrt{3}\cdot r & c1 \\ 2 & 2 & \\ r & \sqrt{3}\cdot r & c1 \\ 2 & 2 & \end{bmatrix}$$

$$\text{zeros}\left(\left\{x+e^z\cdot y-1,x-y-\sin(z)\right\},\{x,y\}\right)$$
$$\begin{bmatrix} e^z\cdot \sin(z)+1 & -(\sin(z)-1) \\ e^z+1 & e^z+1 \end{bmatrix}$$

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.

$$\text{zeros}\left(\left\{e^z \cdot y - 1, y - \sin(z)\right\}, \{y, z\}\right)$$

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 $\sigma, \text{Lista}, \text{Frekv}, \text{CTaso}$

(Datalistan syöte)

zInterval $\sigma, \bar{x}, n, \text{CTaso}$

(Yhteenvetotilaston syöte)

Laskee z -luottamusvälin. Tulosten yhteenvedo tallentuu *stat.results*-muuttujaan. (Katso sivu 185.)

Lisätietoja listassa olevien tyhjien elementtien vaikutuksesta, katso Tyhjät elementitsivulla sivu 248.

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

Tulosmuuttuja	Kuvaus
stat.n	Otoksen keskiarvon sisältävän datasekvenssin pituus
stat. σ	Datasekvenssin <i>Lista</i> tunnettu perusjoukon keskihajonta

zInterval_1Prop

Katalogi > 

zInterval_1Prop $x, n [, CTaso]$

Laskee yhden osuuden z -luottamusvälin. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 185.)

x on ei-negatiivinen kokonaisluku.

Lisätietoja listassa olevien tyhjen elementtien vaikutuksesta, katso Tyhjät elementitsivulla sivu 248.

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

Katalogi > 

zInterval_2Prop $x1, n1, x2, n2 [, CTaso]$

Laskee kahden osuuden z -luottamusvälin. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 185.)

$x1$ ja $x2$ ovat ei-negatiivisia kokonaislukuja.

Lisätietoja listassa olevien tyhjen elementtien vaikutuksesta, katso Tyhjät elementitsivulla sivu 248.

Tulosmuuttuja	Kuvaus
stat.CLower, stat.CUpper	Luottamusväli, joka sisältää jakauman luottamusvälin todennäköisyyden
stat. \hat{p} Diff	Osuuksien välinen laskettu erotus

Tulosmuuttuja	Kuvaus
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

Katalogi > 

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 yhteenvedo tallentuu *stat.results*-
muuttujaan. (Katso sivu 185.)

Lisätietoja listassa olevien tyhjen
elementtien vaikutuksesta, katso Tyhjät
elementtivuorolla sivu 248.

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. $\sigma x2$	<i>Lista 1</i> :n ja <i>Lista 2</i> :n otosten keskihajonnat
stat.n1, stat.n2	Otosten lukumäärä datasekvensseissä
stat.r1, stat.r2	Datasekvenssien <i>Lista 1</i> ja <i>Lista 2</i> tunnetut perusjoukon keskihajonnat

zTest $\mu_0, \sigma, Lista, [Frekv[, Hypot]]$

(Datalistan syöte)

zTest $\mu_0, \sigma, \bar{x}, n[, Hypot]$

(Yhteenvetotilaston syöte)

Suorittaa z-testin frekvenssillä *frekvlista*. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 185.)

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 tyhjen elementtien vaikutuksesta, katso Tyhjät elementitsivulla sivu 248.

Tulosmuuttuja	Kuvaus
stat.z	$(\bar{x} - \mu_0) / (\sigma / \sqrt{n})$
stat.P Value	Pienin todennäköisyys, jolla nollahypoteesi voidaan hylätä
stat. \bar{x}	<i>Lista</i> n sisältämän datasekvenssin otoksen keskiarvo
stat.sx	Datasekvenssin otoksen keskihajonta. Lasketaan vain <i>Data</i> -syötteelle.
stat.n	Otoksen koko

zTest_1Prop

zTest_1Prop $p_0, x, n[, Hypot]$

Laskee yhden osuuden z-testin. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 185.)

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 elementitsivulla sivu 248.

Tulosmuuttuja	Kuvaus
stat.p0	Perusjoukon osuuden hypoteesiarvo
stat.z	Osuudelle laskettu vakio-ohjearvo
stat.PVal	Alin merkitsevyystaso, jolla nollahypoteesi voidaan hylätä
stat. \hat{p}	Arvioitu otoksen osuus
stat.n	Otoksen koko

zTest_2Prop $x1, n1, x2, n2, Hypot$

Laskee kahden osuuden z-testin. Tulosten yhteenvedo tallentuu *stat.results*-muuttujaan. (Katso sivu 185.)

$x1$ ja $x2$ ovat ei-negatiivisia kokonaislukuja.

Testaa $H_0: p1 = p2$ 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 tyhjien elementtien vaikutuksesta, katso Tyhjät elementitsivulla sivu 248.

Tulosmuuttuja	Kuvaus
stat.z	Osuuksien erotukselle laskettu vakio-ohjearvo
stat.PVal	Alin merkitsevyystaso, jolla nollahypoteesi voidaan hylätä
stat. $\hat{p}1$	Arvio ensimmäisen näytteen osuudesta
stat. $\hat{p}2$	Arvio toisen näytteen osuudesta
stat. \hat{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 185.)

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

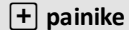
Kun $H_a: \mu_1 > \mu_2$, *Hypot*>0

Lisätietoja listassa olevien tyhjiin elementtien vaikutuksesta, katso Tyhjät elementisivulla sivu 248.

Tulosmuuttuja	Kuvaus
stat.z	Keskiarvojen erotukselle laskettu vakio-ohjearvo
stat.PVal	Alin merkitsevyystaso, jolla nollahypoteesi voidaan hylätä
stat. $\bar{x}1$, stat. $\bar{x}2$	<i>Lista 1</i> :n ja <i>Lista 2</i> :n sisältämien datasekvenssien otosten keskiarvot
stat.sx1, stat.sx2	<i>Lista 1</i> :n ja <i>Lista 2</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

$Listal + Lista2 \Rightarrow lista$
 $Matriisi1 + Matriisi2 \Rightarrow matriisi$

Määrittää listan (tai matriisin), joka sisältää $Listal$:n ja $Lista2$:n (tai $Matriisi1$:n ja $Matriisi2$:n) vastaavien elementtien summat.

$\left\{ 22, \pi, \frac{\pi}{2} \right\} \rightarrow I1$	$\left\{ 22, \pi, \frac{\pi}{2} \right\}$
$\left\{ 10, 5, \frac{\pi}{2} \right\} \rightarrow I2$	$\left\{ 10, 5, \frac{\pi}{2} \right\}$
$I1+I2$	$\{ 32, \pi+5, \pi \}$
$Ans + \{ \pi, 5, \pi \}$	$\{ \pi+32, \pi, 0 \}$
$\begin{bmatrix} a & b \\ c & d \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$	$\begin{bmatrix} a+1 & b \\ c & d+1 \end{bmatrix}$

Argumenttien tulee olla mitoiltaan samanlaisia.

$Laus + Listal \Rightarrow lista$
 $Listal + Laus \Rightarrow lista$

Määrittää listan, joka sisältää $Laus$:n ja $Listal$:n kunkin elementin summat.

$15 + \{ 10, 15, 20 \}$	$\{ 25, 30, 35 \}$
$\{ 10, 15, 20 \} + 15$	$\{ 25, 30, 35 \}$

$Laus + Matriisi1 \Rightarrow matriisi$

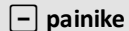
$Matriisi1 + Laus \Rightarrow matriisi$

Laskee matriisin, jossa $Laus$ on lisätty jokaiseen elementtiin $Matriisi1$:n diagonaalimatriisissa. $Matriisi1$:n on oltava neliö.

$20 + \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$	$\begin{bmatrix} 21 & 2 \\ 3 & 24 \end{bmatrix}$
---	--

Huomaa: Käytä merkintää $+$ (piste plus), kun haluat lisätä lausekkeen jokaiseen elementtiin.

-(vähennyslasku)



$Laus1 - Laus2 \Rightarrow lauseke$

Laskee $Laus1$ miinus $Laus2$.

6-2	4
$\pi - \frac{\pi}{6}$	$\frac{5 \cdot \pi}{6}$
	6

-(vähennyslasku)**[-] painike***Listal - Lista2* ⇒ lista

$$\left\{ 22, \pi, \frac{\pi}{2} \right\} - \left\{ 10, 5, \frac{\pi}{2} \right\} = \{ 12, \pi - 5, 0 \}$$

Matriisi1 - Matriisi2 ⇒ matriisi

$$\begin{bmatrix} 3 & 4 \\ 1 & 2 \end{bmatrix} - \begin{bmatrix} 1 & 2 \\ 2 & 2 \end{bmatrix} = \begin{bmatrix} 2 & 2 \\ -1 & 0 \end{bmatrix}$$

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 ⇒ lista

$$15 - \{ 10, 15, 20 \} = \{ 5, 0, -5 \}$$

Listal - Laus ⇒ 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 ⇒ matriisi

$$20 - \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} = \begin{bmatrix} 19 & -2 \\ -3 & 16 \end{bmatrix}$$

Matriisi1 - Laus ⇒ 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 · Laus2* ⇒ lauseke

$$20 \cdot 3 \cdot 4 \cdot 5 = 69$$

Laskee kahden argumentin tulon.

$$x \cdot y \cdot x = x^2 \cdot y$$

Listal · Lista2 ⇒ lista

$$\{ 1, 2, 3 \} \cdot \{ 4, 5, 6 \} = \{ 4, 10, 18 \}$$

Luo listan, joka sisältää *Listal*:n ja *Lista2*:n vastaavien elementtien tulot.

$$\begin{bmatrix} 2 & 3 \\ a & 2 \end{bmatrix} \cdot \begin{bmatrix} a & 2 \\ a & 3 \end{bmatrix} = \begin{bmatrix} 2 \cdot a & b \\ 2 \cdot a & b \end{bmatrix}$$

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* · *Listal* ⇒ *lista**Listal* · *Laus* ⇒ *lista*Määrittää listan, joka sisältää lausekkeen *Laus* ja kunkin *Listal*: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.

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix} \begin{bmatrix} a & d \\ b & e \\ c & f \end{bmatrix}$$

$$\begin{bmatrix} a+2\cdot b+3\cdot c & d+2\cdot e+3\cdot f \\ 4\cdot a+5\cdot b+6\cdot c & 4\cdot d+5\cdot e+6\cdot f \end{bmatrix}$$

$$\pi \cdot \{4,5,6\} \qquad \{4\cdot\pi, 5\cdot\pi, 6\cdot\pi\}$$

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \cdot 0.01 \qquad \begin{bmatrix} 0.01 & 0.02 \\ 0.03 & 0.04 \end{bmatrix}$$

$$\lambda \cdot \text{identity}(3) \qquad \begin{bmatrix} \lambda & 0 & 0 \\ 0 & \lambda & 0 \\ 0 & 0 & \lambda \end{bmatrix}$$

/ (jakolasku)**÷** painike*Laus1* / *Laus2* ⇒ *lauseke*Laskee osamäärän *Laus1* jaettuna *Laus2*:lla.**Huomaa:** Katso myös **Murtolukumalli**, sivu 5.*Listal* / *Lista2* ⇒ *lista*Määrittää listan, joka sisältää osamäärät laskutoimituksista *Listal* jaettuna *Lista2*:lla.

Listojen tulee olla mitoiltaan samanlaisia.

