



TI-*nspire*[™]

TI-Nspire[™] CAS /
TI-Nspire[™] CX CAS
Sovelluksen käsikirja

Tämä opas koskee TI-Nspire[™]-ohjelmiston versiota 3.9. Uusin versio asiakirjoista on saatavilla Internet-sivustolta education.ti.com/guides.

Tärkeitä tietoja

Ellei muuten ilmoiteta ohjelman mukaan liitettyssä 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 - 2014 Texas Instruments Incorporated

Contents

Tärkeitä tietoja	2
Lausekemallit	5
Luettelo aakkosjärjestyksessä	12
A	12
B	21
C	24
D	48
E	59
F	68
G	77
I	83
L	91
M	106
N	114
O	122
P	125
Q	133
R	136
S	150
T	174
U	188
V	189
W	190
X	192
Z	193
Symbolit	201
Tyhjät elementit	227
Matemaattisten lausekkeiden syöttäminen pikavalintojen avulla	229
EOS-järjestelmän (yhtälökäyttöjärjestelmä) hierarkia	231
Virhekoodit ja viestit	233

Varoituskoodit ja -viestit	240
Huolto ja Asiakastuki	243
TI-tuotteiden huolto- ja takuutietoa	243
Index	245

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 203.

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 204.

Esimerkki:

$$2^3 = 8$$

Neliöjuurimalli

ctrl **x²** painikkeet



Huomaa: Katso myös $\sqrt{}$ (neliöjuuri), sivu 214.

Esimerkki:

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

N:s juuri -malli

ctrl ^ painikkeet



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

Esimerkki:

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

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

e eksponenttimalli

e^x painikkeet



e-kantainen eksponenttifunktio korotettuna potenssiin

Huomaa: Katso myös **e^()**, sivu 59.

Esimerkki:

$$e^1 \quad e$$

$$e^1. \quad 2.71828182846$$

Logaritmimalli

ctrl 10^x painike



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

Huomaa: Katso myös **log()**, sivu 102.

Esimerkki:

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

Paloittain määritellyn funktion malli (2-osainen)

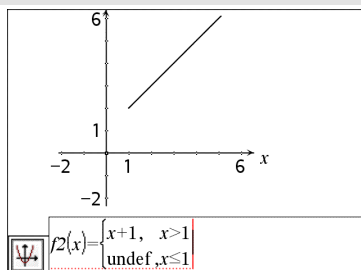
Katalogi >



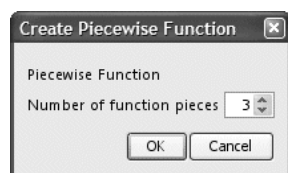
Esimerkki:

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

Huomaa: Katso myös **piecewise()**, sivu 126.



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



Huomaa: Katso myös `piecewise()`, sivu 126.

Esimerkki:

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



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

Huomaa: Katso myös `system()`, sivu 174.

Esimerkki:

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

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

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

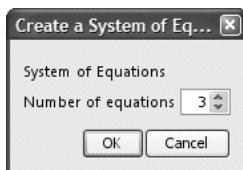
N-osaisen yhtälöryhmän malli

Katalogi > 

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

Esimerkki:

Katso yhtälöparin (2 yhtälöä) mallin esimerkki.



Huomaa: Katso myös `system()`, sivu 174.

Itseisarvon malli

Katalogi > 



Huomaa: Katso myös `abs()`, sivu 12.

Esimerkki:

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

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:

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

Matrisimalli (1 x 2)

Katalogi > 

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

Esimerkki:

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

Matrisimalli (2 x 1)

Katalogi > 

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

Esimerkki:

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

Matrisimalli (m x n)

Katalogi > 

Malli tulee näkyviin määritettyäsi rivien ja sarakkeiden lukumäärän syöttöruutuun.

Create a Matrix ✕

Matrix

Number of rows

Number of columns

Esimerkki:

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

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

Summan malli (Σ)

Katalogi > 

$$\sum_{i=0}^{\begin{bmatrix} 0 \end{bmatrix}} \begin{bmatrix} 0 \end{bmatrix}$$

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

Esimerkki:

$$\sum_{n=3}^7 \begin{bmatrix} n \end{bmatrix} \quad 25$$

Huomaa: Katso myös $\Sigma()$ (**sumSeq**), sivu 215.

Tulon malli (Π)Katalogi > 

$$\frac{d}{d\mathbf{x}} \left(\mathbf{f}(\mathbf{x}) \right)$$

Esimerkki:

$$\frac{d}{dn} \left(\frac{1}{n} \right) = -\frac{1}{n^2}$$

Huomaa: Katso myös $\Pi()$ (**prodSeq**), sivu 214.

Ensimmäisen derivaatan malli

Katalogi > 

$$\frac{d}{d\mathbf{x}} \left(\mathbf{f}(\mathbf{x}) \right)$$

Ensimmäisen derivaatan mallia voi käyttää myös laskettaessa ensimmäinen derivaatta pisteessä.

Esimerkki:

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

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

Huomaa: Katso myös $d()$ (**derivaatta**), sivu 212.

Toisen derivaatan malli

Katalogi > 

$$\frac{d^2}{d\mathbf{x}^2} \left(\mathbf{f}(\mathbf{x}) \right)$$

Toisen derivaatan mallia voi käyttää myös laskettaessa toinen derivaatta pisteessä.

Esimerkki:

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

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

Huomaa: Katso myös $d()$ (**derivaatta**), sivu 212.

N:n:n derivaatan malli

Katalogi > 

$$\frac{d^n}{d\mathbf{x}^n} \left(\mathbf{f}(\mathbf{x}) \right)$$

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

Esimerkki:

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

Huomaa: Katso myös $d()$ (**derivaatta**), sivu 212.

Määrätyn integraalin malli

Katalogi > 

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

Huomaa: Katso myös `integrointi()`, sivu 201.

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 `integral()`, sivu 201.

Esimerkki:

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

Raja-arvon malli

Katalogi > 

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

Esimerkki:

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

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

Huomaa: Katso myös `limit()`, sivu 93.

Luettelo aakkosjärjestyksessä

Komennot, joiden nimiä ei voi järjestää aakkosjärjestykseen (esimerkiksi +, ! ja >), on esitetty tämän kappaleen lopussa alkaen sivulta (sivu 201). Ellei toisin ole mainittu, kaikki tämän kappaleen esimerkit on suoritettu laskimen oletustilassa, eikä mitään muuttujia ole määritetty.

A

abs()

Katalogi > 

abs(LausI)⇒*lauseke*

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

abs(ListaI)⇒*lista*

$$|2-3 \cdot i| \quad \sqrt{13}$$

abs(MatriisiI)⇒*matriisi*

$$|z| \quad |z|$$

Laskee argumentin itseisarvon.

$$|x+y \cdot i| \quad \sqrt{x^2+y^2}$$

Huomaa: Katso myös **Itsesarvon 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)

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

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

NPmt on taulukon maksuerien lukumäärä. Taulukko alkaa ensimmäisestä maksuerästä.

N, I, PV, Pmt, FV, PpY, CpY ja *PmtAt* on kuvattu TVM-argumenttien taulukossa, sivu 186.

- 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-funktiolla.

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 $\Sigma\text{Pm}()$, sivu 216, sekä **bal()**, sivu 21.

and

BooleanLaus1 and BooleanLaus2 \Rightarrow *Boolean lauseke BooleanLista1*

and *BooleanLista2* \Rightarrow *Boolean lista BooleanMatriisi1 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 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 desimaalikonaisluvun, joka on liian suuri etumerkilliselle, 64 bitin binaarimuodolle, laskin käyttää symmetristä modulo-operaatiota, jotta arvo saadaan oikealle alueelle.

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

Heksadesimaalisessa kantaluutilassa:

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

Tärkeää: Nolla, ei O-kirjain.

Binaarisessa kantaluutilassa:

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

Desimaalisessa kantaluutilassa:

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

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

angle()Katalogi > **angle(LausI)**⇒*lauseke*

Laskee argumentin kulman tulkiten argumentin kompleksiluvuksi.

Huomaa: Kaikkia määrittämättömiä muuttujia käsitellään reaali muuttujina.

Astekulmatilassa:

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

Graadikulmatilassa:

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

Radiaanikulmatilassa:

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

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

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

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

angle(ListaI)⇒*lista***angle(MatriisiI)**⇒*matriisi*Laskee listan tai matriisin *Listai*:n tai *MatriisiI*:n elementtien kulmista tulkiten jokaisen elementin kompleksiluvuksi, joka edustaa kaksiulotteista suorakulmakoordinaattipistettä.**ANOVA**Katalogi > **ANOVA** *Listai, Lista2[, Lista3, ..., Lista20][, Lippu]*Suorittaa yksisuuntaisen varianssianalyysin 2-20 perusjoukon keskiarvon vertailua varten. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 169.)*Lippu*=0 datalle, *Lippu*=1 tilastoille

Tulosmuuttuja	Kuvaus
stat.F	F-tilaston arvo
stat.PVal	Alin merkitsevyystaso, jolla nollahypoteesi voidaan hylätä
stat.df	Ryhmiin vapausasteet

Tulosmuuttuja	Kuvaus
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

Katalogi > 

ANOVA2way *List1, Lista2[, Lista3, ..., Lista10][, TasoRivi]*

Laskee kaksisuuntaisen varianssianalyysin 2-10 perusjoukon keskiarvojen vertaamiseksi. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 169.)

TasoRivi=0 lohkolle

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

Tulokset: Lohkomuoto

Tulosmuuttuja	Kuvaus
stat.F	F-tilasto, saraketekijän F-tilasto
stat.PVal	Alin merkitsevyystaso, jolla nollahypoteesi voidaan hylätä
stat.df	Saraketekijän vapausasteet
stat.SS	Saraketekijän nelöiden 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 nelöiden summa

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

SARAKETEKIJÄN tulokset

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

RIVITEKIJÄN tulokset

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

VUOROVAIKUTUKSEN tulokset

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

VIRHEIDEN tulokset

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

ans

ctrl (←) painikkeet

ans ⇒ arvo

56

56

Näyttää viimeksi sievennetyn lausekkeen tuloksen.

56+4

60

60+4



64

approx()

Katalogi > 

approx(Lausl) ⇒ lauseke

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

Tämä vastaa argumentin syöttämistä ja painikkeen   painamista.

approx($\frac{1}{3}$) 0.333333

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

approx({sin(π),cos(π)}) {0,-1}

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

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

approx(Lista) ⇒ lista

approx(Matriisi) ⇒ matriisi

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

approx({sin(π),cos(π)}) {0,-1}

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

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

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

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

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

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

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

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

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

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

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

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

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

Lista[, *tol*]⇒*lista*

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

Matriisi[, *tol*]⇒*matriisi*

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

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

arccos()Katso \cos^{-1} (), sivu 35.**arccosh()**Katso \cosh^{-1} (), sivu 36.**arccot()**Katso \cot^{-1} (), sivu 37.**arccoth()**Katso \coth^{-1} (), sivu 38.

arccsc()**Katso csc^{-1} , sivu 41.****arccsch()****Katso csch^{-1} , sivu 41.****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 funktiotilan määrittystä.

arcLen(Lista1, Muutt, Alku, Loppu) ⇒ lista

Laskee listan jokaisen *Lista1*: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 sec^{-1} , sivu 150.****arcsech()****Katso sech^{-1} , sivu 151.****arcsin()****Katso \sin^{-1} , sivu 160.**

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* liitettyinä *Lista1*:n loppuun.

augment(Matriisi1, Matriisi2)⇒matriisi

Luo uuden matriisin, joka on *Matriisi2* liitettyinä *Matriisi1*:een. Kun käytetään merkkiä " ", matriiseiden rivimäärien on oltava samat, ja *Matriisi2* liitetään *Matriisi1*:een uusina sarakkeina. Ei muuta *Matriisi1*:ä eikä *Matriisi2*:a.

1 2	→ m1	1 2
3 4		3 4
5	→ m2	5
6		6
$\text{augment}(m1, m2)$		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** $\text{avgRC}(x^2 - x + 2, x) \quad 2 \cdot (x - 0.4995)$ **avgRC(Matriisi1, Muutt [=Arvo] [, Askel])⇒matriisi**

Laskee erotusosamäärän eteenpäin (keskimääräisen muutosnopeuden).

 $\text{avgRC}(x^2 - x + 2, x, 0.1) \quad 2 \cdot (x - 0.45)$

Laus1 voi olla käyttäjän määrittämä funktionimi (katso **Func**).

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

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.

N, *I*, *PV*, *Pmt*, *FV*, *PpY*, *CpY* ja *PmtAt* on kuvattu TVM-argumenttien taulukossa, sivu 186.

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

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

bal(4,*tbl*) 1674.27

N, *I*, *PV*, *Pmt*, *FV*, *PpY*, *CpY* ja *PmtAt* on kuvattu TVM-argumenttien taulukossa, sivu 186.

- 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-funktiolla.

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 $\Sigma\text{Pm}()$, sivu 216.

Kokonaisluku1 ►Base2⇒*kokonaisluku*

256►Base2

0b100000000

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

0h1F►Base2

0b11111

Muuttaa *Kokonaisluku1*:n binaariluvuksi.

Binaariluvuissa on aina etuliite 0b ja heksadesimaaliluvuissa etuliite 0h.

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

Negatiiviset luvut näytetään kahden komplementteina. Esimerkki:

-1 näkyy muodossa

0hFFFFFFFFFFFFFFF heksadesimaalisessa kantalukutilassa

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

-2⁶³ näkyy muodossa

0h8000000000000000 heksadesimaalisessa kantalukutilassa

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

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

2⁶³ muuttuu muotoon -2⁶³ ja näkyy muodossa

0h8000000000000000 heksadesimaalisessa kantalukutilassa

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

2⁶⁴ muuttuu muotoon 0 ja näkyy

0h0 heksadesimaalisessa kantalukutilassa

0b0 binaarisessa kantalukutilassa

-2⁶³ - 1 muuttuu muotoon 2⁶³ - 1 ja näkyy muodossa

0h7FFFFFFFFFFFFFFF heksadesimaalisessa

kantalukutilassa

0b111...111 (64 ykköstä) binaarisessa

kantalukutilassa

►Base10 (►Kantaluku10)

Kokonaisluku1 ►Base10⇒*kokonaisluku*

0b10011►Base10

19

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

0h1F►Base10

31

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

0b *binaariluku*0h *heksadesimaaliluku*

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

Binaariluvussa voi olla enintään 64 numeroa.

Heksadesimaaliluvussa voi olla enintään 16 numeroa.

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

►Base16 (►Kantaluku16)

Kokonaisluku1 ►Base16⇒*kokonaisluku*

256►Base16

0h100

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

0b111100001111►Base16

0hFOF

Muuttaa *Kokonaisluku1*:n heksadesimaaliluvuksi.

Binaariluvuissa on aina etuliite 0b ja heksadesimaaliluvuissa etuliite 0h.

0b *binaariluku*0h *heksadesimaaliluku*

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

Binaariluvussa voi olla enintään 64 numeroa.

Heksadesimaaliluvussa voi olla enintään 16 numeroa.

Ilman etuliitettä *Kokonaisluku1*:ä käsitellään

desimaalilukuna (kantaluku 10). Vastaus näkyy heksadesimaalilukuna kantalukutilasta riippumatta.

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

Jos syötät desimaalikononaisluvun, joka on etumerkillisen, 64 bitin binaarimuodon lukualueen ulkopuolella, laskin käyttää symmetristä modulo-operaatiota, jotta arvo saadaan oikealle alueelle.

Lisätietoja, katso ►Base2, sivu 22.

binomCdf()

binomCdf(n,p)⇒luku

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

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

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

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

binomPdf()

binomPdf(n,p)⇒luku

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

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

C

ceiling()Katalogi > **ceiling(LausI)**⇒kokonaisluku

ceiling(.456)

1.

Laskee lähimmän kokonaisluvun, joka on \geq argumentti.

Argumentti voi olla reaali- tai kompleksiluku.

Huomaa: Katso myös **floor()**.

ceiling(ListaI)⇒lista**ceiling(MatriisiI)**⇒matriisi

Laskee listan tai matriisin jokaisen elementin ylärajasta.

ceiling({-3.1,1,2.5})	{-3.,1,3.}
ceiling($\begin{bmatrix} 0 & -3.2 \cdot i \\ 1.3 & 4 \end{bmatrix}$)	$\begin{bmatrix} 0 & -3. \cdot i \\ 2. & 4 \end{bmatrix}$

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

|Muutt=Arvo⇒lauseke

centralDiff(LausI,Muutt [=Arvo][,Lista])⇒lista**centralDiff(ListaI,Muutt [=Arvo][,Askel])**⇒lista**centralDiff(MatriisiI,Muutt [=Arvo][,Askel])**

⇒matriisi

Laskee numeerisen derivaatan käyttäen keskeiserotusosamäärän kaavaa.

Kun *Arvo* määritetään, se ohittaa mahdolliset aikaisemmat muuttujamäärittäykset 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()**.

centralDiff(cos(x),x,h)	$\frac{-\cos(x-h)-\cos(x+h)}{2 \cdot h}$
lim _{h→0} {centralDiff(cos(x),x,h)}	-sin(x)
centralDiff(x ³ ,x,0.01)	3.·(x ² +0.000033)
centralDiff(cos(x),x) _{x=π/2}	-1.
centralDiff(x ² ,x,{0.01,0.1})	{2.·x,2.·x}

cFactor(Laus I, Muutt) ⇒ lauseke

cFactor(Lista I, Muutt) ⇒ lista

cFactor(Matriisi I, Muutt) ⇒ matriisi

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

Laus I: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()**.

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

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

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

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




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

$$\text{cFactor}\left(x^2 + 3 \cdot x\right) \\ (x + \sqrt{3} \cdot i) \cdot (x - \sqrt{3} \cdot i)$$

$$\text{cFactor}\left(x^2 + a \cdot x\right) \\ (x + \sqrt{a} \cdot i) \cdot (x + \sqrt{a} \cdot i)$$

$$\text{cFactor}\left(x^5 + 4 \cdot x^4 + 5 \cdot x^3 - 6 \cdot x - 3\right) \\ x^5 + 4 \cdot x^4 + 5 \cdot x^3 - 6 \cdot x - 3$$

$$\text{cFactor}\left(x^5 + 4 \cdot x^4 + 5 \cdot x^3 - 6 \cdot x - 3, x\right) \\ (x - 0.964673) \cdot (x + 0.611649) \cdot (x + 2.12543) \cdot (x^2 + 1)$$

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

char()Katalogi > **char(Kokonaisluku)**⇒merkki

char(38) " & "

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

char(65) " A "

charPoly()Katalogi > **charPoly(neliömatriisi, Muutt)**⇒polynomilauseke**charPoly(neliömatriisi, Laus)**⇒polynomilauseke**charPoly(neliömatriisi1, Matriisi2)**

⇒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 2wayKatalogi >  **χ^2 2way** *ObsMatriisi***chi22way** *ObsMatriisi*

Laskee χ^2 -testin tarkasteltavan matriisin *ObsMatriisi* sisältämän kaksisuuntaisen lukemataulukon arvojen välisestä assosiaatiosta. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 169.)

Lisätietoja matriisissa olevien tyhjien elementtien vaikutuksesta, katso Tyhjät elementtivuorolla sivu 227.

Tulosmuuttuja	Kuvaus
stat. χ^2	Khin neliö -tilasto: summa (tarkasteltava - odotettu) ² /odotettu
stat.PVal	Alin merkitsevyytaso, jolla nollahypoteesi voidaan hylätä

Tulosmuuttuja	Kuvaus
stat.df	Khin neliö -tilastojen vapausasteet
stat.ExpMat	Odotetun elementtilukumatalukon matriisi, oletuksena nollahypoteesi
stat.CompMat	Elementtien Khin neliö -tilastokontribuutioiden matriisi

χ^2 Cdf()

Katalogi > 

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

Lisätietoja listassa olevien tyhjiin elementtien vaikutuksesta, katso Tyhjät elementitsivulla sivu 227.

χ^2 GOF

Katalogi > 

χ^2 GOF *obsLista*,*expLista*,*df*

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

Suorittaa testin, jolla varmistetaan, että otoksen data on tiettyä jakaamaa vastaavasta perusjoukosta. *obsList* on lukemalista, ja sen tulee sisältää kokonaislukuja. Tulosten yhteenveto tallentuu *stat.results*-muuttuun. (Katso sivu 169.)

Lisätietoja listassa olevien tyhjiin elementtien vaikutuksesta, katso Tyhjät elementitsivulla sivu 227.

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

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 elementtisivulla sivu 227.

clearAZ

$5 \rightarrow b$	5
b	5
ClearAZ	Done
b	b

Poistaa kaikki yksikirjaimiset muuttujat nykyiseltä tehtävääalueelta.

Jos yksi tai useampia muuttujia on lukittu, tämä komento aiheuttaa virheilmoituksen ja poistaa vain lukitsemattomat muuttujat. Katso **unLock**, sivu 189.

ClrErr

Poistaa virhetilan ja nollaa järjestelmän muuttujan *errCode* .

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 **PassErr**, sivu 126, ja **Try**, sivu 183.

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

Esimerkki **ClrErr**-komentosta, katso esimerkki 2 **Try**-komennon kohdalla, sivu 183.

colAugment()Katalogi > **colAugment**(*Matriisi1*, *Matriisi2*)⇒*matriisi*

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

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

colDim()Katalogi > **colDim**(*Matriisi*)⇒*lauseke*

Laskee *Matriisin* sisältämien sarakkeiden lukumäärän.

Huomaa: Katso myös **rowDim()**.

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

colNorm()Katalogi > **colNorm**(*Matriisi*)⇒*lauseke*

Laskee maksimiarvon *Matriisin* sarakkeissa olevien elementtien itseisarvojen summista.

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

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

comDenom()Katalogi > **comDenom**(*LausI*[,*MuuttI*])⇒*lauseke***comDenom**(*ListaI*[,*MuuttI*])⇒*lista***comDenom**(*MatriisiI*[,*MuuttI*])⇒*matriisi*

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

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

comDenom(*Laus1, Muutt*) supistaa osoittajan ja nimittäjän, jotka on lavennettu muuttujalla *Muutt*. Termit ja niiden tekijät lajitellaan siten, että *Muutt* on päämuuttuja. Muuttujan *Muutt* samanlaiset potenssit kerätään. Toimenpiteessä voi esiintyä jonkin verran kerättyjen kertoimien satunnaista tekijöihin jakamista. Verrattuna siihen, että muuttuja *Muutt* jätettäisiin pois, tämä toiminto säästää usein aikaa, muistia ja näyttötilaa, ja samalla lausekkeesta tulee ymmärrettävämpi. Lisäksi tulokseen kohdistuvat seuraavat operaatiot ovat nopeampia eivätkä kuluta muistia yhtä todennäköisesti.

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

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

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

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

$$\frac{x^2 \cdot y \cdot (y+1)+2 \cdot x \cdot y \cdot (y+1)+2 \cdot y \cdot (y+1)}{x^2+2 \cdot x+1}$$

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

$$\frac{y^2 \cdot (x^2+2 \cdot x+2)+y \cdot (x^2+2 \cdot x+2)}{x^2+2 \cdot x+1}$$

Define *comden*(*exprn*)=**comDenom**(*exprn*,*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 ()

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

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

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

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

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

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

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

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

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

$$(x+2)^2+(y+3)^2=10$$

completeSquare ()Katalogi > 

- 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.

$$\text{completeSquare}(3 \cdot x^2 + 2 \cdot y + 7 \cdot y^2 + 4 \cdot x = 3, \{x, y\})$$

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

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

conj()Katalogi > 

conj(Lauseke) ⇒ lauseke

conj(Lista) ⇒ lista

conj(Matriisi) ⇒ matriisi

Laskee argumentin liittokompleksiluvun.

Huomaa: Kaikkia määrittämättömiä muuttujia käsitellään reaali muuttujina.

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

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

$$\text{conj}(z) \quad z$$

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

constructMat()Katalogi > **constructMat**

(Lause, Muutt1, Muutt2, numRivit, numSarakeet)
⇒ matriisi

Laskee matriisin argumentteihin perustuen.

Lause on lauseke muuttujissa *Muutt1* ja *Muutt2*.

Tuloksena olevan matriisin elementit muodostetaan sieventämällä *Lause* jokaisella *Muutt1*:n ja *Muutt2*:n lisätyllä arvolla.

Muutt1:ä lisätään automaattisesti välillä **1** - *numRivit*.

Kullakin rivillä *Muutt2*:a lisätään välillä **1** - *numSarakeet*.

$$\text{constructMat}\left(\frac{1}{i+j}, i, j, 3, 4\right)$$

$$\begin{bmatrix} 1 & 1 & 1 & 1 \\ 2 & 3 & 4 & 5 \\ 1 & 1 & 1 & 1 \\ 3 & 4 & 5 & 6 \\ 1 & 1 & 1 & 1 \\ 4 & 5 & 6 & 7 \end{bmatrix}$$

CopyVarKatalogi > **CopyVar** *Muutt1*, *Muutt2*Define $a(x) = \frac{1}{x}$ Done**CopyVar** *Muutt1.*, *Muutt2.*Define $b(x) = x^2$ Done**CopyVar** *Muutt1*, *Muutt2* kopioi muuttujan *Muutt1* arvon muuttujaan *Muutt2* ja luo tarvittaessa *Muutt2*:n. Muuttujalla *Muutt1* on oltava arvo.CopyVar *a,c*: $c(4)$ $\frac{1}{4}$ Jos *Muutt1* on olemassa olevan käyttäjän määrittämän funktion nimi, kopioi kyseisen funktion määrittämisen funktioon *Muutt2*. Funktio *Muutt1* on määritettävä.CopyVar *b,c*: $c(4)$ 16*Muutt1*:n on oltava muuttujien nimeämissäntöjen mukainen tai epäsuora lauseke, joka sievenny näitä vaatimuksia vastaavaksi muuttujan nimeksi.**CopyVar** *Muutt1.*, *Muutt2.* kopioi kaikki *Muutt1*:n jäsenet. muuttujaryhmä *Var2*:een. ryhmä, *Muutt2*:n luominen. tarvittaessa.

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

Muutt1 tulee olla olemassa olevan muuttujaryhmän nimi, kuten tilastollinen *stat.nn* vastausta tai muuttujaa, jotka on luotu funktiolla **LibShortcut()**. Jos *Muutt2* on jo olemassa, komento korvaa kaikki jäsenet, jotka ovat yhteisiä kummallekin ryhmälle, ja lisää jäsenet, joita ei vielä ole olemassa. Jos yksi tai useampia muuttujan *Muutt2*. jäseniä on lukittu, kaikki muuttujan *Var2*. jäsenet pysyvät muuttumattomina.**corrMat()**Katalogi > **corrMat**(*Lista1*,*Lista2*[,...[,*Lista20*]])Laskee korrelaatiomatriisin laajennetulle matriisille [*Lista1*, *Lista2*, ..., *Lista20*].**►cos**Katalogi > *Laus* ►**cos****Huomaa:** Voit syöttää tämän operaattorin tietokoneen näppäimistöltä kirjoittamalla @>**cos**. $(\sin(x))^2$ ► **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() **painike**

cos(Laus I) ⇒ lauseke

Astekulmatilassa:

$$\cos\left(\frac{\pi}{4}\right) = \frac{\sqrt{2}}{2}$$

cos(Lista I) ⇒ lista

$$\cos(45) = \frac{\sqrt{2}}{2}$$

cos(Laus I) määrittää argumentin kosinin lausekkeena.

$$\cos(\{0,60,90\}) = \left\{1, \frac{1}{2}, 0\right\}$$

cos(Lista I) määrittää listan kaikkien *Listal*:n sisältämien elementtien kosineista.

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

Graadikulmatilassa:

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

Radiaanikulmatilassa:

$$\cos\left(\frac{\pi}{4}\right) = \frac{\sqrt{2}}{2}$$

$$\cos(45^{\circ}) = \frac{\sqrt{2}}{2}$$

cos(neliömatriisi I) ⇒ neliömatriisi

Radiaanikulmatilassa:

Laskee *neliömatriisi 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:

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.

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

$\cos^{-1}(\text{LausI}) \Rightarrow \text{lauseke}$

$\cos^{-1}(\text{Lista}) \Rightarrow \text{lista}$

$\cos^{-1}(\text{LausI})$ määrittää lausekkeena kulman, jonka kosini on LausI .

$\cos^{-1}(\text{Lista})$ laskee listan ListaI :n jokaisen elementin käänteiskosineista.

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

Huomaa: Voit syöttää tämän funktion näppäimistöltä kirjoittamalla $\text{arccos}(\dots)$.

$\cos^{-1}(\text{neliomatriisiI}) \Rightarrow \text{neliomatriisi}$

Astekulmatilassa:

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

Graadikulmatilassa:

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

Radiaanikulmatilassa:

$$\cos^{-1}(\{0, 0.2, 0.5\}) = \left\{ \frac{\pi}{2}, 1.36944, 1.0472 \right\}$$

Radiaanikulmatilassa ja

cos⁻¹()trig **painike**

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

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

suorakulmakompleksimuodossa:

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

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

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

cosh()

Katalogi >

cosh(LausI)⇒*lauseke*

Astekulmatilassa:

cosh(ListaI)⇒*lista*

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

cosh(LausI) määrittää argumentin hyperbolisen kosinin lausekkeena.

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

cosh(neliömatriisi I)⇒*neliömatriisi*

Radiaanikulmatilassa:

Laskee *neliömatriisi I*:n matriisin hyperbolisen kosinin. Tämä ei ole sama kuin kunkin elementin hyperbolisen kosinin laskeminen.

Laskentamenetelmä on kuvattu kohdassa **cos()**.

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

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

421.255	253.909	216.905
327.635	255.301	202.958
226.297	216.623	167.628

cosh⁻¹()

Katalogi >

cosh⁻¹(LausI)⇒*lauseke*

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

cosh⁻¹(ListaI)⇒*lista*

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

cosh⁻¹(LausI) määrittää argumentin käänteisen hyperbolisen kosinin lausekkeena.

cosh⁻¹(ListaI) määrittää listan *Listai*:n kunkin elementin käänteisistä hyperbolisista kosineista.

Huomaa: Voit syöttää tämän funktion näppäimistöltä

kirjoittamalla $\text{arccosh}(\dots)$.

$\cosh^{-1}(\text{neliomatriisi}) \Rightarrow \text{neliomatriisi}$

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

neliomatriisi:n on oltava diagonalisoitavissa.

Vastaus sisältää aina liukulukuja.

Radiaanikulmatilassa ja suorakulmakompleksimuodossa:

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

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

 $\cot()$ 

$\cot(\text{LausI}) \Rightarrow \text{lauseke}$

Astekulmatilassa:

$$\cot(45) \quad 1$$

$\cot(\text{ListaI}) \Rightarrow \text{lista}$

Laskee *LausI*:n kotangentin tai määrittää *lista* *ListaI*:n kaikkien elementtien kotangenteista.

Graadikulmatilassa:

$$\cot(50) \quad 1$$

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

Radiaanikulmatilassa:

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

 $\cot^{-1}()$ 

$\cot^{-1}(\text{LausI}) \Rightarrow \text{lauseke}$

Astekulmatilassa:

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

$\cot^{-1}(\text{ListaI}) \Rightarrow \text{lista}$

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

Graadikulmatilassa:

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

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

Radiaanikulmatilassa:

Huomaa: Voit syöttää tämän funktion näppäimistöltä

cot⁻¹()Katalogi > kirjoittamalla **arccot (...)** .

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

coth()Katalogi > **coth(Laus1)**⇒*lauseke*

$\coth(1.2)$	1.19954
--------------	---------

coth(Lista1)⇒*lista*

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

Laskee *Laus1*:n hyperbolisen kotangentin tai määrittää listan *Lista1*:n kaikkien elementtien hyperbolisista kotangenteista.

coth⁻¹()Katalogi > **coth⁻¹(Laus1)**⇒*lauseke*

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

coth⁻¹(Lista1)⇒*lista*

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

Laskee *Laus1*:n käänteisen hyperbolisen kotangentin tai määrittää listan, joka sisältää *Lista1*: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 sieventyvät numeroarvoiksi.

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

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

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

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

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

Listat & Taulukot -sovelluksessa voit käyttää solun alueita argumenttien tilalla.

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

Tyhjiä elementtejä ei huomioida. Lisätietoja tyhjiistä elementeistä, katso sivu 227.

countIf(*Lista*,*Kriteeri*) \Rightarrow arvo

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

Kriteeri voi olla:

- Arvo, lauseke tai merkijono. Jos kriteerinä käytetään esimerkiksi lukua **3**, laskee lukumäärään vain ne *Listan* elementit, jotka sieventyvät 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 tyhjiistä elementeistä, katso sivu 227.

Huomaa: Katso myös **sumIf()**, sivu 173, ja **frequency()**, sivu 75.

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

Laskee niiden elementtien lukumäärän, jotka ovat yhtä kuin 3.

$\text{countIf}(\{"abc",\text{"def"},\text{"abc"},3\},\text{"def"})$ 1

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

$\text{countIf}(\{x^{-2},x^{-1},1,x,x^2\},x)$ 1

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

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

Laskee lukumäärään 1:n ja 3:n.

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

Laskee lukumäärään 3:n, 5:n ja 7:n.

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

Poly on oltava polynomi yhdessä muuttujassa.

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

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

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

Huomaa: Katso myös **polyRoots()**, sivu 130.

crossP()

Katalogi >

crossP(*Lista1*, *Lista2*)⇒*lista*

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

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

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

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

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

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

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

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

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:

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

csc(*Lista*)⇒*lista*

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

Graadikulmatilassa:

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

Radiaanikulmatilassa:

csc()trig **painike**

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

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

Astekulmatilassa:

csc⁻¹(ListaI) ⇒ lista

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

Laskee kulman, jonka kosekantti on *LausI*, tai määrittää listan, joka sisältää *ListaI*:n kunkin elementin käänteiskosekantit.

Graadikulmatilassa:

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

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

Radiaanikulmatilassa:

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

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

csch()

Katalogi >

csch(LausI) ⇒ lauseke

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

csch(ListaI) ⇒ lista

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

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

csch⁻¹()

Katalogi >

csch⁻¹(LausI) ⇒ lauseke

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

csch⁻¹(ListaI) ⇒ lista

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

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

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

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

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

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



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 reaalitytuloksen kompleksilukumuodossa.

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

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

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.

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

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

Desimaalinen näyttötilassa Kiinteä 2:

$$\begin{array}{l} \text{exact}\left(\text{cSolve}(x^5+4\cdot x^4+5\cdot x^3-6\cdot x-3=0,x)\right) \\ x\cdot(x^4+4\cdot x^3+5\cdot x^2-6)=3 \\ \text{cSolve}(\text{Ans},x) \\ x=-1.11+1.07\cdot i \text{ or } x=-1.11-1.07\cdot i \text{ or } x=2. \end{array}$$

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

$$\text{cSolve}(\text{conj}(z_)=1+i,z_)\qquad \qquad \qquad z_=-1-i$$

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

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

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

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

muuttuja

- tai -

muuttuja = *reaaliluku* tai *ei-reaaliluku*

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


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

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

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 kokonaislukuliiite väliltä 1-255.

Polynomisarjoissa laskun suoritus aika tai muistin

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

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

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

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

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

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

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

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

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 yhtälöistä on ei-polynominen minkä tahansa muuttujan suhteen, mutta kaikki yhtälöt ovat lineaarisia kaikissa ratkaisumuuttujissa, **cSolve()** käyttää Gaussin eliminointia yrittäessään määrittää kaikki ratkaisut.

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

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

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




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

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

$$w_{-}=0.494866 \text{ and } z_{-}=-0.703467$$

$$\text{cSolve}\left(e^{z_{-}}=w_{-} \text{ and } w_{-}=z_{-}^2, \{w_{-}, z_{-}=1+i\}\right)$$

$$w_{-}=0.149606+4.8919 \cdot i \text{ and } z_{-}=1.58805+1 \cdot i$$

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

CubicReg

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

Laskee 3. asteen polynomiregressioy = a · x³ + b · x² + c · x + d listoista X ja Y frekvenssillä *Frekv*. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 169.)

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 Tyhjät elementtisivulla sivu 227.

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	Määrittyskerroin.
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 frekvenssilista.

cumulativeSum()

cumulativeSum(Lista1)⇒lista

$\text{cumulativeSum}(\{1,2,3,4\}) \quad \{1,3,6,10\}$

Laskee listan *Listaa1*: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.

$\begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix} \rightarrow m1$	$\begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix}$
$\text{cumulativeSum}(m1)$	$\begin{bmatrix} 1 & 2 \\ 4 & 6 \\ 9 & 12 \end{bmatrix}$

Tyhjä elementti listassa *Listaa1* tai matriisissa *Matriisi1* tuottaa tyhjän elementin tuloksena olevaan listaan tai matriisiin. Lisätietoja tyhjistä elementeistä, katso sivu 227.

Cycle

Cycle

Funktio, joka laskee yhteen kokonaisluvut väliltä 1-

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.

100 ohittaen luvun 50.

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

g() 5000

Cylind

Katalogi >

Vektori ► **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.

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

cZeros()

Katalogi >

cZeros(Laus, Muutt) ⇒ lista

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

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

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

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

Desimaalien näyttötilassa Kiinteä 3:

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

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

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

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 nollakohdat.

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

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

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

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

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

Määritä rivin 2 juuri:

$$\text{Ans}[2] \quad \begin{bmatrix} \frac{1}{2} \frac{\sqrt{3}}{2} \cdot i & \frac{1}{2} + \frac{\sqrt{3}}{2} \cdot i \end{bmatrix}$$

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

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

$$\begin{bmatrix} 0 & 0 \\ -(\sqrt{1-4 \cdot c_{-} - 1})^2 & -(\sqrt{1-4 \cdot c_{-} - 1}) \\ 4 & 2 \\ -(\sqrt{1-4 \cdot c_{-} + 1})^2 & \sqrt{1-4 \cdot c_{-} + 1} \\ 4 & 2 \end{bmatrix}$$

Voit ottaa mukaan myös tuntemattomia muuttujia, jotka eivät esiinny lausekkeissa. Nämä nollakohtat osoittavat, miten nollakohtien sarjat voivat sisältää mielivaltaisia vakioita, jotka ovat muotoa c_k , jossa k on kokonaislukuilite väliltä 1-255.

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

$$\begin{bmatrix} 0 & 0 & c4 \\ \frac{1}{2} \frac{\sqrt{3}}{2} \cdot i & \frac{1}{2} \frac{\sqrt{3}}{2} \cdot i & c4 \\ \frac{1}{2} + \frac{\sqrt{3}}{2} \cdot i & \frac{1}{2} \frac{\sqrt{3}}{2} \cdot i & c4 \end{bmatrix}$$

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

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

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

$$\begin{bmatrix} e^{w_{-} + i} & e^{w_{-} - i} \\ 2 & 2 \end{bmatrix}$$

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.

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

$$[0.494866 \quad -0.703467]$$

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

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

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

D

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

Lausl ▶DD⇒*arvoListal*▶DD⇒*listaMatriisi1*▶DD⇒*matriisi*

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

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

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:

1▶DD	$\frac{9}{10}^{\circ}$
------	------------------------

Radiaanikulmatilassa:

(1.5)▶DD	85.9437°
----------	----------

Lauseke ►Decimal⇒*lauseke*

Lista ►Decimal⇒*lauseke*

Matriisi ►Decimal⇒*lauseke*

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

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

$\frac{1}{3}$ ►Decimal

0.333333

Define (Määritä)

Define *Muutt* = *Lauseke*

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

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

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

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

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

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

Lohko

EndFunc

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

Lohko

EndPrgm

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

Lohko voi olla joko yksi lauseke tai eri riveillä olevien

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

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

Define (Määrittä)

Katalogi > 

lausekkeiden sarja. *Lohko* voi sisältää myös lausekkeitä ja ohjeita (kuten **If**, **Then**, **Else** ja **For**).