Laus / *Listal* ⇒ *lista**Listal* / *Laus* ⇒ *lista*Määrittää listan, joka sisältää osamäärät laskutoimituksista *Laus* jaettuna *Listal*:llä tai *Listal* jaettuna lausekkeella *Laus*.

$$\frac{2}{3.45} \qquad .57971$$

$$\frac{x^3}{x} \qquad x^2$$

$$\frac{\{1.,2,3\}}{\{4,5,6\}} \qquad \left\{0.25, \frac{2}{5}, \frac{1}{2}\right\}$$

$$\frac{a}{\{3,a,\sqrt{a}\}} \qquad \left\{\frac{a}{3}, 1, \sqrt{a}\right\}$$

$$\frac{\{a,b,c\}}{a\cdot b\cdot c} \qquad \left\{\frac{1}{b\cdot c}, \frac{1}{a\cdot c}, \frac{1}{a\cdot b}\right\}$$

/ (jakolasku)

 painike


Matriisi1 / Laus ⇒ *matriisi*

$$\frac{\begin{bmatrix} a & b & c \end{bmatrix}}{a \cdot b \cdot c} \quad \begin{bmatrix} \frac{1}{b \cdot c} & \frac{1}{a \cdot c} & \frac{1}{a \cdot b} \end{bmatrix}$$

Laskee matriisin, joka sisältää osamäärät laskutoimituksesta *Matriisi1/Laus*.

Huomaa: Käytä merkintää $\cdot /$ (piste jaa), kun haluat jakaa lausekkeen jokaisella elementillä.

^ (potenssi)

 painike

Laus1 ^ Laus2 ⇒ *lauseke*

$$4^2 \quad 16$$

List1 ^ Lista2 ⇒ *lista*

$$\{a,2,c\}^{\{1,b,3\}} \quad \{a,2^b,c^3\}$$

Laskee ensimmäisen argumentin korotettuna toisen argumentin potenssiin.

Huomaa: Katso myös **Eksponenttimalli**, sivu 5.

Jos kyseessä on lista, laskee *List1*:n elementit korotettuna *Lista2*:n vastaavien elementtien potenssiin.

Reaalilukujen alueella murtolukupotenssit, joilla on supistetut eksponentit ja parittomat nimittäjät, käyttävät reaalista aluetta versus pääalue kompleksitilassa.

Laus ^ List1 ⇒ *lista*

$$p^{\{a,2,-3\}} \quad \left\{ p^a, p^2, \frac{1}{p^3} \right\}$$

Laskee lausekkeen *Laus* korotettuna *List1*:n elementtien potenssiin.

List1 ^ Laus ⇒ *lista*

$$\{1,2,3,4\}^{-2} \quad \left\{ 1, \frac{1}{4}, \frac{1}{9}, \frac{1}{16} \right\}$$

Laskee *List1*:n elementit korotettuna *Laus*:n potenssiin.

neliömatriisi1 ^ kokonaisluku ⇒ *matriisi*

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}^2 \quad \begin{bmatrix} 7 & 10 \\ 15 & 22 \end{bmatrix}$$

Laskee *neliömatriisi1*:n korotettuna kokonaisluvun *kokonaisluku* potenssiin.

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}^{-1} \quad \begin{bmatrix} -2 & 1 \\ 3 & -1 \\ 2 & 2 \end{bmatrix}$$

neliömatriisi1:n on oltava neliömatriisi.

Jos *kokonaisluku* = -1, laskee käänteismatriisin.

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}^{-2} \quad \begin{bmatrix} 11 & -5 \\ 2 & 2 \\ -15 & 7 \\ 4 & 4 \end{bmatrix}$$

Jos *kokonaisluku* < -1, laskee käänteismatriisin korotettuna sopivaan positiiviseen potenssiin.

x² (neliö)**x² painike***Laus*l² ⇒ *lauseke*

Laskee argumentin neliön.

4 ²	16
----------------	----

{2,4,6} ²	{4,16,36}
----------------------	-----------

*Listal*² ⇒ *lista*Laskee listan, joka sisältää *Listal*:n elementtien neliöt.

$\begin{bmatrix} 2 & 4 & 6 \\ 3 & 5 & 7 \\ 4 & 6 & 8 \end{bmatrix}^2$	$\begin{bmatrix} 40 & 64 & 88 \\ 49 & 79 & 109 \\ 58 & 94 & 130 \end{bmatrix}$
---	--

*neliömatriisi*l² ⇒ *matriisi*Laskee *neliömatriisi*l:n matriisinelion. Tämä ei ole sama kuin kunkin elementin neliön laskeminen. Käytä merkintää .^2, kun haluat laskea jokaisen elementin neliön.

$\begin{bmatrix} 2 & 4 & 6 \\ 3 & 5 & 7 \\ 4 & 6 & 8 \end{bmatrix} \cdot^2$	$\begin{bmatrix} 4 & 16 & 36 \\ 9 & 25 & 49 \\ 16 & 36 & 64 \end{bmatrix}$
---	--

.+ (piste lisää)**.+ painikkeet***Matriisi*1 .+ *Matriisi*2 ⇒ *matriisi**Laus* .+ *Matriisi*1 ⇒ *matriisi**Matriisi*1 .+ *Matriisi*2 laskee matriisin, joka on *Matriisi*1:n ja *Matriisi*2:n vastaavien elementtiparien summa.*Laus* .+ *Matriisi*1 laskee matriisin, joka on *Laus*:n ja kunkin *Matriisi*1:n elementin summa.

$\begin{bmatrix} a & 2 \\ b & 3 \end{bmatrix} .+ \begin{bmatrix} c & 4 \\ d & 5 \end{bmatrix}$	$\begin{bmatrix} a+c & 6 \\ b+5 & d+3 \end{bmatrix}$
--	--

$x .+ \begin{bmatrix} c & 4 \\ d & 5 \end{bmatrix}$	$\begin{bmatrix} x+c & x+4 \\ x+5 & x+d \end{bmatrix}$
---	--

.- (piste-erotus)**.- painikkeet***Matriisi*1 .- *Matriisi*2 ⇒ *matriisi**Laus* .-*Matriisi*1 ⇒ *matriisi**Matriisi*1 .-*Matriisi*2 laskee matriisin, joka on *Matriisi*1:n ja *Matriisi*2:n vastaavien elementtiparien välinen erotus.*Laus* .-*Matriisi*1 laskee matriisin, joka on *Laus*:n ja kunkin *Matriisi*1:n elementin erotus.

$\begin{bmatrix} a & 2 \\ b & 3 \end{bmatrix} .- \begin{bmatrix} c & 4 \\ d & 5 \end{bmatrix}$	$\begin{bmatrix} a-c & -2 \\ b-d & -2 \end{bmatrix}$
--	--

$x .- \begin{bmatrix} c & 4 \\ d & 5 \end{bmatrix}$	$\begin{bmatrix} x-c & x-4 \\ x-d & x-5 \end{bmatrix}$
---	--

. · (pistetulo)

.	×	painikkeet
---	---	------------

Matriisi1 . · *Matriisi2* ⇒ *matriisi*

$\begin{bmatrix} a & 2 \\ b & 3 \end{bmatrix}$.	$\begin{bmatrix} c & 4 \\ 5 & d \end{bmatrix}$	$\begin{bmatrix} a \cdot c & 8 \\ 5 \cdot b & 3 \cdot d \end{bmatrix}$
--	---	--	--

Laus . · *Matriisi1* ⇒ *matriisi*

x	.	$\begin{bmatrix} a & b \\ c & d \end{bmatrix}$	$\begin{bmatrix} a \cdot x & b \cdot x \\ c \cdot x & d \cdot x \end{bmatrix}$
-----	---	--	--

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* ⇒ *matriisi*

$\begin{bmatrix} a & 2 \\ b & 3 \end{bmatrix}$.	$\left(\begin{bmatrix} c & 4 \\ 5 & d \end{bmatrix} \right)$	$\begin{bmatrix} \frac{a}{c} & \frac{1}{2} \\ \frac{b}{5} & \frac{3}{d} \end{bmatrix}$
--	---	---	--

Laus . / *Matriisi1* ⇒ *matriisi*

x	.	$\left(\begin{bmatrix} c & 4 \\ 5 & d \end{bmatrix} \right)$	$\begin{bmatrix} \frac{x}{c} & \frac{x}{4} \\ \frac{x}{5} & \frac{x}{d} \end{bmatrix}$
-----	---	---	--

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.

. ^ (pistepotenssi)

.	^	painikkeet
---	---	------------

Matriisi1 . ^ *Matriisi2* ⇒ *matriisi*

$\begin{bmatrix} a & 2 \\ b & 3 \end{bmatrix}$.	$\left(\begin{bmatrix} c & 4 \\ 5 & d \end{bmatrix} \right)$	$\begin{bmatrix} a^c & 16 \\ b^5 & 3^d \end{bmatrix}$
--	---	---	---

Laus . ^ *Matriisi1* ⇒ *matriisi*

x	.	$\left(\begin{bmatrix} c & 4 \\ 5 & d \end{bmatrix} \right)$	$\begin{bmatrix} x^c & x^4 \\ x^5 & x^d \end{bmatrix}$
-----	---	---	--

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

$-Laus1 \Rightarrow lauseke$

$-Listal \Rightarrow lista$

$-Matriisi1 \Rightarrow matriisi$

Laskee argumentin negaation.

Kun kyseessä on lista tai matriisi, määrittää kaikkien elementtien negaatiot.

Jos argumentti on binaarinen tai heksadesimaalinen kokonaisluku, negaatio antaa kakkosen komplementin.


-2.43	-2.43
$-\{-1,0.4,1.2\mathbb{E}19\}$	$\{1,-0.4,-1.2\mathbb{E}19\}$
$-a \cdot b$	$a \cdot b$

Binaarisessa kantelukutilassa:

0b100101
0b11111111111111111111111111111111▶

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

% (prosentti)

  painikkeet

$Laus1 \% \Rightarrow lauseke$

$Listal \% \Rightarrow lista$



$Matriisi1 \% \Rightarrow matriisi$

argument

Määrittää 100

Kun kyseessä on lista tai matriisi, määrittää listan tai matriisin, jossa jokainen elementti on jaettu luvulla 100.

Huom: Vastauksen pakottaminen likimääräiseksi:

Kämmenlaite: Paina  .

Windows®: Paina **Ctrl+Enter**.

Macintosh®: Paina **⌘+Enter**.

iPad®: Pidä **enter** ja valitse .

13%	0.13
$\{\{1,10,100\}\}\%$	$\{0.01,0.1,1.\}$

= (on yhtä kuin)

 painike

$Laus1 = Laus2 \Rightarrow Boolean\ lauseke$

$Listal = Lista2 \Rightarrow Boolean\ lista$

$Matriisi1 = Matriisi2 \Rightarrow Boolean\ matriisi$


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.

Esimerkkifunktio, jossa on käytetty matematiikkatestisymboleita: =, ≠, <, ≤, >, ≥

= (on yhtä kuin)

 painike

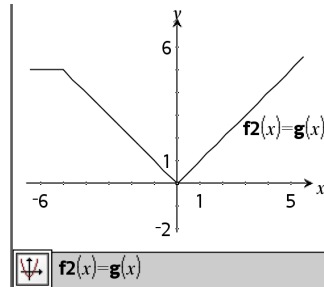
Listojen ja matriisien kohdalla vastauksena on vertailuja elementti elementiltä.

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



```
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.

Listojen ja matriisien kohdalla vastauksena on vertailuja elementti elementiltä.



Katso esimerkki kohdasta "≠" (on yhtä kuin).

≠ (ei yhtä kuin)

  painikkeet

Huomaa: Voit syöttää tämän operaattorin näppäimistöltä kirjoittamalla /=

< (pienempi kuin)

  painikkeet

$Laus1 < Laus2 \Rightarrow \text{Boolean lauseke}$

Katso esimerkki kohdasta “=” (on yhtä kuin).

$List1 < Lista2 \Rightarrow \text{Boolean lista}$

$Matriisi1 < Matriisi2 \Rightarrow \text{Boolean matriisi}$

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).

$List1 \leq Lista2 \Rightarrow \text{Boolean lista}$

$Matriisi1 \leq Matriisi2 \Rightarrow \text{Boolean matriisi}$

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.

Huomaa: Voit syöttää tämän operaattorin näppäimistöltä kirjoittamalla <=

> (suurempi kuin)

ctrl [=] 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)

ctrl [=] 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*BooleanMatriisi1 ⇔ BooleanMatriisi2* antaa vastauksena *Boolean matriisin* $3 \text{ xor } 4$ 7 $3 \Leftrightarrow 4$ -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 näppäimistöltä kirjoittamalla <=>

! (kertoma)

 **painike**

Laus! ⇒ lauseke

$$\begin{array}{l} 5! \qquad \qquad \qquad 120 \\ \{\{5,4,3\}\}! \qquad \qquad \qquad \{120,24,6\} \\ \left(\begin{array}{cc} 1 & 2 \\ 3 & 4 \end{array}\right)! \qquad \qquad \qquad \left[\begin{array}{cc} 1 & 2 \\ 6 & 24 \end{array}\right] \end{array}$$

Lista! ⇒ lista

Matriisi! ⇒ matriisi

Määrittää argumentin kertoman.

Jos kyseessä on lista tai matriisi, määrittää listan tai matriisin elementtien kertomista.

& (liitä)

  **painikkeet**

Merkkijono1 & Merkkijono2 ⇒
merkkijono

"Hello "&"Nick" "Hello Nick"

Antaa vastauksena tekstimerkkijonon, joka on *Merkkijono2* liitettynä *Merkkijono1*:een.

d() (derivaatta)

Katalogi > 

d(Laus!, 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(Lista!, Muutt[, Aste]) ⇒ lista

$$\frac{d}{dy} \left(\frac{d}{dx}(x^2 \cdot y^3) \right) \qquad 6 \cdot y^2 \cdot x$$

d(Matriisi!, Muutt[, Aste]) ⇒ matriisi

$$\frac{d}{dx} \left(\left\{ \begin{array}{ccc} x^2 & x^3 & x^4 \end{array} \right\} \right) \qquad \left\{ 2 \cdot x, 3 \cdot x^2, 4 \cdot x^3 \right\}$$

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 sovitetään 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 10,

Toinen derivaatta, sivu 10 tai

N:s derivaatta, sivu sivu 10.

f() (integraali)

f(Lausl, Muutt[, Ala, Ylä]) ⇒ lauseke

f(Lausl, Muutt[, Vakio]) ⇒ lauseke

Laskee *Lausl*:n integraalin muuttujan *Muutt* suhteen välillä *Ala* - *Ylä* olevista arvoista.

Huomaa: Katso myös **Määrityn** tai **Määrittämättömän integraalin malli**, sivu 11.

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

Laskee antiderivaatan, jos *Ala* ja *Ylä* jätetään pois. Integraalin symbolinen vakio jätetään pois, ellei anna argumenttia *Vakio*.

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

$$\int (a \cdot x^2, x, c)$$

$$\frac{a \cdot x^3}{3} + c$$

Samanarvoisesti kelpaavat antiderivaatat voivat erota numeerisen vakion suhteen. Tällainen vakio on olla naamioitunut—erityisesti, kun antiderivaatta sisältää logaritmeja tai käänteisiä trigonometrisia funktioita. Lisäksi joskus lisätään paloittain määritettyjä vakiolausekkeita, jotta antiderivaatasta saadaan kelvollinen suuremmalle välille kuin tavanomainen kaava.

∫() antaa vastauksena itsensä niille *Laus I:n* paloille, joita se ei pysty määrittämään sisäänrakennettujen funktioidensa ja operaattoreidensa eksplisiittisenä äärellisenä kombinaationa.



Kun annat argumentit *Ala* ja *Ylä*, laskin yrittää paikallistaa mahdolliset epäjatkuvuudet tai epäjatkovat 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.

$$\int b \cdot e^{-x^2} + \frac{a}{x^2+a^2} dx \quad b \cdot \int e^{-x^2} dx + \tan^{-1}\left(\frac{x}{a}\right)$$

Huom: Vastauksen pakottaminen likimääräiseksi:

Kämmenlaite: Paina  .

Windows®: Paina **Ctrl+Enter**.

Macintosh®: Paina **⌘+Enter**.

iPad®: Pidä **enter** ja valitse .

$$\int_{-1}^1 e^{-x^2} dx \quad 1.49365$$

$\int()$ (integraali)

Katalogi >

$\int()$ -komentoa voidaan sijoittaa sisäkkäin usean integraalin suorittamiseksi. Integroinnin raja-arvot voivat riippua niiden ulkopuolella olevista integrointimuuttujista.

Huomaa: Katso myös **nInt()**, sivu 127.

$$\int_0^a \int_0^x \ln(x+y) \, dy \, dx$$
$$\frac{a^2 \cdot \ln(a)}{2} + \frac{a^2 \cdot (4 \cdot \ln(2) - 3)}{4}$$

$\sqrt{}$ (neliöjuuri)

ctrl **x²** painikkeet

$\sqrt{}$ (Lausl) \Rightarrow lauseke

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

$\sqrt{}$ (Listal) \Rightarrow lista

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

Laskee argumentin neliöjuuren.

Kun kyseessä on lista, laskee kaikkien Listal:n elementtien neliöjuuret.

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

Huomaa: Katso myös **Neliöjuurimalli**, sivu 5.

$\prod()$ (tulo)

Katalogi >

\prod (Lausl, 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 (...)**.

Sieventää Lausl:n jokaisen Muutt:n arvon suhteen väliltä Matala - Korkea ja laskee vastausten tulo.

$$\prod_{k=1}^n (k^2) \quad (n!)^2$$

Huomaa: Katso myös **Kertolaskumalli** (Π), sivu 10.

$$\prod_{n=1}^5 \left\{ \left\{ \frac{1}{n}, n, 2 \right\} \right\} \quad \left\{ \frac{1}{120}, 120, 32 \right\}$$

\prod (Lausl, Muutt, Matala, Matala-1) $\Rightarrow 1$

$$\prod_{k=1}^3 (k) \quad 1$$

\prod (Lausl, Muutt, Matala, Korkea)
 $\Rightarrow 1/\prod$ (Lausl, Muutt, Korkea+1, Matala-1), jos Korkea < Matala-1

Esimerkkien kertolaskukaavat on otettu seuraavasta viitteestä:

$\prod()$ (tulo)

Katalogi >

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 >

$\Sigma(\text{Laus } I, \text{ Muutt}, \text{ Matala}, \text{ Korkea})$
 $\Rightarrow \text{lauseke}$

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

Sieventää *Laus I*:n jokaisen muuttujan *Muutt* arvon suhteen väliltä *Matala - Korkea* ja laskee vastausten summan.

Huomaa: Katso myös **Summamalli**, sivu 9.

$$\sum_{n=1}^5 \left(\frac{1}{n}\right) = \frac{137}{60}$$

$$\sum_{k=1}^n (k^2) = \frac{n \cdot (n+1) \cdot (2 \cdot n+1)}{6}$$

$$\sum_{n=1}^{\infty} \left(\frac{1}{n^2}\right) = \frac{\pi^2}{6}$$

$\Sigma(\text{Laus } I, \text{ Muutt}, \text{ Matala}, \text{ Matala}-I) \Rightarrow 0$

$$\sum_{k=4}^3 (k) = 0$$

$\Sigma(\text{Laus } I, \text{ Muutt}, \text{ Matala}, \text{ Korkea})$
 $\Rightarrow \Sigma(\text{Laus } I, \text{ Muutt}, \text{ Korkea}+I, \text{ Matala}-1)$,
 jos *Korkea* < *Matala*-1

$$\sum_{k=4}^1 (k) = -5$$

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.

$$\sum_{k=4}^1 (k) + \sum_{k=2}^4 (k) = 4$$

 $\Sigma\text{Int}()$

Katalogi >

$\Sigma\text{Int}(\text{NPmt } 1, \text{ NPmt } 2, N, I, PV, [\text{Pmt}], [\text{FV}], [\text{PpY}], [\text{CpY}], [\text{PmtAt}], [\text{pyör.Arvo}]) \Rightarrow \text{arvo}$

$$\Sigma\text{Int}(1, 3, 12, 4.75, 20000, ., 12, 12) = -213.48$$

$\Sigma\text{Int}(NPmt1, NPmt2, amortTable) \Rightarrow arvo$

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 204.

- Jos jätät argumentin Pmt pois, sen oletusarvoksi tulee $Pmt = \mathbf{tvmPmt}(N, I, PV, FV, PpY, CpY, PmtAt)$.
- 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 $\mathbf{amortTbl}()$ kuvatun muotoinen, katso sivu 12.

Huomaa: Katso myös $\Sigma\text{Prn}()$, jäljempänä, sekä $\mathbf{Bal}()$, sivu 21.

$tbl := \mathbf{amortTbl}(12, 12, 4.75, 20000., 12, 12)$

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

 $\Sigma\text{Prn}()$

$\Sigma\text{Prn}(NPmt1, NPmt2, N, I, PV, [Pmt], [FV], [PpY], [CpY], [PmtAt], [pyörArvo]) \Rightarrow arvo$

$\Sigma\text{Prn}(1, 3, 12, 4.75, 20000., 12, 12)$ -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 204.

- 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.

$\Sigma Prn(NPmt1, NPmt2, amortTable)$ laskee pääoman summan lyhennystaulukon *amortTable* perusteella. *amortTable*-argumentin on oltava matriisi, joka on kohdassa **amortTbl()** kuvatun muotoinen, katso sivu 12.


Huomaa: Katso myös $\Sigma Int()$, edellä, sekä **Bal()**, sivu 21.

$tbl:=amortTbl(12,12,4.75,20000.,12,12)$

0	0.	0.	20000.
1	-77.49	-1632.43	18367.57
2	-71.17	-1638.75	16728.82
3	-64.82	-1645.1	15083.72
4	-58.44	-1651.48	13432.24
5	-52.05	-1657.87	11774.37
6	-45.62	-1664.3	10110.07
7	-39.17	-1670.75	8439.32
8	-32.7	-1677.22	6762.1
9	-26.2	-1683.72	5078.38
10	-19.68	-1690.24	3388.14
11	-13.13	-1696.79	1691.35
12	-6.55	-1703.37	-12.02

$\Sigma Prn(1,3,tbl)$ -4916.28

(epäsuora operaattori)

  **painikkeet**

muuttNimiMerkkijono

$\#("x"&"y"&"z")$ xyz

Viittaa muuttujaan, jonka nimi on *muuttNimiMerkkijono*. Tällä operaattorilla voit luoda muuttujanimiä funktion sisältä merkkijonojen avulla.