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

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

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

```

Done

```
g(3,-7)

```

3 greater than -7

Done

Define LibPriv (Määrittä LibPriv)

Katalogi > 

Define LibPriv *Muutt* = *Lauseke*

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

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

Lohko

EndFunc

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

Lohko

EndPrgm

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

Huomaa: Katso myös **Define**, sivu 50, ja **Define LibPub**, sivu 51.

Define LibPub (Määrittä 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 50, ja **Define LibPriv**, sivu 51.

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

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

DelVar *Muutt*,

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

Poistaa määritetyn muuttujan tai muuttujaryhmän muistista.

DelVar <i>a</i>	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 189.

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

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

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

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

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

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

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

delVoid()

Katalogi >

delVoid(Lista1)⇒lista

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

Antaa tuloksena listan, jossa on listan *Listal1* sisältö, ja kaikki tyhjät elementit on poistettu.

Lisätietoja tyhjästä elementeistä, katso sivu 227.

derivative()Katso *d()*, sivu 212.**deSolve()**

Katalogi >

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

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·x+c4)·e^{-x}+x²-4·x+6
 $\frac{d^2}{dx^2}(temp)+2\cdot\frac{d}{dx}(temp)+temp-x^2$ 0
 DelVar temp Done

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

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

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

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

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

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

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

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

deSolve(1.astODEandalkuehto, Muutt, riippuvaMuutt) => tietty ratkaisu

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

alkuehto on yhtälö, joka on muotoa:

riippuvaMuutt (riippumatonAlkuarvo) =
riippuvaAlkuarvo

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

deSolve(2.astODEandalkuehto1andalkuehto2, Muutt, riippuvaMuutt) => tietty ratkaisu

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

Käytä *alkuehto1*:lle muotoa:

riippuvaMuutt (riippumatonAlkuarvo) =
riippuvaAlkuarvo

Käytä *alkuehto2*:lle muotoa:

riippuvaMuutt (riippumatonAlkuarvo) =
1.derivaatanAlkuarvo

deSolve(2.astODEandreunaehto1and

reunaehto2, Muutt, riippuvaMuutt) => tietty ratkaisu

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

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

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

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

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

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

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

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

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

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

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



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

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

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

det()Katalogi > **det**(*neliömatriisi*[, *Toleranssi*]) \Rightarrow lausekeLaskee *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   tai **Automaattinen tai likimääräinen** -tilan valintaa Approximate (Likimääräinen), laskut suoritetaan liukulukuaritmetiikalla.
- Jos *Toleranssi* jätetään pois tai sitä ei käytetä, oletusarvoinen toleranssi lasketaan seuraavasti:

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

$\det\begin{pmatrix} a & b \\ c & d \end{pmatrix}$	$a \cdot d - b \cdot c$
$\det\begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}$	-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*) \Rightarrow matriisi**diag**(*rivimatriisi*) \Rightarrow matriisi**diag**(*sarakematriisi*) \Rightarrow matriisi

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

diag(*neliömatriisi*) \Rightarrow rivimatriisiLaskee 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*) \Rightarrow kokonaislukuLaskee *Listan* mitat.

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

dim()

Katalogi >

dim(Matriisi)⇒*lista*

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

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

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

$\dim(\text{"Hello"})$	5
$\dim(\text{"Hello " \& "there"})$	11

Disp

Katalogi >

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

...

Näyttää *Laskin*-sovelluksen historiatietojen sisältämät argumentit. Argumentit näytetään peräkkäin, ja erotinmerkeinä 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	ø
241	ñ
242	ò
243	ó

Done

►DMS

Katalogi >

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

Tulkitsen argumentin kulmana ja näyttää vastaavan DMS-luvun (DDDDDD°MM'SS.ss"). DMS-muoto (asteet, minuutit, sekunnit) on kuvattu kohdissa °, ', " sivulla sivu 219.

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

Astekulmatilassa:

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

komentoa ▶DMS ainoastaan syöterivin lopussa.

domain()

domain(LausI, Muut) ⇒ lauseke

Antaa vastauksena *LausI* 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.

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

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

dominantTerm(Laus1, Muutt [, Piste]) ⇒ lauseke

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

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

Laskee dominanttitermin *Laus1*:n

potenssisarjaesityksestä, kun lauseke on lavennettu *Piste*ellä. Dominanttitermi on se, jonka suuruus kasvaa nopeimmin lähellä arvoa *Muutt = Piste*.

Lausekkeen (*Muutt - Piste*) tuloksena olevalla potenssilla voi olla negatiivinen ja/tai murtolukueksponentti. Tämän potenssin kertoin voi sisältää lausekkeen (*Muutt - Piste*) logaritmeja ja muita *Muutt*:n funktioita, joita hallitsevat kaikki lausekkeen (*Muutt - Piste*) potenssit, joilla on sama eksponentin etumerkki.

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

dominantTerm(...) antaa tuloksena "dominantTerm (...)", ellei se pysty määrittämään tällaista esitystä, kuten olennaisille erikoispisteille, esim. **sin(1/z)**, kun $z=0$, $e^{-1/z}$, 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 sign(...) tai abs(...) reaaliselle kehittelmän muuttujalle tai (-1)^{floor(angle(...))} kompleksille kehittelmän muuttujalle, joka on merkkiin "_" päättyvä muuttuja. Jos tarkoituksesi on käyttää dominanttitermiä vain *Pisteen* toisella puolella oleville arvoille, siinä tapauksessa liitä funktion **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,

$$\text{dominantTerm}(\tan(\sin(x)) - \sin(\tan(x)), x) \quad \frac{x^7}{30}$$

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

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

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

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

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

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

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

joka on asymptoottinen toisen lausekkeen suhteen, esim. $Muutt \rightarrow Piste$. **dominantTerm()** on hyödyllinen myös silloin, kun sarjan ensimmäisen ei-nolla-termin astetta ei tiedetä, etkä halua arvailla iteratiivisesti tai interaktiivisesti ohjelmasilmmukan avulla.

Huomaa: Katso myös **series()**, sivu 153.

dotP()

dotP(Lista1, Lista2)⇒*lauseke*

Laskee kahden listan "pistetulon".

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

dotP(Vektori1, Vektori2)⇒*lauseke*

Laskee kahden vektorin "pistetulon".

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

Kummankin on oltava rivivektoreita, tai kummankin on oltava sarakevektoreita.

E

$e^{\wedge}()$

$e^{\wedge}(Laus1)$ ⇒*lauseke*

Laskee **e**:n arvon korotettuna *Laus1*:n potenssiin.

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

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

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

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

$e^{\wedge}(Lista1)$ ⇒*lista*

Laskee **e**:n arvon korotettuna *Lista1*:n jokaisen elementin potenssiin.

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

$e^{\wedge}()$

☰ painike

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

Laskee *neliömatrissi* l :n matriisiekspotentin. Tämä ei ole sama kuin laskettaessa e korotettuna kunkin elementin mukaiseen potenssiin.

Laskentamenetelmä on kuvattu kohdassa **cos()**.

neliömatrissi l :n on oltava diagonalisoitavissa.

Vastaus sisältää aina liukulukuja.

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

eff()

Katalogi >

eff(nimelliskorko, CpY) ⇒ arvo

eff(5.75,12)

5,90398

Talouseläntoiminto, 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 117.

eigVc()

Katalogi >

eigVc(neliömatrissi) ⇒ matrissi

Suorakulmakompleksimuodossa:

Laskee matriisin, joka sisältää ominaisvektorit reaaliselle tai kompleksiselle *neliömatrissille*, jossa jokainen vastauksen sarake vastaa ominaisarvoa.

Huomaa, että ominaisvektori ei ole yksilöllinen; sitä voidaan skaalata millä tahansa vakiokertoimella.

Ominaisvektorit ovat normaaliomuotoisia, mikä tarkoittaa, että, jos $V = [x_1, x_2, \dots, x_n]$, tällöin:

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

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

-1	2	5	→ mI	-1	2	5
3	-6	9		3	-6	9
2	-5	7		2	-5	7

eigVc(mI)

-0.800906	0.767947	(
0.484029	0.573804+0.052258·i	0.5738
0.352512	0.262687+0.096286·i	0.2626

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

eigV()Katalogi > **eigV(neliömatriisi)**⇒*lista*

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.

Suorakulmakompleksimuodossa:

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

$$\text{eigV}(mI) \\ \{-4.40941, 2.20471+0.763006 \cdot i, 2.20471-0 \cdot i\}$$

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

Else

Katso If, sivu 84.

EiselfKatalogi > **If BooleanLaus1 Then***Lohko1***Eiself BooleanLaus2 Then***Lohko2*

⋮

Eiself 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 $x < 0$ Then

Return -x

ElseIf $x \geq 0$ and $x \neq 10$ Then

Return x

ElseIf $x = 10$ Then

Return 3

EndIf

EndFunc

*Done***EndFor**

Katso For, sivu 73.

EndFunc

Katso Func, sivu 77.

EndIf**Katso If, sivu 84.****EndLoop****Katso Loop, sivu 105.****EndPrgm****Katso Prgm, sivu 132.****EndTry****Katso Try, sivu 183.****EndWhile****Katso While, sivu 192.****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

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

Differentsiaal yhtälö:

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

$$\text{euler}\{0.001 \cdot y \cdot (100 - y), t, y, \{0, 100\}, 10, 1\}$$

0.	1.	2.	3.	4.
10.	10.9	11.8712	12.9174	14.042

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

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

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

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

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

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

Var on riippumaton muuttuja.

$ListOfDepVars$ on riippuvien muuttujien luettelo.

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

$ListOfDepVars0$ on riippuvien muuttujien alkuehtoien luettelo.

$VarStep$ nollasta 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 $VarStep/eulerStep$.

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

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

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

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

Yhtälöryhmä:

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

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

$$euler\left\{ \begin{cases} y1' + 0.1 \cdot y1 \cdot y2 \\ 3 \cdot y2 - y1 \cdot y2 \end{cases}, t, \{y1, y2\}, \{0, 5\}, \{2, 5\}, 1 \right\}$$

0.	1.	2.	3.	4.	5.
2.	1.	1.	3.	27.	243.
5.	10.	30.	90.	90.	-2070.

exact()Katalogi > **exact**(*Laus* *l* [, *Toleranssi*]) \Rightarrow *lauseke***exact**(*Listal* *l* [, *Toleranssi*]) \Rightarrow *lista***exact**(*Matriisi* *l* [, *Toleranssi*]) \Rightarrow *matriisi*

Laskee täsmällisen tilan aritmetiikalla argumentin vastaavan rationaaliluvun, mikäli mahdollista.

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

$\text{exact}(0,25)$	$\frac{1}{4}$
$\text{exact}(0,333333)$	$\frac{333333}{1000000}$
$\text{exact}(0,333333,0,001)$	$\frac{1}{3}$
$\text{exact}(3,5 \cdot x + y)$	$\frac{7 \cdot x}{2} + y$
$\text{exact}(\{0,2,0,33,4,125\})$	$\left\{\frac{1}{5}, \frac{33}{100}, \frac{33}{8}\right\}$

ExitKatalogi > **Exit**

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.

Funktion listaus:

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

►expKatalogi > *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(LausI)**⇒*lauseke*Laskee **e**:n arvon korotettuna *LausI*:n potenssiin.**Huomaa**: Katso myös **e** eksponenttimalli, sivu 6.

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

exp(ListaI)⇒*lista*Laskee **e**:n arvon korotettuna *Listal*:n jokaisen elementin potenssiin.**exp(neliömatriisiI)**⇒*neliömatriisi*

Laskee *neliömatriisiI*:n matriisiekspotentin. Tämä ei ole sama kuin laskettaessa **e** korotettuna kunkin elementin mukaiseen potenssiin.

Laskentamenetelmä on kuvattu kohdassa **cos()**.*neliömatriisiI*:n on oltava diagonalisoitavissa.

Vastaus sisältää aina liukulukuja.

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

$e^{\{1,1,.05\}}$	$\{e,2.71828,1.64872\}$
-------------------	-------------------------

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

exp▶lista()Katalogi > **exp▶list(Laus,Muutt)**⇒*lista*

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

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

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(LausI [, Muutt]) ⇒ lauseke

expand(ListaI [, Muutt]) ⇒ lista

expand(MatriisiI [, Muutt]) ⇒ matriisi

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

Funktion **expand()** tehtävä on muuntaa *LausI* yksinkertaisten termien summaksi ja/tai erotukseksi.

Funktion **factor()** tehtävä sen sijaan on muuntaa *LausI* yksinkertaisten tekijöiden tuloksi ja/tai osamääräksi.

expand(LausI, Muutt) laskee *LausI*: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 äärimäinen vaihtoehto kuin **expand()**.

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

$$\begin{aligned} & \text{expand}\left((x+y+1)^2\right) \\ & \qquad x^2+2\cdot x\cdot y+2\cdot x+y^2+2\cdot y+1 \\ \hline & \text{expand}\left(\frac{x^2-x+y^2-y}{x^2\cdot y^2-x^2\cdot y-x\cdot y^2+x\cdot y}\right) \\ & \qquad \frac{1}{x-1}-\frac{1}{x}-\frac{1}{y-1}-\frac{1}{y} \end{aligned}$$

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

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

expand()

Katalogi >

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

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

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

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

Määrittää *Merkkijonon* sisältämän merkkijonon lausekkeena ja suorittaa toimenpiteen välittömästi.

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

ExpReg

Katalogi >

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

Laskee eksponentiaalisen regressioyhtälön $y = a \cdot (b)^x$ listoista *X* ja *Y* frekvenssillä *Frekv*. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 169.)

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.

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

Lisätietoja listassa olevien tyhjien elementtien vaikutuksesta, katso Tyhjät elementtisivulla sivu 227.

Tulosmuuttuja	Kuvaus
stat.RegEqn	Regressioyhtälö: $a \cdot (b)^x$
stat.a, stat.b	Regressiokertoimet
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 sovitukseen liittyvät jäännökset
stat.XReg	Muokatus <i>X</i> Lista:n sisältämä datapisteiden lista, jota käytetään regressiossa komentojen <i>Frekv</i> , <i>Luokkalista</i> ja <i>Sisällyttä luokat</i> rajoitusten mukaisesti
stat.YReg	Muokatus <i>Y</i> Lista:n sisältämä datapisteiden lista, jota käytetään regressiossa komentojen <i>Frekv</i> , <i>Luokkalista</i> ja <i>Sisällyttä luokat</i> rajoitusten mukaisesti
stat.FreqReg	Komentoja <i>stat.XReg</i> ja <i>stat.YReg</i> vastaava frekvenssilista

F

factor()

factor(LausI[, MuuttI]) ⇒ *lauseke*

factor(ListaI[, MuuttI]) ⇒ *lista*

factor(MatriisiI[, MuuttI]) ⇒ *matriisi*

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

LausI :ä jaetaan tekijöihin mahdollisimman paljon lineaaristen rationaalilukutekijöiden suuntaan ilman, että uusia ei-reaalisia alalausekkeita syntyy. Tämä vaihtoehto on sopiva, jos haluat jakaa lausekkeen tekijöihin useamman kuin yhden muuttujan suhteen.

$$\begin{aligned} \text{factor}(a^3 \cdot x^2 - a \cdot x^2 - a^3 + a) &= a \cdot (a-1) \cdot (a+1) \cdot (x-1) \cdot (x+1) \\ \text{factor}(x^2+1) &= x^2+1 \\ \text{factor}(x^2-4) &= (x-2) \cdot (x+2) \\ \text{factor}(x^2-3) &= x^2-3 \\ \text{factor}(x^2-a) &= x^2-a \end{aligned}$$

`factor(Laus1, Muutt)` laskee *Laus1*:n jaettuna tekijöihin muuttujan *Muutt* suhteen.

Laus1 :ä jaetaan mahdollisimman paljon kohti reaalisia tekijöitä, jotka ovat lineaarisia muuttujassa *Muutt*, vaikka tästä syntyisi irrationaalisia vakioita tai alalausekkeita, jotka ovat irrationaalisia muissa muuttujissa.

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

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

Huomaa: Katso myös `comDenom()`, jossa on kuvattu nopea tapa suorittaa osittainen tekijöihin jako, kun `factor()` ei ole tarpeeksi nopea tai käyttää liikaa muistia.

Huomaa: Katso myös `cFactor()`, jossa on kuvattu täydellinen tekijöihin jako kompleksikertoimiksi esittäessä 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.

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

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

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



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

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

$$\text{factor}(152417172689) = 123457 \cdot 1234577$$

$$\text{isPrime}(152417172689) = \text{false}$$

Pysäytä laskenta käsin,

- **Kämmenlaite:** Pidä -painiketta painettuna ja paina toistuvasti -painiketta.
- **Windows®:** Pidä **F12**-näppäintä pohjassa ja paina toistuvasti **Enter**.
- **Macintosh®:** Pidä **F5**-näppäintä pohjassa ja paina toistuvasti **Enter**.
- **iPad®:** Sovellus näyttää kehotuksen. Voit jatkaa odottamista tai peruuttaa.

Jos haluat pelkästään määrittää, onko jokin luku jaoton, käytä sen sijaan 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,dfNimit)⇒luku, jos *alaraja* ja *yläraja* ovat lukuja, *lista*, jos *alaraja* ja *yläraja* ovat listoja

F Cdf(alaraja,yläraja,dfOsoitt,dfNimit)⇒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 *dfNimitäjälle*.

Aseta $P(X \leq \text{yläraja})$:lle *alaraja* = 0.

Fill

Fill Laus, matriisiMuutt⇒*matriisi*

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

matriisiMuutt on oltava valmiiksi olemassa.

$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$ → <i>amatrix</i>	$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$
Fill 1.01, <i>amatrix</i>	<i>Done</i>
<i>amatrix</i>	$\begin{bmatrix} 1.01 & 1.01 \\ 1.01 & 1.01 \end{bmatrix}$

Fill Laus, listaMuutt⇒*lista*

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

listaMuutt on oltava valmiiksi olemassa.

$\{1,2,3,4,5\}$ → <i>alist</i>	$\{1,2,3,4,5\}$
Fill 1.01, <i>alist</i>	<i>Done</i>
<i>alist</i>	$\{1.01,1.01,1.01,1.01,1.01\}$

FiveNumSummary X , [*Frekv*], [*Luokka*, *Sisällyttä*]

Antaa lyhennetyn version 1 muuttujan tilastoista listalle X . Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 169.)

X edustaa datan sisältävää listaa.

Frekv on valinnainen frekvenssiarvojen lista. Jokainen *Frekv*:n elementti määrittää kunkin vastaavan X :n arvon esiintymisfrekvenssin. Oletusarvo on 1. Kaikkien elementtien on oltava kokonaislukuja 0.

Luokka on numeeristen luokkakoodien lista vastaaville X :n arvoille.

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

Tyhjä elementti jossakin listassa X , *Frekv* tai *Luokka* saa aikaan, että kaikkien listojen vastaava elementti on tyhjä. Lisätietoja tyhjistä elementeistä, katso sivu 227.

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()

floor(*Lausl*) ⇒ kokonaisluku

$\text{floor}(-2.14)$

-3.

Laskee suurimman kokonaisluvun, joka on \leq argumentti. Tämä funktio on identtinen funktion **int()** kanssa.

Argumentti voi olla reaali- tai kompleksiluku.

floor()Katalogi > **floor(Lista l)** ⇒ lista**floor(Matriisi l)** ⇒ matriisi

Määrittää listan tai matriisin jokaisen elementin alarajasta.

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

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

Huomaa: Katso myös **ceiling()** ja **int()**.**fMax()**Katalogi > **fMax(Laus, Muutt)** ⇒ Boolean lauseke**fMax(Laus, Muutt, alaraja)****fMax(Laus, Muutt, alaraja, yläraja)****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.

$$\text{fMax}\left(1-(x-a)^2-(x-b)^2, x\right) \quad x = \frac{a+b}{2}$$

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

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

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

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

$$\text{fMin}\left(1-(x-a)^2-(x-b)^2, x\right) \quad x = -\infty \text{ or } x = \infty$$

$$\text{fMin}\left(0.5 \cdot x^3 - x - 2, x\right)_{x \geq 1} \quad x = 1.$$

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()**.

For *Muutt, Matala, Korkea* [, *Askel*]

Lohko

EndFor

Suurittaa *Lohkon* sisältämät lausekkeet iteratiivisesti jokaiselle muuttujan *Muutt* arvolle, *Matalasta Korkeaan* kohdassa *Askel* määritetyin portain.

Muutt ei saa olla järjestelmän muuttuja.

Askel voi olla positiivinen tai negatiivinen. Oletusarvo on 1.

Lohko voi olla joko yksi lauseke tai sarja lausekkeita, jotka on erotettu toisistaan kaksoispisteellä (:).

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

Define $g()$ =Func	<i>Done</i>
Local <i>tempsum,step,i</i>	
0 → <i>tempsum</i>	
1 → <i>step</i>	
For <i>i,1,100,step</i>	
<i>tempsum+i</i> → <i>tempsum</i>	
EndFor	
EndFunc	
<hr/>	
$g()$	5050

format()Katalogi > **format(Laus[, muotoMerkkijono])**⇒merkkijonoMäärittää 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 säätyy kolmella kerrolliseksi, ja desimaalipiste siirtyy 0, 1 tai 2 numeroa oikealle.

G[n][c]: Muuten sama kuin kiinteä muoto, mutta erottaa myös juuren vasemmalla puolella olevat numerot kolmen ryhmiin. c määrittää ryhmän erotusmerkin, ja sen oletusarvo on pilkku. Jos c on piste, juuri näytetään pilkkuna.

[Rc]: Mihin tahansa edellä mainituista määrittäjistä voidaan liittää Rc-juurilippu, jossa c on yksi merkki, joka määrittää korvauksen kohteen juuripisteestä.

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(Laus1)**⇒lauseke**fPart(Lista1)**⇒lista**fPart(Matriisi1)**⇒matriisi

Laskee argumentin murtolukuosan.

Kun kyseessä on lista tai matriisi, laskee elementtien murtolukuosat.

Argumentti voi olla reaali- tai kompleksiluku.

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

$FPdf(XArvo, dfOsoitt, dfNimit)$ ⇒ luku, jos $XArvo$ on luku, lista, jos $XArvo$ on lista

Laskee F-jakauman todennäköisyyden $XArvo$ n kohdalle määritetyille $dfOsoitt$ jalle (vapausasteet) ja $dfNimit$ äjälle.

freqTable ▶ list()

$freqTable \blacktriangleright list(Lista1, frekvKokonaislukuLista)$ ⇒ lista

Laskee listan, joka sisältää $Lista1$:n elementit lavennettuina $frekvKokonaislukuListan$ määrittämien frekvenssien mukaisesti. Tätä funktiota voidaan käyttää laadittaessa frekvenssitaulukkoa Data & Tilastot -sovelluksessa.

$Lista1$ voi olla mikä tahansa kelvollinen lista.

$frekvKokonaislukuListan$ on oltava samankokoinen kuin $Lista1$ ja sen tulee sisältää ainoastaan ei-negatiivisia kokonaislukuelementtejä. Jokainen elementti määrittää kuinka monta kertaa $Lista1$ -elementti toistetaan tuloslistassa. Nolla-arvo sulkee pois vastaavan $Lista1$ -elementin.

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

Tyhjiä elementtejä ei huomioida. Lisätietoja tyhjiistä elementeistä, katso sivu 227.

$freqTable \blacktriangleright list(\{1,2,3,4\}, \{1,4,3,1\})$	$\{1,2,2,2,3,3,3,4\}$
$freqTable \blacktriangleright list(\{1,2,3,4\}, \{1,4,0,1\})$	$\{1,2,2,2,4\}$

frequency()

$frequency(Lista1, lokerotLista)$ ⇒ lista

Luo listan, joka sisältää $Lista1$:n elementtien lukumäärät. Lukumäärät perustuvat alueisiin (lokeroihin), jotka määritetään kohtaan $lokerotLista$.

Jos $lokerotLista$ on $\{b(1), b(2), \dots, b(n)\}$, määritetyt alueet ovat $\{? \leq b(1), b(1) < ? \leq b(2), \dots, b(n-1) < ? \leq b(n), b(n) > ?\}$. Tuloksena oleva lista on yhden elementin pitempi kuin $lokerotLista$.

$datalist := \{1,2,e,3,\pi,4,5,6, \text{"hello"},7\}$	
$\{1,2,2.71828,3,3.14159,4,5,6, \text{"hello"},7\}$	
$frequency(datalist, \{2.5,4.5\})$	$\{2,4,3\}$

Vastauksen selitys:

2 $Datalistan$ elementtiä on ≤ 2.5

4 $Datalistan$ elementtiä on > 2.5 ja ≤ 4.5

Jokainen vastauksen elementti vastaa niiden *Listal*:n elementtien lukumäärää, jotka ovat kyseisen lokeron alueella. Funktion **countIf()** termeillä ilmaistuna vastaus on { countIf(list, ?≤b(1)), countIf(list, b(1)<?≤b(2)), ..., countIf(list, b(n-1)<?≤b(n)), countIf(list, b(n)>?)}.

Niitä *Listal*:n elementtejä, joita ei voi "lokeroida", ei huomioida. Tyhjiä elementtejä ei myöskään huomioida. Lisätietoja tyhjistä elementeistä, katso sivu 227.

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

Huomaa: Katso myös **countIf()**, sivu 39.

3 *Datalistan* elementtiä on >4.5

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

FTest_2Samp

FTest_2Samp *Listal,Lista2[,Frekv1[,Frekv2[,Hypot]]]*

FTest_2Samp *Listal,Lista2[,Frekv1[,Frekv2[,Hypot]]]*

(*Datalistan* syöte)

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

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

(Yhteenvetotilaston syöte)

Suorittaa kahden otoksen F -testin. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 169.)

Kun $H_a: \sigma_1 > \sigma_2$, aseta *Hypot*>0

Kun $H_a: \sigma_1 \neq \sigma_2$ (oletus), aseta *Hypot* =0

Kun $H_a: \sigma_1 < \sigma_2$, aseta *Hypot*<0

Lisätietoja listassa olevien tyhjien elementtien vaikutuksesta, katso Tyhjät elementtisivulla sivu 227.

Tulosmuuttuja	Kuvaus
stat.F	Laskettu \hat{U} -tilasto datasekvensille
stat.PVal	Alin merkitsevyystaso, jolla nollahypoteesi voidaan hylätä
stat.dfnNumer	osoittajan vapausasteet = n1-1

Tulosmuuttuja	Kuvaus
stat.dDenom	nimittäjän vapausasteet = n2-1
stat.sx1, stat.sx2	Otoksen keskihajonnot <i>Lista 1:n</i> ja <i>Lista 2:n</i> sisältämille datasekvensseille
stat.x1_bar stat.x2_bar	Otoksen keskiarvot <i>Lista 1:n</i> ja <i>Lista 2:n</i> sisältämille datasekvensseille
stat.n1, stat.n2	Otosten koko

Func

Katalogi > 

Func

Lohko

EndFunc

Malli käyttäjän määrittämän funktion luomista varten.

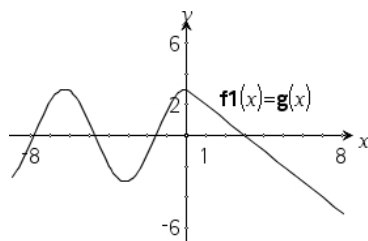
Lohko voi olla yksi lauseke tai sarja lausekkeita, jotka on erotettu toisistaan kaksoispisteellä (:), tai sarja eri riveillä olevia lausekkeita. Funktio voi käyttää **Return**-ohjetta tietyn vastauksen laskemiseen.

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

Määritä paloitain määritelty funktio:

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

Funktion g(x) kuvaajan piirtämisen tulos



G

gcd()

Katalogi > 

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

gcd()Katalogi > 

likimääräinen) -tilassa murtoluvun liukulukujen **gcd** on 1.0.

gcd(Lista1, Lista2)⇒*lista*

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

Laskee *Lista1*:n ja *Lista2*:n toisiaan vastaavien elementtien suurimmat yhteiset jakajat.

gcd(Matriisi1, Matriisi2)⇒*matriisi*

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

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

geomCdf()Katalogi > 

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

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

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

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

geomPdf()Katalogi > 

geomPdf(*p,XArvo*)⇒*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*.

getDenom()Katalogi > **getDenom(LausI)**⇒*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.

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"

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

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 102, ja **unLock**, sivu 189.

<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.

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 155.

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

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,

Tilan nimi	Tilaa vastaava kokonaisluku	Asetuksia vastaavat kokonaisluvut
		21=Kiinteä7, 22=Kiinteä8, 23=Kiinteä9, 24=Kiinteä10, 25=Kiinteä11, 26=Kiinteä12
Kulma	2	1=Radioani, 2=Aste, 3=Graadi
EkspONENTTIMUOTO	3	1=Normaali, 2=Kymmenpotenssi, 3=Tekninen
Reaali- tai kompleksiluku	4	1=Reaali, 2=Suorakulma, 3=Polaarinen
Automaattinen tai likimääräinen.	5	1=Automaattinen, 2=Likimääräinen, 3=Täsmällinen
Vektorimuoto	6	1=Suorakulma, 2=Sylinteri, 3=Pallo
Kantaluku	7	1=Desimaali, 2=Heksagonaalinen, 3=Binaarinen
Yksikköjärjestelmä	8	1=SI, 2=Eng/US

getNum()

Katalogi > 

getNum(LausI) ⇒ 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
$\text{getNum}\left(\frac{1}{x} + \frac{1}{y}\right)$	$x+y$

getType()

Katalogi > 

getType(var) ⇒ merkkijono

Antaa tulokseksi merkkijonon, joka ilmoittaa muuttujan *var* datatyyppin.

Jos muuttujaa *var* ei ole määritelty, tulokseksi tulee merkkijono "EI MITÄÄN".

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

getVarInfo() ⇒ matriisi tai merkkijono

getVarInfo(LibNameString) ⇒ matriisi tai merkkijono

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

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

getVarInfo(KirjNimiMerkkijono) antaa tuloksena tietomatriisin kaikista kirjastossa *KirjNimiMerkkijono* määritetyistä kirjasto-objekteista. *KirjNimiMerkkijono* on oltava merkkijono (lainausmerkkien sisällä oleva teksti) tai merkkijonomuuttuja.

Jos kirjastoa *KirjNimiMerkkijono* ei ole olemassa, esiintyy virhe.

Huomaa vasemmanpuoleinen esimerkki, jossa funktion **getVarInfo()** vastaus on määritetty muuttujaan *vs*. Jos muuttujan *vs* riviä 2 tai riviä 3 yritetään näyttää, tuloksena on "Kelpaamaton lista tai matriisi" -virhe, koska vähintään yksi näiden rivien elementeistä (esimerkiksi muuttuja *b*) sievenny uudelleen matriisiksi.

Tämä virhe voi esiintyä myös käytettäessä *Ans*-muuttujaa funktion **getVarInfo()** tuloksen uudelleenlaskennassa.

Järjestelmä antaa edellä mainitun virheen, koska ohjelmiston nykyinen versio ei tue yleistettyä matriisirakennetta, jossa matriisin elementti voi olla joko matriisi tai lista.

getVarInfo()	"NONE"												
Define x=5	Done												
Lock x	Done												
Define LibPriv y={ 1,2,3 }	Done												
Define LibPub z(x)=3*x ² -x	Done												
getVarInfo()	<table border="1"> <tr> <td>x</td> <td>"NUM"</td> <td>"{}"</td> <td>1</td> </tr> <tr> <td>y</td> <td>"LIST"</td> <td>"LibPriv"</td> <td>0</td> </tr> <tr> <td>z</td> <td>"FUNC"</td> <td>"LibPub"</td> <td>0</td> </tr> </table>	x	"NUM"	"{}"	1	y	"LIST"	"LibPriv"	0	z	"FUNC"	"LibPub"	0
x	"NUM"	"{}"	1										
y	"LIST"	"LibPriv"	0										
z	"FUNC"	"LibPub"	0										
getVarInfo(tmp3)	"Error: Argument must be a string"												
getVarInfo("tmp3")	<table border="1"> <tr> <td>volcyI2</td> <td>"NONE"</td> <td>"LibPub"</td> <td>0</td> </tr> </table>	volcyI2	"NONE"	"LibPub"	0								
volcyI2	"NONE"	"LibPub"	0										

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

Goto *tunnusnimi*

Siirtää ohjauksen tunnukseen *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

Done

Local *temp,i* $0 \rightarrow temp$ $1 \rightarrow i$ Lbl *top* $temp+i \rightarrow temp$ If $i < 10$ Then $i+1 \rightarrow i$ Goto *top*

EndIf

Return *temp*

EndFunc

 $g()$

55

▶ Grad

Laus1 ▶ **Grad** ⇒ *lauseke*

Muuttaa *Laus1*:n graadikulmaan.

Huomaa: Voit syöttää tämän operaattorin tietokoneen näppäimistöltä kirjoittamalla @>**Gr****a****d**.

Astekulmatilassa:

 $(1.5) \blacktriangleright \text{Grad}$ $(1.66667)^{\circ}$

Radiaanikulmatilassa:

 $(1.5) \blacktriangleright \text{Grad}$ $(95.493)^{\circ}$

/

identity()**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

If *BooleanLaus*
lauseke

Define $g(x)=$ Func	<i>Done</i>
If $x<0$ Then	
Return x^2	
EndIf	
EndFunc	

If *BooleanLaus Then*
Lohko

EndIf

$g(-2)$	4
---------	---

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 (:).

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

If *BooleanLaus Then*
Lohko1

Else

Lohko2

EndIf

Define $g(x)=$ Func	<i>Done</i>
If $x<0$ Then	
Return $-x$	
Else	
Return x	
EndIf	
EndFunc	

Jos *BooleanLaus* on tosi, suorittaa *Lohko1*:n ja ohittaa sen jälkeen *Lohko2*:n.

Jos *BooleanLaus* on epätosi, ohittaa *Lohko1*:n, mutta suorittaa *Lohko2*:n.

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

Lohko1 ja *Lohko2* voivat olla yksi lauseke.

If BooleanLaus1 Then

Lohko1

Elsel BooleanLaus2 Then

Lohko2

⋮

Elsel BooleanLausN Then

LohkoN

EndIf

Sallii hypyt. Jos *BooleanLaus1* on tosi, suorittaa *Lohko1*:n. Jos *BooleanLaus1* on epätosi, laskee *BooleanLaus2*:n jne.

Define $g(x)=\text{Func}$ If $x < 5$ Then

Return 5

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

Return 3

EndIf

EndFunc

Done

$g(-4)$	4
$g(10)$	3

ifFn()

ifFn(BooleanLaus, Arvo_Jos_tosi [, Arvo_Jos_epätosi [, Arvo_Jos_tuntematon]]) ⇒ lauseke, lista tai matriisi

Laskee Boolean lausekkeen *BooleanLaus* (kaikille *BooleanLaus*:n elementeille) ja laskee tuloksen noudattaen seuraavia sääntöjä:

- *BooleanLaus* voi testata yhtä arvoa, listaa tai matriisia.
- Jos jokin *BooleanLaus*:n elementti on tosi, laskee vastaavan elementin lausekkeesta *Arvo_Jos_tosi*.
- Jos jokin *BooleanLaus*:n elementti on epätosi, laskee vastaavan elementin lausekkeesta *Arvo_Jos_epätosi*. Jos jätät pois lausekkeen *Arvo_Jos_epätosi*, antaa vastauksen undef.
- Jos *BooleanLaus*:n elementti ei ole tosi eikä epätosi, laskee vastaavan elementin *Arvo_Jos_tuntematon*. Jos jätät pois lausekkeen *Arvo_Jos_tuntematon*, antaa vastauksen undef.
- Jos funktion **ifFn()** toinen, kolmas tai neljäs argumentti on yksi lauseke, Boolean testiä sovelletaan jokaiseen *BooleanLaus*:n position.

Huomaa: Jos sievennetty *BooleanLaus*-lauseke

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

1:n testi-arvo on alle 2.5, joten se vastaa

Arvo_Jos_Tosi-elementti arvolle **5** kopioidaan vastauslistaan.

2:n testi-arvo on alle 2.5, joten se vastaa

Arvo_Jos_Tosi-elementti arvolle **6** kopioidaan vastauslistaan.

3:n testi-arvo ei ole alle 2.5, joten se vastaa *Arvo_Jos_*

Epätosi-elementti arvolle **10** kopioidaan vastauslistaan.

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

Arvo_Jos_tosi on yksi arvo ja vastaa mitä tahansa valittua paikkaa.

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

ifFn()Katalogi > 

sisältää listan tai matriisin, kaikkien muiden lista- tai matriisiargumenttien on oltava samankokoisia, ja myös vastaus on samankokoinen.

Arvo_Jos_epätosi-elementtiä ei ole määritetty. Käytetään merkintää *Undef*.

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

$$\{6, "err"\}$$

Yksi elementti valittu lausekkeesta *Arvo_Jos_tosi*.

Yksi elementti valittu lausekkeesta *Arvo_Jos_epätosi*.

imag()Katalogi > 

imag(LausI) ⇒ lauseke

Laskee argumentin imaginaarisen osan.

Huomaa: Kaikkia määrittämättömiä muuttujia käsitellään reaalimuuttujina. Katso myös **real()**, sivu 140

imag(ListaI) ⇒ lista

Laskee listan elementtien imaginaarisista osista.

imag(MatriisiI) ⇒ matriisi

Laskee matriisin elementtien imaginaarisista osista.

$\text{imag}(1+2 \cdot i)$	2
$\text{imag}(z)$	0
$\text{imag}(x+i \cdot y)$	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()Katalogi > 

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

jossa asteen *Aste* oletusarvo on 1.

Laskee implisiittisen derivaatan yhtälöille, joissa toinen muuttuja määritetään implisiittisesti toisen suhteen.

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

$$\begin{matrix} -x \\ y \end{matrix}$$

Epäsuora operaattori

Katso #(), sivu 217.

inString()Katalogi > **inString**(*srcMerkkijono*, *alaMerkkijono*[, *Alku*])

⇒kokonaisluku

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

Määrittää merkin paikan merkkijonossa *srcMerkkijono*, jonka kohdalla merkkijonon *alaMerkkijono* ensimmäinen esiintymisen alkaa.

Alku, mikäli se on mukana, määrittää merkin paikan sen merkkijonon *srcMerkkijonosisällä*, josta haku alkaa. Oletusarvo = 1 (*srcMerkkijonon* ensimmäinen merkki).

Jos *srcMerkkijono* ei sisällä *alaMerkkijonoa* tai *Alku* on > *srcMerkkijonon* pituus, vastaus on nolla.

int()Katalogi > **int**(*Laus*)⇒kokonaislukuint(-2.5) -3.**int**(*Listal*)⇒listaint([-1.234 0 0.37]) [-2. 0 0.]**int**(*Matriisi l*)⇒matriisi

Laskee suurimman kokonaisluvun, joka on pienempi tai yhtä suuri kuin argumentti. Tämä funktio on identtinen funktion **floor()** kanssa.

Argumentti voi olla reaali- tai kompleksiluku.

Kun kyseessä on lista tai matriisi, laskee kunkin elementin suurimman kokonaisluvun.

intDiv()Katalogi > **intDiv**(*Luku1*, *Luku2*)⇒kokonaislukuintDiv(-7,2) -3**intDiv**(*Listal*, *Listal2*)⇒listaintDiv(4,5) 0**intDiv**(*Matriisi1*, *Matriisi2*)⇒matriisiintDiv({12, -14, -16}, {5, 4, 3}) {2, -3, 5}

Määrittää lausekkeen (*Luku1* ÷ *Luku2*) etumerkillisen kokonaislukuosan.

Listojen ja matriisien kohdalla, laskee lausekkeen (argumentti 1 + argumentti 2) etumerkillisen kokonaislukuosan kullekin elementtiparille.

interpolate ()

Katalogi > **interpolate**(*xValue*, *xList*, *yList*, *yPrimeList*)⇒ *luettelo*

Tämä toiminto tekee seuraavaa:

Kun ilmoitetaan *xList*, *yList=f(xList)* ja *yPrimeList=f'(xList)* jollekin tuntemattomalle funktiolle *f*, käytetään neliöväliarvoa funktion *f* määrittelemiseksi arvossa *xValue*. Oletetaan, että *xList* on luettelo monotonisesti kasvavia tai laskevia numeroita, mutta tämän toiminnon tulokseksi voi tulla arvo, vaikka se ei olisikaan. Tämä toiminto käy läpi *xList*-luettelon etsien väliä [*xList*[*i*], *xList*[*i*+1]] joka sisältää arvon *xValue*. Jos se löytää tällaisen välin, se palauttaa väliarvolasketun arvon funktiolle *f(xValue)*; muuten se antaa tuloksen **undef**.

Luetteloiden *xList*, *yList* ja *yPrimeList* on oltava samankokoiset ≥ 2 ja sisällettävä lausekkeita, jotka sieventyvät luvuiksi.




xValue voi olla määrittämätön muuttuja, luku tai luku luettelo.

Differensiaal yhtälö:

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

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

0.	1.	2.	3.	4.	
5.	3.19499	5.00394	6.99957	9.00593	10

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

Käytä `interpoloi()`-funktiota laskeaksesi *xValue*-listin funktion arvot:

```
xvalueList:=seq(i,i,0,10,0.5)
{0,0.5,1.,1.5,2.,2.5,3.,3.5,4.,4.5,5.,5.5,6.,6.5,7.,7.5,8.,8.5,9.,9.5,10.}
xlist:=mat▶list(rk[1])
{0.,1.,2.,3.,4.,5.,6.,7.,8.,9.,10.}
ylist:=mat▶list(rk[2])
{5.,3.19499,5.00394,6.99957,9.00593,10.9979}
yprimeList:=-3*y+6*t+5|y=yList and t=xList
{-10.,1.41503,1.98819,2.00129,1.98221,2.00621}
interpolat(xvalueList,xlist,yList,yprimeList)
{5.,2.67062,3.19499,4.02782,5.00394,6.00011}
```

inv χ^2 ()Katalogi > **inv χ^2** (*Ala*,*df*)**invchi2**(*Ala*,*df*)

Laskee käänteisen kumulatiivisen χ^2 (chi-neliö) -todennäköisyysfunktion, joka on määritetty vapausasteella *df* tietyllä käyrän alapuoliselle alueelle *Ala*.

invF()

Katalogi > **invF**(*Ala*,*dfOsoitt*,*dfNimit*)

invF()Katalogi > **invF(Ala,dfOsoitt,dfNimit)**

Laskee käänteisen kumulatiivisen F-jakaumafunktion, jolle on määritetty *dfOsoitt* ja *dfNimit*, tietylle käyrän alapuoliselle alueelle *Ala*.

invNorm()Katalogi > **invNorm(Ala[,μ,σ])**

Laskee käänteisen kumulatiivisen normaalijakaumafunktion tietylle normaalijakaumakäyrän alapuolella olevalle *Alalle*, jolle on määritetty μ ja σ .

invt()Katalogi > **invt(Ala,df)**

Laskee käänteisen kumulatiivisen Studentin t-todennäköisyysfunktion, jolle on määritetty vapausaste, *df*, tietylle käyrän alapuoliselle *Alalle*.

iPart()Katalogi > **iPart(Luku)**⇒kokonaisluku**iPart(Lista1)**⇒lista**iPart(Matriisi1)**⇒matriisi

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

Laskee argumentin kokonaisosan.

Listojen ja matriisien kohdalla, laskee jokaisen elementin kokonaisosan.

Argumentti voi olla reaali- tai kompleksiluku.

irr($CF_0, CFLista [, CFFrekv]$) \Rightarrow arvo

Talouselaskentatoiminto, joka laskee investoinnin sisäisen korkokannan.

CF_0 on alkuperäinen kassavirta aikana 0; arvon on oltava kokonaisluku.

$CFLista$ on lista kassavirtamääristä ensimmäisen kassavirran CF_0 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 **mirr()**, sivu 110.

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

isPrime()

isPrime($Luku$) \Rightarrow Boolean vakiolauseke

Määrittää totuusarvon tosi tai epätosi ilmaisten, 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 määrittää, onko $Luku$ jaoton, käytä komentoa **isPrime**() funktion **factor**() sijaan. Se on paljon nopeampi, erityisesti jos $Luku$ ei 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.

$isPrime(5)$	true
$isPrime(6)$	false

Funktio, jolla etsitään seuraava jaoton luku määritetyn luvun jälkeen:

Define $nextprim(n) =$ Func	Done
Loop	
$n + 1 \rightarrow n$	
If $isPrime(n)$	
Return n	
EndLoop	
EndFunc	
$nextprim(7)$	11

isVoid()

Katalogi > 

isVoid(*Muutt*) \Rightarrow *Boolean vakiolauseke*

isVoid(*Laus*) \Rightarrow *Boolean vakiolauseke*

isVoid(*Lista*) \Rightarrow *Boolean vakiolausekkeiden lista*

Määrittää totuusarvon tosi tai epätosi ilmaisten, onko argumentti tyhjä datatyyppi.

Lisätietoja tyhjistä elementeistä, katso sivu 227.

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

L

Lbl

Katalogi > 

Lbl *tunnusnimi*

Määrittää funktion sisällä tunnuksen, jonka nimi on *tunnusnimi*.

Ohjeella **Sirry** *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(*Luku1, Luku2*) \Rightarrow *lauseke*

lcm(*Lista1, Lista2*) \Rightarrow *lista*

lcm(*Matriisi1, Matriisi2*) \Rightarrow *matriisi*

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.

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

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 33.

Muuttujaryhmän poistaminen, katso **DelVar** sivulla sivu 52.

Tässä esimerkissä edellytetään asianmukaisesti tallennettua ja päivitettyä kirjastoasiakirjaa nimeltä **linalg2**, joka sisältää objektit *clearmat*, *gauss1* ja *gauss2*.

getVarInfo("linalg2")

<i>clearmat</i>	"FUNC"	"LibPub "
<i>gauss1</i>	"PRGM"	"LibPriv "
<i>gauss2</i>	"FUNC"	"LibPub "

libShortcut("linalg2", "la")

{*la.clearmat*,*la.gauss2*}

libShortcut("linalg2", "la", 1)

{*la.clearmat*,*la.gauss1*,*la.gauss2*}

limit(LausI, Muutt, Piste[, Suunta]) ⇒ lauseke
raja-arvo(ListaI, Muutt, Piste [, Suunta]) ⇒ lista
raja-arvo(MatriisiI, Muutt, Piste [, Suunta])
 ⇒ matriisi

Laskee pyydetyn raja-arvon.

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

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

Positiivisen ∞ :n ja negatiivisen $-\infty$:n kohdalla olevat raja-arvot muunnetaan aina äärellisen puolen yksipuolisiksi raja-arvoiksi.

Tilanteesta riippuen **limit()** antaa vastauksena itsensä tai 'undef' silloin, kun se ei pysty määrittämään yksilöllistä raja-arvoa. Tämä ei välttämättä tarkoita, että yksilöllistä raja-arvoa ei ole olemassa. undef tarkoittaa, että vastaus on joko tuntematon luku, jonka suuruus on äärellinen tai ääretön, tai se on tällaisten lukujen koko sarja.

limit() käyttää menetelmiä, kuten L'Hopitalin sääntöä, joten on olemassa yksilöllisiä raja-arvoja, joita se ei pysty määrittämään. Jos *LausI* sisältää muita määrittämättömiä muuttujia kuin *Muutt*, joudut mahdollisesti rajoittamaan niitä saadaksesi suppeamman tuloksen.

Raja-arvot voivat olla hyvin herkkiä pyöristysvirheille. Mikäli mahdollista, vältä **Auto or Approximate (Automaattinen tai Likimääräinen)** -tilan Approximate (Likimääräinen) -asetusta ja likiarvoista luvut laskeessasi 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 regressioy = a+b · x listoista X ja Y frekvenssillä $Frekv$. Tulosten yhteenveveto tallentuu *stat.results*-muuttujaan. (Katso sivu 169.)

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 elementitsivulla sivu 227.

Tulosmuuttuja	Kuvaus
stat.RegEqn	Regressioyhtälö: $a+b \cdot x$
stat.a, stat.b	Regressiokertoimet
stat.r ²	Määrittyskerroin
stat.r	Korrelaatiokerroin
stat.Resid	Regressioyhtälön jäännökset
stat.XReg	Muokatus Y Listan sisältämä datapisteiden lista, jota käytetään regressiossa komentojen $Frekv$, $Luokkalista$ ja $Sisällytä$ luokat rajoitusten mukaisesti
stat.YReg	Muokatus X Listan sisältämä datapisteiden lista, jota käytetään regressiossa komentojen $Frekv$, $Luokkalista$ ja $Sisällytä$ luokat rajoitusten mukaisesti
stat.FreqReg	Komentoja $stat.XReg$ ja $stat.YReg$ vastaava frekvenssilista

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

Laskee lineaarisen regressioy = m · x+b listoista X ja Y

frekvenssillä *Frekv*. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 169.)

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 227.

Tulosmuuttuja	Kuvaus
stat.RegEqn	Regressioyhtälö: $m \cdot x + b$
stat.m, stat.b	Regressiokertoimet
stat.r ²	Määrittäyskerroin
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 luokat rajoitusten mukaisesti</i>
stat.YReg	Muokatus X Lista: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

LinRegtIntervals $X, Y[, F[, O[, CTaso]]]$

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 169.)

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 tyhjien elementtien vaikutuksesta, katso Tyhjät elementtsivulla sivu 227.

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
stat.SESlope	Kulmakertoimen keskivirhe
stat.s	Keskivirhe suoran ympärillä

Vain Vaste-tyyppi

Tulosmuuttuja	Kuvaus
[stat.CLower, stat.CUpper]	Keskiarvovasteen luottamusväli
stat.ME	Luottamusvälin virhemarginaali
stat.SE	Keskiarvovasteen keskivirhe
[stat.LowerPred, stat.UpperPred]	Yhden havainnon ennusteväli

Tulosmuuttuja	Kuvaus
stat.MEPred	Ennustevälin virhemarginaali
stat.SEPred	Ennusteen keskivirhe
stat.ȳ	$a + b \cdot X_{\text{Arvo}}$

LinRegtTest

Katalogi > 

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

Laskee lineaarisen regression X - ja Y -listoista ja suorittaa t -testin kulmakertoimen 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 $\text{Frekv}:n$ elementti määrittää kunkin vastaavan datapisteen X ja Y esiintymisfrekvenssin. Oletusarvo on 1. Kaikkien elementtien on oltava kokonaislukuja 0.

Hypot on valinnainen arvo, joka määrittää yhden kolmesta hypoteesista, johon nollahypoteesia ($H_0: \beta = \rho = 0$) testataan.

Kun $H_a: \beta \neq 0$ ja $\rho \neq 0$ (oletus), aseta $\text{Hypot} = 0$

Kun $H_a: \beta < 0$ ja $\rho < 0$, aseta $\text{Hypot} < 0$

Kun $H_a: \beta > 0$ ja $\rho > 0$, aseta $\text{Hypot} > 0$

Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu sivu 169.)

Lisätietoja listassa olevien tyhjien elementtien vaikutuksesta, katso Tyhjät elementitsivulla sivu 227.

Tulosmuuttuja	Kuvaus
stat.RegEqn	Regressioyhtälö: $a + b \cdot x$
stat.t	t -tilasto merkitsevyydestille
stat.PVal	Alin merkitsevyydestaso, jolla nollahypoteesi voidaan hylätä
stat.df	Vapausasteet
stat.a, stat.b	Regressiokertoimet
stat.s	Keskivirhe suoran ympärillä

Tulosmuuttuja	Kuvaus
stat.SESlope	Kulmakertoimen keskiarvo
stat.r ²	Määrittyskerroin
stat.r	Korrelaatiokerroin
stat.Resid	Regressioyhtälön jäännökset

linSolve()

Katalogi > 

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

⇒lista

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

linSolve(Lineaariyht1 ja Lineaariyht2 ja ..., Muutt1

, Muutt2, ...)⇒lista

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

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

⇒lista

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

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

⇒lista

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

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

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

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

⇒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 Argumenttivirhe.

ΔList()

Katalogi > 

ΔList(Lista1)⇒lista

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

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

Δ List()

Katalogi > 

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()

Katalogi > 

list▶mat(Lista [, elementtiäRivillä])⇒matriisi

Laskee matriisin, joka on täytetty rivi riviltä *Listan* elementteillä.

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

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

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

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

▶ln

Katalogi > 

Laus ▶ln⇒lauseke

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

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

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

ln()

  painikkeet

ln(Lausl)⇒lauseke

ln(2.)	0.693147
--------	----------

ln(Listal)⇒lista

Määrittää argumentin luonnollisen logaritmin.

Jos kyseessä on lista, määrittää elementtien luonnolliset logaritmit.

Jos kompleksilukumuodon tila on Reaali:

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

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()**.

neliömatriisi I:n on oltava diagonalisoitavissa.




Vastaus sisältää aina liukulukuja.

Jos kompleksilukumuodon tila on Suorakulma:

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

Radiaanikulmatilassa ja suorakulmakompleksimuodossa:

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

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

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

Laskee logaritmisen regression $y = a + b \cdot \ln(x)$ listoista X ja Y frekvenssillä *Frekv*. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 169.)

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 -dataalle.

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

Lisätietoja listassa olevien tyhjien elementtien vaikutuksesta, katso Tyhjät elementitsivulla sivu 227.

Tulosmuuttuja	Kuvaus
stat.RegEqn	Regressioyhtälö: $a + b \cdot \ln(x)$

Tulosmuuttuja	Kuvaus
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
stat.ResidTrans	Muunnettujen tietojen lineaariseen sovitukseen liittyvät jäännökset
stat.XReg	Muokatus <i>Y Lista</i> :n sisältämä datapisteiden lista, jota käytetään regressiossa komentojen <i>Freqv</i> , <i>Luokkalista</i> ja <i>Sisällytä luokat rajoitusten mukaisesti</i>
stat.YReg	Muokatus <i>Y Lista</i> :n sisältämä datapisteiden lista, jota käytetään regressiossa komentojen <i>Freqv</i> , <i>Luokkalista</i> ja <i>Sisällytä luokat rajoitusten mukaisesti</i>
stat.FreqReg	Komentoja <i>stat.XReg</i> ja <i>stat.YReg</i> vastaava <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

rollcount()

16

rollcount()

3

LockMuutt1[, Muutt2] [, Muutt3] ...

LockMuutt.

Lukitsee määritetyt muuttujat tai muuttujaryhmän.

Lukittuja muuttujia ei voi muokata eikä poistaa.

Et voi lukita tai vapauttaa järjestelmän muuttujaa *Ans*, etkä voi lukita järjestelmän muuttujaryhmiä *stat.* tai *tvm.*

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

Katso **unLock**, sivu 189, ja **getLockInfo()**, sivu 80.

<code>a:=65</code>	65
<code>Lock a</code>	<i>Done</i>
<code>getLockInfo(a)</code>	1
<code>a:=75</code>	"Error: Variable is locked."
<code>DelVar a</code>	"Error: Variable is locked."
<code>Unlock a</code>	<i>Done</i>
<code>a:=75</code>	75
<code>DelVar a</code>	<i>Done</i>

log()

  **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 toinen argumentti jätetään pois, kantalukuna käytetään lukua 10.

Jos kompleksilukumuodon tila on Reaali:

$\log_{10} (\{-3,1,2,5\})$	Error: Non-real result
----------------------------	------------------------

Jos kompleksilukumuodon tila on Suorakulma:

$\log_{10} (\{-3,1,2,5\})$	
$\left\{ \log_{10} (3) + 1.36438 \cdot i, 0.079181, \log_{10} (5) \right\}$	

log(neliömatriisiI[,Laus])⇒*neliömatriisi*

Laskee matriisin kantaluku-*Laus*:n logaritmin *neliömatriisiI*:lle. Tämä ei ole sama kuin kunkin elementin kantaluku-*Laus*-logaritmin laskeminen. Laskentamenetelmä on kuvattu kohdassa **cos()**.

neliömatriisiI:n on oltava diagonalisoitavissa.




Vastaus sisältää aina liukulukuja.

Radiaanikulmatilassa ja suorakulmakompleksimuodossa:

Jos kantalukuargumentti jätetään pois, kantalukuna käytetään lukua 10.

$$\log_{10} \begin{pmatrix} 1 & 5 & 3 \\ 4 & 2 & 1 \\ 6 & -2 & 1 \end{pmatrix}$$

$$\begin{bmatrix} 0.795387+0.753438 \cdot i & 0.003993-0.6474i \\ 0.194895-0.315095 \cdot i & 0.462485+0.27077i \\ -0.115909-0.904706 \cdot i & 0.488304+0.7774i \end{bmatrix}$$

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

►logbase

Laus ►logbase(*Laus1*)⇒*lauseke*

Aiheuttaa sen, että syötteenä oleva Lauseke sievennetään lausekkeeksi, joka käyttää kantaluku-*Laus1*:ä.

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

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

Logistic

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

Laskee logistisen regressioy = $c/(1+a \cdot e^{-bx})$ listoista *X* ja *Y* frekvenssillä *Frekv*. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 169.)

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 elementitsivulla sivu 227.

Tulosmuuttuja	Kuvaus
stat.RegEqn	Regressioyhtälö: $c/(1+a \cdot e^{-bx})$
stat.a, stat.b, stat.c	Regressiokertoimet
stat.Resid	Regressioyhtälön jäännökset
stat.XReg	Muokatus <i>Y Lista</i> :n sisältämä datapisteiden lista, jota käytetään regressiossa komentojen <i>Frekv</i> , <i>Luokkalista</i> ja <i>Sisällytä huokat 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 huokat rajoitusten mukaisesti</i>
stat.FreqReg	Komentoja <i>stat.XReg</i> ja <i>stat.YReg</i> vastaava frekvenssilista

LogisticD

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

Laskee logistisen regression $y = c/(1+a \cdot e^{-bx}+d)$ listoista *X* ja *Y* frekvenssillä *Frekv* käyttäen tiettyä *Iteraatioiden* määrää.

Tulosten yhteenveto tallentuu *stat.results*-muuttuun. (Katso sivu 169.)

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

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

Iteraatio on valinnainen arvo, joka määrittää ratkaisun yrityskertojen enimmäismäärän. Mikäli se jätetään pois, käytetään arvoa 64. Suuremmilla arvoilla saadaan tyypillisesti parempi tarkkuus, mutta suoritus aika on pitempi ja päin vastoin.

Frekv on valinnainen frekvenssiarvojen lista. Jokainen *Frekv*:n elementi 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 elementtivuorolla sivu 227.

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</i> Listan 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</i> Listan 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 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

```
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 `ctrl` `enter` tai **Automaattinen tai likimääräinen**-tilan valintaa Approximate (Likimääräinen), laskut suoritetaan liukulukuaritmeilla.
- Jos *Tol* jätetään pois tai sitä ei käytetä, oletusarvoinen toleranssi lasketaan seuraavasti:
5E-14 · max(dim(*Matriisi*)) · riviNorm(*Matriisi*)

LU-dekomponointialgoritmi käyttää osittaista rivien vaihtoa (pivoting).

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

LU *mI*, *lower*, *upper*, *perm* Done

$$\begin{array}{l} \text{lower} \\ \text{upper} \\ \text{perm} \end{array} \quad \begin{bmatrix} 1 & 0 & 0 \\ \frac{5}{6} & 1 & 0 \\ \frac{1}{2} & \frac{1}{2} & 1 \end{bmatrix}$$

$$\begin{array}{l} \text{lower} \\ \text{upper} \\ \text{perm} \end{array} \quad \begin{bmatrix} 6 & 12 & 18 \\ 0 & 4 & 16 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\begin{array}{l} \text{lower} \\ \text{upper} \\ \text{perm} \end{array} \quad \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\begin{bmatrix} m & n \\ o & p \end{bmatrix} \rightarrow mI \quad \begin{bmatrix} m & n \\ o & p \end{bmatrix}$$

LU *mI*, *lower*, *upper*, *perm* Done

$$\begin{array}{l} \text{lower} \\ \text{upper} \\ \text{perm} \end{array} \quad \begin{bmatrix} 1 & 0 \\ \frac{m}{o} & 1 \end{bmatrix}$$

$$\begin{array}{l} \text{lower} \\ \text{upper} \\ \text{perm} \end{array} \quad \begin{bmatrix} o & p \\ 0 & n - \frac{m \cdot p}{o} \end{bmatrix}$$

$$\begin{array}{l} \text{lower} \\ \text{upper} \\ \text{perm} \end{array} \quad \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

M**matlist()**

matlist(*Matriisi*) ⇒ *lista*

Luo listan, joka on täytetty *Matriisin* elementeillä. Elementit kopioidaan *Matriisista* rivi riviltä.

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

$$\text{matlist}(\begin{bmatrix} 1 & 2 & 3 \end{bmatrix}) \quad \{1,2,3\}$$

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix} \rightarrow mI \quad \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}$$

$$\text{matlist}(mI) \quad \{1,2,3,4,5,6\}$$

max()Katalogi > **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 maksimiaron. Jos argumentteina on kaksi listaa tai matriisia, laskee listan tai matriisin, joka sisältää vastaavien elementtien kunkin parin maksimiaron.

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 tyhjiistä elementeistä, katso sivu 227.

Huomaa: Katso myös **fMax()** ja **min()**.**mean()**Katalogi > **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*

Suorakulmavektorimuodossa:

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

Luo rivivektorin kaikkien *Matriisi1*:n sarakkeiden keskiarvoista.

Jokainen *frekvMatriisin* elementti näyttää *Matriisi1*:n vastaavien elementtien peräkkäisten esiintymien lukumäärän.

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

Tyhjiä elementtejä ei huomioida. Lisätietoja tyhjiistä elementeistä, katso sivu 227.

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

median(Lista[, frekvLista])⇒lauseke

median({0.2,0,1,-0.3,0.4})

0.2

Laskee *Listan* elementtien mediaanin.

Jokainen *frekvLista*n elementti näyttää *Listan* vastaavien elementtien peräkkäisten esiintymien lukumäärän.

median(MatriisiI[, frekvMatriisi])⇒matriisi

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

$$\text{median} \begin{pmatrix} 0.2 & 0 \\ 1 & -0.3 \\ 0.4 & -0.5 \end{pmatrix} \quad [0.4 \quad -0.3]$$

Jokainen *frekvMatriisin* elementti 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 tyhjiä elementeistä, katso sivu 227.

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

Laskee mediaani-mediaani-suorany = (m · x+b)listoista *X* ja *Y* frekvenssillä *Frekv*. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 169.)

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 elementtisivulla sivu 227.

Tulosmuuttuja	Kuvaus
stat.RegEqn	Mediaani-mediaani-suoran yhtälö: $m \cdot x + b$
stat.m, stat.b	Mallin kertoimet
stat.Resid	Mediaani-mediaani-suoran jäännökset
stat.XReg	Muokatus <i>Y Lista</i> :n sisältämä datapisteiden lista, jota käytetään regressiossa komentojen <i>Frekv</i> , <i>Luokkalista</i> ja <i>Sisällytä luokat rajoitusten mukaisesti</i>
stat.YReg	Muokatus <i>Y Lista</i> :n sisältämä datapisteiden lista, jota käytetään regressiossa komentojen <i>Frekv</i> , <i>Luokkalista</i> ja <i>Sisällytä luokat rajoitusten mukaisesti</i>
stat.FreqReg	Komentoja <i>stat.XReg</i> ja <i>stat.YReg</i> vastaava frekvenssilista

mid()

Katalogi > 

mid(lähdemerkkijono, Alku[, Count]) ⇒ merkkijono

Laskee *Count*:n merkit merkkijonosta *lähdemerkkijono* aloittaen merkistä numero *Alku*.

Jos *Count* jätetään pois, tai jos se on suurempi kuin *lähdemerkkijono*, laskee kaikki merkit *lähdemerkkijonosta* aloittaen merkistä numero *Alku*.

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

mid(lähdelista, Alku[, Count]) ⇒ lista

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

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

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

mid(lähdeMerkkijonoLista, Alku[, Count]) ⇒ lista

Laskee *Count*:n merkkijonot merkkijonolistasta *lähdeMerkkijonoLista* aloittaen elementistä numero *Alku*.

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

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

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

min()Katalogi > **min**(*Laus1*, *Laus2*)⇒*lauseke* $\min(2.3, 1.4)$ 1.4 $\min(\{1, 2\}, \{-4, 3\})$ $\{-4, 2\}$ **min**(*Lista1*, *Lista2*)⇒*lista***min**(*Matriisi1*, *Matriisi2*)⇒*matriisi*

Laskee kahden argumentin minimiarvon. Jos argumentteina on kaksi listaa tai matriisia, laskee listan tai matriisin, joka sisältää vastaavien elementtien kunkin parin minimiarvon.

min(*Lista*)⇒*lauseke* $\min(\{0, 1, -7, 1.3, 0.5\})$ -7Laskee *Listan* minimielementin.**min**(*Matriisi1*)⇒*matriisi* $\min\left(\begin{bmatrix} 1 & -3 & 7 \\ -4 & 0 & 0.3 \end{bmatrix}\right)$ $\begin{bmatrix} -4 & -3 & 0.3 \end{bmatrix}$ Luo rivivektorin, joka sisältää *Matriisi1*:n jokaisen sarakkeen minimielementin.**Huomaa:** Katso myös **fMin()** ja **max()**.**mirr()**Katalogi > **mirr**(*tal.arvo*, *uud.invest.arvo*, *CF0*, *CFLista* [, *CFFrekv*]) $list1 := \{6000, -8000, 2000, -3000\}$
 $\{6000, -8000, 2000, -3000\}$

Talouselaskentatoiminto, joka laskee investoinnin modifioidun sisäisen korkokannan.

 $list2 := \{2, 2, 2, 1\}$ $\{2, 2, 2, 1\}$ $\text{mirr}(4.65, 12, 5000, list1, list2)$ 13.41608607*tal.arvo* on kassavirtamäärästä maksettava korkoprosentti.*uud.invest.arvo* on korkokanta, jolla kassavirrat investoidaan uudelleen.*CF0* on alkuperäinen kassavirta aikana 0; arvon on oltava kokonaisluku.*CFLista* on lista kassavirtamäärästä 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 90.

mod()Katalogi > **mod**(*Laus1*, *Laus2*) \Rightarrow lauseke

mod(7,0) 7

mod(*Listal*, *Lista2*) \Rightarrow lista

mod(7,3) 1

mod(*Matriisi1*, *Matriisi2*) \Rightarrow matriisi

mod(-7,3) 2

Laskee ensimmäisen argumentin modulo toinen argumentti identtisten yhtälöiden määrittelyn mukaisesti:

mod(-7,-3) -2

mod(-7,-3) -1

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

mod(x,0) = x

mod(x,y) = x - y 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 142

mRow()Katalogi > **mRow**(*Laus*, *Matriisi1*, *Indeksi*) \Rightarrow matriisi

Luo kopion *Matriisi1*:stä, jossa jokainen rivi *Indeksi* elementti *Matriisi1*:ssä on kerrottu lausekkeella *Laus*.

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

mRowAdd()Katalogi > **mRowAdd**(*Laus*, *Matriisi1*, *Indeksi1*, *Indeksi2*) \Rightarrow matriisi

Luo kopion *Matriisi1*:stä, jossa jokainen rivi *Indeksi2* elementti *Matriisi1*:ssä on korvattu seuraavasti:

mRowAdd(-3, $\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$, 1, 2)	$\begin{bmatrix} 1 & 2 \\ 0 & -2 \end{bmatrix}$
mRowAdd(<i>n</i> , $\begin{bmatrix} a & b \\ c & d \end{bmatrix}$, 1, 2)	$\begin{bmatrix} a & b \\ a \cdot n + c & b \cdot n + d \end{bmatrix}$

Laus · rivi *Indeksi1* + rivi *Indeksi2*

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

Laskee listan Y moninkertaisen lineaarisen regression listojen $X1, X2, \dots, X10$ suhteen. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 169.)

Kaikkien listojen on oltava samankokoisia.

Lisätietoja listassa olevien tyhjiin elementtien vaikutuksesta, katso Tyhjät elementitsivulla sivu 227.

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.ϕLista	\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 169.)

Kaikkien listojen on oltava samankokoisia.

Lisätietoja listassa olevien tyhjiin elementtien vaikutuksesta, katso Tyhjät elementitsivulla sivu 227.

Tulosmuuttuja	Kuvaus
stat.RegEqn	Regressioyhtälö: $b_0 + b_1 \cdot x_1 + b_2 \cdot x_2 + \dots$
stat.ϕ	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

Tulosmuuttuja	Kuvaus
stat.LowerPred, stat.UpperrPred	Yhden havainnon ennusteväli
stat.MEPred	Ennustevälin virhemarginaali
stat.SEPred	Ennusteen keskivirhe
stat.bList	Regressiokertoimien lista, {b0,b1,b3,...}
stat.Resid	Regressioyhtälön jäännökset

MultRegTests

Katalogi > 

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

Moninkertaisen lineaarisen regression testi laskee lineaarisen regression tietystä datasta ja määrittää kertoimille globaalin F -testin tilastot sekä t -testin tilastot.

Tulosten yhteenveto tallentuu *stat.results*-muuttujaan.
(Katso sivu sivu 169.)

Lisätietoja listassa olevien tyhjien elementtien vaikutuksesta, katso Tyhjät elementitsivulla sivu 227.

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 määrityksen kerroin
stat.AdjR ²	Moninkertaisen määrityksen 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

Tulosmuuttuja	Kuvaus
stat.SSError	Virheen neliöiden summa
stat.MSError	Virheen keskineliö
stat.bList	{b0,b1,...} Kertoimien lista
stat.tList	t-tilastojen lista, yksi kullekin bListan sisältämälle kertoimelle
stat.PList	P-arvojen lista kullekin t-tilastolle
stat.SEList	Keskivirheiden lista bListan sisältämille kertoimille
stat.yLista	yLista = b0+b1 · x1+ ...
stat.Resid	Regressioyhtälön jäännökset
stat.sResid	Standardoidut jäännökset; saadaan jakamalla jäännös keskihajonnalla
stat.CookDist	Cookin etäisyys; jäännökseen ja tuottosuhteeseen perustuvan havainnon vaikutus
stat.Leverage	Miten kaukana riippumattoman muuttujan arvot ovat niiden keskiarvoista

N

nand (ei-ja)

  näppäimet

BooleanLaus1 nand *BooleanLaus2* antaa vastauksena

Boolean lausekkeen BooleanList1

nand *BooleanList2* antaa vastauksena *Boolean* listan *BooleanMatriisi1*

nand *BooleanMatriisi2* antaa vastauksena *Boolean* matriisin

Antaa vastauksena loogisen **and** operaation negaation kahdesta argumentista. Antaa vastauksena totuusarvon tosi, epätosi tai yhtälön sievennetyn muodon.

Listojen ja matriisien kohdalla vastauksena on vertailuja elementti elementiltä.

Kokonaisluku1 nand *Kokonaisluku2* ⇒ *kokonaisluku*

Vertaa kahta reaalikokonaislukua bitti bitiltä **nand**-operaation avulla. Sisäisesti kumpikin kokonaisluku muunnetaan etumerkilliseksi, 64 bitin binaariluvuksi. Kun vastaavia bittejä verrataan, tulos on 1, jos kumpikin bitti on 1. Muussa tapauksessa tulos on 0.

$x \geq 3$ 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 $\{3,2,1\}$	$\{-2,-3,-2\}$

Laskettu arvo edustaa bittituloksia ja se näkyy kantaluutilan mukaisesti.

Kokonaisluvut voi syöttää minkä tahansa luvun kantalukena. 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)	$\frac{z!}{c! \cdot (z-c)!}$
Ans	1
nPr(z,c)	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


Laskee listan kombinaatioista kahden listan sisältämiin vastaaviin elementtipareihin perustuen. Argumenttien on oltava samankokoisia listoja.

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

nCr(Matriisi1, Matriisi2)⇒matriisi

Laskee matriisin kombinaatioista kahden matriisin sisältämiin vastaaviin elementtipareihin perustuen. Argumenttien on oltava samankokoisia matriiseja.


nCr($\begin{bmatrix} 6 & 5 \\ 4 & 3 \end{bmatrix}; \begin{bmatrix} 2 & 2 \\ 2 & 2 \end{bmatrix}$)	$\begin{bmatrix} 15 & 10 \\ 6 & 3 \end{bmatrix}$
-----------------------------------------------------------------------------------------------------	--------------------------------------------------

nDerivative()		Katalogi > 
nDerivative (<i>Laus</i> <i>l</i> , <i>Muutt</i> = <i>Arvo</i> [, <i>Aste</i>]) \Rightarrow <i>arvo</i>	$nDerivative(x, x=1)$	1
nDerivative (<i>Laus</i> <i>l</i> , <i>Muutt</i> [, <i>Aste</i>]) <i>Muutt</i> = <i>Arvo</i> \Rightarrow <i>arvo</i>	$nDerivative(x, x) x=0$	undef
	$nDerivative(\sqrt{x-1}, x) x=1$	undef


Laskee numeerisen derivaatan käyttäen automaattisia derivointimenetelmiä.

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


Derivaatan Asteen on oltava **1** tai **2**.

newList()		Katalogi > 
newList (<i>numElementit</i>) \Rightarrow <i>lista</i>	$newList(4)$	{0,0,0,0}

Antaa tuloksena listan, jonka koko on *numElementit*.
Jokainen elementti on nolla.

newMat()		Katalogi > 
newMat (<i>numRivit</i> , <i>numSarakeet</i>) \Rightarrow <i>matriisi</i>	$newMat(2,3)$	$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$

Antaa tuloksena nollamatriisin, jonka koko on *numRivit* ja *numSarakeet*.

nfMax()		Katalogi > 
nfMax (<i>Laus</i> , <i>Muutt</i>) \Rightarrow <i>arvo</i>	$nfMax(-x^2 - 2 \cdot x - 1, x)$	-1.
nfMax (<i>Laus</i> , <i>Muutt</i> , <i>alaraja</i>) \Rightarrow <i>arvo</i>		
nfMax (<i>Laus</i> , <i>Muutt</i> , <i>alaraja</i> , <i>ylaraja</i>) \Rightarrow <i>arvo</i>		
nfMax (<i>Laus</i> , <i>Muutt</i>) <i>alaraja</i> \leq <i>Muutt</i> \leq <i>ylaraja</i> \Rightarrow <i>arvo</i>	$nfMax(0.5 \cdot x^3 - x - 2, x, -5, 5)$	5.

Laskee muuttujan *Muutt* numeerisen ehdotusarvon, jossa lausekkeen *Laus* paikallinen maksimi esiintyy.

Jos ilmoitat *alarajan* ja *ylarajan*, funktio etsii suljetulta väliltä [*alaraja*,*ylaraja*] 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

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()**.

$\text{nfMin}(x^2+2 \cdot x+5, x)$	-1.
$\text{nfMin}(0.5 \cdot x^3 - x - 2, x, -5, 5)$	-5.

nInt()

Katalogi >

nInt(*LausI*, *Muutt*, *Ala*, *Ylä*) \Rightarrow lauseke

Jos integroitava funktio *LausI* ei sisällä muita muuttujia kuin *Muutt*, ja jos *Ala* ja *Ylä* ovat vakioita, positiivinen ∞ tai negatiivinen $-\infty$, tällöin **nInt()** laskee likiarvon lausekkeesta [*LausI*, *Var*, *Ala*, *Ylä*]. Tämä likiarvo on integrandin välillä *Ala* $<$ *Muutt* $<$ *Ylä* olevien joidenkin otosarvojen painotettu keskiarvo.

Tavoitteena on kuusi merkitsevää numeroa. Adaptiivinen algoritmi päättyy, kun näyttää todennäköiseltä, että tavoite on saavutettu, tai kun näyttää epätodennäköiseltä, että lisäotokset tuottaisivat merkittävää parannusta.

Näkyviin tulee viesti (Questionable accuracy (Kyseenalainen tarkkuus)), kun näyttää siltä, että tavoitetta ei ole saavutettu.

Sijoita **nInt()**-komentoa sisäkkäin, jos haluat suorittaa moninkertaisen numeerisen integroinnin. Integroinnin raja-arvot voivat riippua niiden ulkopuolella olevista integrointimuuttujista.

Huomaa: Katso myös **f()**, sivu 201.

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

$\text{nInt}(\cos(x), x, \pi, \pi+1. \text{E}-12)$	-1.04144E-12
$\int_{-\pi}^{\pi+10^{-12}} \cos(x) dx$	$-\sin\left(\frac{1}{1000000000000}\right)$

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

nom()

Katalogi >

nom(*efektiivinenKorko*, *CpY*) \Rightarrow arvo

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

Talouslaskentatoiminto, joka muuntaa efektiivisen vuosikoron *efektiivinenKorko* nimelliskoroksi, kun *CpY* määritetään korkojaksojen lukumääräksi vuodessa.

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

Huomaa: Katso myös **eff()**, sivu 60.

nor (eikä)  **näppäimet**

BooleanLau1 **nor** *BooleanLau2* antaa vastauksena

$x \geq 3$ or $x \geq 4$	$x \geq 3$
--------------------------	------------

Boolean lausekkeen BooleanList1

nor *BooleanList2* antaa vastauksena *Boolean* listan *BooleanMatriisi1*

$x \geq 3$ nor $x \geq 4$	$x < 3$
---------------------------	---------

nor *BooleanMatriisi2* antaa vastauksena *Boolean* matriisin

Antaa vastauksena loogisen **or** operaation negaation kahdesta argumentista. Antaa vastauksena totuusarvon tosi, epätosi tai yhtälön sievennetyn muodon.

Listojen ja matriisien kohdalla vastauksena on vertailuja elementti elementiltä.

Kokonaisluku1 **nor** *Kokonaisluku2* ⇒ *kokonaisluku*

3 or 4	7
--------	---

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.

3 nor 4	-8
---------	----

$\{1,2,3\}$ or $\{3,2,1\}$	$\{3,2,3\}$
----------------------------	-------------

$\{1,2,3\}$ nor $\{3,2,1\}$	$\{-4,-3,-4\}$
-----------------------------	----------------

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).

norm()Katalogi > **norm(Matriisi)**⇒lauseke

$$\text{norm}\begin{pmatrix} a & b \\ c & d \end{pmatrix} \quad \sqrt{a^2+b^2+c^2+d^2}$$

norm(Vektori)⇒lauseke

$$\text{norm}\begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix} \quad \sqrt{30}$$

Laskee Frobeniusin normin.

$$\text{norm}\begin{pmatrix} 1 & 2 \end{pmatrix} \quad \sqrt{5}$$

$$\text{norm}\begin{pmatrix} 1 \\ 2 \end{pmatrix} \quad \sqrt{5}$$

normalLine()Katalogi > **normalLine(LausI,Muutt,Piste)**⇒lauseke

$$\text{normalLine}(x^2,x,1) \quad \frac{3}{2} \cdot \frac{x}{2}$$

normalLine(LausI,Muutt=Piste)⇒lauseke

$$\text{normalLine}((x-3)^2-4,x,3) \quad x=3$$

Määrittää normaalisuoran *LausI*:n kuvaamasta käyrästä pisteessä, joka on määritetty kohtaan *Muutt=Piste*.

$$\text{normalLine}\left(x^{\frac{1}{3}},x=0\right) \quad 0$$

Varmista, että riippumatonta muuttujaa ei ole määritetty. Esimerkiksi, jos $f_1(x)=5$ ja $x:=3$, tällöin

$$\text{normalLine}(\sqrt{|x|},x=0) \quad \text{undef}$$

normalLine(f1(x),x,2) antaa vastauksen "epätosi".**normCdf()**Katalogi > **normCdf(alaraja,yläraja[,μ[,σ]])**⇒luku, jos *yläraja* ja *alaraja* ovat lukuja, *lista*, jos *alaraja* ja *yläraja* ovat listojaLaskee normaalijakauman todennäköisyyden *alarajan* ja *ylärajan* välillä määritetyille μ :lle (oletus=0) ja σ :lle (oletus=1).Kun $P(X \leq \text{yläraja})$, aseta *alaraja* = $-\infty$.**normPdf()**Katalogi > **normPdf(XArvo[,μ,σ])**⇒luku, jos *XArvo* on luku, *lista*, jos *XArvo* on listaLaskee normaalijakauman pistetodennäköisyysfunktion määritetyssä *XArvossa* määritetyille μ :lle ja σ :lle.

nPr()Katalogi > **nPr(Lista1, List2)⇒lista**
$$\text{nPr}\left(\{5,4,3\},\{2,4,2\}\right) \quad \{20,24,6\}$$

Luo listan permutaatioista kahden listan sisältämiin vastaaviin elementtipareihin perustuen. Argumenttien on oltava samankokoisia listoja.

nPr(Matriisi1, Matriisi2)⇒matriisi

$$\text{nPr}\left(\begin{pmatrix} 6 & 5 \\ 4 & 3 \end{pmatrix}, \begin{pmatrix} 2 & 2 \\ 2 & 2 \end{pmatrix}\right) \quad \begin{pmatrix} 30 & 20 \\ 12 & 6 \end{pmatrix}$$

Luo matriisin permutaatioista kahden matriisin sisältämiin vastaaviin elementtipareihin perustuen. Argumenttien on oltava samankokoisia matriiseja.

npv()Katalogi > **npv(Korkoprosentti, CFO, CFLista[, CFFrekv])**

Talouseläntoiminto, joka laskee nettonykyarvon; tulevien ja poistuvien kassavirtojen nykyisten arvojen summan. Jos npv:n tulos on positiivinen, investointi on kannattava.

$$\begin{aligned} \text{list1} &:= \{6000, -8000, 2000, -3000\} \\ &\quad \{6000, -8000, 2000, -3000\} \\ \text{list2} &:= \{2, 2, 2, 1\} \quad \{2, 2, 2, 1\} \\ \text{npv}(10, 5000, \text{list1}, \text{list2}) &\quad 4769.91 \end{aligned}$$

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.

nSolve()Katalogi > **nSolve(Yhtälö, Muutt[=Arvaus])⇒luku tai virhe _merkkijono**
$$\text{nSolve}(x^2 + 5 \cdot x - 25 = 9, x) \quad 3.84429$$
nSolve(Yhtälö, Muutt[=Arvaus], alaraja) ⇒luku tai virhe _merkkijono
$$\text{nSolve}(x^2 = 4, x = -1) \quad -2.$$
$$\text{nSolve}(x^2 = 4, x = 1) \quad 2.$$
nSolve(Yhtälö, Muutt[=Arvaus], alaraja, yläraja) ⇒luku tai virhe _merkkijono

Huomaa: Jos ratkaisuja on useita, voit yrittää löytää tietyn ratkaisun käyttämällä apuna arvausta.

nSolve(*Yhtälö*,*Muutt*[=*Arvaus*]) | *alaraja* ≤ *Muutt*
 ≤ *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 "j" 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 [*1*,]*X*1,[*Frekv*],[*Luokka*,*Sisällytä*]]

OneVar [*n*,]*X*1,*X*2[*X*3[,...],*X*20]]

Laskee yhden muuttujan tilaston enintään 20 listasta. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 169.)

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

X-argumentit ovat datalistoja.

Frekv on valinnainen frekvenssiarvojen lista. Jokainen *Frekv*:*n*

elementti määrittää kunkin vastaavan X :n arvon esiintymisfrekvenssin. Oletusarvo on 1. Kaikkien elementtien on oltava kokonaislukuja 0.

Luokka on numeeristen luokkakoodien lista vastaaville X :n arvoille.

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 tyhjiä elementeistä, katso sivu 227.

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

or (tai)

Katalogi > 

BooleanLaus1 or *BooleanLaus2* antaa vastauksena

$$x \geq 3 \text{ or } x \geq 4$$

$$x \geq 3$$

Boolean lausekkeen BooleanList1

or *BooleanList2* antaa vastauksena *Boolean*

listan BooleanMatriisi1

or *BooleanMatriisi2* antaa vastauksena *Boolean*

matriisiin

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.

Kokonaisluku1orKokonaisluku2=>kokonaisluku

Vertaa kahta reaalikokonaislukua bitti bitiltä or-operaation avulla. Sisäisesti kumpikin kokonaisluku muunnetaan etumerkilliseksi, 64 bitin binaariluvuksi. Kun vastaavia bittejä verrataan, tulos on 1, jos jompikumpi bitti on 1; tulos on 0 vain, jos kumpikin bitti on 0. Laskettu arvo edustaa bittituloksia, ja se näkyy kantelukutilan mukaisesti.

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

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

Jos syötät desimaalikonaisluvun, joka on etumerkillisen, 64 bitin binaarimuodon lukualueen ulkopuolella, laskin käyttää symmetristä modulo-operaatiota, jotta arvo saadaan oikealle alueelle. Lisätietoja, katso **Base2**, sivu 22.

Huomaa: Katso **xor**.

Define $g(x)=$ FuncIf $x \leq 0$ or $x \geq 5$

Goto end

Return $x \cdot 3$

Lbl end

EndFunc

Done

 $g(3)$

9

 $g(0)$

A function did not return a value

Heksadesimaalisessa kantelukutilassa:

0h7AC36 or 0h3D5F

0h7BD7F

Tärkeää: Nolla, ei O-kirjain.