Luo muuttujan xyz tai viittaa siihen.


$10 \rightarrow r$ 10

$"r" \rightarrow s1$ "r"

$\#s1$ 10

Laskee arvon muuttujalle (r), jonka nimi on tallennettu muuttujaan s1.

E (kymmenpotenssimuoto)

 painike

Laus I^E ⇒ lauseke

23000. 23000.

Syöttää luvun kymmenpotenssimuodossa. Luku tulkitaan seuraavasti: *mantissa* × 10^{eksponentti}.

2300000000.+4.1E15 4.1E15

3·10⁴ 30000

Vinkki: Jos haluat syöttää 10-potenssin ilman, että vastauksena on desimaaliluku, käytä komentoa 10^{kokonaisluku}.

Huomaa: Voit syöttää tämän operaattorin tietokoneen näppäimistöä kirjoittamalla @E. Kirjoita esimerkiksi 2.3@E4, kun haluat syöttää 2.3E4.

g (graadi)

 painike

Laus I^g ⇒ lauseke

Aste-, graadi- tai radiaanikulmatilassa.

Lista I^g ⇒ lista

$\cos(50^g)$ $\frac{\sqrt{2}}{2}$

Matriisi I^g ⇒ matriisi

$\cos\{0,100^g,200^g\}$ $\{1,0,-1\}$

Tämän funktion avulla voit määrittää graadikulman ollessasi aste- tai radiaanikulmatilassa.

Kun laskin on radiaanikulmatilassa, kertoo *Laus I*:n arvolla $\pi/200$.

Kun laskin on astekulmatilassa, kertoo *Laus I*:n arvolla $g/100$.

Graadikulmatilassa antaa vastauksena lausekkeen *Laus I* muuttumattomana.

Huomaa: Voit syöttää tämän symbolin tietokoneen näppäimistöä kirjoittamalla @g.

r (radiaani)

 painike


Laus I^r ⇒ lauseke

Aste-, graadi- tai radiaanikulmatilassa:

Lista I^r ⇒ lista

Matriisi I^r ⇒ matriisi

⌠ (radiaani)

 painike

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 ⌠ , jos haluat pakottaa funktion määritelmän yksiköksi radiaanit riippumatta tilasta, joka on käytössä funktion käytön aikana.

Huomaa: Voit syöttää tämän symbolin tietokoneen näppäimistöltä kirjoittamalla $\text{@}\pi$.

$$\cos\left(\frac{\pi}{4^r}\right) \quad \frac{\sqrt{2}}{2}$$

$$\cos\left(\left\{0^r, \frac{\pi}{12}r, (\pi)r\right\}\right) \quad \left\{1, \frac{(\sqrt{3+1})\cdot\sqrt{2}}{4}, -1\right\}$$

° (aste)

 painike

$\text{Laus } l^\circ \Rightarrow \text{lauseke}$

$\text{Listal } l^\circ \Rightarrow \text{lista}$

$\text{Matriisi } l^\circ \Rightarrow \text{matriisi}$

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 $\text{@}d$.

Aste-, graadi- tai radiaanikulmatilassa:

$$\cos(45^\circ) \quad \frac{\sqrt{2}}{2}$$

Radiaanikulmatilassa:

Huom: Vastauksen pakottaminen likimääräiseksi:

Kämmenlaite: Paina  .


Windows®: Paina **Ctrl+Enter**.

Macintosh®: Paina **⌘+Enter**.

iPad®: Pidä **enter** ja valitse .

$$\cos\left(\left\{0, \frac{\pi}{4}, 90^\circ, 30.12^\circ\right\}\right) \quad \{1., 0.707107, 0., 0.864976\}$$

° , ' , " (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 luku

Laskee $dd+(mm/60)+(ss.ss/3600)$.

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 $\odot <$.

$(Magnitudi \ \angle \ Kulma) \Rightarrow$ kompleksiarvo
(polaarinen syöte)

Syöttää kompleksilukuarvon $(r \angle \theta)$ polaarisisä muodosssa. *Kulma* tulkitaan nykyisen kulmatila-asetuksen mukaisesti.

Radiaanikulmatilassa ja kun vektorimuoto on asetettu valintaan: suorakulma

$$\left[5 \ \angle 60^{\circ} \ \angle 45^{\circ} \right] \left[\frac{5 \cdot \sqrt{2}}{4} \ \frac{5 \cdot \sqrt{6}}{4} \ \frac{5 \cdot \sqrt{2}}{2} \right]$$

sylinteri

$$\left[5 \ \angle 60^{\circ} \ \angle 45^{\circ} \right] \left[\frac{5 \cdot \sqrt{2}}{2} \ \angle \frac{\pi}{3} \ \frac{5 \cdot \sqrt{2}}{2} \right]$$

pallo

$$\left[5 \ \angle 60^{\circ} \ \angle 45^{\circ} \right] \left[5 \ \angle \frac{\pi}{3} \ \angle \frac{\pi}{4} \right]$$



Radiaanikulmatilassa ja suorakulmakompleksimuodossa:

$$5+3 \cdot i \cdot \left(10 \ \angle \frac{\pi}{4} \right) \quad 5-5 \cdot \sqrt{2} + (3-5 \cdot \sqrt{2}) \cdot i$$

∠ (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)

KatsoTyhjät elementit, sivu 248.

_ (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

Voit käyttää valmiiksi määritettyjä yksiköitä tai luoda omia yksiköitä.

Katalogista. Napsauta ensin  ja sen jälkeen **Matemaattiset operaattorit**.

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[^](Lauseke) ⇒ lauseke

$10^{1.5}$	31.6228
------------	---------

10[^](Lista) ⇒ 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 *Lista*:n elementtien potenssiin.

10[^](neliomatriisi) ⇒ neliomatriisi

Laskee luvun 10 korotettuna *neliomatriisi*:n potenssiin. Tämä ei ole sama kuin kunkin elementin 10-potenssiin korottamisen laskeminen.

Laskentamenetelmä on kuvattu kohdassa **cos()**.

neliomatriisi:n on oltava diagonalisoitavissa. Vastaus sisältää aina liukulukuja.

$10^{\begin{bmatrix} 1 & 5 & 3 \\ 4 & 2 & 1 \\ 6 & -2 & 1 \end{bmatrix}}$	$\begin{bmatrix} 1.14336\text{E}7 & 8.17155\text{E}6 & 6.67589\text{E}6 \\ 9.95651\text{E}6 & 7.11587\text{E}6 & 5.81342\text{E}6 \\ 7.65298\text{E}6 & 5.46952\text{E}6 & 4.46845\text{E}6 \end{bmatrix}$
---	--

^{^-1}(käänteisluku)

Katalogi >

Lauseke ^{^-1} ⇒ lauseke

$(3.1)^{-1}$	0.322581
--------------	----------

Lista ^{^-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 *Lista*:n elementtien käänteisluvut.

neliomatriisi ^{^-1} ⇒ neliomatriisi

Laskee *neliomatriisi*:n käänteisluvun.

neliomatriisi:n oltava ei-singulaarinen neliomatriisi.

$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}^{-1}$	$\begin{bmatrix} -2 & 1 \\ 3 & -1 \\ 2 & 2 \end{bmatrix}$
$\begin{bmatrix} 1 & 2 \\ a & 4 \end{bmatrix}^{-1}$	$\begin{bmatrix} -2 & 1 \\ a-2 & a-2 \\ a & -1 \\ 2 \cdot (a-2) & 2 \cdot (a-2) \end{bmatrix}$

Laus | Booleanlaus1 [andBooleanlaus2]...

$$x+1|x=3 \quad 4$$

Laus | Booleanlaus1 [orBooleanlaus2]...

$$x+y|x=\sin(y) \quad \sin(y)+y$$

$$x+y|\sin(y)=x \quad x+y$$

Rajoittava ("|")-symboli toimii binaarisena operaattorina. Operaattorin | vasemalla 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.

$$x^3-2\cdot x+7 \rightarrow f(x) \quad Done$$

$$f(x)|x=\sqrt{3} \quad \sqrt{3}+7$$

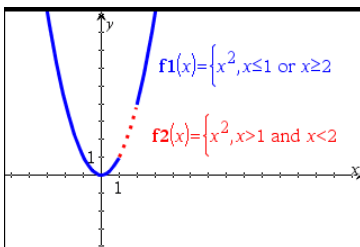
$$(\sin(x))^2+2\cdot\sin(x)-6|\sin(x)=d \quad d^2+2\cdot d-6$$

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.



$$\text{solve}(x^2-1=0,x)|x>0 \text{ and } x<2 \quad x=1$$

$$\sqrt{x}\cdot\sqrt{\frac{1}{x}}|x>0 \quad 1$$

$$\sqrt{x}\cdot\sqrt{\frac{1}{x}} \quad \sqrt{\frac{1}{x}}\cdot\sqrt{x}$$



| (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.

$$\text{solve}(x^2-1=0,x)|x\neq 1 \quad x=-1$$

→ (tallenna)

  painike

Laus → *Muutt*

Lista → *Muutt*

Matriisi → *Muutt*

Laus → *Funktio(Param1,...)*

Lista → *Funktio(Param1,...)* *Matriisi* → *Funktio(Param1,...)*

$\frac{\pi}{4}$	→ myvar	$\frac{\pi}{4}$
$2 \cdot \cos(x)$	→ y1(x)	Done
{1,2,3,4}	→ lst5	{1,2,3,4}
$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}$	→ matg	$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}$
"Hello"	→ str1	"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.

$myvar := \frac{\pi}{4}$	$\frac{\pi}{4}$
$y1(x) := 2 \cdot \cos(x)$	<i>Done</i>
$lst5 := \{1, 2, 3, 4\}$	$\{1, 2, 3, 4\}$
$matg := \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}$	$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}$
$str1 := "Hello"$	"Hello"

© (kommentti)

ctrl  painikkeet

© [*teksti*]

© käsittelee *tekstin* kommenttirivinä, minkä avulla voit lisätä merkintöjä luomiisi 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 määrittysten 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

0b, 0h**0b** *binaariluku***0h** *heksadesimaaliluku*

Määrittää binaari- (0b) tai heksadesimaaliluvun (0h). Syöttääksesi binaari- tai heksadesimaaliluvun sinun on syötettävä etuliite 0b tai 0h riippumatta kantalukutilasta. Ilman etuliitettä lukua käsitellään desimaalilukuna (kantaluku 10).

Tulokset näytetään kantalukutilan mukaisesti.

Desimaalisessa kantalukutilassa:

$0b10+0hF+10$	27
---------------	----

Binaarisessa kantalukutilassa:

$0b10+0hF+10$	0b11011
---------------	---------

Heksadesimaalisessa kantalukutilassa:



$0b10+0hF+10$	0h1B
---------------	------

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 taulukkodatasta.

Funktion **delVoid()** avulla voit poistaa tyhjä elementit listasta. Funktion **isVoid()** avulla voit testata tyhjää elementtiä. Lisätietoja, katso **delVoid()**, sivu 56, ja **isVoid()**, sivu 99.

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.

$l1:=\{1,2,3,4,5\}; l2:=\{2,_,3,5,6,6\}$	$\{2,_,3,5,6,6\}$
LinRegMx l1,l2	Done
stat.Resid	$\{0.434286,_, -0.862857, -0.011429, 0.44\}$
stat.XReg	$\{1,_,3,4,5\}$
stat.YReg	$\{2,_,3,5,6,6\}$
stat.FreqReg	$\{1,_,1,1,1,1\}$

Poisjätetty luokka regressioissa aiheuttaa tyhjän elementin jäännöksen vastaavalle elementille.

$l1:=\{1,3,4,5\}; l2:=\{2,3,5,6,6\}$	$\{2,3,5,6,6\}$
cat:={ "M", "M", "F", "F" }; incl:={ "F" }	{ "F" }
LinRegMx l1,l2,1,cat,incl	Done
stat.Resid	$\{_,_,0,0,0\}$
stat.XReg	$\{_,_,4,5\}$
stat.YReg	$\{_,_,5,6,6\}$
stat.FreqReg	$\{_,_,1,1,1\}$

Frekvenssi 0 regressioissa aiheuttaa tyhjän elementin jäännöksen vastaavalle elementille.

$l1:=\{1,3,4,5\}; l2:=\{2,3,5,6,6\}$	$\{2,3,5,6,6\}$
LinRegMx l1,l2,{1,0,1,1}	Done
stat.Resid	$\{0.069231,_, -0.276923, 0.207692\}$
stat.XReg	$\{1,_,4,5\}$
stat.YReg	$\{2,_,5,6,6\}$
stat.FreqReg	$\{1,_,1,1,1\}$

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:
π	<code>pi</code>
θ	<code>theta</code>
∞	<code>infinity</code>
\leq	<code><=</code>
\geq	<code>>=</code>
\neq	<code>/=</code>
\Rightarrow (looginen seuraus)	<code>=></code>
\Leftrightarrow (looginen kaksoisseuraus, XNOR)	<code><=></code>
\rightarrow (tallennusoperaattori)	<code>=:</code>
<code> </code> (itseisarvo)	<code>abs (...)</code>
$\sqrt{\quad}$	<code>sqrt (...)</code>
$d(\quad)$	<code>derivative (...)</code>
$\int(\quad)$	<code>integral (...)</code>
$\Sigma(\quad)$ (yhteenlaskumalli)	<code>sumSeq (...)</code>
$\Pi(\quad)$ (tulon malli)	<code>prodSeq (...)</code>
$\sin^{-1}(\quad), \cos^{-1}(\quad), \dots$	<code>arcsin (...), arccos (...), ...</code>
<code>ΔList()</code>	<code>deltaList (...)</code>
<code>ΔtmpCnv()</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>

Syötettävä kohde:	Pikavalinta:
° (asteet)	@d
° (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 (\times^r), alaindeksi ([]), transponointi (\top)
 - 5 Potenssiin korotus, potenssioperaattori (^)
 - 6 Negaatio (-)
 - 7 Merkkijonojen ketjutus (&)
 - 8 Kertolasku (•), jakolasku (/)
 - 9 Yhteenlasku (+), vähennyslasku (-)
 - 10 Yhtäläisyysuhteet: on yhtä kuin (=), ei ole yhtä kuin (\neq tai \neq), pienempi kuin (<), pienempi tai yhtä suuri kuin (\leq tai \leq), suurempi tai yhtä suuri kuin (>), suurempi tai yhtä suuri kuin (\geq tai \geq)
 - 11 Looginen **not**
 - 12 Looginen **and**
 - 13 Looginen **or**
 - 14 **xor, nor, ei-ja**
 - 15 Looginen seuraus (\Rightarrow)
 - 16 Looginen kaksoisseuraus, XNOR (\Leftrightarrow)
 - 17 Rajoittava operaattori ("|")
 - 18 Tallenna (\rightarrow)
-

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 komennoksi 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 $2^{(3^2)}$, josta saadaan 512. Tämä on eri asia kuin $(2^3)^2$, 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.

Virhekoodit ja viestit

Kun tapahtuu virhe, virhekoodi määritetään muuttujaan *errCode*. Käyttäjän määrittämällä ohjelmilla ja funktioilla voidaan tutkia *errCode*-muuttujaa virheen syyn määrittämiseksi. Esimerkki muuttujan *errCode* käytöstä on esitetty Esimerkissä 2 **Try**-komennon kohdalla, sivu 200.

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 hakuvälillä määritettäessä.

180 Keskeytys

Näppäintä  tai  on painettu pitkän laskutoimituksen tai ohjelman suorituksen aikana.

Virhekoodi Kuvaus

- 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 Datatyyppi 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 nolllalla
- 260 Määrittelyjoukkovirhe
- Argumentin on kuuluttava tiettyyn määrittelyjoukkoon. Esimerkiksi **rand(0)** ei kelpaa.
- 270 Muuttujan nimi jo käytössä
- 280 Else ja Elseif eivät kelpaa lohkon If..EndIf ulkopuolella
- 290 EndTry-komenosta puuttuu sopiva Else-lauseke
- 295 Liian monta iteraatiota
- 300 Odotettavissa 2- tai 3-elementtinen lista tai matriisi
- 310 Funktion **nSolve** ensimmäisen argumentin on oltava yhden muuttujan yhtälö. Se ei voi sisältää muuta arvotonta muuttujaa kuin käsittelyn kohteena olevan muuttujan.
- 320 solve- tai 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 Indeksialueen ulkopuolella
- 360 Epäsuora merkijono 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 **Local**-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 polaarinen 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 lohkoksta lf.. Endlf
- 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 Argumenttivirhe
- 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

Virhekoodi Kuvaus

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:

- sto →
- :=
- **Define**

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:

- **Define**
- :=
- sto →

määrittääksesi funktion.

1100 Ei-reaalinen laskutoimitus

Esimerkiksi, jos ohjelmisto on Real-asetuksessa, $\sqrt{(-1)}$ ei kelpaa.

Virhekoodi Kuvaus

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 Argumenttinvirhe

Ensimmäisen argumentin on oltava toisen argumentin sisällä oleva polynomi lauseke. Jos toinen argumentti jätetään pois, ohjelmisto yrittää valita oletusarvon.

1150 Argumenttinvirhe

Ensimmäisten kahden argumentin on oltava kolmannen argumentin sisällä olevia polynomi lausekkeita. Jos kolmas argumentti jätetään pois, ohjelmisto yrittää valita oletusarvon.

1160 Virheellinen kirjaston polkunimi

Polkunimen on oltava muodossa xxx\yyy, jossa:

- xxx-osassa voi olla 1-16 merkkiä.
- yyy-osassa voi olla 1–15 merkkiä.

Lisätietoja löydät ohjekirjan osasta Kirjasto.

1170 Kirjaston polkunimeä on käytetty väärin

- Polkunimelle ei voi määrittää arvoa komennon **Define**, := tai sto → avulla.
- Polkunimeä ei voi määrittää paikalliseksi muuttujaksi tai käyttää funktion tai ohjelman parametrina.

1180 Virheellinen kirjastomuuttujan nimi.

Tarkista, että nimi:

- ei sisällä pistettä
- ei ala alaviivalla
- ei ylitä 15 merkkiä

Lisätietoja löydät ohjekirjan osasta Kirjasto.

1190 Kirjastodokumenttia ei löydy:

- Tarkista, että kirjasto on MyLib-kansiossa.
- Päivitä kirjastot.

Lisätietoja löydät ohjekirjan osasta Kirjasto.

1200 Kirjastomuuttujaa ei löydy:

- Tarkista, että kirjaston ensimmäisessä tehtävässä on kirjastomuuttuja.
- Tarkista, että kirjastomuuttujaksi on määritetty LibPub tai LibPriv.
- Päivitä kirjastot.

Lisätietoja löydät ohjekirjan osasta Kirjasto.

Virhekoodi Kuvaus

- 1210 Virheellinen kirjaston pikavalintanimi.
- 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 209.

Varoituskoodi	Viesti
10000	Laskutoimitus voi antaa väärää ratkaisuja.
10001	Yhtälön derivointi voi antaa väärän yhtälön.
10002	Kyseenalainen ratkaisu
10003	Kyseenalainen tarkkuus
10004	Laskutoimituksessa voi hävitä ratkaisuja.
10005	<code>cSolve</code> voi määrittää lisää nollakohtia.
10006	<code>Solve</code> voi määrittää lisää nollia.
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">• <code>solve(Yhtälö, Muut=Arvaus) alaraja<Muut<yläraja</code>• <code>solve(Yhtälö, Muut) alaraja<Muut<yläraja</code>• <code>solve(Yhtälö, Muut=Arvaus)</code>
10008	Tuloksen määrittelyjoukko voi olla pienempi kuin syötteen määrittelyjoukko.
10009	Tuloksen määrittelyjoukko voi olla suurempi kuin syötteen määrittelyjoukko.
10012	Ei-reaalinen laskutoimitus
10013	∞^0 tai undef^0 korvattu arvolla 1
10014	undef^0 korvattu arvolla 1
10015	1^∞ tai 1^undef korvattu arvolla 1
10016	1^undef korvattu arvolla 1
10017	Laskualueen ylitys korvattu arvolla ∞ tai $-\infty$
10018	Laskutoimitus vaatii ja laskee 64-bittisen arvon.
10019	Muisti täynnä, sievennystä ei mahdollisesti suoriteta loppuun.
10020	<code>Trig</code> -funktion argumentti on liian suuri täsmälliseen sievennykseen.
10021	Syöte sisältää määrittämättömän parametrin.

Varoituskoodi	Viesti
	Tulos ei ehkä ole voimassa kaikille mahdollisille parametrialueille.
10022	Ratkaisu on mahdollinen, kun määritetään oikeat ala- ja ylärajat.
10023	Skalaari on kerrottu identtisellä matriisilla.
10024	Tulos saatu käyttämällä likimääräistä aritmetiikkaa.
10025	Ekvivalenssia ei voida varmistaa EXACT-tilassa.
10026	Ehto saatetaan ohittaa. Määrittele ehto muodossa "\' 'Muuttuja MatemaattinenTestiSymboli Vakio' tai näiden muotojen yhdistelmänä, esimerkiksi 'x<3 ja x>-12'

Huolto ja Asiakastuki

TI-tuotteiden huolto- ja takuutietoa

Tietoa TI-tuotteista ja niiden huollostar

Lisätietoja TI-tuotteista ja niiden huollosta saa sähköpostin kautta tai TI-laskimien kotisivulta.

sähköpostiosoite: ti-cares@ti.com

internet-osoite: education.ti.com

Huolto- ja takuutietoa

Tietoja takuuajan kestosta ja takuuehdoista sekä tuotteen huollosta löytyy tuotteen mukana seuraavasta takuuselosteesta tai paikalliselta Texas Instruments-vähittäismyyjältä/jälleenmyyjältä.

Index

	-	
- , vähennyslasku[*]		220
	!	
!, kertoma		231
	"	
", sekuntimuoto		240
	#	
#, epäsuora operaattori		237, 253
	%	
%, prosentti		226
	&	
&, liitä		231
	*	
*, kertolasku		221
	,	
, jaoton		241
, minuuttimuoto		240
	.	
., piste-erotus		224
.*, pistetulo		225
./, pisteosamäärä		225
.^, pistepotenssi		225
., pistesumma		224
	:	
:=, määritä		246
	^	
\wedge^{-1} , käänteisluku		243

\wedge , potenssi	223
$-$	
$_$ yksikön merkki	241
$_$ yksikön nimi	241
$ $	
$ $, (rajoittava operaattori)	244
$+$	
$+$, yhteenlasku	220
$/$	
$/$, jakolasku[*]	222
\neq	
\neq , ei yhtä kuin[*]	227
$=$	
$=$, on yhtä kuin	226
$>$	
$>$, suurempi kuin	229
Π	
Π , tulo[*]	234
Σ	
$\Sigma()$, summa[*]	235
$\Sigma\text{Int}()$	235
$\Sigma\text{Prn}()$	236