Binaarisessa kantelukutilassa:

0b100101 or 0b100

0b100101

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

Heksadesimaalisessa syötteessä voi olla korkeintaan 16 numeroa.

ord()Katalogi > **ord**(*Merkkijono*) \Rightarrow *kokonaisluku***ord**(*Lista*) \Rightarrow *lista*

Määrittää merkkijonon *Merkkijono* ensimmäisen merkin numerokoodin tai luo listan jokaisen listaelementin ensimmäisistä merkeistä.

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

P**P>Rx()**Katalogi > **P>Rx**(*rLaus*, θ *Laus*) \Rightarrow *lauseke***P>Rx**(*rLista*, θ *Lista*) \Rightarrow *lista***P>Rx**(*rMatriisi*, θ *Matriisi*) \Rightarrow *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 $^{\circ}$, $^{\text{G}}$ tai $^{\text{r}}$.

Huomaa: Voit syöttää tämän funktion tietokoneen näppäimistöltä kirjoittamalla **P@>Rx** (...).

Radianikulmatilassa:

$\text{P}\>\text{R}\text{x}(r, \theta)$	$\cos(\theta) \cdot r$
$\text{P}\>\text{R}\text{x}(4, 60^{\circ})$	2
$\text{P}\>\text{R}\text{x}\left\{\{-3, 10, 1.3\}, \left\{\frac{\pi}{3}, \frac{-\pi}{4}, 0\right\}\right\}$	$\left\{\frac{-3}{2}, 5 \cdot \sqrt{2}, 1.3\right\}$

P>Ry()Katalogi > **P>Ry**(*rLaus*, θ *Laus*) \Rightarrow *lauseke***P>Ry**(*rLista*, θ *Lista*) \Rightarrow *lista***P>Ry**(*rMatriisi*, θ *Matriisi*) \Rightarrow *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 $^{\circ}$, $^{\text{G}}$ tai $^{\text{r}}$.

Huomaa: Voit syöttää tämän funktion tietokoneen näppäimistöltä kirjoittamalla **P@>Ry** (...).

Radianikulmatilassa:

$\text{P}\>\text{R}\text{y}(r, \theta)$	$\sin(\theta) \cdot r$
$\text{P}\>\text{R}\text{y}(4, 60^{\circ})$	$2 \cdot \sqrt{3}$
$\text{P}\>\text{R}\text{y}\left\{\{-3, 10, 1.3\}, \left\{\frac{\pi}{3}, \frac{-\pi}{4}, 0\right\}\right\}$	$\left\{\frac{-3 \cdot \sqrt{3}}{2}, 5 \cdot \sqrt{2}, 0\right\}$

PassErr

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 29, ja **Try**, sivu 183.

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

Esimerkki **PassErr**-komentosta, katso esimerkki 2 **Try**-komennon kohdalla, sivu 183.

piecewise()

piecewise(*Laus1* [, *Ehto1* [, *Laus2* [, *Ehto2* [, ...]]])

Laatii määritelmät paloittain määrittelylle funktiolle listan muodossa. Voit luoda paloittain määriteltyjen funktioiden määrittäksiä myös mallin avulla.

Define $p(x) = \begin{cases} x, & x > 0 \\ \text{undef}, & x \leq 0 \end{cases}$	<i>Done</i>
$p(1)$	1
$p(-1)$	undef

Huomaa: Katso myös **Paloittain määrittelyn funktion malli**, sivu 7.

poissCdf()

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($\lambda, XArvo$) \Rightarrow luku, jos $XArvo$ on luku, lista, jos $XArvo$ on lista

Laskee todennäköisyyden diskreetille Poissonin jakaumalle, jolla on määritetty keskiarvo λ .

►Polar (►Polaarinen)

Vektori ►Polar

$[1 \ 3.]$ ►Polar [3.16228 \angle 1.24905]

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

$[x \ y.]$ ►Polar
$$\left[\sqrt{x^2+y^2} \quad \angle \frac{\pi \cdot \text{sign}(y)}{2} - \tan^{-1}\left(\frac{x}{y}\right) \right]$$

Näyttää *vektorin* polaarisisä muodossa $[r \ \angle \theta]$.

Vektorin on oltava kooltaan 2, ja se voi olla rivi tai sarake.

Huomaa: ►Polar on näyttömuodon ohje, ei muunnosfunktio. Voit käyttää komentoa ainoastaan syötearvojen lopussa, eikä se päivitä *ans*:n arvoa.

Huomaa: Katso myös ►Rect, sivu 140.

kompleksi.Arvo ►Polaarinen

Radiaanikulmatilassa:

Näyttää *kompleksiVektorin* polaarisisä muodossa.

$(3+4 \cdot i)$ ►Polar
$$e^{i \cdot \left(\frac{\pi}{2} - \tan^{-1}\left(\frac{3}{4}\right) \right)} \cdot 5$$

- Astekulmatilassa vastauksena on $(r \ \angle \theta)$.
- Radiaanikulmatilassa vastauksena on $re^{i\theta}$.

*kompleksiArvo*lla voi olla mikä tahansa kompleksilukumuoto. Syöte $re^{i\theta}$ aiheuttaa kuitenkin virheen astekulmatilassa.

$\left(\left(4 \ \angle \frac{\pi}{3} \right) \right)$ ►Polar
$$e^{\frac{i \cdot \pi}{3}} \cdot 4$$

Huomaa: Polaarisisä syötteessä $(r \ \angle \theta)$ on käytettävä sulkeita.

Graadikulmatilassa:

$(4 \cdot i)$ ►Polar $(4 \ \angle \ 100)$

Astekulmatilassa:

$(3+4 \cdot i)$ ►Polar
$$\left(5 \ \angle \ 90 - \tan^{-1}\left(\frac{3}{4}\right) \right)$$

polyCoeffs()Katalogi > **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.

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

$$\text{polyCoeffs}((x-1)^2 \cdot (x+2)^3) \quad \{1, 4, 1, -10, -4, 8\}$$

Laventaa polynomien ja valitsee *x*:n poisjätetyille muuttujalle *Muutt*.

$$\text{polyCoeffs}((x+y+z)^2, x) \quad \{1, 2 \cdot (y+z), (y+z)^2\}$$

$$\text{polyCoeffs}((x+y+z)^2, y) \quad \{1, 2 \cdot (x+z), (x+z)^2\}$$

$$\text{polyCoeffs}((x+y+z)^2, z) \quad \{1, 2 \cdot (x+y), (x+y)^2\}$$

polyDegree()Katalogi > **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.

$$\text{polyDegree}(5) \quad 0$$

$$\text{polyDegree}(\ln(2) + \pi, x) \quad 0$$

Vakiopolynomit

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

$$\text{polyDegree}((x-1)^2 \cdot (x+2)^3) \quad 5$$

$$\text{polyDegree}((x+y^2+z^3)^2, x) \quad 2$$

$$\text{polyDegree}((x+y^2+z^3)^2, y) \quad 4$$

polyDegree()Katalogi > 

$\text{polyDegree}((x-1)^{10000}, x)$	10000
---------------------------------------	-------

Asteesta voidaan ottaa juuri, vaikka kertoimista ei voida. Tämä johtuu siitä, että asteesta voidaan ottaa juuri laventamatta polynomia.

polyEval()Katalogi > **polyEval(Lista1, Laus1)⇒lauseke****polyEval(Lista1, Lista2)⇒lauseke**

Tulkitsee ensimmäisen argumentin laskeva-asteisen polynomin kertoimeksi ja antaa vastauksena polynomin, josta on laskettu toisen argumentin arvo.

$\text{polyEval}\{a, b, c\}, x$	$a \cdot x^2 + b \cdot x + c$
$\text{polyEval}\{1, 2, 3, 4\}, 2$	26
$\text{polyEval}\{1, 2, 3, 4\}, \{2, -7\}$	$\{26, -262\}$

polyGcd()Katalogi > **polyGcd(Laus1, Laus2)⇒lauseke**

Laskee kahden argumentin suurimman yhteisen jakajan.

Laus1:n ja *Laus2*:n on oltava polynomilausekkeita.

Lista-, matriisi- ja Boolean argumentteja ei sallita.

$\text{polyGcd}(100, 30)$	10
$\text{polyGcd}(x^2-1, x-1)$	$x-1$
$\text{polyGcd}(x^3-6 \cdot x^2+11 \cdot x-6, x^2-6 \cdot x+8)$	$x-2$

polyQuotient()Katalogi > **polyQuotient(Poly1, Poly2 [, Muutt1])⇒lauseke**

Laskee polynomin *Poly1* osamäärän jaettuna 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.

$\text{polyQuotient}(x-1, x-3)$	1
$\text{polyQuotient}(x-1, x^2-1)$	0
$\text{polyQuotient}(x^2-1, x-1)$	$x+1$
$\text{polyQuotient}(x^3-6 \cdot x^2+11 \cdot x-6, x^2-6 \cdot x+8)$	x

polyQuotient()

Katalogi > 

$\text{polyQuotient}((x-y)\cdot(y-z), x+y+z, x)$	$y-z$
$\text{polyQuotient}((x-y)\cdot(y-z), x+y+z, y)$	$2\cdot x-y+2\cdot z$
$\text{polyQuotient}((x-y)\cdot(y-z), x+y+z, z)$	$-(x-y)$

polyRemainder()

Katalogi > **polyRemainder**(Poly1, Poly2 [, Muutt]) ⇒ lauseke

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.

$\text{polyRemainder}(x-1, x-3)$	2
$\text{polyRemainder}(x-1, x^2-1)$	$x-1$
$\text{polyRemainder}(x^2-1, x-1)$	0
$\text{polyRemainder}((x-y)\cdot(y-z), x+y+z, x)$	$-(y-z)\cdot(2\cdot y+z)$
$\text{polyRemainder}((x-y)\cdot(y-z), x+y+z, y)$	$-2\cdot x^2-5\cdot x\cdot z-2\cdot z^2$
$\text{polyRemainder}((x-y)\cdot(y-z), x+y+z, z)$	$(x-y)\cdot(x+2\cdot y)$

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

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

Kertoinlistaan.

Huomaa: Katso myös **cPolyRoots()**, sivu 40.

PowerReg

PowerReg $X, Y [, Frekv] [, Luokka, Sisällyttä]$

Laskee potenssiregression $y = (a \cdot (x)^b)$ listoista X ja Y frekvenssillä $Frekv$. Tulosten yhteenvedo tallentuu *stat.results*-muuttujaan. (Katso sivu 169.)

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 elementitsivulla sivu 227.

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	Muokatus Y Lista:n sisältämä datapisteiden lista, jota käytetään regressiossa komentojen <i>Frekv</i> , <i>Luokkalista</i> ja <i>Sisällyttä</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>Sisällyttä</i> luokat rajoitusten mukaisesti
stat.FreqReg	Komentoja <i>stat.XReg</i> ja <i>stat.YReg</i> vastaava frekvenssilista

Prgm

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 määritysten syöttämisestä löytyvät tuotteen ohjekirjan Laskin-osiosta.

Laske GCD ja näytä välitulokset.

```
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 214.

Product (PI)

Katso $\Pi()$, sivu 214.

product()

product(Lista[, Alku[, loppu]])⇒lauseke

Laskee *Listan* sisältämien elementtien tulon. *Alku* ja *Loppu* ovat valinnaisia. Ne määrittävät elementtien alueen.

product({1,2,3,4})	24
product({2,x,y})	2·x·y
product({4,5,8,9},2,3)	40

product()Katalogi > **product(MatriisiI[, Alku[, loppu]])**⇒matriisi

Laskee rivivektorin, joka sisältää *MatriisiI*:n sarakkeiden elementtien tulot. *Alku* ja *Loppu* ovat valinnaisia. Ne määrittävät rivialueen.

Tyhjiä elementtejä ei huomioida. Lisätietoja tyhjiistä elementeistä, katso sivu 227.

$$\text{product} \left(\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix} \right) \quad [28 \ 80 \ 162]$$

$$\text{product} \left(\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix}, 1, 2 \right) \quad [4 \ 10 \ 18]$$

propFrac()Katalogi > **propFrac(LausI[, Muutt])**⇒lauseke

propFrac(rationaali_luku) laskee *rationaali_luvun* kokonaisluvun summana ja murtolukuna, jolla on sama etumerkki ja suurempi nimittäjä kuin osoittaja.

propFrac(rationaali_lauseke, Muutt) laskee aitojen suhdelukujen summan ja polynomin muuttujan *Muutt* suhteen. Muuttujan *Muutt* aste nimittäjässä on suurempi kuin muuttujan *Muutt* aste osoittajassa kussakin aidossa suhdeluvussa. Muuttujan *Muutt* samanlaiset potenssit kerätään. Termit ja niiden tekijät lajitellaan siten, että *Muutt* on päämuuttuja.

Jos *Muutt* jätetään pois, varsinaisen murtoluvun lavennus tehdään pääasiallisimmalla muuttujalla. Polynomiosan kertoimet tehdään sen jälkeen aidoiksi ensin niiden pääasiallisimman muuttujan suhteen ja niin edelleen.

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

propFrac()-funktion avulla voit esittää sekalukuja ja suorittaa sekalukujen yhteen- ja vähennyslaskua.

$$\text{propFrac} \left(\frac{4}{3} \right) \quad 1 + \frac{1}{3}$$

$$\text{propFrac} \left(-\frac{4}{3} \right) \quad -1 - \frac{1}{3}$$

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

$$\text{propFrac}(\text{Ans}) \quad \frac{1}{x + 1} + x + \frac{1}{y + 1} + y$$

$$\text{propFrac} \left(\frac{11}{7} \right) \quad 1 + \frac{4}{7}$$

$$\text{propFrac} \left(3 + \frac{1}{11} + 5 + \frac{3}{4} \right) \quad 8 + \frac{37}{44}$$

$$\text{propFrac} \left(3 + \frac{1}{11} - \left(5 + \frac{3}{4} \right) \right) \quad -2 - \frac{29}{44}$$

Q

QR *Matriisi, qMatNimi, rMatNimi[, Tol]*

Suorittaa Householderin QR-dekomponoinnin reaali- tai kompleksilukumatriisista. Tuloksena olevat Q- ja R-matriisit tallentuvat määritettyihin *MatNimi*-muuttujiin. Q-matriisi on jakamaton. R-matriisi on yläkolmiomatriisi.

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

- Jos käytät painikkeita **ctrl** **enter** tai **Automaattinen tai likimääräinen** -tilan valintaa Approximate (Likimääräinen), laskut suoritetaan liukulukuaritmetiikalla.
- Jos *Tol* jätetään pois tai sitä ei käytetä, oletusarvoinen toleranssi lasketaan seuraavasti:
 $5E-14 \cdot \max(\dim(\text{Matriisi})) \cdot \text{rowNorm}(\text{Matriisi})$

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, Sisällyttä]*

Laskee 2. asteen polynomiregressioy = a · x² + b · x + c listaista *X* ja *Y* frekvenssillä *Frekv*. Tulosten yhteenveto tallentuu

stat.results-muuttujaan. (Katso sivu 169.)

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 elementitsivulla sivu 227.

Tulosmuuttuja	Kuvaus
stat.RegEqn	Regressioyhtälö: $a \cdot x^2 + b \cdot x + c$
stat.a, stat.b, stat.c	Regressiokertoimet
stat.R ²	Määrittyskerroin
stat.Resid	Regressioyhtälön jäännökset
stat.XReg	Muokatus <i>Y</i> Lista:n sisältämä datapisteiden lista, jota käytetään regressiossa komentojen <i>Frekv</i> , <i>Luokkalista</i> ja <i>Sisällytä</i> huokat rajoitusten mukaisesti
stat.YReg	Muokatus <i>Y</i> Lista:n sisältämä datapisteiden lista, jota käytetään regressiossa komentojen <i>Frekv</i> , <i>Luokkalista</i> ja <i>Sisällytä</i> huokat rajoitusten mukaisesti
stat.FreqReg	Komentoja <i>stat.XReg</i> ja <i>stat.YReg</i> vastaava frekvenssilista

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

Laskee 4. asteen polynomiregression

$y = a \cdot x^4 + b \cdot x^3 + c \cdot x^2 + d \cdot x + e$ listaista *X* ja *Y* frekvenssillä *Frekv*.

Tulosten yhteenveto tallentuu *stat.results*-muuttujaan.

(Katso sivu sivu 169.)

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.

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

Lisätietoja listassa olevien tyhjiä elementtien vaikutuksesta, katso Tyhjät elementtisivulla sivu 227.

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	Määrittyskerroin
stat.Resid	Regressioyhtälön jäännökset
stat.XReg	Muokatuun Y Listan sisältämä datapisteiden lista, jota käytetään regressiossa komentojen <i>Frekv</i> , <i>Luokkalista</i> ja <i>Sisällytyt luokat rajoitusten mukaisesti</i>
stat.YReg	Muokatuun X Listan sisältämä datapisteiden lista, jota käytetään regressiossa komentojen <i>Frekv</i> , <i>Luokkalista</i> ja <i>Sisällytyt luokat rajoitusten mukaisesti</i>
stat.FreqReg	Komentoja <i>stat.XReg</i> ja <i>stat.YReg</i> vastaava frekvenssilista

R

R►Pθ()

R►Pθ (x Laus, y Laus) ⇒ lauseke

Astekulmatilassa:

R►Pθ (x Listan, y Listan) ⇒ lista

R►Pθ(x, y) $90 \cdot \text{sign}(y) - \tan^{-1}\left(\frac{x}{y}\right)$

R►Pθ (x Matriisi, y Matriisi) ⇒ matriisi

Määrittää vastaavan θ -koordinaatin (x, y)-parin argumenteille.

Graadikulmatilassa:

R►P0()Katalogi > 

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

Huomaa: Voit syöttää tämän funktion tietokoneen näppäimistöltä kirjoittamalla **R@>Ptheta (...)** .

$$\text{R►P0}(x,y) \quad 100 \cdot \text{sign}(y) - \tan^{-1}\left(\frac{x}{y}\right)$$

Radiaanikulmatilassa:

$$\text{R►P0}(3,2) \quad \tan^{-1}\left(\frac{2}{3}\right)$$

$$\text{R►P0}\left([3 \ -4 \ 2], \left[0 \ \frac{\pi}{4} \ 1.5\right]\right) \\ \left[0 \ \tan^{-1}\left(\frac{16}{\pi}\right) + \frac{\pi}{2} \ 0.643501\right]$$

R►Pr()Katalogi > 

R►Pr (*xLaus*, *yLaus*) ⇒ *lauseke*

R►Pr (*xLista*, *yLista*) ⇒ *lista*

R►Pr (*xMatriisi*, *yMatriisi*) ⇒ *matriisi*

Määrittää (*x*,*y*)-parin argumenttien vastaavan r-koordinaatin.

Huomaa: Voit syöttää tämän funktion tietokoneen näppäimistöltä kirjoittamalla **R@>Pr (...)** .

Radiaanikulmatilassa:

$$\text{R►Pr}(3,2) \quad \sqrt{13}$$

$$\text{R►Pr}(x,y) \quad \sqrt{x^2+y^2}$$

$$\text{R►Pr}\left([3 \ -4 \ 2], \left[0 \ \frac{\pi}{4} \ 1.5\right]\right) \\ \left[3 \ \frac{\sqrt{\pi^2+256}}{4} \ 2.5\right]$$

►RadKatalogi > 

Laus ► **Rad** ⇒ *lauseke*

Muuntaa argumentin radiaanikulmayksikköön.

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

Astekulmatilassa:

$$(1.5) \text{►Rad} \quad (0.02618)^r$$

Graadikulmatilassa:

$$(1.5) \text{►Rad} \quad (0.023562)^r$$

rand()

[Katalogi >](#) 

rand()⇒*lauseke*

rand(Kokeiden lkm)⇒*lista*

rand() määrittää satunnaisluvun väliltä 0 ja 1.

rand(Kokeiden lkm) määrittää listan, joka sisältää *Kokeiden lkm* satunnaislukua väliltä 0 ja 1.

RandSeed 1147	<i>Done</i>
rand(2)	{0.158206,0.717917}

randBin()

[Katalogi >](#) 

randBin(*n, p*)⇒*lauseke*

randBin(*n, p, Kokeiden lkm*)⇒*lista*

randBin(*n, p*) määrittää satunnaisreaaliluvun määritetystä binomijakaumasta.

randBin(*n, p, Kokeiden lkm*) luo listan, joka sisältää *Kokeiden lkm* satunnaisreaalilukua määritetystä binomijakaumasta.

randBin(80,0.5)	42
randBin(80,0.5,3)	{41,32,39}

randInt()

[Katalogi >](#) 

randInt(alaraja,yläraja)⇒*lauseke*

randInt(alaraja,yläraja,Kokeiden lkm)⇒*lista*

randInt(alaraja,yläraja) määrittää satunnaiskokonaisluvun, joka on kokonaislukujen *alarajan* ja *ylärajan* välissä.

randInt(alaraja,yläraja,Kokeiden lkm) luo listan, joka sisältää *Kokeiden lkm* satunnaiskokonaislukua määritetyltä alueelta.

randInt(3,10)	5
randInt(3,10,4)	{9,7,5,8}

randMat()

[Katalogi >](#) 

randMat(numRivit, numSarakeet)⇒*matriisi*

Laskee määrätynkokoisen kokonaislukumatriisin väliltä -9 ja 9.

Kummankin argumentin tulee sieventyä kokonaisluvuksi.

RandSeed 1147	<i>Done</i>									
randMat(3,3)	<table border="1"><tr><td>8</td><td>-3</td><td>6</td></tr><tr><td>-2</td><td>3</td><td>-6</td></tr><tr><td>0</td><td>4</td><td>-6</td></tr></table>	8	-3	6	-2	3	-6	0	4	-6
8	-3	6								
-2	3	-6								
0	4	-6								

Huomaa: Tämän matriisin arvot muuttuvat aina, kun

randMat()

Katalogi >

painat **enter**.**randNorm()**

Katalogi >

randNorm(μ, σ) \Rightarrow lauseke**randNorm**($\mu, \sigma, kokeidenLkm$) \Rightarrow lista

Määrittää desimaaliluvun määritetystä normaali jakaumasta. Luku voi olla mikä tahansa reaali luku, mutta se keskittyy voimakkaasti välille $[\mu - 3 \cdot \sigma, \mu + 3 \cdot \sigma]$.

randNorm($\mu, \sigma, kokeidenLkm$) laskee listan, joka sisältää *kokeidenLkm* desimaalilukua määritetystä normaali jakaumasta.

RandSeed 1147	Done
randNorm(0,1)	0.492541
randNorm(3,4.5)	-3.54356

randPoly()

Katalogi >

randPoly(*Muutt*, *Aste*) \Rightarrow lauseke

Määrittää polynomin muuttujasta *Muutt*, joka on määritettyä *Astetta*. Kertoimet ovat satunnaisia kokonaislukuja väliltä -9 ja 9. Alussa oleva kertoin ei ole nolla.

Asteen on oltava 0-99.

RandSeed 1147	Done
randPoly(x,5)	$-2 \cdot x^5 + 3 \cdot x^4 - 6 \cdot x^3 + 4 \cdot x - 6$

randSamp()

Katalogi >

randSamp(*List*, *Kokeiden lkm*, *eiKorv*) \Rightarrow lista

Laskee listan, joka sisältää satunnaisotoksen *Kokeiden lkm* kokeesta, jotka on otettu *List*asta, ja valinnaisena on otoksen korvaaminen (*eiKorv*=0) tai ei otoksen korvaamista (*eiKorv*=1). Oletusarvona on otoksen korvaaminen.

Define list3={1,2,3,4,5}	Done
Define list4=randSamp(list3,6)	Done
list4	{2,3,4,3,1,2}

RandSeed *Luku*

Jos *Luku* = 0, asettaa siemenluvut satunnaislukugeneraattorin tehdasasetuksiin. Jos *Luku* ≠ 0, sillä luodaan kaksi siemenlukua, jotka tallentuvat järjestelmän muuttujiin seed1 ja seed2.

RandSeed 1147

Done

rand()

0.158206

real() (reaali)**real**(*Lausl*) ⇒ *lauseke*

real(2+3·i)

2

Määrittää argumentin reaali- osan.

real(z)

z

Huomaa: Kaikkia määrittämättömiä muuttujia käsitellään reaali- muuttujina. Katso myös **imag()**, sivu 86.

real(x+i·y)

x

real(*Listal*) ⇒ *lista*

real({a+i·b,3,i})

{a,3,0}

Määrittää kaikkien elementtien reaali- osat.

real(*Matriisil*) ⇒ *matriisi*real($\begin{pmatrix} a+i·b & 3 \\ c & i \end{pmatrix}$) $\begin{bmatrix} a & 3 \\ c & 0 \end{bmatrix}$

Määrittää kaikkien elementtien reaali- osat.

▶ **Rect***Vektori* ▶ **Rect**

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

Näyttää *Vektorin* suorakulmam muodossa [x, y, z].

Vektorin koon on oltava 2 tai 3, ja se voi olla rivi tai sarake.

 $\left(\begin{matrix} 3 & \angle \frac{\pi}{4} & \angle \frac{\pi}{6} \end{matrix} \right)$ ▶ **Rect** $\begin{bmatrix} 3\sqrt{2} & 3\sqrt{2} & 3\sqrt{3} \\ 4 & 4 & 2 \end{bmatrix}$ $\begin{bmatrix} a & \angle b & \angle c \end{bmatrix}$ $\begin{bmatrix} a \cdot \cos(b) \cdot \sin(c) & a \cdot \sin(b) \cdot \sin(c) & a \cdot \cos(c) \end{bmatrix}$

Huomaa: ▶**Rect** on näyttömuodon ohje, ei muunnosfunktio. Voit käyttää komentoa ainoastaan syöterivin lopussa, eikä se päivitä *ans*:n arvoa.

Huomaa: Katso myös ▶**Polar**, sivu 127.

kompleksiArvo ▶ **Rect**

Näyttää *kompleksiArvon* suorakulmam muodossa a+bi. *kompleksiArvon* muoto voi olla mikä tahansa kompleksilukumuoto. Syöte $re^{i\theta}$ aiheuttaa kuitenkin virheen astekulmatilassa.

Radiaanikulmatilassa:

Huomaa: Polaarisessa syötteessä ($r \angle \theta$) on käytettävä sulkeita.

$\left(4 \cdot e^{\frac{\pi}{3}}\right)$ ►Rect	$4 \cdot e^{\frac{\pi}{3}}$
$\left(\left(4 \angle \frac{\pi}{3}\right)\right)$ ►Rect	$2+2 \cdot \sqrt{3} \cdot i$

Graadikulmatilassa:

$\left(\left(1 \angle 100\right)\right)$ ►Rect	i
------------------------------------------------	-----

Astekulmatilassa:

$\left(\left(4 \angle 60\right)\right)$ ►Rect	$2+2 \cdot \sqrt{3} \cdot i$
-----------------------------------------------	------------------------------

Huomaa: Voit syöttää merkin \angle valitsemalla sen Katalogin symboliiluettelosta.

ref()

ref(MatriisiI[, Tol])⇒matriisi

Määrittää *MatriisiI*:n rivi-echelon-muodon.

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 **ctrl** **enter** tai **Automaattinen tai likimääräinen** -tilan valintaa Approximate (Likimääräinen), laskut suoritetaan liukulukuaritmetiikalla.
- Jos *Tol* jätetään pois tai sitä ei käytetä, oletusarvoinen toleranssi lasketaan seuraavasti:
 $5E-14 \cdot \max(\dim(\text{MatriisiI})) \cdot \text{rowNorm}(\text{MatriisiI})$

Vältä määrittämättömiä elementtejä matriisissa *MatriisiI*. Tuloksena voi olla odottamattomia vastauksia.

Jos esimerkiksi *a* on määrittämätön seuraavassa

ref($\begin{pmatrix} -2 & -2 & 0 & -6 \\ 1 & -1 & 9 & -9 \\ -5 & 2 & 4 & -4 \end{pmatrix}$)	$\begin{bmatrix} 1 & \frac{-2}{5} & \frac{-4}{5} & \frac{4}{5} \\ 0 & 1 & \frac{4}{7} & \frac{11}{7} \\ 0 & 0 & 1 & \frac{-62}{71} \end{bmatrix}$
-----------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------

$\begin{bmatrix} a & b \\ c & d \end{bmatrix} \rightarrow mI$	$\begin{bmatrix} a & b \\ c & d \end{bmatrix}$
ref(<i>mI</i>)	$\begin{bmatrix} 1 & \frac{d}{c} \\ 0 & 1 \end{bmatrix}$

lausekkeessa, näkyviin tulee varoitusviesti, ja vastaus näkyy seuraavasti:

$$\text{ref} \left(\begin{bmatrix} a & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \right) \quad \begin{bmatrix} 1 & \frac{1}{a} & 0 \\ & a & \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Varoitus näkyy, koska yleistetty elementi $1/a$ ei olisi mahdollinen yhtälölle $a=0$.

Voit välttää tämän tallentamalla etukäteen arvon a :lle tai käyttämällä rajoittavaa ("|")-operaattoria arvon korvaamiseksi, kuten on esitetty seuraavassa esimerkissä.

$$\text{ref} \left(\begin{bmatrix} a & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \mid a=0 \right) \quad \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}$$

Huomaa: Katso myös **rref()**, sivu 149.

remain(Laus1, Laus2) ⇒ lauseke

remain(Lista1, Lista2) ⇒ lista

remain(Matriisi1, Matriisi2) ⇒ matriisi

Laskee ensimmäisen argumentin jäännöksen toisen argumentin suhteen seuraavien identtisten yhtälöiden määrittelyn mukaisesti:

$\text{remain}(x,0)$ x

$\text{remain}(x,y)$ $x-y \cdot \text{iPart}(x/y)$

Huomaa, että seurauksena **remain(-x,y)** = **-remain(x,y)**. Vastaus on joko nolla tai samanmerkkinen kuin ensimmäinen argumentti.

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

$\text{remain}(7,0)$	7
$\text{remain}(7,3)$	1
$\text{remain}(-7,3)$	-1
$\text{remain}(7,-3)$	1
$\text{remain}(-7,-3)$	-1
$\text{remain}(\{12,-14,16\},\{9,7,-5\})$	$\{3,0,1\}$

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

Request *kehotemerkkijono, var[, DispFlag [, statusVar]]*

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

Ohjelmointikomento: Keskeyttää ohjelman ja näyttää valintaruudun, jossa on viesti *kehotemerkkijono* sekä syöttöruutu käyttäjän antamaa vastausta varten.

Kun käyttäjä kirjoittaa vastauksen ja napsauttaa **OK**-painiketta, syöttöruudun sisältö määritetään muuttujaan *muutt*.

Jos käyttäjä napsauttaa **Peruuta**-painiketta, ohjelma etenee hyväksymättä mitään syötteitä. Ohjelma käyttää muuttujan *muutt* aikaisempaa arvoa, jos *muutt* on jo määritetty.

Valinnainen *NäytäLippu*-argumentti voi olla mikä tahansa lauseke.

- Jos *NäytäLippu* jätetään pois, tai jos se sievennyy arvoksi **1**, kehoteviesti ja käyttäjän vastaus näkyvät laskimen historiassa.
- Jos *NäytäLippu* sievennyy arvoon **0**, kehoite ja vastaus eivät näy historiassa.

Valinnainen *statusVar* argumentti antaa ohjelmalle tavan määritellä, kuinka käyttäjä on sulkenut valintaikkunan. Huomaa, että *statusVar* vaatii *DispFlag*-argumentin.

- Jos käyttäjä on napsauttanut **OK** tai painanut **Enter** tai **Ctrl+Enter**, muuttuja *statusVar* asetetaan arvoon **1**.
- Muuten muuttuja *statusVar* asetetaan arvoon **0**.

Funk()-argumentin avulla ohjelma voi tallentaa käyttäjän vastauksen funktion määritelmäksi. Tämä syntaksi toimii ikään kuin käyttäjä suorittaisi komennon:

```
Define funk(arg1, ...argn) = käyttäjän vastaus
```

Sen jälkeen ohjelma voi käyttää määritettyä funktiota *funk()*. Komennon *kehotemerkkijono* pitäisi opastaa käyttäjää syöttämään sopiva *käyttäjän vastaus*, joka määrittää loppuun funktion määritelmän.

Määritä ohjelma:

```
Define request_demo()=Prgm
```

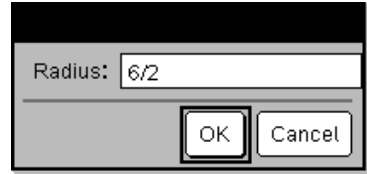
```
Request "Säde: ",r
```

```
Disp "Ala = ",pi*r^2
```

```
EndPrgm
```

Suorita ohjelma ja kirjoita vastaus:

```
request_demo()
```



Vastaus **OK**-näppäimen painamisen jälkeen:

Säde: 6/2

Ala = 28.2743

Määritä ohjelma:

```
Define polynomial()=Prgm
```

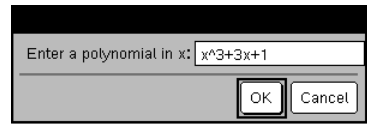
```
Request "Syötä polynomi x:ään:",p(x)
```

```
Disp "Reaalilukujuret ovat:",polyRoots(p(x),x)
```

```
EndPrgm
```

Suorita ohjelma ja kirjoita vastaus:

```
polynomi()
```





Vastaus **OK**-näppäimen painamisen jälkeen:

Syötä polynomi x:ään: x^3+3x+1

Huomaa: **Request** -komentoa voi käyttää käyttäjän määrittämän ohjelman sisällä mutta ei funktion sisällä.

Reaalilukujuuret ovat: {-0.322185}

Pysäytä ohjelma, joka sisältää **Request** komennon 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**, sivu 144.

RequestStr*kehote*merkkijono, *muutt[, NäytäLippu]*

Määritä ohjelma:

Ohjelmointikomento: Toimii samalla tavalla kuin **Request**-komennon ensimmäinen syntaksi paitsi, että käyttäjän vastaus tulkitaan aina merkkijonoksi. **Request**-komento kuitenkin tulkitsee vastauksen lausekkeeksi, ellei käyttäjä merkitse sitä lainausmerkkien ("") sisään.

```
Define requestStr_demo()=Prgm
```

```
RequestStr "Oma nimesi: ",nimi,0
```

```
Disp "Vastauksessa on ",dim(nimi)," merkkiä."
```



```
EndPrgm
```

Huomaa: **RequestStr**-komentoa voi käyttää käyttäjän määrittämän ohjelman sisällä mutta ei funktion sisällä.

Suorita ohjelma ja kirjoita vastaus:

```
requestStr_demo()
```

Pysäytä ohjelma, joka sisältää **RequestStr** komennon 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.



Vastaus **OK**-näppäimen painamisen jälkeen (Huomaa, että jos *NäytäLippu*-argumentti on **0**, kehote ja vastaus eivät näy historiassa):

```
requestStr_demo()
```

Vastauksessa on 5 merkkiä.

Huomaa: Katso myös **Request**, sivu 143.

Return [*Laus*]

Antaa lausekkeen *Laus* funktion tuloksena.
Käytetään lohkon **Func...EndFunc** sisällä.

Huomaa: Voit poistua ohjelmasta käyttämällä **Return**-komentoa ilman argumenttia lohkon **Prgm...EndPrgm** sisällä.

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