$\sqrt{\quad}$	
$\sqrt{\quad}$, neliöjuuri[*]	234
\int	
\int , integraali[*]	232

\leq , pienempi tai yhtä suuri kuin	228
\geq , suurempi tai yhtä suuri kuin	229
\blacktriangleright , muunna graadikulmaksi[Grad]	91
\blacktriangleright , muunna yksiköt[*]	242
\blacktriangleright . asteen regressio, QuartReg	148
\blacktriangleright approxFraction()	18
\blacktriangleright cos, näytä kulman kosini[cos]	35
\blacktriangleright Cylind, näytä vektorina sylinterikoordinaatistossa[Cylind]	49
\blacktriangleright DD, näytä desimaalikulmana[DD]	52
\blacktriangleright Desimaali, näytä tulos desimaalilukuna[Desimaali]	53
\blacktriangleright DMS, näytä asteina/ minuutteina/ sekunteina[DMS]	60
\blacktriangleright exp, näytä e:n arvolla[exp]	70
\blacktriangleright Kantaluku10, näytä desimaalikonaislukuna[Kantaluku10]	23
\blacktriangleright Kantaluku16, näytä heksadesimaalilukuna[Kantaluku16]	24
\blacktriangleright Kantaluku2, näytä binaarisena[Kantaluku2]	22
\blacktriangleright Polaarinen, näytä polaarisenä vektorina[Polaarinen]	138
\blacktriangleright Rad, muunna radiaanikulmaan	149
\blacktriangleright Rect, näytä suorakulmavektorina	153
\blacktriangleright sin, näytä kulman sini[sin]	175
\blacktriangleright Sphere, näytä pallonmuotoisena vektorina[Sphere]	183
\Rightarrow , looginen seuraus[*]	230, 250
\rightarrow , tallenna	245
\Leftrightarrow , looginen kaksoisseurauk [*]	230
©, kommentti	246
°, astekulmamuuoto[*]	239

°, astetta/minuuttia/sekuntia[*]	240
0	
Ob, binaarinen indikaattori	247
Oh, heksadesimaalinen indikaattori	247
1	
10^(), kymmenen potenssi	243
2	
2 otoksen F-testi	82
2. asteen regressio, QuadReg	146
3	
3. asteen regressio, CubicReg	47
A	
abs(), itseisarvo	12
alaraja, floor()	78
alaviiva, _	241
alimatriisi, subMat()	188, 190
amortTb(), lyhennystaulukko	12, 21
and, Boolean operator	13
angle(), kulma	14
ANOVA, yksisuuntainen varianssianalyysi	14
ANOVA2way, kaksisuuntainen varianssianalyysi	15
ans, viimeinen vastaus	17
approx(), likimääräinen	17
approxRational()	18
arccos()	18
arccosh()	18
arccot()	18
arccoth()	19
arccsc()	19
arccsch()	19
arcLen(), kaaren pituus	19
arcsec()	19
arcsech()	19
arcsin()	19
arcsinh()	20
arctan()	20

arctanh()	20
arkuskosini, $\cos^{-1}()$	37
arkussini, $\sin^{-1}()$	176
arkustangentti, $\tan^{-1}()$	191
asetta	
tila, setMode()	170
asetukset, hae nykyiset	87
aste/minuutti/sekunti-muoto	240
astekulmamuoto, -	239
augment(), lisää/ketjuta	20
avgRC(), keskimääräinen muutosnopeus	20

B

binaarinen	
indikaattori, 0b	247
näytä, 4Kantaluku2	22
binomCdf()	24, 96-97
binomPdf()	25
Boolean operators	
and	13
Booleen operaattorit	
\Rightarrow	230, 250
\Leftrightarrow	230
ei	130
ei-ja	124
eikä	128
tai	135
xor	210

C

Cdf()	76
ceiling(), yläraja	25
centralDiff()	26
cFactor(), kompleksilukutekijä	26
char(), merkkijono	27
charPoly()	28
χ^2 2way	28
clearAZ	30
ClrErr, poista virhe	30
colAugment	31
colDim(), matriisin sarakemitat	31
colNorm(), matriisin sarakenormi	31

<code>comDenom()</code> , yhteinen nimittäjä	32
<code>completeSquare()</code> , <code>complete square</code>	33
<code>conj()</code> , liittokompleksiluku	33
<code>constructMat()</code> , luo matriisi	34
<code>corrMat()</code> , korrelaatiomatriisi	35
<code>cos⁻¹()</code> , arkuskosini	37
<code>cos()</code> , kosini	36
<code>cosh⁻¹()</code> , hyperbolinen arkuskosini	38
<code>cosh()</code> , hyperbolinen kosini	38
<code>cot⁻¹()</code> , arkuskotangenti	39
<code>cot()</code> , kotangenti	39
<code>coth⁻¹()</code> , hyperbolinen arkuskotangenti	40
<code>coth()</code> , hyperbolinen kotangenti	40
<code>count()</code> , laske listan kohtien lukumäärä	40
<code>countif()</code> , laske listan yksiköiden lukumäärä ehdollisesti	41
<code>cPolyRoots()</code>	42
<code>crossP()</code> , ristitulo	42
<code>csc⁻¹()</code> , käänteiskosekanti	43
<code>csc()</code> , kosekanti	42
<code>csch⁻¹()</code> , käänteinen hyperbolinen kosekanti	43
<code>csch()</code> , hyperbolinen kosekanti	43
<code>cSolve()</code> , ratkaise kompleksiyhtälö	44
<code>CubicReg</code> , 3. asteen regressio	47
<code>cumulativeSum()</code> , kumulatiivinen summa	48
<code>cycle</code> , <code>Cycle</code>	48
<code>Cycle</code> , <code>cycle</code>	48
<code>cZeros()</code> , kompleksiset nollakohdat	49

D

<code>d()</code> , ensimmäinen derivaatta	231
<code>dbd()</code> , päivämäärien väliset päivät	52
<code>Define</code> (Määritä)	53
<code>Define LibPriv</code> (Määritä LibPriv)	54
<code>Define LibPub</code> (Määritä LibPub)	55
<code>Define</code> , määritä	53
<code>deltaList()</code>	55
<code>deltaTmpCnv()</code>	55
<code>DelVar</code> , poista muuttuja	56
<code>delVoid()</code> , poista tyhjät elementit	56
derivaatat	
ensimmäinen derivaatta, <code>d()</code>	231
numeerinen derivaatta, <code>nDeriv()</code>	127

numeerinen derivaatta, nDerivative()	126
derivaatta tai n:s derivaatta	
malli	10
desimaali	
kokonaislukunäyttö, 4Kantaluku10	23
kulmanäyttö, ►DD	52
deSolve(), ratkaisu	57
det(), matriisin determinantti	59
diag(), diagonaalimatriisi	59
dim(), mitta	59
Disp, näytä tiedot	60, 166
dominantTerm(), dominanttitermi	62
dominanttitermi, dominantTerm()	62
dotP(), pistetulo	63

E

e eksponentti	
malli	6
e korotettuna potenssiin, e^()	64, 70
E, eksponentti	238
e, näytä lauseke e:n arvolla	70
e^(), e korotettuna potenssiin	64
efektiivinen korko, eff()	64
eff), muunna nimelliskorko efektiiviseksi koroksi	64
ei-ja, Boolean operaattori	124
ei yhtä kuin, ≠	227
ei, Boolean operaattori	130
eigVc(), ominaisvektori	65
eigVl(), ominaisarvo	65
eikä, Boolean operaattori	128
eksklusioio " "-operaattorilla	244
eksponentiaalinen regressio, ExpReg	73
eksponentit	
malli	5
eksponentti, E	238
else if, Elsef	66
Elsef, else if	66
end	
for, EndFor	79
funktio, EndFunc	83
ohjelma, EndPrgm	143
silmukka, EndLoop	114
try, EndTry	200

while, EndWhile	210
end while, EndWhile	210
EndTry, kokeilun loppu	200
EndWhile, end while	210
ensimmäinen derivaatta	
malli	10
EOS (yhtälökäyttöjärjestelmä)	252
epäsuora operaattori (#)	253
epäsuora operaattori, #	237
etumerkki, sign()	173
euler(), Euler function	67
exact(), täsmällinen	69
Exit, lopeta	69
exp(), e korotettuna potenssiin	70
exp►list(), lauseke listaksi	71
expand(), lavenna	71
expr(), merkkijono lausekkeeksi	73, 112
ExpReg, eksponentiaalinen regressio	73

F

factor(), tekijä	74
Fill, matriisin täyttö	76
FiveNumSummary	77
floor(), alaraja	78
fMax(), funktion maksimi	78
fMin(), funktion minimi	79
For	79
for, For	79
For, for	79
format(), merkkijonon muoto	80
fpart(), funktion osa	81
freqTable()	81
frequency()	82
Frobeniusin normi, norm()	129
Func, funktio	83
Func, ohjelmafunktio	83
funktio loppu, EndFunc	83
funktio	
käyttäjän määrittämät	53
maksimi, fMax()	78
minimi, fMin()	79
ohjelmafunktio, Func	83
osa, fpart()	81

funktiot ja muuttujat	
kopioiminen	34

G

g, graadit	238
gcd(), suurin yhteinen jakaja	84
geomCdf()	84
geomPdf()	85
Get	85
getDenom(), hae/laske nimittäjä	86
getLangInfo(), hae/anna kielitiedot	86
getLockInfo(), testaa muuttujan tai muuttujaryhmän lukitustilan	87
getMode(), hae tila-asetukset	87
getNum(), hae/laske luku	88
GetStr	89
getType(), get type of variable	89
getVarInfo(), hae/laske muuttujien tiedot	89
Goto, siirry	90
graadimuoto, g	238

H

hae/laske	
luku, getNum()	88
muuttujien tiedot, getVarInfo()	86, 89
nimittäjä, getDenom()	86
heksadesimaali	
näytä, 4Kantaluku16	24
heksadesimaalinen	
indikaattori, Oh	247
hyperbolinen	
areatangenti, $\tanh^{-1}()$	193
arkuskosini, $\cosh^{-1}()$	38
arkussini, $\sinh^{-1}()$	177
kosini, $\cosh()$	38
sini, $\sinh()$	177
tangenti, $\tanh()$	192

I

identity(), identtinen matriisi	91
identtinen matriisi, identity()	91
If, jos	91
iffn()	92

imag(), imaginaarinen osa	93
imaginaarinen osa, imag()	93
ImpDif(), implisiittinen derivaatta	93
implisiittinen derivaatta, Impdif()	93
Input, syöte	93
inString(), merkkijonon sisällä	94
int(), kokonaisluku	94
intDiv(), kokonaisluvun jakaminen	95
integraali, ‰	232
interpolate(), interpoloi	95
invF()	96
invNorm(), käänteinen kumulatiivinen normaalijakauma)	97
invt()	97
Inv χ^2 ()	96
iPart(), kokonaisosa	97
irr(), sisäinen palautuksen määrä sisäinen palautuksen määrä, irr()	98
isPrime(), jaottoman testi	98
isVoid(), tyhjän testi	99
itseisarvo malli	8

J

jakakaumafunktiot	
binomCdf()	96-97
invNorm()	97
invt()	97
jakaumafunktiot	
binomCdf()	24
binomPdf()	25
Inv χ^2 ()	96
normCdf()	130
normPdf()	130
poissCdf()	138
poissPdf()	138
tCdf()	194
tPdf()	199
χ^2 2way()	28
χ^2 Cdf()	29
χ^2 GOF()	29
χ^2 Pdf()	30
jakolasku, P	222
jaoton,	241

jaottoman luvun testi, isPrime()	98
jos käytettävissäsi on	244
jos, EndIf	91
jos, If	91
jäännös, remain()	156

K

kaaren pituus, arcLen()	19
kahden muuttujan tulokset, TwoVar	204
kertolasku, *	221
kertoma, !	231
keskiarvo, mean()	116