```
Define factorial (n)=
Func
Local answer,counter
1 → answer
For counter,1,n
answer·counter → answer
EndFor
Return answer
EndFunc
```

factorial (3)

6

right()

right(*List1* [, *Num*])⇒*lista*

Määrittää oikeanpuoleisimmat *Num*-elementit, jotka sisältyvät *List1*:een.

Jos jätät pois komennon *Num*, määrittää kaiken listasta *List1*.

right(*lähdemerkkijono* [, *Num*])⇒*merkkijono*

Määrittää oikeanpuoleisimmat *Num*-merkit, jotka sisältyvät merkkijonoon *lähdemerkkijono*.

Jos jätät pois komennon *Num*, määrittää kaiken merkkijonosta *lähdemerkkijono*.

right(*Vertailu*)⇒*lauseke*

Laskee yhtälön tai epäyhtälön oikean puolen.

right{ {1,3,-2,4},3} {3,-2,4}

right("Hello",2) "lo"

right($x < 3$) 3

rk23()

rk23(*ExprVardepVar*, {*Var0VarMax*},
depVar0VarStep [,
diftoI])⇒*matriisi*

rk23(*SystemOfExpr*, *Var*, *ListOfDepVars*, {*Var0*,
VarMax}, *ListOfDepVars0*, *VarStep* [, *diftoI*])
⇒*matriisi*

rk23(*ListOfExpr*, *Var*, *ListOfDepVars*, {*Var0*,

Differentiaaliyhtälö:

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

rk23($0.001 \cdot y \cdot (100 - y)$, t, y , {0,100}, 10, 1)

0.	1.	2.	3.	4.
10.	10.9367	11.9493	13.042	14.2

Jos haluat nähdä koko vastauksen, paina ▲ ja siirrä

$VarMax$), $ListOfDepVars0$, $VarStep$ [, $difIol$])
 \Rightarrow matriisi

Käyttää Runge-Kutta -menetelmää ryhmän ratkaisuun

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

muuttujalla $\text{depVar}(\text{Var}0)=\text{depVar}0$ välillä $[\text{Var}0, \text{VarMax}]$. Antaa tulokseksi matriisin, jonka ensimmäinen rivi määrittelee Var tulosarvot muuttujan $VarStep$ määrittelemällä tavalla. Toinen rivi määrittää ensimmäisen ratkaisukomponentin arvon vastaavissa Var -arvoissa, ja niin edelleen.

$Expr$ 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.

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

$ListOfDepVars0$ on riippuvien muuttujien alkuehtojen luettelo.

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

jos $VarStep$ sievenee nolaksi, ratkaisut lasketaan "Runge-Kutta" muuttujan Var arvoilla.

$difIol$ on virhetoleranssi (oletuksena 0.001).

sen jälkeen osoitinta painikkeilla \leftarrow ja \rightarrow .

Sama yhtälö, jossa $difIol$ on asetettu arvoon 1.E-6

$$\text{rk23}\left(0.001 \cdot y \cdot (100-y), t, y, \{0, 100\}, 10, 1, 1.E-6\right)$$

0.	1.	2.	3.	4.
10.	10.9367	11.9495	13.0423	14.2189

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

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

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

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

$$\{10., 10.9367, 11.9494, 13.0423, 14.2189, 15.4\}$$

$$\text{Yhtälöryhmä: } \begin{cases} yI' = -yI + 0.1 \cdot yI \cdot y2 \\ y2' = 3 \cdot y2 - yI \cdot y2 \end{cases} \text{ kun}$$

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

$$\text{rk23}\left(\begin{cases} -yI + 0.1 \cdot yI \cdot y2 \\ 3 \cdot y2 - yI \cdot y2 \end{cases}, t, \{yI, y2\}, \{0, 5\}, \{2, 5\}, 1\right)$$

0.	1.	2.	3.	4.
2.	1.94103	4.78694	3.25253	1.82848
5.	16.8311	12.3133	3.51112	6.27245

rotate()Katalogi > 

kierto tapahtuu oikealle. Oletusarvo on -1 (kierrä oikealle yksi elementti).

rotate(*MerkkijonoI* [, *KiertojenLkm*]) ⇒ *merkkijono*

Luo *MerkkijonoI*:n kopion, jota on kierretty oikealle tai vasemmalle *KiertojenLkm*:n merkkien määrittämän määrän. Ei muuta *MerkkijonoI*:ä.

Jos *KiertojenLkm* on positiivinen, kierto tapahtuu vasemmalle. Jos *KiertojenLkm* on negatiivinen, kierto tapahtuu oikealle. Oletusarvo on -1 (kierrä oikealle yksi merkki).

rotate("abcd")	"dabc"
rotate("abcd", -2)	"cdab"
rotate("abcd", 1)	"bcda"

round()Katalogi > 

round(*LausI* [, *numerot*]) ⇒ *lauseke*

Pyöristää argumentin jättäen desimaalipisteen jälkeen määritetyn määrän numeroita.

numerot-ohjeen on oltava kokonaisluku väliltä 0-12.

Jos *numerot*-ohjetta ei oteta mukaan, pyöristää argumentin 12 merkittävään numeroon tarkkuudelle.

Huomaa: Näytettävät numerot-tila voi vaikuttaa näyttöön.

round(*ListaiI* [, *numerot*]) ⇒ *lista*

Pyöristää elementtien listan jättäen desimaalipisteen jälkeen määritetyn määrän numeroita.

round(*MatriisiI* [, *numerot*]) ⇒ *matriisi*

Pyöristää elementtien matriisin jättäen desimaalipisteen jälkeen määritetyn määrän numeroita.

round(1.234567, 3)	1.235
--------------------	-------

round({ π, √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()Katalogi > 

rowAdd(*MatriisiI*, *rIndeksi1*, *rIndeksi2*) ⇒ *matriisi*

Luo kopion *MatriisiI*:stä, jossa rivi *rIndeksi2* on korvattu rivien *rIndeksi1* ja *rIndeksi2* summalla.

rowAdd($\begin{bmatrix} 3 & 4 \\ -3 & -2 \end{bmatrix}$, 1, 2)	$\begin{bmatrix} 3 & 4 \\ 0 & 2 \end{bmatrix}$
rowAdd($\begin{bmatrix} a & b \\ c & d \end{bmatrix}$, 1, 2)	$\begin{bmatrix} a & b \\ a+c & b+d \end{bmatrix}$

rowDim()

Katalogi >

rowDim(Matriisi)⇒*lauseke*Laskee *Matriisin* sisältämien rivien lukumäärän.**Huomaa:** Katso myös **colDim()**, sivu 30.

$\begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix} \rightarrow m1$	$\begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix}$
rowDim(<i>m1</i>)	3

rowNorm()

Katalogi >

rowNorm(Matriisi)⇒*lauseke*Laskee *Matriisin* riveillä olevien elementtien itseisarvojen summien maksimin.**Huomaa:** Kaikkien matriisien elementtien tulee sieventyä luvuiksi. Katso myös **colNorm()**, sivu 30.

rowNorm($\begin{bmatrix} -5 & 6 & -7 \\ 3 & 4 & 9 \\ 9 & -9 & -7 \end{bmatrix}$)	25
------------------------------------------------------------------------------------	----

rowSwap()

Katalogi >

rowSwap(Matriisi1, rIndeksi1, rIndeksi2)⇒*matriisi*Antaa tuloksena *Matriisi1*:n jossa rivien *rIndeksi1* ja *rIndeksi2* paikkoja on vaihdettu.

$\begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix} \rightarrow mat$	$\begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix}$
rowSwap(<i>mat</i> ,1,3)	$\begin{bmatrix} 5 & 6 \\ 3 & 4 \\ 1 & 2 \end{bmatrix}$

ref()



Katalogi >

ref(Matriisi1[, Tol])⇒*matriisi*Määrittää *Matriisi1*:n sievennetyn rivi-echelonmuodon.

$\text{ref}\left(\begin{bmatrix} -2 & -2 & 0 & -6 \\ 1 & -1 & 9 & -9 \\ -5 & 2 & 4 & -4 \end{bmatrix}\right)$	$\begin{bmatrix} 1 & 0 & 0 & \frac{66}{71} \\ 0 & 1 & 0 & \frac{147}{71} \\ 0 & 0 & 1 & \frac{-62}{71} \end{bmatrix}$
---------------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------

Valinnaisesti kaikkia 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.

$\text{ref}\left(\begin{bmatrix} a & b \\ c & d \end{bmatrix}\right)$	$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$
-----------------------------------------------------------------------	------------------------------------------------

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

Huomaa: Katso myös **ref()**, sivu 141.

S

sec()

 -painike

sec(LausI) ⇒ lauseke

Astekulmatilassa:

sec(ListaI) ⇒ lista

Laskee *LausI*:n sekantin tai määrittää listan, joka sisältää *Listal*:n kaikkien elementtien sekantit.

$$\sec(45) \quad \sqrt{2}$$

$$\sec(\{1, 2, 3, 4\}) \quad \left\{ \frac{1}{\cos(1)}, 1.00081, \frac{1}{\cos(4)} \right\}$$

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

sec⁻¹()

 -painike

sec⁻¹(LausI) ⇒ lauseke

Astekulmatilassa:

sec⁻¹(ListaI) ⇒ lista

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

Määrittää kulman, jonka sekantti on *LausI*, tai määrittää listan, joka sisältää *Listal*:n kaikkien elementtien käänteissekantit.

Graadikulmatilassa:

$$\sec^{-1}(\sqrt{2}) \quad 50$$

Huomaa: Vastaus lasketaan aste-, graadi- tai radiaanikulmana käytössä olevan

Radiaanikulmatilassa:

sec⁻¹()

trig -painike

kulmatila-asetuksen mukaisesti.

$$\text{sec}^{-1}(\{1,2,5\}) \quad \left\{0, \frac{\pi}{3}, \cos^{-1}\left(\frac{1}{5}\right)\right\}$$

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

sech()

Katalogi >

sech(LausI) ⇒ lauseke

$$\text{sech}(3) \quad \frac{1}{\cosh(3)}$$

sech(ListaI) ⇒ lista

$$\text{sech}(\{1,2,3,4\}) \quad \left\{\frac{1}{\cosh(1)}, 0.198522, \frac{1}{\cosh(4)}\right\}$$

Laskee *LausI*:n hyperbolisen sekantin tai määrittää listan, joka sisältää *Listai*:n kaikkien elementtien hyperboliset sekantit.

sech⁻¹()

Katalogi >

sech⁻¹(LausI) ⇒ lauseke

Radiaanikulma- ja suorakulmakompleksitilassa:

sech⁻¹(ListaI) ⇒ lista

$$\text{sech}^{-1}(1) \quad 0$$

$$\text{sech}^{-1}(\{1,-2,2,1\}) \quad \left\{0, \frac{2 \cdot \pi}{3} \cdot i, 8. \text{E} - 15 + 1.07448 \cdot i\right\}$$

Laskee *LausI*:n käänteisen hyperbolisen sekantin tai määrittää listan, joka sisältää *Listai*:n kaikkien elementtien käänteiset hyperboliset sekantit.

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

seq() (sekv)

Katalogi >

seq(Laus, Muutt, Matala, Korkea[, Askeli]) ⇒ lista

$$\text{seq}(n^2, n, 1, 6) \quad \{1, 4, 9, 16, 25, 36\}$$

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.

$$\text{seq}\left(\frac{1}{n}, n, 1, 10, 2\right) \quad \left\{1, \frac{1}{3}, \frac{1}{5}, \frac{1}{7}, \frac{1}{9}\right\}$$

$$\text{sum}\left(\text{seq}\left(\frac{1}{n^2}, n, 1, 10, 1\right)\right) \quad \frac{1968329}{1270080}$$

Oletusarvo *Step* = 1.

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* portailta *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* portailta *VarStep*, laskee *ListOfDepVars*(*Var*) muuttujan *Var* vastaaville arvoille käyttäen *Expr*-kaavaa ja *MatrixOfInitTerms* -luetteloa, ja antaa tulokset matriisina.

Muuttujan *Var* alkuperäinen sisältö on edelleen tallessa funktion **seqGen()** suorittamisen jälkeen

Oletusarvo *VarStep* = 1.

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:

$$\text{seqGen} \left(u(n-1)+2, n, u, \{1, 5\}, \{a\} \right)$$

$$\{ a, a+2, a+4, a+6, a+8 \}$$

Kahden sekvenssin ryhmä:

$$\text{seqGen} \left(\left\{ \frac{1}{n}, \frac{u^2(n-1)}{2} + u(n-1) \right\}, n, \{u1, u2\}, \{1, 5\}, \left[\begin{array}{c} - \\ 2 \end{array} \right] \right)$$

$$\left[\begin{array}{cccccc} 1 & \frac{1}{2} & \frac{1}{3} & \frac{1}{4} & \frac{1}{5} & \\ & 2 & 3 & 4 & 5 & \\ 2 & 2 & \frac{3}{2} & \frac{13}{2} & \frac{19}{2} & \end{array} \right]$$

Huomaa: Muuttujaa Tyhjä () yllä olevassa

aloitusterimatriisissa käytetään ilmoittamaan, että $u(1)$:n aloitustermi on laskettu käyttämällä täsmällistä sekvenssikaavaa $u(1) = 1/n$.

seqn()

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 $u(1)=2$.

$$\text{seqn}\left(\frac{u(n-1)}{n}, \{2\}, 6\right) \\ \left\{2, 1, \frac{1}{3}, \frac{1}{12}, \frac{1}{60}, \frac{1}{360}\right\}$$

$$\text{seqn}\left(\frac{1}{n^2}, 6\right) \quad \left\{1, \frac{1}{4}, \frac{1}{9}, \frac{1}{16}, \frac{1}{25}, \frac{1}{36}\right\}$$

series()

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

$$\text{series}\left(\frac{1-\cos(x-1)}{(x-1)^2}, x, -4, 1\right) \quad \frac{1}{2} \frac{(x-1)^2}{24} + \frac{(x-1)^4}{720}$$

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

$$\text{series}\left(\left(1+\frac{1}{n}\right)^n, n, 2, \infty\right) \quad e - \frac{e}{2 \cdot n} + \frac{11 \cdot e}{24 \cdot n^2}$$

Laskee yleisen tyypistetyt potensisarjaesityksen *LausI*: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.

Pisteen oletusarvo on 0. *Piste* voi olla ∞ tai $-\infty$, jolloin lavennus tapahtuu asteen *Aste* kautta lausekkeessa $1/(Muutt - Piste)$.

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 **sign**(...) tai **abs**(...) reaalille kehitelmän muuttujalle tai $(-1)^{\text{floor}(\text{angle}(\dots))}$ kompleksille kehitelmän muuttujalle, joka on merkkiin "_" päättyvä muuttuja. Jos tarkoituksesi on käyttää sarjaa vain *Pisteen* toisella puolella oleville arvoille, siinä tapauksessa liitä sopiva lauseke, "**|Muutt > Piste**", "**|Muutt < Piste**", "**|Muutt \geq Piste**" tai "**|Muutt \leq Piste**", jotta saat yksinkertaisemman vastauksen.

series() voi antaa symbolisia likiarvoja määrittämättömille integraaleille ja määrättyille integraaleille, joille ei muuten voida saada symbolisia ratkaisuja.

series() jakautuu 1. argumentin listoihin ja matriiseihin.

series() on **taylor()**-funktion yleistetty versio.

Kuten on havainnollistettu oikeanpuoleisessa viimeisessä esimerkissä, tuloksen alapuolella olevat **series**(...)-funktion tuottamat näyttöruutit voivat järjestää termit uudelleen siten, että dominanttitermi ei ole vasemmanpuoleisin.

Huomaa: Katso myös **dominantTerm()**, sivu 58.

$$\text{series}\left(\tan^4\left(\frac{1}{x}\right), x, 5\right), x > 0 \quad \frac{\pi}{2} - x + \frac{x^3}{3} - \frac{x^5}{5}$$

$$\text{series}\left(\int \frac{\sin(x)}{x} dx, x, 6\right) \quad x - \frac{x^3}{18} + \frac{x^5}{600}$$

$$\text{series}\left(\int_0^x \sin(x \cdot \sin(t)) dt, x, 7\right) \quad \frac{x^3}{2} - \frac{x^5}{24} + \frac{29 \cdot x^7}{720}$$

$$\text{series}\left(\left(1 + e^x\right)^2, x, 2, 1\right) \quad (e+1)^2 + 2 \cdot e \cdot (e+1) \cdot (x-1) + e \cdot (2 \cdot e+1) \cdot (x-1)^2$$

setMode(*tilaNimiKokonaisluku*,
asetusKokonaisluku) ⇒ *kokonaisluku*
setMode(*lista*) ⇒ *kokonaislukulista*

Toimii vain funktiossa tai ohjelmassa.

setMode(*tilaNimiKokonaisluku*,
asetusKokonaisluku) asettaa tilan
tilaNimiKokonaisluku väliaikaisesti uuteen
asetukseen *asetusKokonaisluku* ja määrittää
kokonaisluvun, joka vastaa kyseisen tilan
alkuperäistä asetusta. Muutos on rajoitettu
ohjelman/funktion suorittamisen ajalle.

tilaNimiKokonaisluku määrittää asetettavan tilan.
Sen on oltava jokin alla olevan taulukon tilaa
kuvaavista kokonaisluvuista.

asetusKokonaisluku määrittää tilan uuden asetuksen.
Sen on oltava jokin seuraavista asetettavalle tilalle
varatuista asetusta kuvaavista kokonaisluvuista.

setMode(*lista*)-komennolla voit muuttaa useita
asetuksia. *lista* sisältää tilaa ja asetusta kuvaavat
kokonaislukuparit. **setMode**(*lista*) luo samanlaisen
listan, jonka kokonaislukuparit kuvaavat alkuperäisiä
tiloja ja asetuksia.

Jos olet tallentanut kaikki tila-asetukset komennolla
getMode(0) → *muutt*, voit palauttaa nämä asetukset
komennolla **setMode**(*muutt*) aina funktion tai
ohjelman sulkemiseen saakka. Katso **getMode**(), sivu
80.

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.

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	<i>Done</i>
Disp approx(π)	
setMode(1,16)	
Disp approx(π)	
EndPrgm	
<hr/>	
<i>prog1</i> ()	
	3.14159
	3.14
	<i>Done</i>

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()

Katalogi > 

shift(KokonaislukuI[,SiirtojenLkm])⇒kokonaisluku

Siirtää binaarisen kokonaisluvun bittejä. Voit syöttää *KokonaislukuI*:n minä tahansa kantalukuna; 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 bittiä. Vasemmalle tapahtuvassa siirrossa vasemmanpuoleisin bitti pudotetaan, ja 0 lisätään vastaamaan

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 merkki etumerkki 0b ja heksadesimaaliluvun edelle 0h (nolla, ei O-kirjain).

oikeanpuoleisinta bittiä.

Esimerkki siirrosta oikealle:

Jokainen bitti siirtyy oikealle.

```
0b0000000000000111101011000011010
```

Lisää 0:n, jos vasemmanpuoleisin bitti on 0,

tai 1:n, jos vasemmanpuoleisin bitti on 1.

tuottaa:

```
0b0000000000000111101011000011010
```

Vastaus näkyy kantelukutilan mukaisesti. Alkunollia ei näytetä.

shift(Lista1 [,SiirtojenLkm])⇒*lista*

Luo *Listal*:n kopion, jota on siirretty oikealle tai vasemmalle *SiirtojenLkm*:n elementtien määrittämän määrän. Ei muuta *Listal*:ä.

Jos *SiirtojenLkm* on positiivinen, siirto tapahtuu vasemmalle. Jos *SiirtojenLkm* on negatiivinen, siirto tapahtuu oikealle. Oletusarvo on -1 (siirrä yksi elementti oikealle).

Elementit, jotka siirto vie *listan* alkuun tai loppuun, asettuvat symboliksi "undef".

shift(Merkkijono1 [,SiirtojenLkm])⇒*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.

Desimaalisessa kantelukutilassa:

shift({ 1,2,3,4 })	{ undef,1,2,3 }
shift({ 1,2,3,4 },-2)	{ undef,undef,1,2 }
shift({ 1,2,3,4 },2)	{ 3,4,undef,undef }

shift("abcd")	" abc "
shift("abcd",-2)	" ab "
shift("abcd",1)	"bcd "

sign() (etumerkki)Katalogi > **sign(LausI)**⇒lauseke

$$\text{sign}(-3.2) \quad -1.$$

sign(ListaI)⇒lista

$$\text{sign}(\{2,3,4,-5\}) \quad \{1,1,1,-1\}$$

sign(MatriisiI)⇒matriisi

$$\text{sign}(1+|x|) \quad 1$$

Kun kyseessä on reaali- tai kompleksilukulauseke *LausI*, antaa vastauksena *LausI*/**abs(LausI)**, kun *LausI* ≠ 0.

Jos kompleksilukutila on Reaali:

Vastaus on 1, jos *LausI* on positiivinen. Vastaus on -1, jos *LausI* on negatiivinen.

$$\text{sign}(\begin{bmatrix} -3 & 0 & 3 \end{bmatrix}) \quad \begin{bmatrix} -1 & \pm 1 & 1 \end{bmatrix}$$

sign(0) edustaa kompleksialueen yksikköpiiriä.

Kun kyseessä on lista tai matriisi, määrittää kaikkien elementtien etumerkit.

simult()Katalogi > **simult(kerroinMatriisi, vakioVektori[, tol])**⇒matriisi

Ratkaise yhtälöstä x ja y:

Määrittää sarakevektorin, joka sisältää lineaarisen yhtälöryhmän ratkaisut.

$$x + 2y = 1$$

$$3x + 4y = -1$$

Huomaa: Katso myös **linSolve()**, sivu 98.

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

kerroinMatriisin on oltava neliömatriisi, joka sisältää yhtälöiden kertoimet.

Ratkaisu on x=-3 and y=2.

vakioVektorissa on oltava sama rivimäärä (sama koko) kuin *kerroinMatriisissa* ja sen tulee sisältää vakiot.

Ratkaise:

$$ax + by = 1$$

$$cx + dy = 2$$

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.

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} \rightarrow \text{mat}x1 \quad \begin{bmatrix} a & b \\ c & d \end{bmatrix}$$

$$\text{simult}\left(\text{mat}x1, \begin{bmatrix} 1 \\ 2 \end{bmatrix}\right) \quad \begin{bmatrix} -(2 \cdot b - d) \\ a \cdot d - b \cdot c \\ 2 \cdot a - c \\ a \cdot d - b \cdot c \end{bmatrix}$$

- Jos asetat **Automaattinen tai likimääräinen** -tilan valintaan Approximate (Likimääräinen), laskut suoritetaan liukulukuaritmetiikalla.
- Jos *Tol* jätetään pois tai sitä ei käytetä, oletusarvoinen toleranssi lasketaan

seuraavasti:

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

simult(kerroinMatriisi, vakioMatriisi[, tol])

\Rightarrow matriisi

Ratkaisee lineaarisia yhtälöryhmiä, joissa jokaisessa ryhmässä on samat yhtälöiden kertoimet mutta eri vakiot.

Jokaisen vakioMatriisin sarakkeen tulee sisältää jonkin yhtälöryhmän vakiot. Jokainen tulosmatriisin sarakke sisältää vastaavan ryhmän ratkaisun.

Ratkaise:

$$x + 2y = 1$$

$$3x + 4y = -1$$

$$x + 2y = 2$$

$$3x + 4y = -3$$

$$\text{simult}\left(\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}, \begin{bmatrix} 1 & 2 \\ -1 & -3 \end{bmatrix}\right) \quad \begin{bmatrix} -3 & -7 \\ 2 & \frac{9}{2} \end{bmatrix}$$

Ensimmäisessä ryhmässä $x=-3$ ja $y=2$. Toisessa ryhmässä $x=-7$ ja $y=9/2$.

Laus **sin**

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

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

sin alentaa kaikkia lausekkeen

$\cos(\dots)$ modulo $1 - \sin(\dots)^2$

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

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

$$\cos(x)^2 \sin \quad 1 - (\sin(x))^2$$

sin(LausI) \Rightarrow lauseke

Astekulmatilassa:

sin() -painike**sin(Lista1)**⇒*lista***sin(Laus1)** määrittää argumentin sinin lausekkeena.**sin(Lista1)** määrittää listan, joka sisältää *Listal*:n kaikkien elementin sinit.**Huomaa:** Argumentti tulkitaan aste-, graadi- tai radiaanikulmaksi käytössä olevan kulmatilan mukaisesti. Voit ohittaa kulmatila-asetuksen väliaikaisesti käyttämällä merkintää °, G tai r.

$\sin\left(\frac{\pi}{4}\right)$	$\frac{\sqrt{2}}{2}$
----------------------------------	----------------------

$\sin(45)$	$\frac{\sqrt{2}}{2}$
------------	----------------------

$\sin(\{0,60,90\})$	$\left\{0, \frac{\sqrt{3}}{2}, 1\right\}$
---------------------	-------------------------------------------

Graadikulmatilassa:

$\sin(50)$	$\frac{\sqrt{2}}{2}$
------------	----------------------

Radiaanikulmatilassa:

$\sin\left(\frac{\pi}{4}\right)$	$\frac{\sqrt{2}}{2}$
----------------------------------	----------------------

$\sin(45^\circ)$	$\frac{\sqrt{2}}{2}$
------------------	----------------------

sin(neliömatriisi1)⇒*neliömatriisi*Määrittää *neliömatriisi1*:n matriisin sinin. Tämä ei ole sama kuin kunkin elementin sinin laskeminen. Laskentamenetelmä on kuvattu kohdassa **cos()**.*neliömatriisi1*:n on oltava diagonalisoitavissa.

Vastaus sisältää aina liukulukuja.

Radiaanikulmatilassa:

$\sin\begin{pmatrix} 1 & 5 & 3 \\ 4 & 2 & 1 \\ 6 & -2 & 1 \end{pmatrix}$	$\begin{bmatrix} 0.9424 & -0.04542 & -0.031999 \\ -0.045492 & 0.949254 & -0.020274 \\ -0.048739 & -0.00523 & 0.961051 \end{bmatrix}$
--------------------------------------------------------------------------	--------------------------------------------------------------------------------------------------------------------------------------

sin⁻¹() -painike**sin⁻¹(Laus1)**⇒*lauseke***sin⁻¹(Lista1)**⇒*lista***sin⁻¹(Laus1)** määrittää lausekkeena kulman, jonka sini on *Laus1*.**sin⁻¹(Lista1)** määrittää listan, joka sisältää *Listal*:n kaikkien elementtien käänteissinit.**Huomaa:** Vastaus lasketaan aste-, graadi- tai radiaanikulmana käytössä olevan kulmatila-asetuksen mukaisesti.

Astekulmatilassa:

$\sin^{-1}(1)$	90
----------------	----

Graadikulmatilassa:

$\sin^{-1}(1)$	100
----------------	-----

Radiaanikulmatilassa:

sin⁻¹()
 **-painike**

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

sin⁻¹(neliomatriisi)⇒neliomatriisi

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

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

$$\sin^{-1}(\{0,0,2,0.5\}) \quad \{0,0.201358,0.523599\}$$

Radiaanikulmatilassa ja suorakulmakompleksimuodossa:

$$\sin^{-1}\left(\begin{pmatrix} 1 & 5 \\ 4 & 2 \end{pmatrix}\right) \begin{bmatrix} -0.174533-0.12198 \cdot i & 1.74533-2.35591 \cdot i \\ 1.39626-1.88473 \cdot i & 0.174533-0.593162 \cdot i \end{bmatrix}$$

sinh()
Katalogi > 

sinh(LausI)⇒lauseke

$$\sinh(1.2) \quad 1.50946$$

sinh(ListaI)⇒lista

$$\sinh(\{0,1,2,3\}) \quad \{0,1.50946,10.0179\}$$

sinh(LausI) laskee argumentin hyperbolisen kosinin lausekkeena.

sinh(ListaI) määrittää listan, joka sisältää *ListaI*:n jokaisen elementin hyperbolisen sinin.

sinh(neliomatriisi)⇒neliomatriisi

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

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

Radiaanikulmatilassa:

$$\sinh\left(\begin{pmatrix} 1 & 5 & 3 \\ 4 & 2 & 1 \\ 6 & -2 & 1 \end{pmatrix}\right) \begin{bmatrix} 360.954 & 305.708 & 239.604 \\ 352.912 & 233.495 & 193.564 \\ 298.632 & 154.599 & 140.251 \end{bmatrix}$$

sinh⁻¹()
Katalogi > 

sinh⁻¹(LausI)⇒lauseke

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

sinh⁻¹(ListaI)⇒lista

$$\sinh^{-1}(\{0,2,1,3\}) \quad \{0,1.48748,\sinh^{-1}(3)\}$$

sinh⁻¹(LausI) laskee argumentin käänteisen hyperbolisen sinin lausekkeena.

sinh⁻¹(ListaI) määrittää listan, joka sisältää *ListaI*:n jokaisen elementin käänteiset hyperboliset sinin.

Huomaa: Voit syöttää tämän funktion näppäimistöltä

kirjoittamalla **arcsinh (...)**.

sinh⁻¹(neliomatriisi I) ⇒ **neliomatriisi**

Laskee **neliomatriisi I**:n matriisin käänteisen hyperbolisen sinin. Tämä ei ole sama kuin kunkin elementin käänteisen hyperbolisen sinin laskeminen.

Laskentamenetelmä on kuvattu kohdassa **cos()**.

neliomatriisi I:n on oltava diagonalisoitavissa.

Vastaus sisältää aina liukulukuja.

Radiaaనిkulmatilassa:

$$\sinh^{-1} \begin{pmatrix} 1 & 5 & 3 \\ 4 & 2 & 1 \\ 6 & -2 & 1 \end{pmatrix} \begin{bmatrix} 0.041751 & 2.15557 & 1.1582 \\ 1.46382 & 0.926568 & 0.112557 \\ 2.75079 & -1.5283 & 0.57268 \end{bmatrix}$$

SinReg

SinReg $X, Y [, [Iteraatio] [, [Jakso] [, Luokka, Sisällyttä]]$

Laskee siniregression listoista X ja Y . Tulosten yhteenveto tallentuu *stat.results*-muuttuun. (Katso sivu 169.)

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

X ja Y ovat riippumattomien ja riippuvien muuttujien listoja.

Iteraatio on arvo, joka määrittää ratkaisun yrityskertojen (1-16) enimmäismäärän. Mikäli sitä ei määritetä, oletuksena käytetään arvoa 8. Suuremmilla arvoilla saadaan tyypillisesti parempi tarkkuus, mutta suoritus aika on pitempi ja päin vastoin.

Jakso määrittää arvioidun jakson. Mikäli sitä ei käytetä, X :n arvojen välisen eron tulisi olla sama ja arvojen tulisi olla peräkkäisessä järjestyksessä. Jos määrität *Jakson*, x :n arvojen väliset erot voivat olla erisuuria.

Luokka on luokkakoodien lista vastaavalle X - ja Y -datalle..

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

SinReg:n tulos esitetään aina radiaaneina riippumatta kulmatilan asetuksesta.

Lisätietoja listassa olevien tyhjien elementtien vaikutuksesta, katso Tyhjät elementtisivulla sivu 227.

Tulosmuuttuja	Kuvaus
stat.RegEqn	Regressioyhtälö: $a \cdot \sin(bx+c)+d$

Tulosmuuttuja	Kuvaus
stat.a, stat.b, stat.c, stat.d	Regressiokertoimet
stat.Resid	Regressioyhtälön jäännökset
stat.XReg	Muokatus <i>Y Lista</i> :n sisältämä datapisteiden lista, jota käytetään regressiossa komentojen <i>Frekv</i> , <i>Luokkalista</i> ja <i>Sisällytä luokat rajoitusten mukaisesti</i>
stat.YReg	Muokatus <i>Y Lista</i> :n sisältämä datapisteiden lista, jota käytetään regressiossa komentojen <i>Frekv</i> , <i>Luokkalista</i> ja <i>Sisällytä luokat rajoitusten mukaisesti</i>
stat.FreqReg	Komentoja <i>stat.XReg</i> ja <i>stat.YReg</i> vastaava frekvenssilista

solve()

Katalogi > 

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

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

solve(*Epäyhtälö*, *Muutt*) ⇒ *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.

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*.

$$\text{solve}(a \cdot x^2 + b \cdot x + c = 0, x)$$

$$x = \frac{\sqrt{b^2 - 4 \cdot a \cdot c} - b}{2 \cdot a} \text{ or } x = \frac{-\left(\sqrt{b^2 - 4 \cdot a \cdot c} + b\right)}{2 \cdot a}$$

$$\text{Ans} | a=1 \text{ and } b=1 \text{ and } c=1$$

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

$$\text{solve}((x-a) \cdot e^x = x \cdot (x-a), x)$$

$$x = a \text{ or } x = -0.567143$$

$$(x+1) \cdot \frac{x-1}{x-1} + x - 3 \quad 2 \cdot x - 2$$

$$\text{solve}(5 \cdot x - 2 \geq 2 \cdot x, x)$$

$$x \geq \frac{2}{3}$$

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 reaalityyppiset ratkaisut eivät 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.

Reaalityyppisissä murtolukupotenssissa, joilla on pariton nimittäjä, viittaavat ainoastaan reaalityyppiseen. Muussa tapauksessa haaralausekkeet, kuten murtolukupotenssit, logaritmit ja käänteiset trigonometriset funktiot, viittaavat vain päähaaraan. Näin ollen **solve()** tuottaa vain ratkaisuja, jotka vastaavat tätä yhtä reaalityyppistä tai päähaaraa.

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

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

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

solve(*{Yht1, Yht2* [...]) {*MuuttTaiArvaus1*, *MuuttTaiArvaus2* [, ...]}) ⇒ Boolean lauseke

Antaa ehdotuksia reaalityyppisistä ratkaisuksista

exact(**solve**($(x-a) \cdot e^x = x \cdot (x-a), x$))
 $e^x + x = 0$ or $x = a$

Radiaaniikulmatilassa:

solve($\tan(x) = \frac{1}{x}, x$), $x > 0$ and $x < 1$
 $x = 0.860334$

solve($x = x + 1, x$) false
solve($x = x, x$) true

$2 \cdot x - 1 \leq 1$ and **solve**($x^2 \neq 9, x$) $x \neq -3$ and $x \leq 1$

Radiaaniikulmatilassa:

solve($\sin(x) = 0, x$) $x = n1 \cdot \pi$

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

solve($\sqrt{x} = 2, x$) false

solve($\sqrt{-x} = 2, x$) $x = -4$

solve($y = x^2 - 2$ and $x + 2 \cdot y = -1, \{x, y\}$)
 $x = \frac{-3}{2}$ and $y = \frac{1}{4}$ or $x = 1$ and $y = -1$

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*-komentoon on oltava muodossa:

muuttuja

- tai -

muuttuja = reaaliluku tai ei-reaaliluku

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 reaalilukuratkaisut.

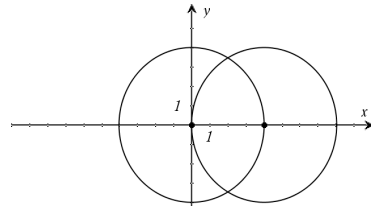
Oletetaan esimerkiksi, että origossa on ympyrän säde r , ja toinen ympyrän säde r on keskitetty kohtaan, jossa ensimmäinen ympyrä leikkaa positiivisen x -akselin. Määritä leikkauskohdat **solve()**-funktion avulla.

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 kokonaislukuilite välillä 1-255.

Polynomisarjoissa laskun suoritus aika tai muistin käyttö voivat riippua merkittävästi ratkaisumuuttujien



$$\text{solve}\left(x^2+y^2=r^2 \text{ and } (x-r)^2+y^2=r^2, \{x,y\}\right)$$

$$x=\frac{r}{2} \text{ and } y=\frac{\sqrt{3}\cdot r}{2} \text{ or } x=\frac{r}{2} \text{ and } y=\frac{-\sqrt{3}\cdot r}{2}$$

$$\text{solve}\left(x^2+y^2=r^2 \text{ and } (x-r)^2+y^2=r^2, \{x,y,z\}\right)$$

$$x=\frac{r}{2} \text{ and } y=\frac{\sqrt{3}\cdot r}{2} \text{ and } z=c1 \text{ or } x=\frac{r}{2} \text{ and } y=-\frac{\sqrt{3}\cdot r}{2} \text{ and } z=c1$$

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

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 reaaliulukratkaisuja.

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

Jokainen ratkaisumuuttuja alkaa arvausarvostaan, mikäli se on määritetty; muussa tapauksessa se alkaa arvosta 0.0.




Arvausten avulla voit etsiä lisäratkaisuja yksi kerrallaan. Suppenemista varten arvauksen on mahdollisesti oltava melko lähellä ratkaisua.

$$\text{solve}\left(x+e^z \cdot y=1 \text{ and } x-y=\sin(z),\{x,y\}\right)$$

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

$$\text{solve}\left(e^z \cdot y=1 \text{ and } -y=\sin(z),\{y,z\}\right)$$

$$y=2.812\text{E-}10 \text{ and } z=21.9911 \text{ or } y=0.001871$$

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

$$\text{solve}\left(e^z \cdot y=1 \text{ and } -y=\sin(z),\{y,z=2 \cdot \pi\}\right)$$

$$y=0.001871 \text{ and } z=6.28131$$

SortA

SortA *Lista1* [, *Lista2*] [, *Lista3*] ...

SortA *Vektori1* [, *Vektori2*] [, *Vektori3*] ...

Lajittelee ensimmäisen argumentin elementit nousevaan järjestykseen.

Jos otat mukaan lisäargumentteja, lajittelee kunkin argumentin elementit siten, että niiden uudet paikat vastaavat ensimmäisen argumentin elementtien uusia paikkoja.

Kaikkien argumenttien on oltava lista- tai vektorinimiä. Kaikkien argumenttien on oltava samankokoisia.

Ensimmäisen argumentin sisältämät tyhjät elementit

$\{2,1,4,3\} \rightarrow \text{list1}$	$\{2,1,4,3\}$
SortA list1	Done
list1	$\{1,2,3,4\}$
$\{4,3,2,1\} \rightarrow \text{list2}$	$\{4,3,2,1\}$
SortA list2,list1	Done
list2	$\{1,2,3,4\}$
list1	$\{4,3,2,1\}$

siirtyvät alas. Lisätietoja tyhjiistä elementeistä, katso sivu 227.

SortD

SortD *List1* [, *List2*] [, *List3*] ...

$\{2,1,4,3\} \rightarrow list1$ $\{2,1,4,3\}$

SortD *Vektori1* [, *Vektori2*] [, *Vektori3*] ...

$\{1,2,3,4\} \rightarrow list2$ $\{1,2,3,4\}$

Muuten samanlainen kuin **SortA** paitsi, että **SortD** lajittelee elementit laskevaan järjestykseen.

SortD *list1*, *list2* *Done*

Ensimmäisen argumentin sisältämät tyhjat elementit siirtyvät alas. Lisätietoja tyhjiistä elementeistä, katso sivu 227.

list1 $\{4,3,2,1\}$

list2 $\{3,4,1,2\}$

Sphere

Vektori ▶ **Sphere**

Huom: Vastauksen pakottaminen likimääräiseksi:

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

Kämmenlaite: Paina **ctrl enter**.

Näyttää rivi- tai sarakevektorin pallonmuotoisena [ρ $\angle \theta$ $\angle \phi$].

Windows®: Paina **Ctrl+Enter**.

Macintosh®: Paina **⌘+Enter**.

Vektorin on oltava kooltaan 3, ja se voi olla rivi- tai sarakevektori.

iPad®: Pidä **enter** ja valitse .

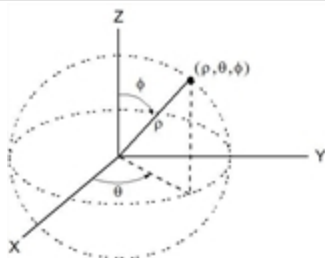
Huomaa: ▶**Sphere** on näyttömuodon ohje, ei muunnosfunktio. Voit käyttää komentoa ainoastaan syöterivin lopussa.

$[1 \ 2 \ 3] \triangleright \text{Sphere}$
 $[3.74166 \ \angle 1.10715 \ \angle 0.640522]$

$\left(\begin{matrix} 2 & \angle \frac{\pi}{4} & 3 \end{matrix} \right) \triangleright \text{Sphere}$
 $[3.60555 \ \angle 0.785398 \ \angle 0.588003]$

Paina **enter**

$\left(\begin{matrix} 2 & \angle \frac{\pi}{4} & 3 \end{matrix} \right) \triangleright \text{Sphere}$
 $\left[\sqrt{13} \ \angle \frac{\pi}{4} \ \angle \sin^{-1} \left(\frac{2 \cdot \sqrt{13}}{13} \right) \right]$

**sqrt()****sqrt(LausI)** ⇒ lauseke

$$\sqrt{4} \qquad 2$$

sqrt(ListaI) ⇒ lista

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

Laskee argumentin neliöjuuren.

Kun kyseessä on lista, laskee kaikkien *Listai*:n elementtien neliöjuuret.

Huomaa: Katso myös **Neliöjuurimalli**, sivu 5.

stat.results

Näyttää tilastollisen laskutoimituksen tulokset.

Vastaukset näytetään nimiarvoparien sarjana.

Näytetyt nimenomaiset nimet riippuvat viimeksi sievennetystä tilastofunktiosta tai komennosta.

Voit kopioida nimen tai arvon ja liittää sen muihin paikkoihin.

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.

$$xlist:=\{1,2,3,4,5\} \quad \{1,2,3,4,5\}$$

$$ylist:=\{4,8,11,14,17\} \quad \{4,8,11,14,17\}$$

LinRegMx xlist,ylist,1: stat.results

"Title"	"Linear Regression (mx+b)"
"RegEqn"	"m*x+b"
"m"	3.2
"b"	1.2
"r ² "	0.996109
"r"	0.998053
"Resid"	" {... } "

stat.values	"Linear Regression (mx+b)"
	"m*x+b"
	3.2
	1.2
	0.996109
	0.998053
	" {-0.4,0.4,0.2,0,-0.2} "

stat.a	stat.dfDenom	stat.MedianY	stat.Q3X	stat.SSBlock
stat.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.̄ _x
stat.b9	stat.FBlock	Stat.Ç	stat.Σ _x ²	stat.̄ _{x1}
stat.b10	stat.Fcol	stat.Ç ₁	stat.Σ _{xy}	stat.̄ _{x2}
stat.bList	stat.FInteract	stat.Ç ₂	stat.Σ _y	stat.̄ _x Diff
stat.χ ²	stat.FreqReg	stat.ÇDiff	stat.Σ _y ²	stat.̄ _x List
stat.c	stat.Frow	stat.PList	stat.s	stat.XReg
stat.CLower	stat.Leverage	stat.PVal	stat.SE	stat.XVal
stat.CLowerList	stat.LowerPred	stat.PValBlock	stat.SEList	stat.XValList
stat.CompList	stat.LowerVal	stat.PValCol	stat.SEPred	stat.ȳ
stat.CompMatrix	stat.m	stat.PValInteract	stat.sResid	stat.ȳ
stat.CookDist	stat.MaxX	stat.PValRow	stat.SEslope	stat.ȳList

stat.CUpper	stat.MaxY	stat.Q1X	stat.sp	stat.YReg
stat.CUpperList	stat.ME	stat.Q1Y	stat.SS	
stat.d	stat.MedianX			

Huomaa: Aina kun Listat & Taulukot -sovellus laskee tilastolaskujen vastauksia, se kopioi "stat." -ryhmän muuttujat "stat#." -ryhmään, jossa # on luku, jota lisätään automaattisesti. Tällä tavoin voit säilyttää aikaisemmat tulokset suorittaessasi useita laskutoimituksia.

stat.values

Katalogi > 

stat.values

Katso esimerkki kohdassa **stat.results**.

Näyttää matriisin viimeksi sievennetylle 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

Laskee *Lista*:n sisältämien elementtien perusjoukon keskihajonnan.

Jokainen *frekvLista*n elementti näyttää *Listan* vastaavien elementtien peräkkäisten esiintymien lukumäärän.

Huomaa: *Listassa* tulee olla vähintään kaksi elementtiä. Tyhjiä elementtejä ei huomioida. Lisätietoja tyhjiistä elementeistä, katso sivu 227.

stDevPop(MatriisiI[, frekvMatriisiI]) ⇒ matriisi

Laskee rivivektorin *MatriisiI*:n sarakkeiden perusjoukon keskihajonnoista.

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

Huomaa: *MatriisiI* :ssä on oltava vähintään kaksi riviä. Tyhjiä elementtejä ei huomioida. Lisätietoja

Radiaanikulma- ja automaattisissa tiloissa:

$$\text{stDevPop}\left\{\left\{a, b, c\right\}\right\} = \frac{\sqrt{2 \cdot\left(a^2-a \cdot(b+c)+b^2-b \cdot c+c^2\right)}}{3}$$

$$\text{stDevPop}\left\{\left\{1,2,5,-6,3,-2\right\}\right\} = \frac{\sqrt{465}}{6}$$

$$\text{stDevPop}\left\{\left\{1.3,2.5,-6.4\right\},\left\{3,2,5\right\}\right\} = 4.11107$$

$$\text{stDevPop}\left(\begin{array}{ccc} 1 & 2 & 5 \\ -3 & 0 & 1 \\ 5 & 7 & 3 \end{array}\right) = \left[\begin{array}{ccc} 4 \cdot \sqrt{6} & \sqrt{78} & 2 \cdot \sqrt{6} \\ 3 & 3 & 3 \end{array}\right]$$

$$\text{stDevPop}\left(\begin{array}{cc} -1.2 & 5.3 \\ 2.5 & 7.3 \\ 6 & -4 \end{array}\right) = \left[\begin{array}{cc} 4 & 2 \\ 3 & 3 \\ 1 & 7 \end{array}\right]$$

$$\left[2.52608 \quad 5.21506\right]$$

tyhjästä elementeistä, katso sivu 227.

stDevSamp()

stDevSamp(*Lista*[, *frekvLista*])⇒*lauseke*

Laskee *Lista*n sisältämien elementtien otoksen keskihajonnan.

Jokainen *frekvLista*n elementti näyttää *Lista*n vastaavien elementtien peräkkäisten esiintymien lukumäärän.

Huomaa: *Listassa* tulee olla vähintään kaksi elementtiä. Tyhjiä elementtejä ei huomioida. Lisätietoja tyhjästä elementeistä, katso sivu 227.

stDevSamp(*MatriisiI*[, *frekvMatriisiI*])⇒*matriisi*

Laskee rivivektorin *MatriisiI*:n sarakkeiden otosten keskihajonnoista.

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

Huomaa: *MatriisiI* :ssä on oltava vähintään kaksi riviä. Tyhjiä elementtejä ei huomioida. Lisätietoja tyhjästä elementeistä, katso sivu 227.

$$\text{stDevSamp}\left\{\{a,b,c\}\right\} = \frac{\sqrt{3 \cdot (a^2 - a \cdot (b+c) + b^2 - b \cdot c + c^2)}}{3}$$

$$\text{stDevSamp}\left\{\{1,2,5,-6,3,-2\}\right\} = \frac{\sqrt{62}}{2}$$

$$\text{stDevSamp}\left\{\{1.3,2.5,-6.4\},\{3,2,5\}\right\} = 4.33345$$

$$\text{stDevSamp}\left(\begin{pmatrix} 1 & 2 & 5 \\ -3 & 0 & 1 \\ 5 & 7 & 3 \end{pmatrix}\right) = \begin{bmatrix} 4 & \sqrt{13} & 2 \end{bmatrix}$$

$$\text{stDevSamp}\left(\begin{pmatrix} -1.2 & 5.3 \\ 2.5 & 7.3 \\ 6 & -4 \end{pmatrix}, \begin{bmatrix} 4 & 2 \\ 3 & 3 \\ 1 & 7 \end{bmatrix}\right) = \begin{bmatrix} 2.7005 & 5.44695 \end{bmatrix}$$

Stop

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 progI()=Prgm Done
  For i,1,10,1
  If i=5
  Stop
  EndFor
EndPrgm
progI() Done
i 5
```

string() (merkkijono)Katalogi > **string(Laus)** ⇒ merkkijono

Sieventää lausekkeen *Laus* ja antaa vastauksen merkkijonona.

<code>string(1.2345)</code>	"1.2345"
<code>string(1+2)</code>	"3"
<code>string(cos(x)+√3)</code>	"cos(x)+√(3)"

subMat()Katalogi > 

subMat(MatriisiI[, alkurivi] [, alkusarake] [, loppurivi] [, loppusarake])
⇒ matriisi

Laskee *MatriisiI*:n määritetyn alimatriisin.

Oletusarvot: *alkurivi*=1, *alkusarake*=1, *loppurivi*=viimeinen rivi, *endCol*=viimeinen sarake.

$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix} \rightarrow m1$	$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$
<code>subMat(m1,2,1,3,2)</code>	$\begin{bmatrix} 4 & 5 \\ 7 & 8 \end{bmatrix}$
<code>subMat(m1,2,2)</code>	$\begin{bmatrix} 5 & 6 \\ 8 & 9 \end{bmatrix}$

Sum (Sigma)Katso $\Sigma()$, sivu 215.**sum()**Katalogi > **sum(Lista[, Alku[, Loppu]])** ⇒ lauseke

Laskee *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 227.

<code>sum({1,2,3,4,5})</code>	15
<code>sum({a,2·a,3·a})</code>	6·a
<code>sum(seq(n,n,1,10))</code>	55
<code>sum({1,3,5,7,9},3)</code>	21

sum()Katalogi > **sum(Matriisi[, Alku[, Loppu]])**⇒matriisiLaskee rivivektorin, joka sisältää *Matriisi1*:n sarakkeiden elementtien summat.*Alku* ja *Loppu* ovat valinnaisia. Ne määrittävät rivialueen.Mikä tahansa tyhjä argumentti tuottaa tyhjän vastauksen. *Matriisi1*:ssä olevia tyhjiä elementtejä ei huomioida. Lisätietoja tyhjiä elementeistä, katso sivu 227.

sum	$\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{pmatrix}$	[5 7 9]
sum	$\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix}$	[12 15 18]
sum	$\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix}, 2, 3$	[11 13 15]

sumIf()Katalogi > **sumIf(Lista, Kriteeri[, SummaLista])**⇒arvoLaskee kaikkien niiden *Listan* sisältämien elementtien kumuloituneen summan, jotka vastaavat määritettyjä kriteereitä *Kriteeri*. 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 niitä *Listan* elementtejä, jotka ovat alle 10.

Kun jokin *Listan* elementti vastaa kriteereitä *Kriteeri*, elementti lisätään kumuloituvaan summaan. Jos sisällytät funktion *summaListan*, summaan lisätäänkin sen sijaan vastaava *summaListan* elementti.Listat & Taulukot -sovelluksessa voit käyttää solualueita *Listan* ja *summaListan* tilalla.

Tyhjiä elementtejä ei huomioida. Lisätietoja tyhjiä elementeistä, katso sivu 227.

sumIf	$\{\{1, 2, e, 3, \pi, 4, 5, 6\}, 2.5 < ? < 4.5\}$	$e + \pi + 7$
sumIf	$\{\{1, 2, 3, 4\}, 2 < ? < 5, \{10, 20, 30, 40\}\}$	70

sumlf()

Katalogi >

Huomaa: Katso myös **countlf()**, sivu 39.

sumSeq()

Katso $\Sigma()$, sivu 215.

system()

Katalogi >

system(Yht1 [, Yht2 [, Yht3 [, ...]])

$\text{solve}\left(\begin{cases} x+y=0 \\ x-y=8 \end{cases}, x, y\right)$ $x=4$ and $y=-4$

system(Laus1 [, Laus2 [, Laus3 [, ...]])

Laskee yhtälöryhmän listaksi muotoiltuna. Voit luoda yhtälöryhmän myös mallin avulla.

Huomaa: Katso myös **Yhtälöryhmä**, sivu 8.

T

T (transponoi)

Katalogi >

Matriisi $I^T \Rightarrow$ *matriisi*

Laskee *Matriisi* I :n transponoidun liittokompleksimatriisin.

Huomaa: Voit syöttää tämän operaattorin tietokoneen näppäimistöltä kirjoittamalla @t.

$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}^T$	$\begin{bmatrix} 1 & 4 & 7 \\ 2 & 5 & 8 \\ 3 & 6 & 9 \end{bmatrix}$
$\begin{bmatrix} a & b \\ c & d \end{bmatrix}^T$	$\begin{bmatrix} a & c \\ b & d \end{bmatrix}$
$\begin{bmatrix} 1+i & 2+i \\ 3+i & 4+i \end{bmatrix}^T$	$\begin{bmatrix} 1-i & 3-i \\ 2-i & 4-i \end{bmatrix}$

tan()

-painike

tan(Laus1) \Rightarrow *lauseke*

Astekulmatilassa:

tan(Lista1) \Rightarrow *lista*

$\tan\left(\frac{\pi}{4}\right)$ 1

tan(Laus1) laskee argumentin tangentin lausekkeena.

$\tan(45)$ 1

tan(Lista1) määrittää *Lista1*: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,undef}

Radiaanikulmatilassa:

$\tan\left(\frac{\pi}{4}\right)$	1
$\tan(45^\circ)$	1
$\tan\left(\left\{\pi, \frac{\pi}{3}, -\pi, \frac{\pi}{4}\right\}\right)$	{0,√3,0,1}

Radiaanikulmatilassa:

$\tan\left(\begin{array}{ccc} 1 & 5 & 3 \\ 4 & 2 & 1 \\ 6 & -2 & 1 \end{array}\right)$	$\begin{bmatrix} -28.2912 & 26.0887 & 11.1142 \\ 12.1171 & -7.83536 & -5.48138 \\ 36.8181 & -32.8063 & -10.4594 \end{bmatrix}$
----------------------------------------------------------------------------------------	--------------------------------------------------------------------------------------------------------------------------------

tan(neliömatriisiI)⇒neliömatriisi

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

neliömatriisiI:n on oltava diagonalisoitavissa.

Vastaus sisältää aina liukulukuja.

tan⁻¹(LausI)⇒lauseke**tan⁻¹(ListI)⇒lista**

tan⁻¹(LausI) laskee kulman, jonka tangenti on *LausI*, määrittäen vastauksen lausekkeena.

tan⁻¹(ListI) luo listan *ListI*:n jokaisen elementin käänteistangenteista.

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

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

tan⁻¹(neliömatriisiI)⇒neliömatriisi

Laskee *neliömatriisiI*:n matriisin käänteistangentin.

Astekulmatilassa:

$\tan^{-1}(1)$	45
----------------	----

Graadikulmatilassa:

$\tan^{-1}(1)$	50
----------------	----

Radiaanikulmatilassa:

$\tan^{-1}(\{0,0,2,0,5\})$	{0,0.197396,0.463648}
----------------------------	-----------------------

Radiaanikulmatilassa:

tan⁻¹()

trig -painike

Tämä ei ole sama kuin kunkin elementin käänteisen tangentin laskeminen. Laskentamenetelmä on kuvattu kohdassa **cos()**.

neliömatriisi I:n on oltava diagonalisoitavissa.

Vastaus sisältää aina liukulukuja.

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

-0.083658	1.26629	0.62263
0.748539	0.630015	-0.070012
1.68608	-1.18244	0.455126

tangentLine()

Katalogi >

tangentLine(LausI, Muutt, Piste) ⇒ lauseke

tangentLine(LausI, Muutt=Piste) ⇒ lauseke

Määrittää tangenttisuoran *LausI:n* kuvaamasta käyrästä pisteessä, joka on määritetty kohtaan *Muutt=Piste*.

Varmista, että riippumatonta muuttujaa ei ole määritetty. Esimerkiksi, jos $f_1(x)=5$ and $x:=3$, tällöin

tangentLine(f1(x), x, 2) antaa vastauksen "epätosi".

tangentLine($x^2, x, 1$)	$2 \cdot x - 1$
tangentLine($(x-3)^2 - 4, x=3$)	-4
tangentLine($\frac{1}{x^3}, x=0$)	$x=0$
tangentLine($\sqrt{x^2-4}, x=2$)	undef
$x:=3$: tangentLine($x^2, x, 1$)	5

tanh()

Katalogi >

tanh(LausI) ⇒ lauseke

tanh(ListaI) ⇒ lista

tanh(LausI) laskee argumentin hyperbolisen tangentin lausekkeena.

tanh(ListaI) luo listan *Listai:n* jokaisen elementin hyperbolisista tangenteista.

tanh(neliömatriisiI) ⇒ 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:

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

-0.097966	0.933436	0.425972
0.488147	0.538881	-0.129382
1.28295	-1.03425	0.428817

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:

$$\begin{array}{l} \text{tanh}^{-1}(0) \\ \text{tanh}^{-1}(\{1,2,1,3\}) \end{array} \quad \begin{array}{l} 0 \\ \left\{ \text{undef}, 0.518046 - 1.5708 \cdot i, \frac{\ln(2)}{2} - \frac{\pi}{2} \cdot i \right\} \end{array}$$

Radiaanikulmatilassa ja suorakulmakompleksimuodossa:

$$\begin{array}{l} \text{tanh}^{-1} \begin{pmatrix} 1 & 5 & 3 \\ 4 & 2 & 1 \\ 6 & -2 & 1 \end{pmatrix} \\ \begin{bmatrix} -0.099353 + 0.164058 \cdot i & 0.267834 - 1.4908 \\ -0.087596 - 0.725533 \cdot i & 0.479679 - 0.94730 \\ 0.511463 - 2.08316 \cdot i & -0.878563 + 1.7901 \end{bmatrix} \end{array}$$

Jos haluat nähdä koko vastauksen, paina  ja siirrä sen jälkeen osoitinta painikkeilla  ja .**taylor(Laus I, Muutt, Aste[, Piste])**⇒*lauseke*Laskee pyydetyn Taylorin polynomin. Polynomi sisältää kokonaislukuasteiden ei-nollatermejä nollasta arvoon *Aste* saakka komennossa (*Muutt* miinus *Piste*). **taylor()** antaa vastauksena itsensä, jos tämän asteen tyypistettyjä potenssisarjoja ei ole, tai jos tässä vaadittaisiin negatiivisia tai murtolukueksponentteja. Käytä substituutiota ja/tai väliaikaista kertomista funktion (*Muutt* miinus *Piste*) potenssilla, kun haluat määrittää yleisempiä potenssisarjoja.*Piste* en oletusarvo on nolla, ja se on lavennuspiste.

$$\begin{array}{l} \text{taylor}(\sqrt{x}, x, 2) \\ \text{taylor}(e^t, t, 4) | t = \sqrt{x} \\ \text{taylor}\left(\frac{1}{x(x-1)}, x, 3\right) \\ \text{expand}\left(\frac{\text{taylor}\left(\frac{x}{x(x-1)}, x, 4\right)}{x}, x\right) \end{array} \quad \begin{array}{l} \text{taylor}(\sqrt{x}, x, 2, 0) \\ \frac{3}{24} + \frac{x^2}{6} + \frac{x}{2} + \sqrt{x} + 1 \\ \text{taylor}\left(\frac{1}{x(x-1)}, x, 3, 0\right) \\ -x^3 - x^2 - x - \frac{1}{x} - 1 \end{array}$$

tCdf(alaraja,yläraja,df)⇒luku, jos *alaraja* ja *yläraja* ovat lukuja, *lista*, jos *alaraja* ja *yläraja* ovat listoja

Laskee Studentin *t*-todennäköisyysjakauman *alarajan* ja *ylärajan* välillä määritetylle vapausasteelle *df*.

Kun $P(X \leq \text{yläraja})$, aseta *alaraja* = $-\infty$.

tCollect(Lausl)⇒*lauseke*

Laskee lausekkeen, jossa tulot ja sinien ja kosinien kokonaislukupotenssit on muunnettu useiden kulmien sinien ja kosinien lineaariseksi kombinaatioksi, kulmien summiksi ja kulmien erotuksiksi. Transformaatio muuntaa trigonometriset polynomit niiden harmonisten lineaariseksi kombinaatioksi.

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.

$$\frac{\text{tCollect}(\cos(\alpha)^2)}{\text{tCollect}(\sin(\alpha) \cdot \cos(\beta))} = \frac{\frac{\cos(2 \cdot \alpha) + 1}{2}}{\frac{\sin(\alpha - \beta) + \sin(\alpha + \beta)}{2}}$$

tExpand(Lausl)⇒*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

$$\frac{\text{tExpand}(\sin(3 \cdot \phi))}{\text{tExpand}(\cos(\alpha - \beta))} = \frac{4 \cdot \sin(\phi) \cdot (\cos(\phi))^2 - \sin(\phi)}{\cos(\alpha) \cdot \cos(\beta) + \sin(\alpha) \cdot \sin(\beta)}$$

transformaatiot. Joskus **tCollect()**-funktion soveltaminen **tExpand()**-funktion tulokseen, tai päin vastoin, kahdessa eri vaiheessa yksinkertaistaa lauseketta.

Huomaa: Astitilan skaalaus arvolla $\pi/180$ häiritsee **tExpand()**-funktion kykyä tunnistaa laajennettavia muotoja. Parhaan tuloksen saavuttamiseksi **tExpand()**-funktiota tulee käyttää radiaanikulmatilassa.

Text*kehotmerkkijono[, Näytälippu]*

Ohjelmointikomento: Keskeyttää ohjelman ja näyttää merkkijonon *kehotmerkkijono* valintaruudussa.

Kun käyttäjä valitsee **OK**-näppäimen, ohjelman suoritus jatkuu.


Valinnainen *lippu*-argumentti voi olla mikä tahansa lauseke.

- Jos *NäytäLippu* jätetään pois, tai jos se sievennyy arvoksi **1**, tekstimuotoinen viesti lisätään laskimen historiaan.
- Jos *NäytäLippu* sievennyy arvoon **0**, tekstimuotoista viestiä ei lisätä historiaan.

Jos ohjelma vaatii käyttäjän kirjoittaman vastauksen, katso **Request**, sivu **143**, tai **RequestStr**, sivu **144**.

Huomaa: Tätä komentoa voi käyttää käyttäjän määrittämän ohjelman sisällä mutta ei funktion sisällä.

Määritä ohjelma, joka keskeytyy ja näyttää kunkin viidestä satunnaisluvusta valintaruudussa.

Paina mallin Prgm...EndPrgm jokaisen rivin lopussa näppäintä  näppäimen **enter** sijaan. Tietokoneen näppäimistöillä **Alt**-näppäintä pidetään alhaalla ja painetaan **Enter**.

```
Define text_demo()=Prgm
```

```
For i,1,5
```

```
    strinfo:="Random number " & string  
(rand(i))
```

```
    Text strinfo
```

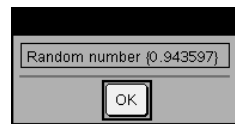
```
EndFor
```

```
EndPrgm
```

Suorita ohjelma:

```
text_demo()
```

Esimerkki yhdestä valintaruudusta:



TInterval

Katalogi > **TInterval** *Lista[,Frekv[,CTaso]]*

(Datalistan syöte)

TInterval $\bar{x}, Sx, n[,CTaso]$

(Yhteenvetotilaston syöte)

Laskee *t*-luottamusvälin. Tulosten yhteenvedo tallentuu *stat.results*-muuttujaan. (Katso sivu 169.)

Lisätietoja listassa olevien tyhjiin elementtien vaikutuksesta, katso Tyhjät elementitsivulla sivu 227.

Tulosmuuttuja	Kuvaus
stat.CLower, stat.CUpper	Tuntemattoman perusjoukon keskiarvon luottamusväli
stat. \bar{x}	Satunnaisesta normaali jakaumasta olevan datasekvenssin otoksen keskiarvo
stat.ME	Virhemarginaali
stat.df	Vapausasteet
stat. α	Otoksen keskihajonta
stat.n	Otoksen keskiarvon sisältävän datasekvenssin pituus

TInterval_2Samp

Katalogi > **TInterval_2Samp** *Lista1,Lista2[,Frekv1[,Frekv2[,CTaso[,Poolaus]]]]*

(Datalistan syöte)

TInterval_2Samp $\bar{x}1, Sx1, n1, \bar{x}2, Sx2, n2[,CTaso[,Poolaus]]$

(Yhteenvetotilaston syöte)

Laskee kahden otoksen *t*-luottamusvälin. Tulosten yhteenvedo tallentuu *stat.results*-muuttujaan. (Katso sivu 169.)

Poolaus=1 poolaa varianssit; *Poolaus=0* ei poolaa variansseja.

Lisätietoja listassa olevien tyhjiin elementtien vaikutuksesta, katso Tyhjät elementitsivulla sivu 227.

Tulosmuuttuja	Kuvaus
stat.CLower, stat.CUpper	Luottamusväli, joka sisältää jakauman luottamusvälin todennäköisyyden
stat.x1-x2	Satunnaisesta normaalijakaumasta olevien datasekvenssien otosten keskiarvot
stat.ME	Virhemarginaali
stat.df	Vapausasteet
stat.x1, stat.x2	Satunnaisesta normaalijakaumasta olevien datasekvenssien otosten keskiarvot
stat.alpha1, stat.alpha2	Lista 1:n ja Lista 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

Merkin Δ saat Katalogin symboleista.

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

Muuttaa *Laus*:n määrittämän lämpötila-alueen

Δ tmpCnv()



Katalogi > 

(kahden lämpötila-arvon välisen eron) yksiköstä toiseen. Kelpaavat lämpötilayksiköt ovat:

_°CCelsius
_°FFahrenheit
_°KKelvin
_°RRankine

Δ tmpCnv(100·_°C,_°F)	180·_°F
Δ tmpCnv(180·_°F,_°C)	100·_°C
Δ tmpCnv(100·_°C,_°K)	100·_°K
Δ tmpCnv(100·_°F,_°R)	100·_°R
Δ tmpCnv(1·_°C,_°F)	1.8·_°F

Voit syöttää merkin ° valitsemalla sen symbolipaletista tai kirjoittamalla e.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()**.

Huomaa: Lämpötilan yksiköt voi valita Katalogista.

tPdf()

Katalogi > 

tPdf(*XArvo*,*df*) ⇒ luku, jos *XArvo* on luku, lista, jos *XArvo* on lista

Laskee todennäköisyystiheysfunktion (pdf) Studentin *t*-jakaumalle määritetyllä *x*:n arvolla ja määritetyillä vapausasteilla *df*.

trace()

Katalogi > 

trace(neliömatriisi) ⇒ 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· <i>a</i>

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 233.

lohko1 ja *lohko2* voivat olla joko yksi lauseke tai sarja lausekkeita, jotka on erotettu toisistaan kaksoispisteellä (:).

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

Esimerkki 2

Jos halaut nähdä komentojen **Try**, **ClrErr** ja **PassErr** toiminnan, syötä oikealla näkyvä *eigenvals()*-ohjelma. Suorita ohjelma suorittamalla kukin seuraavista lausekkeista.

$$\text{eigenvals}\left(\begin{bmatrix} -3 \\ -41 \\ 5 \end{bmatrix}, \begin{bmatrix} -1 & 2 & -3.1 \end{bmatrix}\right)$$

$$\text{eigenvals}\left(\begin{bmatrix} 1 & 2 & 3 \end{bmatrix}, \begin{bmatrix} 1 \\ 2 \end{bmatrix}\right)$$

Huomaa: Katso myös **ClrErr**, sivu 29, ja **PassErr**, sivu 126.

```
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:",eigV(a*b)
```

Else

```
If errCode=230 Then
  Disp "Error: Product of A·B must be a square
matrix"
```

ClrErr

Else

PassErr

EndIf

EndTry

EndPrgm

tTest**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 169.)

Testaa $H_0: \mu = \mu_0$, jonkin seuraavan vaihtoehdon suhteen:

Kun $H_a: \mu < \mu_0$, aseta *Hypot*<0

Kun $H_a: \mu \neq \mu_0$ (oletus), aseta *Hypot*=0

Kun $H_a: \mu > \mu_0$, aseta *Hypot*>0

Lisätietoja listassa olevien tyhjien elementtien vaikutuksesta, katso Tyhjät elementtisivulla sivu 227.

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}	Listan sisältämän datasekvenssin otoksen keskiarvo
stat.sx	Datasekvenssin otoksen keskihajonta
stat.n	Otoksen koko

tTest_2Samp**tTest_2Samp** $\text{Lista1}, \text{Lista2}, \text{Frekv1}, \text{Frekv2}, \text{Hypot}$
[,Poolaus]]]

(Datalistan syöte)

tTest_2Samp $\bar{x}1, s_x1, n1, \bar{x}2, s_x2, n2, [\text{Hypot}, \text{Poolaus}]$

(Yhteenvetotilaston syöte)

Suurittaa kahden otoksen t -testin. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 169.)

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 tyhjien elementtien vaikutuksesta, katso Tyhjät elementitsivulla sivu 227.

Tulosmuuttuja	Kuvaus
stat.t	Keskiarvojen erotukselle laskettu vakio-ohjearvo
stat.PVal	Alin merkitsevyytaso, jolla nollahypoteesi voidaan hylätä
stat.df	t-tilaston vapausasteet
stat.x1, stat.x2	Listat 1:n ja Listat 2:n sisältämien datasekvenssien otosten keskiarvot
stat.sx1, stat.sx2	Listat 1:n ja Listat 2:n sisältämien datasekvenssien otosten keskihajonnat
stat.n1, stat.n2	Otosten koko
stat.sp	Poolattu keskihajonta. Laskettu, kun <i>Poolaus</i> =1

tvmFV()

$tvmFV(N, I, PV, Pmt, [PpY], [CpY], [PmtAt]) \Rightarrow arvo$

$tvmFV(120, 5, 0, -500, 12, 12)$ 77641.1

Talouselaskentafunktio, joka laskee rahan tulevan arvon.

Huomaa: TVM-funktioissa käytetyt argumentit on kuvattu TVM-argumenttien taulukossa, sivu 186. Katso myös **amortTbl()**, sivu 12.

tvmI()

$tvmI(N, PV, Pmt, FV, [PpY], [CpY], [PmtAt]) \Rightarrow arvo$

$tvmI(240, 100000, -1000, 0, 12, 12)$ 10.5241

tvm()Katalogi > 

Talouselaskentafunktio, joka laskee vuosikoron.

Huomaa: TVM-funktioissa käytetyt argumentit on kuvattu TVM-argumenttien taulukossa, sivu 186.
Katso myös **amortTbl()**, sivu 12.

tvmN()Katalogi > 

tvmN($I, PV, Pmt, FV, [PpY], [CpY], [PmtAft]$) $\Rightarrow arvo$

$tvmN(5, 0, -500, 77641, 12, 12)$ 120.

Talouselaskentafunktio, joka laskee maksuerien lukumäärän.

Huomaa: TVM-funktioissa käytetyt argumentit on kuvattu TVM-argumenttien taulukossa, sivu 186.
Katso myös **amortTbl()**, sivu 12.

tvmPmt()Katalogi > 

tvmPmt($N, I, PV, FV, [PpY], [CpY], [PmtAft]$) $\Rightarrow arvo$

$tvmPmt(60, 4, 30000, 0, 12, 12)$ -552.496

Talouselaskentafunktio, joka laskee jokaisen maksuerän määrän.

Huomaa: TVM-funktioissa käytetyt argumentit on kuvattu TVM-argumenttien taulukossa, sivu 186.
Katso myös **amortTbl()**, sivu 12.

tvmPV()Katalogi > 

tvmPV($N, I, Pmt, FV, [PpY], [CpY], [PmtAft]$) $\Rightarrow arvo$

$tvmPV(48, 4, -500, 30000, 12, 12)$ -3426.7

Talouselaskentafunktio, joka laskee nykyarvon.

Huomaa: TVM-funktioissa käytetyt argumentit on kuvattu TVM-argumenttien taulukossa, sivu 186.
Katso myös **amortTbl()**, sivu 12.

TVM-argumentti*	Kuvaus	Datatyyppi
N	Maksuerien lukumäärä	reaaliluku

TVM-argumentti*	Kuvaus	Datatyyppi
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 **tv.m.pv** ja **tv.m.pmt**), joita käytetään *Laskin*-sovelluksen talouslaskentatoiminnossa. Talouslaskentafunktioiden argumenttien arvot tai vastaukset eivät kuitenkaan tallennu TVM-muuttujiin.

TwoVar

Katalogi > 

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

Laskee kahden muuttujan tilastot. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 169.)

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

X ja Y ovat riippumattomien ja riippuvien muuttujien listoja.

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

Luokka on luokkakoodien lista vastaavalle X - ja Y -datalle.

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

Tyhjä elementti jossakin listassa X , *Frekv* tai *Luokka* saa aikaan, että kaikkien listojen vastaava elementti on tyhjä. Tyhjä elementti jossakin listassa $X1$ - $X20$ saa aikaan, että kaikkien listojen vastaava elementti on tyhjä. Lisätietoja tyhjästä elementeistä, katso sivu 227.

Tulosmuuttuja	Kuvaus
stat. \bar{x}	x:n arvojen keskiarvo
stat. x	x:n arvojen summa
stat. x2	x ² :n arvojen summa
stat.sx	x:n otoksen keskihajonta
stat. x	x:n perusjoukon keskihajonta
stat.n	Datapisteiden lukumäärä
stat. \bar{y}	y:n arvojen keskiarvo
stat. Σy	y:n arvojen summa
stat. Σy^2	y ² :n arvojen summa
stat.sy	y:n otoksen keskihajonta
stat. σy	y:n perusjoukon keskihajonta
stat. Σxy	x · y -arvojen summa
stat.r	Korrelaatiokerroin
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 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. $\Sigma(x-\bar{x})^2$	x:n keskiarvon poikkeamien neliöiden summa
stat. $\Sigma(y-\bar{y})^2$	y:n keskiarvon poikkeamien neliöiden summa

U

unitV()

Katalogi > **yksikköV(Vektori1)**⇒vektori

Laskee joko rivi- tai sarakkeyksikkövektorin riippuen *Vektori1*:n muodosta.

Vektori1 :n on oltava joko yksirivinen matriisi tai yksisarakeinen matriisi.

$$\text{unitV}\left(\begin{bmatrix} a & b & c \end{bmatrix}\right) = \begin{bmatrix} \frac{a}{\sqrt{a^2+b^2+c^2}} & \frac{b}{\sqrt{a^2+b^2+c^2}} & \frac{c}{\sqrt{a^2+b^2+c^2}} \end{bmatrix}$$

$$\text{unitV}\left(\begin{bmatrix} 1 & 2 & 1 \end{bmatrix}\right) = \begin{bmatrix} \frac{\sqrt{6}}{6} & \frac{\sqrt{6}}{3} & \frac{\sqrt{6}}{6} \end{bmatrix}$$

$$\text{unitV}\left(\begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}\right) = \begin{bmatrix} \frac{\sqrt{14}}{14} \\ \frac{14}{\sqrt{14}} \\ 7 \\ \frac{3\sqrt{14}}{14} \end{bmatrix}$$

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 102, ja **getLockInfo()**, sivu 80.

<code>a:=65</code>	65
<code>Lock a</code>	Done
<code>getLockInfo(a)</code>	1
<code>a:=75</code>	"Error: Variable is locked."
<code>DelVar a</code>	"Error: Variable is locked."
<code>Unlock a</code>	Done
<code>a:=75</code>	75
<code>DelVar a</code>	Done

V

varPop()

Katalogi > **varPop**(Lista[, frekvLista])⇒lauseke

Laskee *Listan* perusjoukon varianssin.

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

<code>varPop({5,10,15,20,25,30})</code>	875
	12
<code>Ans*1.</code>	72.9167

Huomaa: Listassa tulee olla vähintään kaksi elementtiä.

Jos jokin elementti jommassakummassa listassa on tyhjä, kyseistä elementtiä ei huomioida, eikä toisessa listassa olevaa vastaavaa elementtiä myöskään huomioida. Lisätietoja tyhjiä elementteistä, katso sivu 227.

varSamp()

varSamp(Lista[, frekvLista]) ⇒ lauseke

Laskee Listan otosten varianssin.

Jokainen frekvListan elementti näyttää Listan vastaavien elementtien peräkkäisten esiintymien lukumäärän.

Huomaa: Listassa tulee olla vähintään kaksi elementtiä.

Jos jokin elementti jommassakummassa listassa on tyhjä, kyseistä elementtiä ei huomioida, eikä toisessa listassa olevaa vastaavaa elementtiä myöskään huomioida. Lisätietoja tyhjiä elementteistä, katso sivu 227.

varSamp(MatriisiI[, frekvMatriisiI]) ⇒ matriisi

Laskee rivivektorin, joka sisältää MatriisiI:n kaikkien sarakkeiden otoksen varianssin.

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

Jos jokin elementti jommassakummassa matriisissa on tyhjä, kyseistä elementtiä ei huomioida, eikä toisessa matriisissa olevaa vastaavaa elementtiä myöskään huomioida. Lisätietoja tyhjiä elementteistä, katso sivu 227.

Huomaa: MatriisiI:ssä on oltava vähintään kaksi elementtiä.

$\text{varSamp}(\{a, b, c\})$

$$\frac{a^2 - a \cdot (b+c) + b^2 - b \cdot c + c^2}{3}$$

$\text{varSamp}(\{1, 2, 5, 6, 3, -2\})$

$$\frac{31}{2}$$

$\text{varSamp}(\{1, 3, 5\}, \{4, 6, 2\})$

$$\frac{68}{33}$$

$\text{varSamp}\left(\begin{bmatrix} 1 & 2 & 5 \\ -3 & 0 & 1 \\ .5 & .7 & 3 \end{bmatrix}\right)$ [4.75 1.03 4]

$\text{varSamp}\left(\begin{bmatrix} -1.1 & 2.2 \\ 3.4 & 5.1 \\ -2.3 & 4.3 \end{bmatrix}, \begin{bmatrix} 6 & 3 \\ 2 & 4 \\ 5 & 1 \end{bmatrix}\right)$ [3.91731 2.08411]

W

warnCodes ()

Katalogi > 


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 240.

	$\text{warnCodes}\left(\text{solve}\left(\sin(10 \cdot x) = \frac{x^2}{x}, x\right), \text{warn}\right)$ $x = -0.84232 \text{ or } x = -0.706817 \text{ or } x = -0.2852$
<i>warn</i>	{ 10007, 10009 }

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

when()

Katalogi > 

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.

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 piiryy vain jollekin välille.

when() on hyödyllinen komento rekursiivisten funktioiden määrittämisessä.

$\text{when}(x < 0, x + 3), x = 5$	undef
------------------------------------	-------

$\text{when}(n > 0, n \cdot \text{factorial}(n - 1), 1) \rightarrow \text{factorial}(n)$	<i>Done</i>
$\text{factorial}(3)$	6
3!	6

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 määritysten syöttämisestä löytyvät tuotteen ohjekirjan Laskin-osiosta.

Define $sum_of_recip(n) = \text{Func}$ Local $i, tempsum$ $1 \rightarrow i$ $0 \rightarrow tempsum$ While $i \leq n$ $tempsum + \frac{1}{i} \rightarrow tempsum$ $i + 1 \rightarrow i$

EndWhile

Return $tempsum$

EndFunc

Done

 $sum_of_recip(3)$ $\frac{11}{6}$ **X**

BooleanLaus1 **xor** *BooleanLaus2* antaa vastauksena *Boolean* lausekkeen *BooleanList1*

xor *BooleanList2* antaa vastauksena *Boolean* listan *BooleanMatriisi1*

xor *BooleanMatriisi2* antaa vastauksena *Boolean* matriisin

Määrittää totuusarvoksi tosi, jos *BooleanLaus1* on tosi ja *BooleanLaus2* on epätosi, tai päin vastoin.

Määrittää totuusarvoksi epätosi, jos kumpikin argumentti on tosi tai kumpikin on epätosi. Antaa vastauksena sievennetyn *Boolean* lausekkeen, jos kummankaan argumentin totuusarvoa ei voi määrittää todeksi tai epätodeksi.

Huomaa: Katso **or**, sivu 123.

Kokonaisluku1 **xor** *Kokonaisluku2* \Rightarrow *kokonaisluku*

Vertaa kahta reaalikokonaislukua bitti bitiltä **xor**-operaation avulla. Sisäisesti kumpikin kokonaisluku muunnetaan etumerkilliseksi, 64 bitin binaariluvuksi. Kun vastaavia bittejä verrataan, tulos on 1, jos jompikumpi bitti (mutta ei molemmat) on 1; tulos on 0, jos kumpikin bitti on 0 tai kumpikin bitti on 1. Laskettu

 $true \text{ xor } true$ false $5 > 3 \text{ xor } 3 > 5$ true

Heksadesimaalisessa kantalukutilassa:

Tärkeää: Nolla, ei O-kirjain.

 $0h7AC36 \text{ xor } 0h3D5F$ 0h79169

Binaarisessa kantalukutilassa:

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).

Jos syötät desimaalikokonaisluvun, joka on etumerkillisen, 64 bitin binaarimuodon lukualueen ulkopuolella, laskin käyttää symmetristä modulooperaatiota, jotta arvo saadaan oikealle alueelle.

Lisätietoja, katso ▶ **Base2**, sivu 22.

Huomaa: Katso or, sivu 123.

0b100101 xor 0b100

0b100001

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

Z

zeros()

zeros(*Laus*, *Muutt*)⇒*lista*

zeros(*Laus*, *Muutt*=*Arvaus*)⇒*lista*

Määrittää listan muuttujan *Muutt* mahdollisista reaalityökaluista, 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.

$$\text{zeros}\left(a \cdot x^2 + b \cdot x + c, x\right) = \left\{ \frac{\sqrt{b^2 - 4 \cdot a \cdot c} - b}{2 \cdot a}, \frac{-\left(\sqrt{b^2 - 4 \cdot a \cdot c} + b\right)}{2 \cdot a} \right\}$$

$$a \cdot x^2 + b \cdot x + c | x = \text{Ans}[2] \quad 0$$

$$\text{exact}\left(\text{zeros}\left(a \cdot \left(e^x + x\right) \cdot \left(\text{sign}(x) - 1\right), x\right)\right) \quad \left\{ \left\{ \right\} \right\}$$

$$\text{exact}\left(\text{solve}\left(a \cdot \left(e^x + x\right) \cdot \left(\text{sign}(x) - 1\right) = 0, x\right)\right)$$

$$e^x + x = 0 \text{ or } x > 0 \text{ or } a = 0$$

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.

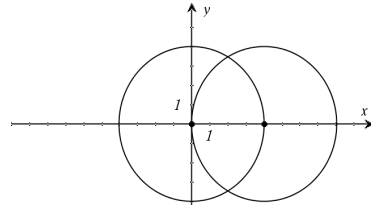
Oletetaan esimerkiksi, että origossa on ympyrän säde r , ja toinen ympyrän säde r on keskitetty kohtaan, jossa ensimmäinen ympyrä leikkaa positiivisen x -akselin. Määritä leikkauskohdat **zeros()**-funktion avulla.

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. Sylinterien nollakohdat havainnollistavat, miten nollakohtien sarjat voivat sisältää mielivaltaisia vakioita, jotka ovat muotoa c_k , jossa k on kokonaislukuliite väliltä 1-255.

Polynomisarjoissa laskutoimituksen suoritus aika tai muistin käyttö voivat riippua merkittävästi



$$\text{zeros}\left(\left\{x^2+y^2-r^2, (x-r)^2+y^2-r^2\right\}, \{x,y\}\right)$$

$$\begin{bmatrix} r & -\sqrt{3}\cdot r \\ 2 & 2 \\ r & \sqrt{3}\cdot r \\ 2 & 2 \end{bmatrix}$$

Määritä rivin 2 juuri:

$$\text{Ans}[2] \quad \begin{bmatrix} r & \sqrt{3}\cdot r \\ 2 & 2 \end{bmatrix}$$

$$\text{zeros}\left(\left\{x^2+y^2-r^2, (x-r)^2+y^2-r^2\right\}, \{x,y,z\}\right)$$

$$\begin{bmatrix} r & -\sqrt{3}\cdot r & c1 \\ 2 & 2 & \\ r & \sqrt{3}\cdot r & c1 \\ 2 & 2 & \end{bmatrix}$$

tuntemattomien muuttujien järjestyksestä. Jos ensimmäinen valintasi kuluttaa muistia, tai et jaksaa odottaa vastausta, yritä järjestää muuttujat uudelleen lausekkeisiin ja/tai *MuuttuTaiArvaus*-listaan.

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

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

Jokainen tuntematon muuttuja alkaa arvausarvostaan, mikäli se on määritetty; muussa tapauksessa se alkaa arvosta 0.0.

Arvausten avulla voit etsiä lisää nollakohtia yksi kerrallaan. Suppenemista varten arvauksen on mahdollisesti oltava melko lähellä nollakohtaa.

$$\text{zeros}\left(\left\{x+e^z \cdot y-1, x-y-\sin(z)\right\}, \{x, y\}\right)$$

$$\begin{bmatrix} e^z \cdot \sin(z)+1 & -(\sin(z)-1) \\ e^z+1 & e^z+1 \end{bmatrix}$$

$$\text{zeros}\left(\left\{e^z \cdot y-1, y-\sin(z)\right\}, \{y, z\}\right)$$

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 σ , *Lista*, *Frekv*, *CTaso*]

(Datalistan syöte)

zInterval σ , \bar{x} , *n* [, *CTaso*]

(Yhteenvetotilaston syöte)

Laskee z -luottamusvälin. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 169.)

Lisätietoja listassa olevien tyhjien elementtien vaikutuksesta, katso Tyhjät elementit sivulla 227.

Tulosmuuttuja	Kuvaus
stat.CLower, stat.CUpper	Tuntemattoman perusjoukon keskiarvon luottamusväli
stat. \bar{x}	Satunnaisesta normaalijakaumasta olevan datasekvenssin otoksen keskiarvo
stat.ME	Virhemarginaali
stat.sx	Otoksen keskihajonta
stat.n	Otoksen keskiarvon sisältävän datasekvenssin pituus
stat. 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 169.)

x on ei-negatiivinen kokonaisluku.

Lisätietoja listassa olevien tyhjiä elementtien vaikutuksesta, katso Tyhjät elementitsivulla sivu 227.

Tulosmuuttuja	Kuvaus
stat.CLower, stat.CUpper	Luottamusväli, joka sisältää jakauman luottamusvälin todennäköisyyden
stat. \hat{C}	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 169.)

$x1$ ja $x2$ ovat ei-negatiivisia kokonaislukuja.

Lisätietoja listassa olevien tyhjiä elementtien vaikutuksesta, katso Tyhjät elementitsivulla sivu 227.

Tulosmuuttuja	Kuvaus
stat.CLower, stat.CUpper	Luottamusväli, joka sisältää jakauman luottamusvälin todennäköisyyden
stat.ÇDiff	Osuuksien välinen laskettu erotus
stat.ME	Virhemarginaali
stat.Ç1	Arvio ensimmäisen näytteen osuudesta
stat.Ç2	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 169.)

Lisätietoja listassa olevien tyhjien elementtien vaikutuksesta, katso Tyhjät elementtisivulla sivu 227.

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. $\alpha x1$, stat. $\alpha x2$	<i>Lista 1:n</i> ja <i>Lista 2:n</i> 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 yhteenveito tallentuu *stat.results*-muuttujaan. (Katso sivu 169.)Testaa $H_0: \mu = \mu_0$, jonkin seuraavan vaihtoehdon suhteen:Kun $H_a: \mu < \mu_0$, aseta *Hypot*<0Kun $H_a: \mu \neq \mu_0$ (oletus), aseta *Hypot*=0Kun $H_a: \mu > \mu_0$, aseta *Hypot*>0

Lisätietoja listassa olevien tyhjien elementtien vaikutuksesta, katso Tyhjät elementitsivulla sivu 227.

Tulosmuuttuja	Kuvaus
stat.z	$(\bar{x} - \mu_0) / (\sigma / \sqrt{n})$
stat.P Value	Pienin todennäköisyys, jolla nollahypoteesi voidaan hylätä
stat. \bar{x}	Listan sisältämän datasekvenssin otoksen keskiarvo
stat.sx	Datasekvenssin otoksen keskihajonta. Lasketaan vain <i>Data</i> -syötteelle.
stat.n	Otoksen koko

zTest_1Prop $p_0, x, n[, Hypot]$ Laskee yhden osuuden z-testin. Tulosten yhteenveito tallentuu *stat.results*-muuttujaan. (Katso sivu 169.) x on ei-negatiivinen kokonaisluku.Testaa $H_0: p = p_0$ jonkin seuraavan vaihtoehdon suhteen:Kun $H_a: p > p_0$, aseta *Hypot*>0Kun $H_a: p \neq p_0$ (oletus), aseta *Hypot*=0Kun $H_a: p < p_0$, aseta *Hypot*<0

Lisätietoja listassa olevien tyhjien elementtien vaikutuksesta, katso Tyhjät elementitsivulla sivu 227.

Tulosmuuttuja	Kuvaus
stat.p0	Perusjoukon osuuden hypoteesiarvo
stat.z	Osuudelle laskettu vakio-ohjearvo
stat.PVal	Alin merkitsevyystaso, jolla nollahypoteesi voidaan hylätä
stat.Ç	Arvioitu otoksen osuus
stat.n	Otoksen koko

zTest_2Prop

Katalogi > 

zTest_2Prop $x1, n1, x2, n2[, Hypot]$

Laskee kahden osuuden z -testin. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 169.)

$x1$ 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 elementtisivulla sivu 227.

Tulosmuuttuja	Kuvaus
stat.z	Osuuksien erotukselle laskettu vakio-ohjearvo
stat.PVal	Alin merkitsevyystaso, jolla nollahypoteesi voidaan hylätä
stat.Ç1	Arvio ensimmäisen näytteen osuudesta
stat.Ç2	Arvio toisen näytteen osuudesta
stat.Ç	Poolattu arvio otoksen osuudesta
stat.n1, stat.n2	Kokeissa 1 ja 2 otettujen otosten lukumäärä

zTest_2Samp

Katalogi > 

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 169.)