keskihajonta, stdDev()	186-187, 207
keskimmäinen merkkijono, mid()	119
keskimääräinen muutosnopeus, avgRC()	20
kieli	
hae kielitiedot	86
kierrä, rotate()	160-161
kirjasto	
luo pikavalinnat objekteihin	100
kokonaisluku, int()	94
kokonaisluvun jakaminen, intDiv()	95
kokonaisuosa, iPart()	97
kombinaatiot, nCr()	125
kommentti, ©	246
kompleksi	
liitto, conj()	33
nollakohdat, cZeros()	49
ratkaise, cSolve()	44
tekijä, cFactor()	26
kopioi muuttuja tai funktio, CopyVar	34
korkoerien summa	235
korrelaatiomatriisi, corrMat()	35
korvaaminen " " -operaattorilla	244
kosini	
näytä kulman	35
kosini, cos()	36
kotangentti, cot()	39
kulma, angle()	14
kumulatiivinen summa, cumulativeSum()	48
kun, when()	209
kymmenen potenssi, 10^()	243
käyttäjän määrittämät funktiot	53

käyttäjän määrittämät funktiot ja ohjelmat	54-55
käänteinen kumulatiivinen normaalijakauma (invNorm())	97
käänteinen, \wedge^{-1}	243
käänteisluku, \wedge^{-1}	243

L

lajitteleminen	
laske järjestys, SortD	183
nouseva järjestys, SortA	183
laske listan kohtien lukumäärä, count()	40
laske listan yksiköiden lukumäärä ehdollisesti, countif()	41
laske polynomi, polyEval()	140
laske päivämäärien väliset päivät, dbd()	52
lausekkeet	
lauseke listaksi, exp►list()	71
merkkijono lausekkeeksi, expr()	73, 112
lavenna, expand()	71
Lbl, tunnus	99
lcm, pienin yhteinen jaettava	99
left(), vasen	100
LibPriv	54
LibPub	55
libShortcut(), luo pikavalinnat kirjasto-objekteihin	100
liitä, &	231
likimääräinen, approx()	17
limit() tai lim(), raja-arvo	101
lineaarinen regressio, LinRegAx	103
lineaarinen regressio, LinRegBx	102, 104
LinRegBx, lineaarinen regressio	102
LinRegMx, lineaarinen regressio	103
LinRegtIntervals, lineaarinen regressio	104
LinRegtTest	105
linSolve()	106
Δlist(), listaerotus	107
list►mat(), listasta matriisiksi	107
lista, laske kohtien lukumäärä	40
lista, laske yksiköiden lukumäärä ehdollisesti	41
listasta matriisiksi, list►mat()	107
listat	
erotukset listassa, @list()	107
erotus, @list()	107
keskimäinen merkkijono, mid()	119
kumulatiivinen summa, cumulativeSum()	48

lajittele laskevaan järjestykseen, SortD	183
lajittele nousevaan järjestykseen, SortA	183
lauseke listaksi, exp►list()	71
listasta matriisiksi, list►mat()	107
lisää/ketjuta, augment()	20
maksimi, max()	116
matriisi listaksi, mat►list()	115
minimi, min()	119
pistetulo, dotP()	63
ristitulo, crossP()	42
tulo, product()	144
tyhjät elementit	248
uusi, newList()	126
yhteenlasku, sum()	189
lisää/ketjuta, augment()	20
ln(), luonnollinen logaritmi	108
LnReg, logaritminen regressio	109
Local, paikallinen muuttuja	110
Lock, lukitse muuttuja tai muuttujaryhmä	110
Logaritmi	
malli	6
logaritminen regressio, LnReg	109
logaritmit	108
Logistic, logistinen regressio	112
LogisticD, logistinen regressio	113
logistinen regressio, Logistic	112
logistinen regressio, LogisticD	113
looginen kaksoisseuraus, ⇔	230
looginen seuraus, ⇒	230, 250
Loop, silmukka	114
lopeta, Exit	69
loppu	91
loppu jos, EndIf	91
LU, matriisin ala-ylä-dekomponointi	115
lukujono	
kierrä, rotate()	160-161
oikea, right()	158-159
lukujonot	
oikea, right()	95
luo matriisi, constructMat()	34
luonnollinen logaritmi, ln()	108
lyhennystaulukko, amortTbl()	12, 21

M

maksimi, <code>max()</code>	116
mallit	
derivaatta tai n:s derivaatta	10
eksponentti	6
eksponentti	5
ensimmäinen derivaatta	10
itseisarvo	8
Logaritmi	6
matriisi (1 × 2)	8
matriisi (2 × 1)	9
matriisi (2 × 2)	8
matriisi (m × n)	9
murtoluku	5
määrittämätön integraali	11
määrätty integraali	11
n:s juuri	6
nelöjuuri	5
paloittain määritelty funktio (2-osainen)	6
paloittain määritelty funktio (N-osainen)	7
raja-arvo	11
sum (G)	9
toinen derivaatta	10
tulo (P)	10
yhtälöpari (2 yhtälöä)	7
yhtälöryhmä (N yhtälöä)	8
<code>mat▶list()</code> , matriisi listaksi	115
matriisi (1 × 2)	
malli	8
matriisi (2 × 1)	
malli	9
matriisi (2 × 2)	
malli	8
matriisi (m × n)	
malli	9
matriisi listaksi, <code>mat▶list()</code>	115
matriisit	
ala-ylä-dekomponointi, LU	115
alimatriisi, <code>subMat()</code>	188, 190
determinantti, <code>det()</code>	59
diagonaali, <code>diag()</code>	59
identtinen, <code>identity()</code>	91
kumulatiivinen summa, <code>cumulativeSum()</code>	48

listasta matriisiksi, list►mat()	107
lisää/ketjuta, augment()	20
maksimi, max()	116
matriisi listaksi, mat►list()	115
minimi, min()	119
mitta, dim()	59
ominaisarvo, eigVl()	65
ominaisvektori, eigVc()	65
piste-erotus, .-	224
pisteosamäärä, ./	225
pistepotenssi, .^	225
pistesumma, .+	224
pistetulo, .*	225
QR-hajottaminen, QR	146
rivi-echelon-muoto, ref()	153
rivien kertominen ja yhteenlasku, mRowAdd()	121
rivien yhteenlasku, rowAdd()	163
rivikoko, rowDim()	163
rivinormi, rowNorm()	163
rivinvaihto, rowSwap()	163
rivioperaatio, mRow()	121
sarakemitat, colDim()	31
sarakenormi, colNorm()	31
satunnainen, randMat()	151
transponoi, T	190
tulo, product()	144
täyttäminen, Fill	76
uusi, newMat()	126
yhteenlasku, sum()	189
max(), maksimi	116
mean(), keskiarvo	116
mediaani-mediaani-suoran regressio, MedMed	117
mediaani, median()	117
median(), mediaani	117
MedMed, mediaani-mediaani-suoran regressio	117
merkit	
merkkijono, char()	27
numeerinen koodi, ord()	136
merkkijono	
mitta, dim()	59
pituus	59
merkkijono, char()	27
merkkijonon muoto, format()	80

merkkijonon pituus	59
merkkijonon sisällä, inString()	94
merkkijonot	
epäsuora operaattori, #	237
keskimmäinen merkkijono, mid()	119
käyttö muuttujanimien luomisessa	253
lauseke merkkijonoksi, string()	188
liitä, &	231
merkkijono lausekkeeksi, expr()	73, 112
merkkijono, char()	27
merkkikoodi, ord()	136
muoto, format()	80
muotoilu	80
siirrä, shift()	172
sisällä, inString()	94
vasen, left()	100
mid(), keskimmäinen merkkijono	119
min(), minimi	119
minimi, min()	119
minuuttimuoto,	240
mirr(), modifioitu sisäinen korkokanta	120
mitta, dim()	59
mod(), modulo	120
modifioitu sisäinen korkokanta, mirr()	120
modulo, mod()	120
Moninkertaisen lineaarisen regression t-testi	123
mRow(), matriisin rivioperaatio	121
mRowAdd(), matriisin rivien kertominen ja yhteenlasku	121
MultReg	121
MultRegIntervals()	122
MultRegTests()	123
murtoluvut	
malli	5
propFrac	145
muunna	
4Grad	91
yksiköt	242
muuntaminen	
►Rad	149
muutoin, Else	91
muuttuja	
nimen luominen merkkijonosta	253

muuttujat	
paikallinen, Local	110
poista kaikki yksikirjaimiset	30
poista, DelVar	56
muuttujat ja funktiot	
kopiointi	34
muuttujat, lukitseminen ja vapauttaminen	87, 110, 207
muuttujien ja muuttujaryhmien lukitseminen	110
muuttujien ja muuttujaryhmien vapauttaminen	207
määrittäminen	
julkinen funktio tai ohjelma	55
yksityinen funktio tai ohjelma	54
määrittämätön integraali	
malli	11
määritä, Define	53
määrätty integraali	
malli	11

N

n:s juuri	
malli	6
nCr(), kombinaatiot	125
nDerivative(), numeerinen derivaatta	126
negatio, negatiivisten lukujen syöttäminen	253
neliöjuuri	
malli	5
neliöjuuri, ‡()	184, 234
nettonykyarvo, npv()	132
newList(), uusi lista	126
newMat(), uusi matriisi	126
nffMax(), numeerisen funktion maksimi	127
nffMin(), numeerisen funktion minimi	127
nimelliskorko, nom()	128
nimittäjä	32
nInt(), numeerinen integraali	127
nollakohdat, zeroes()	211
nom(), muuta efektiivinen korko nimelliskoroksi	128
norm(), Frobeniusin normi	129
normaalijakauman todennäköisyys, normCdf()	130
normaalisuora, normalLine()	129
normalLine()	129
normCdf()	130
normPdf()	130

nPr(), permutaatiot	131
npv(), nettonykyarvo	132
nSolve(), numeerinen ratkaisu	132
numeerinen	
derivaatta, nDeriv()	127
derivaatta, nDerivative()	126
integraali, nInt()	127
ratkaisu, nSolve()	132
näppäimistön pikavalinnat	250
näyttö asteina/ minuutteina/ sekunteina, ▶DMS	60
näyttö pallonmuotoisena vektorina, ▶Sphere	183
näyttö vektorina sylinterikoordinaatistossa, ▶Cylind	49
näytä	
asteina/ minuutteina/ sekunteina, ▶DMS	60
binaarisena, 4Kantaluku2	22
desimaalikokonaislukuna, 4Kantaluku10	23
desimaalikulmana, ▶DD	52
heksadesimaalilukuna, 4Kantaluku16	24
pallonmuotoinen vektori, ▶Sphere	183
polaarinen vektori, 4Polaarinen	138
suorakulmavektorina, ▶Rect	153
vektorina sylinterikoordinaatistossa, 4Cylind	49
näytä tiedot, Disp	60, 166

O

objektit	
luo pikavalinnat kirjastoon	100
Odota-komento	208
ohita virhe, PassErr	137
ohjelmat	
julkisen kirjaston määrittäminen	55
yksityisen kirjaston määrittäminen	54
ohjelmat ja ohjelmointi	
kokeilun loppu, EndTry	200
lopeta ohjelma, EndPrgm	143
näytä I/O-näyttö, Disp	60, 166
poista virhe, ClrErr	30
try, Try	200
ohjelmointi	
määritä ohjelma, Prgm	143
näytä tiedot, Disp	60, 166
ohita virhe, PassErr	137
oikea, right()	95, 158-159

ominisarvo, eigVl()	65
ominaisvektori, eigVc()	65
on yhtä kuin, =	226
OneVar, yhden muuttujan tilastot	133
operaattorit	
sievennysjärjestys	252
ord(), numeerinen merkkikoodi	136

P

P►Rx(), x:n suorakulmakoordinaatti	136
P►Ry(), y:n suorakulmakoordinaatti	136