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 tyhjien elementtien vaikutuksesta, katso Tyhjät elementitsivulla sivu 227.

Tulosmuuttuja	Kuvaus
stat.z	Keskiarvojen erotukselle laskettu vakio-ohjearvo
stat.PVal	Alin merkitsevyytaso, jolla nollahypoteesi voidaan hylätä
stat.x1, stat.x2	<i>Lista1:n</i> ja <i>Lista2:n</i> sisältämien datasekvenssien otosten keskiarvot
stat.sx1, stat.sx2	<i>Lista1:n</i> ja <i>Lista2:n</i> sisältämien datasekvenssien otosten keskihajonnat
stat.n1, stat.n2	Otosten koko

Symbolit

+ (yhteenlasku)

 painike

$Laus1 + Laus2 \Rightarrow lauseke$

Laskee kahden argumentin summan.

56	56
56+4	60
60+4	64
64+4	68
68+4	72

$Lista1 + Lista2 \Rightarrow lista$

$Matriisi2 \Rightarrow matriisi$

Määrittää listan (tai matriisin), joka sisältää $Lista1$:n ja $Lista2$:n (tai $Matriisi1$:n ja $Matriisi2$:n) vastaavien elementtien summat.

Argumenttien tulee olla mitoiltaan samanlaisia.

$\left\{ 22, \pi, \frac{\pi}{2} \right\} \rightarrow I1$	$\left\{ 22, \pi, \frac{\pi}{2} \right\}$
$\left\{ 10, 5, \frac{\pi}{2} \right\} \rightarrow I2$	$\left\{ 10, 5, \frac{\pi}{2} \right\}$
$I1+I2$	$\{ 32, \pi+5, \pi \}$
$Ans + \{ \pi, -5, -\pi \}$	$\{ \pi+32, \pi, 0 \}$
$\begin{bmatrix} a & b \\ c & d \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$	$\begin{bmatrix} a+1 & b \\ c & d+1 \end{bmatrix}$

$Laus + Lista1 \Rightarrow lista$

$Lista1 + Laus \Rightarrow lista$

Määrittää listan, joka sisältää $Laus$:n ja $Lista1$: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)

 painike

$Laus1 - Laus2 \Rightarrow lauseke$

Laskee $Laus1$ miinus $Laus2$.

6-2	4
$\pi - \frac{\pi}{6}$	$\frac{5 \cdot \pi}{6}$

-(vähennyslasku) painike $Lista1 - Lista2 \Rightarrow lista$

$$\left\{ 22, \pi, \frac{\pi}{2} \right\} - \left\{ 10, 5, \frac{\pi}{2} \right\} = \left\{ 12, \pi - 5, 0 \right\}$$

 $Matriisi1 - Matriisi2 \Rightarrow matriisi$

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

Vähentää $Lista2:n$ (tai $Matriisi2:n$) jokaisen elementin $Lista1:n$ (tai $Matriisi1:n$) vastaavasta elementistä ja antaa tuloksena vastaukset.

Argumenttien tulee olla mitoiltaan samanlaisia.

 $Laus - Lista1 \Rightarrow lista$

$$15 - \{10, 15, 20\} = \{5, 0, -5\}$$

 $Lista1 - Laus \Rightarrow lista$

$$\{10, 15, 20\} - 15 = \{-5, 0, 5\}$$

Vähentää jokaisen $Lista1:n$ elementin lausekkeesta $Laus$ tai vähentää lausekkeen $Laus$ jokaisesta $Lista1:n$ elementistä ja antaa vastauksena tuloslistan.

 $Laus - Matriisi1 \Rightarrow matriisi$

$$20 - \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} = \begin{bmatrix} 19 & -2 \\ -3 & 16 \end{bmatrix}$$

 $Matriisi1 - Laus \Rightarrow matriisi$

$Laus - Matriisi1$ laskee matriisin $Laus$ kertaa identtinen matriisi miinus $Matriisi1$. $Matriisi1:n$ on oltava neliö.

$Matriisi1 - Laus$ laskee lausekkeen $Laus$ matriisin kerrottuna identtisellä matriisilla, joka on vähennetty $Matriisi1:stä$. $Matriisi1:n$ on oltava neliö.

Huomaa: Käytä merkintää $-$ (piste miinus), kun haluat vähentää lausekkeen jokaisesta elementistä.

·(kertolasku) painike $Laus1 \cdot Laus2 \Rightarrow lauseke$

$$2 \cdot 3 \cdot 45 = 6 \cdot 9$$

Laskee kahden argumentin tulon.

$$x \cdot y \cdot x = x^2 \cdot y$$

 $Lista1 \cdot Lista2 \Rightarrow lista$

$$\{1, 2, 3\} \cdot \{4, 5, 6\} = \{4, 10, 18\}$$

Luo listan, joka sisältää $Lista1:n$ ja $Lista2:n$ vastaavien elementtien tulot.

$$\left\{ \frac{2}{a}, \frac{3}{2} \right\} \cdot \left\{ a^2, \frac{b}{3} \right\} = \left\{ 2 \cdot a, \frac{b}{2} \right\}$$

Listojen tulee olla mitoiltaan samanlaisia.

 $Matriisi1 \cdot Matriisi2 \Rightarrow matriisi$

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix} \cdot \begin{bmatrix} a & d \\ b & e \\ c & f \end{bmatrix}$$

Laskee $Matriisi1:n$ ja $Matriisi2:n$ matriisitulon.

$Matriisi1:n$ sarakkeiden lukumäärän on oltava sama kuin $Matriisi2:n$ rivien lukumäärä.

$$\begin{bmatrix} a+2 \cdot b+3 \cdot c & d+2 \cdot e+3 \cdot f \\ 4 \cdot a+5 \cdot b+6 \cdot c & 4 \cdot d+5 \cdot e+6 \cdot f \end{bmatrix}$$

· (kertolasku)**✕ painike** $Laus \cdot Lista \Rightarrow lista$

$$\pi \cdot \{4,5,6\}$$

$$\{4 \cdot \pi, 5 \cdot \pi, 6 \cdot \pi\}$$

 $Lista1 \cdot Laus \Rightarrow lista$

Määrittää listan, joka sisältää lausekkeen *Laus* ja kunkin *Lista1*:n elementin tulon.

 $Laus \cdot Matriisi1 \Rightarrow matriisi$

$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \cdot 0.01$	$\begin{bmatrix} 0.01 & 0.02 \\ 0.03 & 0.04 \end{bmatrix}$
-----------------------------------------------------------	------------------------------------------------------------

Määrittää matriisin, joka sisältää lausekkeen *Laus* ja kunkin *Matriisi1*:n elementin tulon.

$\lambda \cdot \text{identity}(3)$	$\begin{bmatrix} \lambda & 0 & 0 \\ 0 & \lambda & 0 \\ 0 & 0 & \lambda \end{bmatrix}$
------------------------------------	---------------------------------------------------------------------------------------

Huomaa: Käytä merkintää \cdot (piste kerro), kun haluat kertoa lausekkeen jokaisella elementillä.

/ (jakolasku)**÷ painike** $Laus1 / Laus2 \Rightarrow lauseke$

$\frac{2}{3.45}$.57971
------------------	--------

Laskee osamäärän *Laus1* jaettuna *Laus2*:lla.

Huomaa: Katso myös **Murtolukumalli**, sivu 5.

$\frac{x^3}{x}$	x^2
-----------------	-------

 $Lista1 / Lista2 \Rightarrow lista$

$\frac{\{1,2,3\}}{\{4,5,6\}}$	$\left\{0.25, \frac{2}{5}, \frac{1}{2}\right\}$
-------------------------------	-------------------------------------------------

Määrittää listan, joka sisältää osamäärät laskutoimituksista *Lista1* jaettuna *Lista2*:lla.

Listojen tulee olla mitoiltaan samanlaisia.

 $Laus / Lista1 \Rightarrow lista$

$\frac{a}{\{3,a,\sqrt{a}\}}$	$\left\{\frac{a}{3}, 1, \sqrt{a}\right\}$
------------------------------	-------------------------------------------

 $Lista1 / Laus \Rightarrow lista$

Määrittää listan, joka sisältää osamäärät laskutoimituksista *Laus* jaettuna *Lista1*:llä tai *Lista1* jaettuna lausekkeella *Laus*.

$\frac{\{a,b,c\}}{a \cdot b \cdot c}$	$\left\{\frac{1}{b \cdot c}, \frac{1}{a \cdot c}, \frac{1}{a \cdot b}\right\}$
---------------------------------------	--------------------------------------------------------------------------------

 $Matriisi1 / Laus \Rightarrow matriisi$

$\frac{\begin{bmatrix} a & b & c \end{bmatrix}}{a \cdot b \cdot c}$	$\begin{bmatrix} \frac{1}{b \cdot c} & \frac{1}{a \cdot c} & \frac{1}{a \cdot b} \end{bmatrix}$
---------------------------------------------------------------------	-------------------------------------------------------------------------------------------------

Laskee matriisin, joka sisältää osamäärät laskutoimituksesta *Matriisi1*/*Laus*.

Huomaa: Käytä merkintää $/$ (piste jaa), kun haluat jakaa lausekkeen jokaisella elementillä.

$Laus1 \wedge Laus2 \Rightarrow lauseke$

$$4^2 \qquad 16$$

$Lista1 \wedge Lista2 \Rightarrow lista$

$$\{a,2,c\} \{1,b,3\} \qquad \{a,2^b,c^3\}$$

Laskee ensimmäisen argumentin korotettuna toisen argumentin potenssiin.

Huomaa: Katso myös **Eksponenttimalli**, sivu 5.

Jos kyseessä on lista, laskee $Lista1$:n elementit korotettuna $Lista2$:n vastaavien elementtien potenssiin.

Reaalilukujen alueella murtolukupotenssit, joilla on supistetut eksponentit ja parittomat nimittäjät, käyttävät reaalista aluetta versus pääalue kompleksitilassa.

$Laus \wedge Lista1 \Rightarrow lista$

$$p \{a,2,-3\} \qquad \left\{ p^a, p^2, \frac{1}{p^3} \right\}$$

Laskee lausekkeen $Laus$ korotettuna $Lista1$:n elementtien potenssiin.

$Lista1 \wedge Laus \Rightarrow lista$

$$\{1,2,3,4\}^2 \qquad \left\{ 1, \frac{1}{4}, \frac{1}{9}, \frac{1}{16} \right\}$$

Laskee $Lista1$:n elementit korotettuna $Laus$:n potenssiin.

$neliömatriisi1 \wedge kokonaisluku \Rightarrow matriisi$

Laskee $neliömatriisi1$:n korotettuna kokonaisluvun $kokonaisluku$ potenssiin.

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}^2 \qquad \begin{bmatrix} 7 & 10 \\ 15 & 22 \end{bmatrix}$$

$neliömatriisi1$:n on oltava neliömatriisi.

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}^{-1} \qquad \begin{bmatrix} -2 & 1 \\ 3 & -1 \\ 2 & 2 \end{bmatrix}$$

Jos $kokonaisluku = -1$, laskee käänteismatriisin.

Jos $kokonaisluku < -1$, laskee käänteismatriisin korotettuna sopivaan positiiviseen potenssiin.

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}^{-2} \qquad \begin{bmatrix} 11 & -5 \\ 2 & 2 \\ -15 & 7 \\ 4 & 4 \end{bmatrix}$$

x² (neliö)**x² painike***Laus*l² ⇒ lauseke

4 ²	16
----------------	----

Laskee argumentin neliön.

{2,4,6} ²	{4,16,36}
----------------------	-----------

*Lista*l² ⇒ listaLaskee listan, joka sisältää *Lista*l:n elementtien neliöt.

<table border="1"> <tr><td>2</td><td>4</td><td>6</td></tr> <tr><td>3</td><td>5</td><td>7</td></tr> <tr><td>4</td><td>6</td><td>8</td></tr> </table> ²	2	4	6	3	5	7	4	6	8	<table border="1"> <tr><td>40</td><td>64</td><td>88</td></tr> <tr><td>49</td><td>79</td><td>109</td></tr> <tr><td>58</td><td>94</td><td>130</td></tr> </table>	40	64	88	49	79	109	58	94	130
2	4	6																	
3	5	7																	
4	6	8																	
40	64	88																	
49	79	109																	
58	94	130																	

*neliö*matriisi l² ⇒ matriisiLaskee *neliö*matriisi l:n matriisinelion. Tämä ei ole sama kuin kunkin elementin neliön laskeminen. Käytä merkintää .[^]2, kun haluat laskea jokaisen elementin neliön.

<table border="1"> <tr><td>2</td><td>4</td><td>6</td></tr> <tr><td>3</td><td>5</td><td>7</td></tr> <tr><td>4</td><td>6</td><td>8</td></tr> </table> . [^] 2	2	4	6	3	5	7	4	6	8	<table border="1"> <tr><td>4</td><td>16</td><td>36</td></tr> <tr><td>9</td><td>25</td><td>49</td></tr> <tr><td>16</td><td>36</td><td>64</td></tr> </table>	4	16	36	9	25	49	16	36	64
2	4	6																	
3	5	7																	
4	6	8																	
4	16	36																	
9	25	49																	
16	36	64																	

.+ (piste lisää)**.+ painikkeet***Matriisi*1 .+ *Matriisi*2 ⇒ matriisi

<table border="1"> <tr><td>a</td><td>2</td></tr> <tr><td>b</td><td>3</td></tr> </table> .+ <table border="1"> <tr><td>c</td><td>4</td></tr> <tr><td>5</td><td>d</td></tr> </table>	a	2	b	3	c	4	5	d	<table border="1"> <tr><td>a+c</td><td>6</td></tr> <tr><td>b+5</td><td>d+3</td></tr> </table>	a+c	6	b+5	d+3
a	2												
b	3												
c	4												
5	d												
a+c	6												
b+5	d+3												

Laus .+ *Matriisi*1 ⇒ matriisi*Matriisi*1 .+ *Matriisi*2 laskee matriisin, joka on*Matriisi*1:n ja *Matriisi*2:n vastaavien elementtiparien summa.

x .+ <table border="1"> <tr><td>c</td><td>4</td></tr> <tr><td>5</td><td>d</td></tr> </table>	c	4	5	d	<table border="1"> <tr><td>x+c</td><td>x+4</td></tr> <tr><td>x+5</td><td>x+d</td></tr> </table>	x+c	x+4	x+5	x+d
c	4								
5	d								
x+c	x+4								
x+5	x+d								

Laus .+ *Matriisi*1 laskee matriisin, joka on *Laus*:n ja kunkin *Matriisi*1:n elementin summa.**.- (piste-erotus)****.- painikkeet***Matriisi*1 .- *Matriisi*2 ⇒ matriisi

<table border="1"> <tr><td>a</td><td>2</td></tr> <tr><td>b</td><td>3</td></tr> </table> .- <table border="1"> <tr><td>c</td><td>4</td></tr> <tr><td>d</td><td>5</td></tr> </table>	a	2	b	3	c	4	d	5	<table border="1"> <tr><td>a-c</td><td>-2</td></tr> <tr><td>b-d</td><td>-2</td></tr> </table>	a-c	-2	b-d	-2
a	2												
b	3												
c	4												
d	5												
a-c	-2												
b-d	-2												

Laus .- *Matriisi*1 ⇒ matriisi*Matriisi*1 .- *Matriisi*2 laskee matriisin, joka on*Matriisi*1:n ja *Matriisi*2:n vastaavien elementtiparien välinen erotus.

x .- <table border="1"> <tr><td>c</td><td>4</td></tr> <tr><td>d</td><td>5</td></tr> </table>	c	4	d	5	<table border="1"> <tr><td>x-c</td><td>x-4</td></tr> <tr><td>x-d</td><td>x-5</td></tr> </table>	x-c	x-4	x-d	x-5
c	4								
d	5								
x-c	x-4								
x-d	x-5								

Laus .- *Matriisi*1 laskee matriisin, joka on *Laus*:n ja kunkin *Matriisi*1:n elementin erotus.

. (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*

$\begin{bmatrix} a & 2 \\ b & 3 \end{bmatrix}$.	/	$\begin{bmatrix} c & 4 \\ 5 & d \end{bmatrix}$		$\begin{bmatrix} \frac{a}{c} & \frac{1}{2} \\ \frac{b}{5} & \frac{3}{d} \end{bmatrix}$
------------------------------------------------	---	---	------------------------------------------------	--	----------------------------------------------------------------------------------------

Laus . / *Matriisi1* ⇒ *matriisi*

x	.	/	$\begin{bmatrix} c & 4 \\ 5 & d \end{bmatrix}$		$\begin{bmatrix} \frac{x}{c} & \frac{x}{4} \\ \frac{x}{5} & \frac{x}{d} \end{bmatrix}$
-----	---	---	------------------------------------------------	--	----------------------------------------------------------------------------------------

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*

$\begin{bmatrix} a & 2 \\ b & 3 \end{bmatrix}$.	^	$\begin{bmatrix} c & 4 \\ 5 & d \end{bmatrix}$		$\begin{bmatrix} a^c & 16 \\ b^5 & 3^d \end{bmatrix}$
------------------------------------------------	---	---	------------------------------------------------	--	-------------------------------------------------------

Laus . ^ *Matriisi1* ⇒ *matriisi*

x	.	^	$\begin{bmatrix} c & 4 \\ 5 & d \end{bmatrix}$		$\begin{bmatrix} x^c & x^4 \\ x^5 & x^d \end{bmatrix}$
-----	---	---	------------------------------------------------	--	--------------------------------------------------------

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 ⇒ lauseke

-2.43		-2.43
-------	--	-------

-Lista1 ⇒ lista

$-\{ -1, 0, 4, 1.2 \in \mathbb{R} \}$		$\{ 1, -0, 4, -1.2 \in \mathbb{R} \}$
---------------------------------------	--	---------------------------------------

-Matriisi1 ⇒ *matriisi*

$-a \cdot b$		$a \cdot b$
--------------	--	-------------

Laskee argumentin negaation.

- (negaatio)



Kun kyseessä on lista tai matriisi, määrittää kaikkien elementtien negaatiot.

Jos argumentti on binaarinen tai heksadesimaalinen kokonaisluku, negaatio antaa kakkosen komplementin.

Binaarisessa kantalukutilassa:

```
-0b100101
0b11111111111111111111111111111111▶
```

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

% (prosentti)



Laus1 % ⇒ lauseke

Lista1 % ⇒ lista

Matriisi1 % ⇒ 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)



Laus1 = Laus2 ⇒ Boolean lauseke

Lista1 = Lista2 ⇒ Boolean lista

Matriisi1 = Matriisi2 ⇒ Boolean matriisi

Antaa totuusarvon tosi, jos *Laus1* määritetään olevan yhtä kuin *Laus2*.

Antaa totuusarvon epätosi, jos *Laus1* määritetään olevan ei yhtä kuin *Laus2*.

Kaikissa muissa tapauksissa vastauksena on yhtälön sievennetty muoto.

Listojen ja matriisien kohdalla vastauksena on vertailuja elementti elementiltä.

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

Esimerkkifunktio, jossa on käytetty matematiikkatestitasymboleita: =, ≠, <, ≤, >, ≥

```
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
```

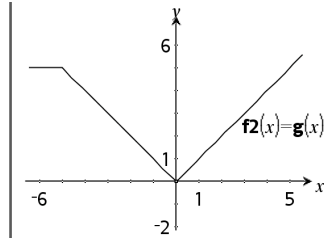
Done


Funktion g(x) kuvaajan piirtämisen tulos

= (on yhtä kuin)

 painike

osiosta.



 $f2(x) = g(x)$

≠ (ei yhtä kuin)

 painikkeet

$Laus1 \neq Laus2 \Rightarrow$ Boolean lauseke

Katso esimerkki kohdasta "=" (on yhtä kuin).

$Lista1 \neq Lista2 \Rightarrow$ 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.

Huomaa: Voit syöttää tämän operaattorin näppäimistöltä kirjoittamalla \neq

< (pienempi kuin)

 painikkeet

$Laus1 < Laus2 \Rightarrow$ Boolean lauseke

Katso esimerkki kohdasta "=" (on yhtä kuin).

$Lista1 < Lista2 \Rightarrow$ Boolean lista

$Matriisi1 < Matriisi2 \Rightarrow$ Boolean matriisi

Antaa totuusarvon tosi, jos $Laus1$ määritetään olevan pienempi kuin $Laus2$.

Antaa totuusarvon epätosi, jos $Laus1$ määritetään olevan suurempi tai yhtä suuri kuin $Laus2$.

< (pienempi kuin)

ctrl [=] painikkeet

Kaikissa muissa tapauksissa vastauksena on yhtälön sievennetty muoto.

Listojen ja matriisien kohdalla vastauksena on vertailuja elementti elementiltä.

≤ (pienempi tai yhtä suuri kuin)

ctrl [=] painikkeet

$Laus1 \leq Laus2 \Rightarrow$ Boolean lauseke

Katso esimerkki kohdasta "=" (on yhtä kuin).

$Lista1 \leq Lista2 \Rightarrow$ Boolean lista

$Matriisi1 \leq Matriisi2 \Rightarrow$ 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

$Laus1 > Laus2 \Rightarrow$ Boolean lauseke

Katso esimerkki kohdasta "=" (on yhtä kuin).

$Lista1 > Lista2 \Rightarrow$ Boolean lista

$Matriisi1 > Matriisi2 \Rightarrow$ 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** $Laus1 \geq Laus2 \Rightarrow$ Boolean lausek

Katso esimerkki kohdasta "=" (on yhtä kuin).

 $Lista1 \geq Lista2 \Rightarrow$ Boolean lista $Matriisi1 \geq Matriisi2 \Rightarrow$ Boolean matriisiAntaa 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 \geq **\Rightarrow (looginen seuraus)****ctrl [=] näppäimet** $BooleanLaus1 \Rightarrow BooleanLaus2$ antaa vastauksena Boolean lausekkeen $5 > 3$ or $3 > 5$ true $BooleanList1 \Rightarrow BooleanList2$ antaa vastauksena Boolean listan $5 > 3 \Rightarrow 3 > 5$ false 3 or 4 7 $BooleanMatriisi1 \Rightarrow BooleanMatriisi2$ antaa vastauksena Boolean matriisin $3 \Rightarrow 4$ -4 $\{1, 2, 3\}$ or $\{3, 2, 1\}$ $\{3, 2, 3\}$ $Kokonaisluku1 \Rightarrow Kokonaisluku2$ antaa vastauksena kokonaisluvun $\{1, 2, 3\} \Rightarrow \{3, 2, 1\}$ $\{-1, -1, -3\}$

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 \Rightarrow

↔ (looginen kaksoisseuraus, XNOR)**ctrl [=] näppäimet***BooleanLaus1* ↔ *BooleanLaus2* antaa vastauksena
Boolean lausekkeen $5 > 3 \text{ xor } 3 > 5$ true $5 > 3 \Leftrightarrow 3 > 5$ false*BooleanList1* ↔ *BooleanList2* antaa vastauksena
Boolean listan $3 \text{ xor } 4$ 7 $3 \Leftrightarrow 4$ -8*BooleanMatriisi1* ↔ *BooleanMatriisi2* antaa
vastauksena *Boolean matriisin* $\{1,2,3\} \text{ xor } \{3,2,1\}$ $\{2,0,2\}$ $\{1,2,3\} \Leftrightarrow \{3,2,1\}$ $\{-3,-1,-3\}$ *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***Laus1* ⇒ *lauseke* $5!$ 120*Lista1* ⇒ *lista* $\{\{5,4,3\}\}!$ $\{120,24,6\}$ *Matriisi1* ⇒ *matriisi* $\begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}!$ $\begin{bmatrix} 1 & 2 \\ 6 & 24 \end{bmatrix}$

Määrittää argumentin kertoman.

Jos kyseessä on lista tai matriisi, määrittää listan tai matriisin elementtien kertomista.

& (liitä)**ctrl [&] painikkeet***Merkkijono1* & *Merkkijono2* ⇒ *merkkijono* $\text{"Hello " \& "Nick"}$ "Hello Nick"

Antaa vastauksena tekstimerkkijonon, joka on *Merkkijono2* liitettyinä *Merkkijono1*:een.

d() (derivaatta)Katalogi >  $d(\text{LausI}, \text{Muutt}[, \text{Aste}]) \Rightarrow \text{lauseke}$ $d(\text{ListaI}, \text{Muutt}[, \text{Aste}]) \Rightarrow \text{lista}$ $d(\text{MatriisiI}, \text{Muutt}[, \text{Aste}]) \Rightarrow \text{matriisi}$

Laskee ensimmäisen argumentin ensimmäisen derivaatan muuttujan *Muutt* suhteen.

Asteen, mikäli se otetaan mukaan, on oltava kokonaisluku. Jos aste on pienempi kuin nolla, vastaus on antiderivaatta.

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

d() ei noudata normaalia sievennysmekanismia, jossa argumentit sievennetään täydellisesti, minkä jälkeen sovitetaan funktion määritelmää näihin täydellisesti sievennetyihin argumentteihin. Sen sijaan funktio **d()** suorittaa seuraavat vaiheet:

1. Sievennä toista argumenttia vain sen verran, että sen tuloksena ei ole ei-muuttuja.
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.

$$\frac{d}{dx}(f(x) \cdot g(x)) = \frac{d}{dx}(f(x)) \cdot g(x) + \frac{d}{dx}(g(x)) \cdot f(x)$$

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

$$\frac{d}{dx} \left\{ x^2, x^3, x^4 \right\} = \left\{ 2 \cdot x, 3 \cdot x^2, 4 \cdot x^3 \right\}$$

∫() (integraali)Katalogi >  $∫(\text{LausI}, \text{Muutt}[, \text{Ala}, \text{Ylä}]) \Rightarrow \text{lauseke}$ $∫(\text{LausI}, \text{Muutt}[, \text{Vakio}]) \Rightarrow \text{lauseke}$

Laskee *LausI*:n integraalin muuttujan *Muutt* suhteen välillä *Ala* - *Ylä* olevista arvoista.

Huomaa: Katso myös **Määrittyn** tai

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

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*.

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

$$\int (a \cdot x^2, x, c)$$

$$\frac{a \cdot x^3}{3} + c$$

Samanarvoisesti kelpaavat antiderivaatat voivat erota numeerisen vakion suhteen. Tällainen vakio on olla naamioitunut – erityisesti, kun antiderivaatta sisältää logaritmeja tai käänteisiä trigonometrisia funktioita. Lisäksi joskus lisätään paloittain määritettyjä vakiolausekkeita, jotta antiderivaatasta saadaan kelvollinen suuremmalle välille kuin tavanomainen kaava.

∫() antaa vastauksena itsensä niille *Laus1*:n paloille, joita se ei pysty määrittämään sisäänrakennettujen funktioidensa ja operaattoreidensa eksplisiittisenä äärellisenä kombinaationa.

$$\int b \cdot e^{-x^2} + \frac{a}{x^2 + a^2} dx \quad b \cdot \int e^{-x^2} dx + \tan^{-1}\left(\frac{x}{a}\right)$$

Kun annat argumentit *Ala* ja *Ylä*, laskin yrittää paikallistaa mahdolliset epäjatkuvuudet tai 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.

Huom: Vastauksen pakottaminen likimääräiseksi:

Kämmenlaite: Paina .

Windows®: Paina **Ctrl+Enter**.

Macintosh®: Paina **⌘+Enter**.

iPad®: Pidä **enter** ja valitse .

$$\int_{-1}^1 e^{-x^2} dx \quad 1.49365$$

∫()-komentoa voidaan sijoittaa sisäkkäin usean integraalin suorittamiseksi. Integroinnin raja-arvot voivat riippua niiden ulkopuolella olevista integrointimuuttujista.

Huomaa: Katso myös **nInt()**, sivu 117.

$$\int_0^a \int_0^x \ln(x+y) dy dx$$

$$\frac{a^2 \cdot \ln(a)}{2} + \frac{a^2 \cdot (4 \cdot \ln(2) - 3)}{4}$$

√(Laus1) ⇒ lauseke

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

√(Lista1) ⇒ lista

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

Laskee argumentin neliöjuuren.

Kun kyseessä on lista, laskee kaikkien Lista1:n elementtien neliöjuuret.

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

Huomaa: Katso myös **Neliöjuurimalli**, sivu 5.

Π(Laus1, Muutt, Matala, Korkea) ⇒ lauseke

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

Sieventää Laus1:n jokaisen Muutt:n arvon suhteen väliltä Matala - Korkea ja laskee vastausten tulon.

Huomaa: Katso myös **Kertolaskumalli (Π)**, sivu 10.

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

$$\prod_{k=1}^n (k^2) \quad (n!)^2$$

$$\prod_{n=1}^5 \left\{ \left\{ \frac{1}{n}, n, 2 \right\} \right\} \quad \left\{ \frac{1}{120}, 120, 32 \right\}$$

$\Pi()$ (tulo)Katalogi >  $\Pi(Laus\ I, Muutt, Matala, Matala-1) \Rightarrow 1$

$$\prod_{k=4}^3 (k) \quad 1$$

 $\Pi(Laus\ I, Muutt, Matala, Korkea)$ $\Rightarrow \mathbf{1}\Pi(Laus\ I, Muutt, Korkea+1, Matala-1)$, jos $Korkea < Matala-1$

Esimerkkien kertolaskukaavat on otettu seuraavasta viitteestä:

Ronald L. Graham, Donald E. Knuth, and Oren Patashnik. *Concrete Mathematics: A Foundation for Computer Science*. Reading, Massachusetts: Addison-Wesley, 1994.

$$\prod_{k=4}^1 \left(\frac{1}{k}\right) \quad 6$$

$$\prod_{k=4}^1 \left(\frac{1}{k}\right) \cdot \prod_{k=2}^4 \left(\frac{1}{k}\right) \quad \frac{1}{4}$$

 $\Sigma()$ (summa)Katalogi >  $\Sigma(Laus\ I, Muutt, Matala, Korkea) \Rightarrow$ 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.

$$\sum_{n=1}^5 \left(\frac{1}{n}\right) \quad \frac{137}{60}$$

$$\sum_{k=1}^n (k^2) \quad \frac{n \cdot (n+1) \cdot (2 \cdot n+1)}{6}$$

$$\sum_{n=1}^{\infty} \left(\frac{1}{n^2}\right) \quad \frac{\pi^2}{6}$$

 $\Sigma(Laus\ I, Muutt, Matala, Matala-1) \Rightarrow 0$

$$\sum_{k=4}^3 (k) \quad 0$$

 $\Sigma(Laus\ I, Muutt, Matala, Korkea)$ $\Rightarrow \Sigma(Laus\ I, Muutt, Korkea+1, Matala-1)$, jos $Korkea < Matala-1$

Esimerkkien yhteenlaskukaavat on otettu seuraavasta viitteestä:

Ronald L. Graham, Donald E. Knuth, and Oren Patashnik. *Concrete Mathematics: A Foundation*

$$\sum_{k=4}^1 (k) \quad -5$$

$$\sum_{k=4}^1 (k) + \sum_{k=2}^4 (k) \quad 4$$

for Computer Science. Reading, Massachusetts:
Addison-Wesley, 1994.

 $\Sigma\text{Int}()$

$\Sigma\text{Int}(NPmt1, NPmt2, N, I, PV, [Pmt], [FV], [PpY], [CpY], [PmtAt], [pyörArvo]) \Rightarrow arvo$

$\Sigma\text{Int}(1,3,12,4.75,20000,,12,12)$ -213.48

$\Sigma\text{Int}(NPmt1, NPmt2, amortTable) \Rightarrow arvo$

Lyhennystoiminto, joka laskee koron summan määritetyn maksueräjakson ajalta.

$NPmt1$ ja $NPmt2$ määrittävät maksujakson alku- ja loppurajat.

$N, I, PV, Pmt, FV, PpY, CpY$ ja $PmtAt$ on kuvattu TVM-argumenttien taulukossa, sivu 186.

- 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-funktiolla.

$pyörArvo$ 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 $\mathbf{amortTbl}()$ kuvatus muotoinen, katso sivu 12.

Huomaa: Katso myös $\Sigma\text{Pm}()$, 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

 $\Sigma\text{Pm}()$

$\Sigma\text{Pm}(NPmt1, NPmt2, N, I, PV, [Pmt], [FV], [PpY], [CpY], [PmtAt], [pyörArvo]) \Rightarrow arvo$

$\Sigma\text{Pm}(1,3,12,4.75,20000,,12,12)$ -4916.28

$\Sigma\text{Pm}(NPmt1, NPmt2, amortTable) \Rightarrow arvo$

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 186.

- 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 $PmtAt$ oletusarvat ovat samat kuin TVM-funktiolla.

$pyörArvo$ määrittää pyöristyksessä käytettävien desimaalien määrän. Oletusarvo=2.

$\Sigma Pm(NPmt1, NPmt2, amortTable)$ laskee pääoman summan lyhennystaulukon $amortTable$ perusteella. $amortTable$ -argumentin on oltava matriisi, joka on kohdassa **amortTbl()** kuvatus 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

$\Sigma Pm(1,3,tbl)$ -4916.28

(epäsuora operaattori)

  **painikkeet**

muuttNimiMerkkijono

Viittaa muuttujaan, jonka nimi on $muuttNimiMerkkijono$. Tällä operaattorilla voit luoda muuttujanimiä funktion sisältä merkkijonojen avulla.

$\#("x"&"y"&"z")$ xyz

Luo muuttujan xyz tai viittaa siihen.

$10 \rightarrow r$ 10

"r" $\rightarrow s1$ "r"

$\#s1$ 10

Laskee arvon muuttujalle (r), jonka nimi on tallennettu muuttujaan s1.

E (kymmenpotenssimuoto) **painike***Laus*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 näppäimistöltä kirjoittamalla @E. Kirjoita esimerkiksi 2.3@E4, kun haluat syöttää 2.3E4.

g (graadi) **painike***Laus*g⇒*lauseke*

Aste-, graadi- tai radiaanikulmatilassa.

*Lista*g⇒*lista*

$$\cos(50^g) \quad \frac{\sqrt{2}}{2}$$
*Matriisi*g⇒*matriisi*

$$\cos(\{0, 100^g, 200^g\}) \quad \{1, 0, -1\}$$

Tämän funktion avulla voit määrittää graadikulman ollessasi aste- tai radiaanikulmatilassa.

Kun laskin on radiaanikulmatilassa, kertoo *Laus*g:n arvolla π/200.

Kun laskin on astekulmatilassa, kertoo *Laus*g:n arvolla g/100.

Graadikulmatilassa antaa vastauksena lausekkeen *Laus*g muuttumattomana.

Huomaa: Voit syöttää tämän symbolin tietokoneen näppäimistöltä kirjoittamalla @g.

r (radiaani) **painike***Laus*r⇒*lauseke*

Aste-, graadi- tai radiaanikulmatilassa:

*Lista*r⇒*lista*

$$\cos\left(\frac{\pi}{4^r}\right) \quad \frac{\sqrt{2}}{2}$$
*Matriisi*r⇒*matriisi*

$$\cos\left(\left\{0^r, \frac{\pi}{12} \cdot r, (\pi)^r\right\}\right) \quad \left\{1, \frac{(\sqrt{3+1}) \cdot \sqrt{2}}{4}, -1\right\}$$

Tämän funktion avulla voit määrittää radiaanikulman ollessasi aste- tai graadikulmatilassa.

Kun laskin on astekulmatilassa, kertoo argumentin arvolla 180/π.

r (radiaani)

Radiaanikulmatilassa antaa vastauksena argumentin muuttumattomana.

Kun laskin on graadikulmatilassa, kertoo argumentin arvolla $200/\pi$.

Vinkki: Käytä komentoa r , jos haluat pakottaa funktion määritelmän yksiköksi radiaanit riippumatta tilasta, joka on käytössä funktion käytön aikana.

Huomaa: Voit syöttää tämän symbolin tietokoneen näppäimistöltä kirjoittamalla @r.

° (aste)

Lausl°⇒lauseke

Listal°⇒lista

Matriisil°⇒matriisi

Tämän funktion avulla voit määrittää astekulman ollessasi graadi- tai radiaanikulmatilassa.

Kun laskin on radiaanikulmatilassa, kertoo argumentin arvolla $\pi/180$.

Astekulmatilassa antaa vastauksena argumentin muuttumattomana.

Kun laskin on graadikulmatilassa, kertoo argumentin arvolla 10/9.

Huomaa: Voit syöttää tämän symbolin tietokoneen näppäimistöltä kirjoittamalla @d.

Aste-, graadi- tai radiaanikulmatilassa:

$$\cos(45^\circ) \quad \frac{\sqrt{2}}{2}$$

Radiaanikulmatilassa:

Huom: Vastauksen pakottaminen likimääräiseksi:

Kämmenlaite: Paina **ctrl** **enter**.

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)

dd°mm'ss.ss"⇒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

Astekulmatilassa:

$$\begin{array}{r} 25^\circ 13' 17.5'' \quad 25.2215 \\ 25^\circ 30' \quad \frac{51}{2} \end{array}$$

asteina/minuutteina/sekunteina nykyisestä kulmatilasta riippumatta.

- Syöttää kellonajan tunteina/minuutteina/sekunteina.

Huomaa: Merkitse sekuntien ss.ss perään kaksi heittomerkkiä ("), ei lainausmerkkiä (").

∠ (kulma)

[Säde,∠θ_Kulma]⇒vektori
(polaarinen syöte)

[Säde,∠θ_Kulma,Z_Koordinaatti]⇒vektori
(lieriömäinen syöte)

[Säde,∠θ_Kulma,∠θ_Kulma]⇒vektori
(pallonmuotoinen syöte)

Laskee koordinaatit vektorina riippuen vektorimuotoilan asetuksesta: suorakulma, sylinteri tai pallo.

Huomaa: Voit syöttää tämän symbolin tietokoneen näppäimistöltä kirjoittamalla @<.

(Magnitudi ∠ Kulma)⇒kompleksiArvo
(polaarinen syöte)

Syöttää kompleksilukuarvon ($r∠\theta$) polaarisisästä muodossa. *Kulma* tulkitaan nykyisen kulmatilasetuksen mukaisesti.

Radiaanikulmatilassa ja kun vektorimuoto on asetettu valintaan:

suorakulma

$$\left[5 \quad \angle 60^\circ \quad \angle 45^\circ \right] \left[\frac{5 \cdot \sqrt{2}}{4} \quad \frac{5 \cdot \sqrt{6}}{4} \quad \frac{5 \cdot \sqrt{2}}{2} \right]$$

sylinteri

$$\left[5 \quad \angle 60^\circ \quad \angle 45^\circ \right] \left[\frac{5 \cdot \sqrt{2}}{2} \quad \angle \frac{\pi}{3} \quad \frac{5 \cdot \sqrt{2}}{2} \right]$$



pallo

$$\left[5 \quad \angle 60^\circ \quad \angle 45^\circ \right] \left[5 \quad \angle \frac{\pi}{3} \quad \angle \frac{\pi}{4} \right]$$

Radiaanikulmatilassa ja suorakulmakompleksimuodossa:

$$5+3 \cdot i \cdot \left(10 \angle \frac{\pi}{4} \right) \quad 5-5 \cdot \sqrt{2} + (3-5 \cdot \sqrt{2}) \cdot i$$

Huom: Vastauksen pakottaminen likimääräiseksi:

Kämmenlaite: Paina  .

Windows®: Paina **Ctrl+Enter**.

Macintosh®: Paina **⌘+Enter**.

iPad®: Pidä **enter** ja valitse .

$$5+3 \cdot i \cdot \left(10 \angle \frac{\pi}{4} \right) \quad -2.07107-4.07107 \cdot i$$

' (jaoton)

[?] painike

muuttuja'
muuttuja''

Syöttää jaottoman symbolin differentiaaliyhtälöön.
Yksi jaottoman symboli tarkoittaa 1. asteen differentiaaliyhtälöä, kaksi jaottoman symbolia tarkoittaa 2. asteen differentiaaliyhtälöä jne.

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

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

_ (alaviiva edustaa tyhjää elementtiä)

KatsoTyhjät elementit, sivu 227.

_ (alaviiva edustaa yksikön nimeä)

[ctrl] [] painikkeet

Laus_Yksikkö

Määrittää lausekkeen *Laus* yksiköt. Kaikkien yksiköiden nimien alussa on oltava alaviiva.

Voit käyttää valmiiksi määritettyjä yksiköitä tai luoda omia yksiköitä. Valmiiksi määritetyt yksiköt on esitetty vakioiden ja mittausyksiköiden moduulissa. Voit valita yksiköiden nimet Katalogista tai kirjoittaa ne suoraan.

Muuttuja_

Kun *Muuttujalla* ei ole arvoa, sitä käsitellään ikään kuin se edustaisi kompleksilukua. Oletusarvoisesti, ilman merkkiä *_*, muuttujaa käsitellään reaali-lukuna.

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()**.

3·_m▶_ft

9.84252·_ft

Huomaa: Muunnoksen symboli ▶ löytyy Katalogista.

Napsauta ensin $\int \Sigma$ ja sen jälkeen **Matemaattiset operaattorit**.

Olettaen, että *z* on määrittämätön:

$\text{real}(z)$	<i>z</i>
$\text{real}(z_)$	$\text{real}(z_)$
$\text{imag}(z)$	0
$\text{imag}(z_)$	$\text{imag}(z_)$

▶ (muunna)

[ctrl] [] 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^()

10^ (Lausl) ⇒ lauseke

$$10^{1.5} \quad 31.6228$$

10^ (Listal) ⇒ lista

$$10^{\{0, 2, 2, a\}} \quad \left\{ 1, \frac{1}{100}, 100, 10^a \right\}$$

Laskee luvun 10 korotettuna argumentin potenssiin.