paikallinen muuttuja, Local	110
paikallinen, Local	110
Palaa, return	158
paloittain määritelty funktio (2-osainen)	
malli	6
paloittain määritelty funktio (N-osainen)	
malli	7
PassErr, ohita virhe	137
Pdf()	81
permutaatiot, nPr()	131
piecewise()	138
pienempi tai yhtä suuri kuin, {	228
pienin yhteinen jaettava, lcm	99
pikavalinnat, näppäimistö	250
piste	
erotus, .-	224
osamäärä, ./	225
potenssi, .^	225
summa, .+	224
tulo, .*	225
tulo, dotP()	63
poissCdf()	138
poissPdf()	138
poista	
poista, ClrErr	30
tyhjät elementit listasta	56
poistaminen	
muuttuja, DelVar	56
polaarinen	
koordinaatti, R►Pr()	149
koordinaatti, R►Pθ()	149
vektorinäyttö, 4Polaarinen	138

polyCoeff()	139
polyDegree()	140
polyEval(), laske polynomi	140
polyGcd()	141
polynomit	
laske, polyEval()	140
satunnainen, randPoly()	152
PolyRoots()	142
potenssi, ^	223
potenssiregressio, PowerReg	142, 156-157, 195
PowerReg, potenssiregressio	142
Prgm, määritä ohjelma	143
prodSeq()	144
product(), tulo	144
propFrac, varsinainen murtoluku	145
prosentti, %	226
PyydäMerkkij	157
Pyyntö	156
pyöristä, round()	162
päivämäärien väliset päivät, dbd()	52
pääoman maksuerien summa	236

Q

QR-hajottaminen, QR	146
QR, QR-hajottaminen (tekijöihin jako)	146
QuadReg, 2. asteen regressio	146
QuartReg, 4. asteen regressio	148

R

R, radiaani	238
R•Pr(), polaarinen koordinaatti	149
R•Pθ(), polaarinen koordinaatti	149
radiaani, R	238
rahan aika-arvo, Korkeo	203
rahan aika-arvo, maksuerien lukumäärä	203
rahan aika-arvo, maksuerä	203
rahan aika-arvo, nykyarvo	204
rahan aika-arvo, Tuleva arvo	202
raja-arvo	
lim()	101
limit()	101
malli	11

rajoitettava operaattori " "	244
rajoitettava operaattori, laskemisjärjestys	252
rand(), satunnaisluku	150
randBin, satunnaisluku	150
randInt(), satunnainen kokonaisluku	150
randMat(), satunnaismatriisi	151
randNorm(), satunnainen normaalijakauma	151
randPoly(), satunnaispolynomi	152
randSamp()	152
RandSeed, satunnaisluvun siemenluku	152
ratkaise, solve()	179
ratkaisu, deSolve()	57
reaali, real()	152
real(), reaali	152
ref(), rivi-echelon-muoto	153
RefreshProbeVars	155
regressiot	
2. asteen, QuadReg	146
3. asteen, CubicReg	47
4. asteen, QuartReg	148
eksponentiaalinen, ExpReg	73
lineaarinen regressio, LinRegAx	103
lineaarinen regressio, LinRegBx	102, 104
logaritminen, LnReg	109
Logistinen	112
logistinen, Logistic	113
mediaani-mediaani-suora, MedMed	117
MultReg	121
potenssiregressio, PowerReg	142, 156-157, 195
sinimuotoinen, SinReg	178
remain(), jäännös	156
return, Palaa	158
right(), oikea	158
right, right()	33, 67, 209
ristitulo, crossP()	42
rivi-echelon-muoto, ref()	153
rk23(), Runge Kutta -funktio	159
rotate(), kierrä	160-161
round(), pyöristä	162
rowAdd(), matriisin rivien yhteenlasku	163
rowDim(), matriisin rivikoko	163
rowNorm(), matriisin rivinormi	163
rowSwap(), matriisin rivinvaihto	163

rref(), sievennetty rivi-echelon-muoto	163
ryhmät, lukitseminen ja vapauttaminen	110, 207
ryhmät, lukitustilan testaaminen	87

S

samanaikaiset yhtälöt, simult()	174
sarja, series()	168
satunnainen	
matriisi, randMat()	151
normaalijakauma, randNorm()	151
polynomi, randPoly()	152
siemenluku, RandSeed	152
satunnaisotos	152
sec ⁻¹ (), käänteissekantti	164
sec(), sekantti	164
sech ⁻¹ (), käänteinen hyperbolinen sekantti	165
sech(), hyperbolinen sekantti	165
sekaluvut, käyttäen funktiota propFrac()	145
sekuntimuoto, "	240
sekvenssi, seq()	166
seq(), sekvenssi	166
seqGen()	167
seqn()	168
sequence, seq()	167-168
series(), sarja	168
setMode(), aseta tila	170
shift(), siirrä	172
sievennetty rivi-echelon-muoto, rref()	163
sievennysjärjestys	252
sign(), etumerkki	173
siirry, Goto	90
silmukan loppu, EndLoop	114
silmukka, Loop	114
simult(), samanaikaiset yhtälöt	174
sin ⁻¹ (), arkussini	176
sin(), sini	175
sinh ⁻¹ (), hyperbolinen arkussini	177
sinh(), hyperbolinen sini	177
sini	
näytä kulman	175
sini, sin()	175
siniregressio, SinReg	178
SinReg, siniregressio	178

<code>solve()</code> , ratkaise	179
<code>SortA</code> , lajittele nousevaan järjestykseen	183
<code>SortD</code> , lajittele laskevaan järjestykseen	183
<code>sqrt()</code> , neliöjuuri	184
<code>stat.results</code>	185
<code>stat.values</code>	186
<code>stdDevPop()</code> , perusjoukon keskihajonta	186
<code>stdDevSamp()</code> , otoksen keskihajonta	187
Stop-komento	188
<code>string()</code> , lauseke merkkijonoksi	188
<code>strings</code>	
<code>right, right()</code>	33, 67, 209
<code>studentint-todennäköisyysjakauma, tCdf()</code>	194
<code>studentint-todennäköisyystiheys, tPdf()</code>	199
<code>subMat()</code> , alimatriisi	188, 190
<code>sum()</code> , yhteenlasku	189
<code>sumIf()</code>	189
<code>summa (G)</code>	
malli	9
<code>summa, S()</code>	235
<code>sumSeq()</code>	190
suorakulmavektorinäyttö, ►Rect	153
suurempi kuin, >	229
suurempi tai yhtä suuri kuin, 	229
suurin yhteinen jakaja, <code>gcd()</code>	84
<code>syöte, Input</code>	93

T

<code>t-testi, tTest</code>	201
<code>T, transponoi</code>	190
<code>tai (Boolean), tai</code>	135
<code>tai, Boolean operaattori</code>	135
tallentaminen	
symboli, &	245-246
<code>talouselaskentafunktiot, tvnFV()</code>	202
<code>talouselaskentafunktiot, tvnI()</code>	203
<code>talouselaskentafunktiot, tvnN()</code>	203
<code>talouselaskentafunktiot, tvnPmt()</code>	203
<code>talouselaskentafunktiot, tvnPv()</code>	204
<code>tan⁻¹()</code> , arkustangentti	191
<code>tan()</code> , tangentti	191
<code>tangentLine()</code>	192
<code>tangentti, tan()</code>	191

tangenttisuora, tangentLine()	192
tanh ⁻¹ (), hyperbolinen areatangentti	193
tanh(), hyperbolinen tangentti	192
taylor(), Taylorin polynomi	194
Taylorin polynomi, taylor()	194
tCdf(), studentint-todennäköisyysjakauma	194
tCollect(), trigonometrinen keruu	194
tekijä, factor()	74
Teksti-komento	195
Test_2S, 2 otoksen F-testi	82
tExpand(), trigonometrinen laajennus	195
tila-asetukset, getMode()	87
tilastot	
kahden muuttujan tulokset, TwoVar	204
kertoma, !	231
keskiarvo, mean()	116
keskihajonta, stdDev()	186-187, 207
kombinaatiot, nCr()	125
mediaani, median()	117
permutaatiot, nPr()	131
satunnainen normaalijakauma, randNorm()	151
satunnainen siemenluku, RandSeed	152
varianssi, variance()	207
yhden muuttujan tilastot, OneVar	133
tilat	
asetus, setMode()	170
TInterval, t-luottamusväli	196
TInterval_2Samp, kahdenotoksen t-luottamusvälin	197
ΔtmpCnv() [tmpCnv]	198
tmpCnv()	198
todennäköisyystiheys, normPdf()	130
toinen derivaatta	
malli	10
tPdf(), studentint-todennäköisyystiheys	199
trace()	199
transponoi, T	190
trigonometrinen keruu, tCollect()	194
trigonometrinen laajennus, tExpand()	195
Try, virheenkäsittelykomento	200
tTest, t-testi	201
tTest_2Samp, kahden otoksen t-testi	201
tulo (P)	
malli	10

tulo, P()	234
tulo, product()	144
tulokset, tilastot	185
tulosarvot, tilastot	186
tunnus, Lbl	99
TVM-argumentit	204
TVM-funktioiden argumentit	204
tvmFV()	202
tvmI()	203
tvmN()	203
tvmPmt()	203
tvmPV()	204
TwoVar, kahden muuttujan tulokset	204
tyhjä, testi	99
tyhjän testi, isVoid()	99
tyhjät elementit	248
tyhjät elementit, poista	56
täsmällinen, exact()	69

U

unitV(), yksikkövektori	206
unLock, vapauta muuttuja tai muuttujaryhmä	207
uusi	
lista, newList()	126
matriisi, newMat()	126

V

vaihto, shift()	172
vakio	
funktiossa solve()	180
vakiot	
funktiossa cSolve()	46
funktiossa cZeros()	51
funktiossa deSolve()	57
funktiossa solve()	182
pikavalinnat	250
varianssi, variance()	207

W

warnCodes(), Warning codes	209
varoituskoodit ja -viestit	261
varPop()	207

varSamp(), otoksen varianssi	207
varsinainen murtoluku, propFrac	145
vasen, left()	100
vastaus	
näytä e:n arvolla	70
näytä kulman kosini	35
näytä kulman sini	175
vastaus (viimeinen), ans	17
vektorit	
näyttö vektorina sylinterikoordinaatistossa, ►Cylind	49
pistetulo, dotP()	63
ristitulo, crossP()	42
yksikkö, unitV()	206
when(), kun	209
while, While	210
While, while	210
virheet ja vianmäärittäminen	
ohita virhe, PassErr	137
poista virhe, ClrErr	30
vähennyslasku, N	220

X

x:n suorakulmakoordinaatti, P►Rx()	136
x ² , neliö	224
XNOR	230
xor, Boolean eksklusiivinen tai (or)	210

Y

y:n suorakulmakoordinaatti, P►Ry()	136
yhden muuttujan tilastot, OneVar	133
yhteenlasku, +	220
yhteenlasku, sum()	189
yhteinen nimittäjä, comDenom()	32
Yhtälökäyttöjärjestelmä (EOS)	252
yhtälöpari (2 yhtälöä)	
malli	7
yhtälöryhmä (N yhtälöä)	
malli	8
yksikkövektori, unitV()	206
yksiköt	
muunna	242
yläraja, ceiling()	-25-26, 42

Z

zeroes(), nollakohdat	211
zInterval, z-luottamusväli	214
zInterval_1Prop, yhden osuuden z-luottamusväli	215
zInterval_2Prop, kahden osuuden z-luottamusväli	215
zInterval_2Samp, kahden näytteen z-luottamusväli	216
zTest	217
zTest_1Prop, yhden osuuden z-testi	217
zTest_2Prop, kahden osuuden z-testi	218
zTest_2Samp, kahden otoksen z-testi	219

X

χ^2 Cdf()	29
χ^2 GOF	29
χ^2 Pdf()	30