Jos kyseessä on lista, laskee luvun 10 korotettuna Listal:n elementtien potenssiin.

10^ (neliömatriisi l) ⇒ neliömatriisi

Laskee luvun 10 korotettuna neliömatriisi l:n potenssiin. Tämä ei ole sama kuin kunkin elementin 10-potenssiin korottamisen laskeminen.

Laskentamenetelmä on kuvattu kohdassa **cos()**.

neliömatriisi l:n on oltava diagonalisoitavissa.

Vastaus sisältää aina liukulukuja.

$$10^{\begin{bmatrix} 1 & 5 & 3 \\ 4 & 2 & 1 \\ 6 & -2 & 1 \end{bmatrix}} \quad \begin{bmatrix} 1.14336\text{E}7 & 8.17155\text{E}6 & 6.67589\text{E}6 \\ 9.95651\text{E}6 & 7.11587\text{E}6 & 5.81342\text{E}6 \\ 7.65298\text{E}6 & 5.46952\text{E}6 & 4.46845\text{E}6 \end{bmatrix}$$

Laus | ^-1=>lauseke

$$(3.1)^{-1} \quad 0.322581$$

List | ^-1=>lista

$$\{a, 4, 0.1, x, -2\}^{-1} \quad \left\{ \frac{1}{a}, \frac{1}{4}, -10, \frac{1}{x}, \frac{-1}{2} \right\}$$

Laskee argumentin käänteisluvun.

Jos kyseessä on lista, laskee *List*:n elementtien käänteisluvut.

neliömatrissi | ^-1=>neliömatrissi

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}^{-1} \quad \begin{bmatrix} -2 & 1 \\ 3 & -1 \\ 2 & 2 \end{bmatrix}$$

Laskee *neliömatrissi*:n käänteisluvun.

neliömatrissi:n oltava ei-singulaarinen neliömatrissi.

$$\begin{bmatrix} 1 & 2 \\ a & 4 \end{bmatrix}^{-1} \quad \begin{bmatrix} -2 & 1 \\ a-2 & a-2 \\ a & -1 \\ 2 \cdot (a-2) & 2 \cdot (a-2) \end{bmatrix}$$

| (rajoittava operaattori)

Laus | **|** *Booleanlaus* | **|** *andBooleanlaus* | ...

$$x+1|x=3 \quad 4$$

Laus | **|** *Booleanlaus* | **|** *orBooleanlaus* | ...

$$x+y|x=\sin(y) \quad \sin(y)+y$$

Rajoittava ("|")-symboli toimii binaarisena operaattorina. Operaattorin | vasemmalla puolella oleva operandi on lauseke. Operaattorin | oikealla puolella oleva operandi määrittää yhden tai useampia suhteita, joiden tarkoitus on vaikuttaa lausekkeen sieventämiseen. Operaattorin | jäljessä olevat useat suhteet on yhdistettävä loogisilla operaattoreilla "and" tai "or".

$$x+y|\sin(y)=x \quad x+y$$

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*.

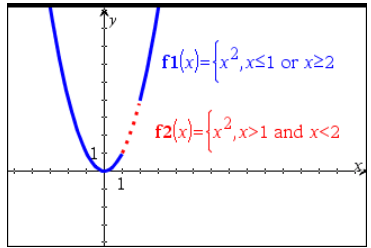
$$x^3-2 \cdot x+7 \rightarrow f(x) \quad Done$$

$$f(x)|x=\sqrt{3} \quad \sqrt{3+7}$$

$$(\sin(x))^2+2 \cdot \sin(x)-6|\sin(x)=d \quad d^2+2 \cdot d-6$$

Väljen rajoitusten muoto on yksi tai useampia epäyhtälöitä, jotka on yhdistetty loogisilla operaattoreilla "and" or "or". Väljen rajoitusehdot sallivat myös sievennyksen, joka saattaisi muuten olla kelpaamaton tai ei laskettavissa.

$\text{solve}(x^2-1=0,x) x>0 \text{ and } x<2$	$x=1$
$\sqrt{x} \cdot \frac{1}{x} x>0$	1
$\sqrt{x} \cdot \frac{1}{x}$	$\frac{1}{x} \cdot \sqrt{x}$



Pois sulkemisessa käytetään suhdeoperaattoria "ei ole yhtä kuin" (\neq tai \neq), jonka avulla jokin tietty arvo suljetaan pois. Näitä käytetään pääasiallisesti täsmällisen ratkaisun poissulkemiseen käytettäessä funktioita **cSolve()**, **cZeros()**, **fMax()**, **fMin()**, **solve()**, **zeros()**, jne.

$\text{solve}(x^2-1=0,x) x\neq 1$	$x=-1$
-----------------------------------	--------

→ (tallenna)

Laus → *Muutt*

Lista → *Muutt*

Matriisi → *Muutt*

Laus → *Funktio*(*Param1*,...)

Lista → *Funktio*(*Param1*,...) *Matriisi* → *Funktio*(*Param1*,...)

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

$\frac{\pi}{4} \rightarrow \text{myvar}$	$\frac{\pi}{4}$
$2 \cdot \cos(x) \rightarrow y1(x)$	<i>Done</i>
$\{1,2,3,4\} \rightarrow \text{lst5}$	$\{1,2,3,4\}$
$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix} \rightarrow \text{matg}$	$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}$
<i>"Hello"</i> → <i>str1</i>	<i>"Hello"</i>

edelleen.

Huomaa: Voit syöttää tämän operaattorin näppäimistöltä kirjoittamalla = : pikavalintana. Kirjoita esimerkiksi `pi/4 =: myvar.`

:= (määritä)

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.

<code>myvar:=</code>	$\frac{\pi}{4}$	<code></code>	$\frac{\pi}{4}$												
<code>y1(x):=</code>	$2 \cdot \cos(x)$	<code></code>	Done												
<code>lst5:=</code>	{ 1,2,3,4 }	<code></code>	{ 1,2,3,4 }												
<code>matg:=</code>	<table border="1"><tr><td>1</td><td>2</td><td>3</td></tr><tr><td>4</td><td>5</td><td>6</td></tr></table>	1	2	3	4	5	6	<code></code>	<table border="1"><tr><td>1</td><td>2</td><td>3</td></tr><tr><td>4</td><td>5</td><td>6</td></tr></table>	1	2	3	4	5	6
1	2	3													
4	5	6													
1	2	3													
4	5	6													
<code>str1:=</code>	"Hello"	<code></code>	"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 määritysten syöttämisestä löytyvät tuotteen ohjekirjan Laskin-osiosta.

Define $g(n)=$ Func

© Declare variables

Local *i,result*

result:=0

For *i,1,n,1* ©Loop *n times*

result:=result+i²

EndFor

Return *result*

EndFunc

Done

$g(3)$

14

0b *binaariluku***0h** *heksadesimaaliluku*

Määrittää binaari- (Ob) tai heksadesimaaliluvun (Oh).
Syöttääksesi binaari- tai heksadesimaaliluvun sinun on syötettävä etuliite Ob tai Oh riippumatta kantalukutilasta. Ilman etuliitettä lukua käsitellään desimaalilukuna (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 Kuvaaajien piirtäminen taulukkodatasta.

Funktion **delVoid()** avulla voit poistaa tyhjät elementit listasta. F funktion **isVoid()** avulla voit testata tyhjää elementtiä. Lisätietoja, katso **delVoid()**, sivu 53, ja **isVoid()**, sivu 91.

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.

<code> _</code>	<code>_</code>
<code>gcd(100,_)</code>	<code>_</code>
<code>3+_</code>	<code>_</code>
<code>{5,_} - {3,6,9}</code>	<code>{2,_}</code>

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

<code>sum({2,_3,5,6,6})</code>	16.6
<code>median({1,2,_,_,3})</code>	2
<code>cumulativeSum({1,2,_,4,5})</code>	<code>{1,3,_,7,12}</code>
<code>cumulativeSum($\begin{pmatrix} 1 & 2 \\ 3 & - \\ 5 & 6 \end{pmatrix}$)</code>	$\begin{bmatrix} 1 & 2 \\ 4 & - \\ 9 & 8 \end{bmatrix}$

SortA ja **SortD** siirtävät kaikki ensimmäisen argumentin sisältämät tyhjät elementit viimeiseksi.

<code>{5,4,3,_,1} → list1</code>	<code>{5,4,3,_,1}</code>
<code>{5,4,3,2,1} → list2</code>	<code>{5,4,3,2,1}</code>
<code>SortA list1,list2</code>	<code>Done</code>
<code>list1</code>	<code>{1,3,4,5,_,}</code>
<code>list2</code>	<code>{1,3,4,5,2}</code>

Regressioissa X- tai Y-listassa oleva tyhjä elementti aiheuttaa tyhjän elementin jäännöksen vastaavalle elementille.

$\{1,2,3,_,5\} \rightarrow list1$	$\{1,2,3,_,5\}$
$\{1,2,3,4,5\} \rightarrow list2$	$\{1,2,3,4,5\}$
SortD list1,list2	Done
list1	$\{5,3,2,1,_\}$
list2	$\{5,3,2,1,4\}$
<hr/>	
$II:=\{1,2,3,4,5\}; I2:=\{2,_,3,5,6,6\}$	$\{2,_,3,5,6,6\}$
LinRegMx II,I2	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	$\{1,_,1,1,1\}$

Poisjätetty luokka regressioissa aiheuttaa tyhjän elementin jäännöksen vastaavalle elementille.

$II:=\{1,3,4,5\}; I2:=\{2,3,5,6,6\}$	$\{2,3,5,6,6\}$
cat:={"M","M","F","F"}; incl:={"F"}	$\{ "F" \}$
LinRegMx II,I2,1,cat,incl	Done
stat.Resid	$\{_, -, 0, ., 0\}$
stat.XReg	$\{_, -, 4, ., 5\}$
stat.YReg	$\{_, -, 5, ., 6, 6\}$
stat.FreqReg	$\{_, -, 1, ., 1, .\}$

Frekvenssi 0 regressioissa aiheuttaa tyhjän elementin jäännöksen vastaavalle elementille.

$II:=\{1,3,4,5\}; I2:=\{2,3,5,6,6\}$	$\{2,3,5,6,6\}$
LinRegMx II,I2,{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>	<code>@n1, @n2, ...</code>

Syötettävä kohde:	Pikavalinta:
(kokonaislukuvakiot)	
i (imaginaarinen vakio)	@i
e (luonnollisen logaritmin kantaluku e)	@e
ϵ (kymmenpotenssimuoto)	@E
T (transponointi)	@t
r (radiaanit)	@r
$^{\circ}$ (asteet)	@d
$^{\circ}$ (graadit)	@g
\angle (kulma)	@<
\blacktriangleright (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 sulkuryhmyityksiä 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 (T)
 - 5 Potenssiin korotus, potenssioperaattori (^)
 - 6 Negaatio (-)
 - 7 Merkkijonojen ketjutus (&)
 - 8 Kertolasku (\cdot), jakolasku (/)
 - 9 Yhteenlasku (+), vähennyslasku (-)
Yhtäläisyysuhteet: on yhtä kuin (=), ei ole yhtä kuin (\neq tai \neq),
 - 10 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 komennoiksi eikä implisiittiseksi kertolaskuksi. Esimerkiksi $a(b+c)$ on funktio a , jota sievennetään lausekkeella $b+c$. Kun haluat kertoa lausekkeen $b+c$ muuttujalla a , käytä eksplisiittistä kertolaskua: $a*(b+c)$.

Epäsuora operaattori

Epäsuora operaattori (#) muuttaa merkijonon 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 neljännessä 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 $(-)$ ja syöttämällä luku sen perään. Jälkioperaattorit ja eksponenttillausekkeet 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 183.

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: <ul style="list-style-type: none">• ei ala numerolla• ei sisällä välilyöntejä tai erikoismerkkejä• ei käytä alaviivaa tai pistettä väärällä tavalla• ei ylitä pituusrajoituksia Lisätietoja löytyy ohjekirjan Laskin-sovellusta käsittelevästä osasta.
160	Argumentin on oltava lauseke.
165	Paristovirta ei riitä lähettämiseen tai vastaanottamiseen. Asenna uudet paristot ennen lähettämistä tai vastaanottoa.
170	Raja Alarajan on oltava ylärajaa pienempi hakuväliä määritettäessä.
180	Keskeytys Näppäintä esc tai Ctrl on painettu pitkän laskutoimituksen tai ohjelman suorituksen aikana.
190	Kehämääritys

Virhekoodi Kuvaus

- Tämä viesti tulee näkyviin muistin loppumisen välttämiseksi muuttujien arvojen äärettömän korvauksen aikana sievennetyssä lausekkeita. Virheen aiheuttaa esimerkiksi lauseke $a+1->a$, jossa a on määrittämätön muuttuja.
- 200 Määrittelyalueen lauseke ei kelpaa.
- Tämän virheilmoituksen voi aiheuttaa esimerkiksi lauseke $\text{solve}(3x^2-4=0,x) | x<0$ 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 Dimensoiden yhteensopivuusvirhe
- Kahdella tai useammalla argumentilla on oltava sama dimensio. Esimerkiksi lauseke $[1,2]+[1,2,3]$ aiheuttaa dimensoiden 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 lf..Endlf ulkopuolella
- 290 EndTry-komennosta 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 merkkijono ei kelpaa muuttujan nimeksi.
- 380 Määrittämätön Ans
- Edellinen laskutoimitus ei luonut Ans:n arvoa, tai aikaisempaa laskutoimitusta ei ole syötetty.
- 390 Epäkelpo tehtävä
- 400 Epäkelpo tehtävän arvo
- 410 Virheellinen komento
- 430 Virheellinen nykyisille tila-asetuksille

Virhekoodi Kuvaus

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 kertomerkitömiän 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 lmuutt ei kelpaa.
575	Virheellinen polaarin kompleksiluku
580	Virheellinen ohjelmaviittaus Ohjelmiin ei voi viitata funktioiden tai lausekkeiden sisällä, kuten $1+p(x)$, jossa p on ohjelma.
600	Kelpaamaton taulukko
605	Yksikötä 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 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 (

Virhekoodi Kuvaus

690	Puuttuva)
700	Puuttuva "
710	Puuttuva]
720	Puuttuva }
730	Syntaksista puuttuu lohkon alku tai loppu
740	Then puuttuu lohokosta 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 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 komennosta: <ul style="list-style-type: none">• <code>sto →</code>• <code>:=</code>• Define määrittääksesi muuttujille arvot.

Virhekoodi Kuvaus

965	Käyttöjärjestelmälle ei ole lisenssiä
970	Muuttuja on käytössä, joten viittaukset tai muutokset eivät ole sallittuja
980	Muuttuja on suojattu
990	Virheellinen muuttujan nimi Varmista, että nimi ei ylitä pituusrajoituksia
1000	Ikkunamuuttujien määrittelyalue
1010	Zoomaus
1020	Sisäinen virhe
1030	Suojatun muistin rikkomus
1040	Toimintoa ei tueta Tämä toiminto vaatii tietokonealgebrajärjestelmän Kokeile TI-Nspire™ CAS -järjestelmää.
1045	Operaattoria ei tueta. Tämä operaattori vaatii tietokonealgebrajärjestelmän. Kokeile TI-Nspire™ CAS -järjestelmää.
1050	Toimintoa ei tueta. Tämä operaattori vaatii tietokonealgebrajärjestelmän. Kokeile TI-Nspire™ CAS -järjestelmää.
1060	Syötetyn argumentin on oltava numeerinen. Vain numeerisia arvoja sisältävät syötteet sallitaan.
1070	Trig-funktion argumentti on liian suuri täsmälliseen sievennykseen
1080	Ans-muuttujan käyttöä ei tueta. Tämä sovellus ei tue Ans-muuttujan käyttöä.
1090	Funktiota ei ole määritetty. Käytä jotakin seuraavista komennoista: <ul style="list-style-type: none">• Define• :=• sto → määrittääksesi funktion.
1100	Ei-reaalinen laskutoimitus 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).
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 polynomilauseke. Jos toinen argumentti jätetään pois, ohjelmisto yrittää valita oletusarvon.
1150	Argumenttinvirhe Ensimmäisten kahden argumentin on oltava kolmannen argumentin sisällä olevia polynomilausekkeita. Jos kolmas argumentti jätetään pois, ohjelmisto yrittää valita oletusarvon.
1160	Virheellinen kirjaston polkunimi Polkunimen on oltava muodossa xxx\yyy, jossa: <ul style="list-style-type: none">• xxx-osassa voi olla 1-16 merkkiä.

Virhekoodi Kuvaus

- 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.
- 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:
- Trigonometrinen muunnosten operaattoreita ei tueta aste- eikä graadikulmatilassa.
- 1250 Argumenttiovirhe
- Käytä lineeariyhtälöryhmää.
- Esimerkki kahden lineeariyhtälön ryhmästä, jossa on muuttujat x ja y :
- $$3x + 7y = 5$$

Virhekoodi Kuvaus

$$2y-5x=-1$$

- 1260 Argumenttivistä:
nMin:n tai **nMax:n** ensimmäisen argumentin on oltava yhden muuttujan sisältämä lauseke. Se ei voi sisältää muuta arvotonta muuttujaa kuin käsittelyn kohteena olevan muuttujan.
- 1270 Argumenttivistä
Derivaatan asteen on oltava 1 tai 2.
- 1280 Argumenttivistä
Käytä yhden muuttujan sisältämää laajennetussa muodossa olevaa polynomia.
- 1290 Argumenttivistä
Käytä yhden muuttujan sisältämää polynomia.
- 1300 Argumenttivistä
Polynomien kertoimien tuloksena on oltava numeerinen arvo.
- 1310 Argumenttivistä:
Yhtä tai useampaa funktion argumenttia ei voi sieventää.
- 1380 Argumenttivistä:
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 191.

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	cSolve voi määrittää lisää nollakohtia.
10006	Solve voi määrittää lisää nollia.
10007	Lisää ratkaisuja voi olla olemassa. Yritä määrittämällä oikeat ala- ja ylärajat ja/tai arvaamalla. Esimerkkejä solve() käytöstä: <ul style="list-style-type: none">• solve(Yhtälö, Muut=Arvaus)alaraja<Muut<yläraja• solve(Yhtälö, Muut)alaraja<Muut<yläraja• solve(Yhtälö, Muut=Arvaus)
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	Trig-funktion argumentti on liian suuri täsmälliseen sievennykseen.
10021	Syöte sisältää määrittämättömän parametrin.

Varoituskoodi	Viesti
	Tulos ei ehkä ole voimassa kaikille mahdollisille parametriarvoille.
10022	Ratkaisu on mahdollinen, kun määritetään oikeat ala- ja ylärajat.
10023	Skalaari on kerrottu identtisellä matriisilla.
10024	Tulos saatu käyttämällä liikimääräistä 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[*]	201
	!	
!, kertoma	211
	"	
", sekuntimuoto	219
	#	
#, epäsuora operaattori	217, 232
	%	
%, prosentti	207
	&	
&, liitä	211
	*	
*, kertolasku	202
	,	
, jaoton	221
, minuuttimuoto	219

.	
-, piste-erotus	205
*, pistetulo	206
/, pisteosamäärä	206
^, pistepotenssi	206
+, pistesumma	205
:	
:=, määritä	225
^	
^-1, käänteisluku	223
^, potenssi	204
—	
_ , yksikön merkki	221
_ , yksikön nimi	221
, (rajoittava operaattori)	223
+	
+, yhteenlasku	201
/	
/, jakolasku[*]	203
≠	
≠, ei yhtä kuin[*]	208

	$=$	
$=$, on yhtä kuin		207
	$>$	
$>$, suurempi kuin		209
	Π	
Π , tulo[*]		214
	Σ	
$\Sigma()$, summa[*]		215
$\Sigma\text{Int}()$		216
$\Sigma\text{Prn}()$		216
	$\sqrt{\quad}$	
$\sqrt{\quad}$, neliöjuuri[*]		214
	\int	
\int , integraali[*]		212
	\leq	
\leq , pienempi tai yhtä suuri kuin		209
	\geq	
\geq , suurempi tai yhtä suuri kuin		210
	\blacktriangleright	
\blacktriangleright , muunna graadikulmaksi[Grad]		83
\blacktriangleright , muunna yksiköt[*]		221

► . asteen regressio, QuartReg	135
► approxFraction()	18
► cos, näytä kulman kosini[cos]	33
► Cylind, näytä vektorina sylinterikoordinaatistossa[Cylind]	46
► DD, näytä desimaalikulmana[DD]	49
► Desimaali, näytä tulos desimaalilukuna[Desimaali]	50
► DMS, näytä asteina/ minuutteina/ sekunteina[DMS]	56
► exp, näytä e:n arvolla[exp]	64
► Kantaluku10, näytä desimaalikonaislukuna[Kantaluku10]	23
► Kantaluku16, näytä heksadesimaalilukuna[Kantaluku16]	23
► Kantaluku2, näytä binaarisena[Kantaluku2]	22
► Polaarinen, näytä polaarisenä vektorina[Polaarinen]	127
► Rad, muunna radiaanikulmaan[Rad]	137
► Rect, näytä suorakulmavektorina[Rect]	140
► sin, näytä kulman sini[sin]	159
► Sphere, näytä pallonmuotoisena vektorina[Sphere]	167

⇒

⇒ , looginen seuraus[*]	210, 229
-------------------------------	----------

→

→ , tallenna	224
--------------------	-----

↔

↔ , looginen kaksoisseuraus[*]	211
--------------------------------------	-----

©

© , kommentti	225
---------------------	-----

°

° , astekulmamuoto[*]	219
-----------------------------	-----

° , astetta/ minuuttia/ sekuntia[*]	219
-------------------------------------------	-----

0

0b, binaarinen indikaattori	226
0h, heksadesimaalinen indikaattori	226

1

10 [^] (), kymmenen potenssi	222
----------------------------------------------	-----

2

2 otoksen F-testi	76
2. asteen regressio, QuadReg	134

3

3. asteen regressio, CubicReg	44
-------------------------------------	----

A

abs(), itseisarvo	12
alaraja, floor()	71
alaviiva, _	221
alimatriisi, subMat()	172, 174
amortTbl(), 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()	18

arccsc()	19
arccsch()	19
arcLen(), kaaren pituus	19
arcsec()	19
arcsech()	19
arcsin()	19
arcsinh()	20
arctan()	20
arctanh()	20
arkuskosini, $\cos^{-1}()$	35
arkussini, $\sin^{-1}()$	160
arkustangentti, $\tan^{-1}()$	175
aseta	
tila, setMode()	155
asetukset, hae nykyiset	80
aste/ minuutti/sekunti-muoto	219
astekulmamuoto, -	219
augment(), lisää/ketjuta	20
avgRC(), keskimääräinen muutosnopeus	20

B

binaarinen	
indikaattori, 0b	226
näytä, 4Kantaluku2	22
binomCdf()	24
binomPdf()	24
Boolean operators	
and	13
Booleen operaattorit	
\Rightarrow	210, 229
\Leftrightarrow	211
ei	120
ei-ja	114
eikä	118
tai	123

C

Cdf()	70
ceiling(), yläraja	25
centralDiff()	25
cFactor(), kompleksilukutekijä	26
char(), merkkijono	27
charPoly()	27
χ^2 2way	27
clearAZ	29
ClrErr, poista virhe	29
colAugment	30
colDim(), matriisin sarakemitat	30
colNorm(), matriisin sarakenormi	30
comDenom(), yhteinen nimittäjä	30
completeSquare(), complete square	31
conj(), liittokompleksiluku	32
constructMat(), luo matriisi	32
corrMat(), korrelaatiomatriisi	33
\cos^{-1} , arkuskosini	35
cos(), kosini	34
\cosh^{-1} (), hyperbolinen arkuskosini	36
cosh(), hyperbolinen kosini	36
\cot^{-1} (), arkuskotangenti	37
cot(), kotangenti	37
\coth^{-1} (), hyperbolinen arkuskotangenti	38
coth(), hyperbolinen kotangenti	38
count(), laske listan kohtien lukumäärä	38
countif(), laske listan yksiköiden lukumäärä ehdollisesti	39
cPolyRoots()	40
crossP(), ristitulo	40
\csc^{-1} (), käänteiskosekanti	41
csc(), kosekanti	40
csch^{-1} (), käänteinen hyperbolinen kosekanti	41

<code>csch()</code> , hyperbolinen kosekanti	41
<code>cSolve()</code> , ratkaise kompleksiyhtälö	42
<code>CubicReg</code> , 3. asteen regressio	44
<code>cumulativeSum()</code> , kumulatiivinen summa	45
<code>cycle</code> , <code>Cycle</code>	45
<code>Cycle</code> , <code>cycle</code>	45
<code>cZeros()</code> , kompleksiset nollakohdat	46

D

<code>d()</code> , ensimmäinen derivaatta	212
<code>dbd()</code> , päivämäärien väliset päivät	49
<code>Define</code> (Määritä)	50
<code>Define LibPriv</code> (Määritä <code>LibPriv</code>)	51
<code>Define LibPub</code> (Määritä <code>LibPub</code>)	51
<code>Define</code> , määritä	50
<code>deltaList()</code>	52
<code>deltaTmpCnv()</code>	52
<code>DelVar</code> , poista muuttuja	52
<code>delVoid()</code> , poista tyhjät elementit	53
derivaatat	
ensimmäinen derivaatta, <code>d()</code>	212
numeerinen derivaatta, <code>nDeriv()</code>	116-117
numeerinen derivaatta, <code>nDerivative()</code>	116
derivaatta tai n:s derivaatta	
malli	10
desimaali	
kokonaislukunäyttö, <code>4Kantaluku10</code>	23
kulmanäyttö, <code>►DD</code>	49
<code>deSolve()</code> , ratkaisu	53
<code>det()</code> , matriisin determinantti	55
<code>diag()</code> , diagonaalimatriisi	55
<code>dim()</code> , mitta	55
<code>Disp</code> , näytä tiedot	56
<code>dominantTerm()</code> , dominanttitermi	58
dominanttitermi, <code>dominantTerm()</code>	58

dotP(), pistetulo	59
--------------------------	----

E

e eksponentti	
malli	6
e korotettuna potenssiin, $e^{\wedge}()$	59, 65
E, eksponentti	218
e, näytä lauseke e:n arvolla	64
$e^{\wedge}()$, e korotettuna potenssiin	59
efektiivinen korko, eff()	60
eff), muunna nimelliskorko efektiiviseksi koroksi	60
ei-ja, Boolean operaattori	114
ei yhtä kuin, \neq	208
ei, Boolean operaattori	120
eigVc(), ominaisvektori	60
eigVl(), ominaisarvo	61
eikä, Boolean operaattori	118
eksklusioio "]"-operaattorilla	223
eksponentiaalinen regressio, ExpReg	67
eksponentit	
malli	5
eksponentti, E	218
else if, Elseif	61
else, Else	84
Elseif, else if	61
end	
for, EndFor	73
funktio, EndFunc	77
if, EndIf	84
ohjelma, EndPrgm	132
silmukka, EndLoop	105
try, EndTry	183
while, EndWhile	192
end if, EndIf	84
end while, EndWhile	192

EndTry, kokeilun loppu	183
EndWhile, end while	192
ensimmäinen derivaatta	
malli	10
EOS (yhtälökäyttöjärjestelmä)	231
epäsuora operaattori (#)	232
epäsuora operaattori, #	217
etumerkki, sign()	158
euler(), Euler function	62
exact(), täsmällinen	64
Exit, lopeta	64
exp(), e korotettuna potenssiin	65
expList(), lauseke listaksi	65
expand(), lavenna	66
expr(), merkkijono lausekkeeksi	67, 103
ExpReg, eksponentiaalinen regressio	67

F

factor(), tekijä	68
Fill, matriisin täyttö	70
FiveNumSummary	71
floor(), alaraja	71
fMax(), funktion maksimi	72
fMin(), funktion minimi	72
For	73
for, For	73
For, for	73
format(), merkkijonon muoto	74
fpart(), funktion osa	74
freqTable()	75
frequency()	75
Frobeniusin normi, norm()	119
Func, funktio	77
Func, ohjelmafunktio	77
funktion loppu, EndFunc	77

funktiot	
käyttäjän määrittämät	50
maksimi, fMax()	72
minimi, fMin()	72
ohjelmafunktio, Func	77
osa, fpart()	74
funktiot ja muuttajat	
kopiointi	33

G

g, graadit	218
gcd(), suurin yhteinen jakaja	77
geomCdf()	78
geomPdf()	78
getDenom(), hae/laske nimittäjä	79
getLangInfo(), hae/anna kielitiedot	79
getLockInfo(), testaa muuttujan tai muuttujaryhmän lukitustilan	80
getMode(), hae tila-asetukset	80
getNum(), hae/laske luku	81
getType(), get type of variable	81
getVarInfo(), hae/laske muuttujien tiedot	82
Goto, siirry	83
graadimuoto, g	218

H

hae/laske	
luku, getNum()	81
muuttujien tiedot, getVarInfo()	79, 82
nimittäjä, getDenom()	79
heksadesimaali	
näytä, 4Kantaluku16	23
heksadesimaalinen	
indikaattori, Oh	226

hyperbolinen

areatangenti, $\tanh^{-1}()$	177
arkuskosini, $\cosh^{-1}()$	36
arkussini, $\sinh^{-1}()$	161
kosini, $\cosh()$	36
sini, $\sinh()$	161
tangenti, $\tanh()$	176

I

identity(), identtinen matriisi	83
identtinen matriisi, identity()	83
if, If	84
If, if	84
ifFn()	85
imag(), imaginaarinen osa	86
imaginaarinen osa, imag()	86
ImpDif(), implisiittinen derivaatta	86
implisiittinen derivaatta, Impdif()	86
Input, syöte	86
inString(), merkkijonon sisällä	87
int(), kokonaisluku	87
intDiv(), kokonaisluvun jakaminen	87
integraali, %o	212
interpolate(), interpolate	88
invF()	88
invNorm(), käänteinen kumulatiivinen normaalijakauma	89
inv()	89
Inv χ^2 ()	88
iPart(), kokonaisosa	89
irr(), sisäinen korkokanta	
sisäinen korkokanta, irr()	90
isPrime(), jaottoman testi	90
isVoid(), testaa tyhjät	91
itseisarvo	
malli	8

J

jakaumafunktiot

binomCdf()	24
binomPdf()	24
invNorm()	89
invt()	89
Inv χ^2 ()	88
normCdf()	119
normPdf()	119
poissCdf()	126
poissPdf()	127
tCdf()	178
tPdf()	182
χ^2 2way()	27
χ^2 Cdf()	28
χ^2 GOF()	28
χ^2 Pdf()	29
jakolasku, P	203
jaoton,	221
jaottoman luvun testi, isPrime()	90
jos käytettävissäsi on	223
jäännös, remain()	142

K

kaaren pituus, arcLen()	19
kahden muuttujan tulokset, TwoVar	187
kertolasku, *	202
kertoma, !	211
keskiarvo, mean()	107
keskihajonta, stdDev()	170-171, 189
keskimmäinen merkkijono, mid()	109
keskimääräinen muutosnopeus, avgRC()	20

kieli	
hae kielitiedot	79
kierrä, rotate()	147
kirjasto	
luo pikavalinnat objekteihin	92
kokonaisluku, int()	87
kokonaisluvun jakaminen, intDiv()	87
kokonaisuosa, iPart()	89
kombinaatiot, nCr()	115
kommentti, ©	225
kompleksi	
liitto, conj()	32
nollakohdat, cZeros()	46
ratkaise, cSolve()	42
tekijä, cFactor()	26
kopioi muuttuja tai funktio, CopyVar	33
korkeerien summa	216
korrelaatiomatriisi, corrMat()	33
korvaaminen "]"-operaattorilla	223
kosini	
näytä kulman	33
kosini, cos()	34
kotangenti, cot()	37
kulma, angle()	14
kumulatiivinen summa, cumulativeSum()	45
kun, when()	191
kymmenen potenssi, 10^()	222
käyttäjän määrittämät funktiot	50
käyttäjän määrittämät funktiot ja ohjelmat	51
käänteinen kumulatiivinen normaalijakauma (invNorm())	89
käänteinen, ^-1	223
käänteisluku, ^-1	223

L

lajitteleminen	
laskeva järjestys, SortD	167
nouseva järjestys, SortA	166
laske listan kohtien lukumäärä, count()	38
laske listan yksiköiden lukumäärä ehdollisesti, countif()	39
laske polynomi, polyEval()	129
laske päivämäärien väliset päivät, dbd()	49
lausekkeet	
lauseke listaksi, exp▶list()	65
merkkijono lausekkeeksi, expr()	67, 103
lavenna, expand()	66
Lbl, tunnus	91
lcm, pienin yhteinen jaettava	91
left(), vasen	92
LibPriv	51
LibPub	51
libShortcut(), luo pikavalinnat kirjasto-objekteihin	92
liitä, &	211
likimääräinen, approx()	17
limit() tai lim(), raja-arvo	93
lineaarinen regressio, LinRegAx	94
lineaarinen regressio, LinRegBx	94-95
LinRegBx, lineaarinen regressio	94
LinRegMx, lineaarinen regressio	94
LinRegIntervals, lineaarinen regressio	95
LinRegtTest	97
linSolve()	98
Δlist(), listaerotus	98
list▶mat(), listasta matriisiksi	99
lista, laske kohtien lukumäärä	38
lista, laske yksiköiden lukumäärä ehdollisesti	39
listasta matriisiksi, list▶mat()	99

listat	
erotukset listassa, @list()	98
erotus, @list()	98
keskimäinen merkkijono, mid()	109
kumulatiivinen summa, cumulativeSum()	45
lajittele laskevaan järjestykseen, SortD	167
lajittele nousevaan järjestykseen, SortA	166
lauseke listaksi, exp►list()	65
listasta matriisiksi, list►mat()	99
lisää/ketjuta, augment()	20
maksimi, max()	107
matriisi listaksi, mat►list()	106
minimi, min()	110
pistetulo, dotP()	59
ristitulo, crossP()	40
tulo, product()	132
tyhjät elementit	227
uusi, newList()	116
yhteenlasku, sum()	172-173
lisää/ketjuta, augment()	20
ln(), luonnollinen logaritmi	99
LnReg, logaritminen regressio	100
Local, paikallinen muuttuja	101
Lock, lukitse muuttuja tai muuttujaryhmä	102
Logaritmi	
malli	6
logaritminen regressio, LnReg	100
logaritmit	99
Logistic, logistinen regressio	103
LogisticD, logistinen regressio	104
logistinen regressio, Logistic	103
logistinen regressio, LogisticD	104
looginen kaksoisseuraus, ⇔	211
looginen seuraus, ⇒	210, 229
Loop, silmukka	105

lopeta, Exit	64
LU, matriisin ala-ylä-dekomponointi	106
luo matriisi, constructMat()	32
luonnollinen logaritmi, ln()	99
lyhennystaulukko, amortTbl()	12, 21

M

maksimi, max()	107
mallit	
derivaatta tai n:s derivaatta	10
e eksponentti	6
eksponentti	5
ensimmäinen derivaatta	10
itseisarvo	8
Logaritmi	6
matriisi (1 × 2)	9
matriisi (2 × 1)	9
matriisi (2 × 2)	8
matriisi (m × n)	9
murtoluku	5
määrittämätön integraali	11
määrätty integraali	11
n:s juuri	6
neliöjuuri	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
mat►list(), matriisi listaksi	106
matriisi (1 × 2)	
malli	9

matriisi (2×1)	
malli	9
matriisi (2×2)	
malli	8
matriisi ($m \times n$)	
malli	9
matriisi listaksi, mat►list()	106
matriisit	
ala-ylä-dekomponointi, LU	106
alimatriisi, subMat()	172, 174
determinantti, det()	55
diagonaali, diag()	55
identtinen, identity()	83
kumulatiivinen summa, cumulativeSum()	45
listasta matriisiksi, list►mat()	99
lisää/ketjuta, augment()	20
maksimi, max()	107
matriisi listaksi, mat►list()	106
minimi, min()	110
mitta, dim()	55
ominaisarvo, eigVl()	61
ominaisvektori, eigVc()	60
piste-erotus, .-	205
pisteosamäärä, ./	206
pistepotenssi, .^	206
pistesumma, .+	205
pistetulo, .*	206
QR-hajottaminen, QR	134
rivi-echelon-muoto, ref()	141
rivien kertominen ja yhteenlasku, mRowAdd()	111
rivien vaihto, rowSwap()	149
rivien yhteenlasku, rowAdd()	148
rivikoko, rowDim()	149
rivioperaatio, mRow()	111
row norm, rowNorm()	149

sarakemitat, colDim()	30
sarakenormi, colNorm()	30
satunnaisluvut, randMat()	138
sievennetty rivi-echelon-muoto, rref()	149
transponoi, T	174
tulo, product()	132
täyttäminen, Fill	70
uusi, newMat()	116
yhteenlasku, sum()	172-173
max(), maksimi	107
mean(), keskiarvo	107
mediaani-mediaani-suoran regressio, MedMed	108
mediaani, median()	108
median(), mediaani	108
MedMed, mediaani-mediaani-suoran regressio	108
merkit	
merkkijono, char()	27
numeerinen koodi, ord()	125
merkkijono	
mitta, dim()	55
pituus	55
merkkijono, char()	27
merkkijonon muoto, format()	74
merkkijonon pituus	55
merkkijonon sisällä, inString()	87
merkkijonot	
epäsuora operaattori, #	217
keskimmäinen merkkijono, mid()	109
kierrä, rotate()	147
käyttö muuttujanimien luomisessa	232
lauseke merkkijonoksi, string()	172
liitä, &	211
merkkijono lausekkeeksi, expr()	67, 103
merkkijono, char()	27
merkkikoodi, ord()	125

muoto, format()	74
muotoilu	74
oikea, right()	145
siirrä, shift()	156
sisällä, InString	87
vasen, left()	92
mid(), keskimäinen merkkijono	109
min(), minimi	110
minimi, min()	110
minuuttimuoto,	219
mirr(), modifioitu sisäinen korkokanta	110
mitta, dim()	55
mod(), modulo	111
modifioitu sisäinen korkokanta, mirr()	110
modulo, mod()	111
Moninkertaisen lineaarisen regression t-testi	113
mRow(), matriisin rivioperaatio	111
mRowAdd(), matriisin rivien kertominen ja yhteenlasku	111
MultReg	112
MultRegIntervals()	112
MultRegTests()	113
murtoluvut	
malli	5
propFrac	133
muunna	
4Grad	83
4Rad	137
yksiköt	221
muuttuja	
nimen luominen merkkijonosta	232
muuttujat	
paikallinen, Local	101
poista kaikki yksikirjaimiset	29
poista, DelVar	52

muuttujat ja funktiot	
kopioiminen	33
muuttujat, lukitseminen ja vapauttaminen	80, 102, 189
muuttujien ja muuttujaryhmien lukitseminen	102
muuttujien ja muuttujaryhmien vapauttaminen	189
määrittäminen	
julkinen funktio tai ohjelma	51
yksityinen funktio tai ohjelma	51
määrittämätön integraali	
malli	11
määritä, Define	50
määrätty integraali	
malli	11

N

n:s juuri	
malli	6
nCr(), kombinaatiot	115
nDerivative(), numeerinen derivaatta	116
negaatio, negatiivisten lukujen syöttäminen	232
neliöjuuri	
malli	5
neliöjuuri, $\ddagger()$	168, 214
nettonykyarvo, npv()	121
newList(), uusi lista	116
newMat(), uusi matriisi	116
nfMax(), numeerisen funktion maksimi	116
nfMin(), numeerisen funktion minimi	117
nimelliskorko, nom()	117
nimittäjä	30
nInt(), numeerinen integraali	117
nollakohdat, zeroes()	193
nom), muuta efektiivinen korko nimelliskoroksi	117
norm(), Frobeniusin normi	119
normaalijakauman todennäköisyys, normCdf()	119

normaalisuora, normalLine()	119
normalLine()	119
normCdf()	119
normPdf()	119
nPr(), permutaatiot	120
npv(), nettonykyarvo	121
nSolve(), numeerinen ratkaisu	121
numeerinen	
derivaatta, nDeriv()	116-117
derivaatta, nDerivative()	116
integraali, nInt()	117
ratkaisu, nSolve()	121
näppäimistön pikavalinnat	229
näyttö asteina/ minuutteina/ sekunteina, ▶DMS	56
näyttö pallonmuotoisena vektorina, ▶Sphere	167
näyttö vektorina sylinterikoordinaatistossa, ▶Cylind	46
näytä	
asteina/ minuutteina/ sekunteina, ▶DMS	56
binaarisena, 4Kantaluku2	22
desimaalikononaislukuna, 4Kantaluku10	23
desimaalikulmana, ▶DD	49
heksadesimaalilukuna, 4Kantaluku16	23
pallonmuotoinen vektori, ▶Sphere	167
polaarinen vektori, 4Polaarinen	127
suorakulmavektori, ▶Rect	140
vektorina sylinterikoordinaatistossa, 4Cylind	46
näytä tiedot, Disp	56

O

objektit	
luo pikavalinnat kirjastoon	92
ohita virhe, PassErr	126
ohjelmat	
julkisen kirjaston määrittäminen	51
yksityisen kirjaston määrittäminen	51

ohjelmat ja ohjelmointi	
kokeilun loppu, EndTry	183
lopeta ohjelma, EndPrgm	132
näytä I/O-näyttö, Disp	56
poista virhe, ClrErr	29
try, Try	183
ohjelmointi	
määritä ohjelma, Prgm	132
näytä tiedot, Disp	56
ohita virhe, PassErr	126
oikea, right()	145
ominaisarvo, eigVl()	61
ominaisvektori, eigVc()	60
on yhtä kuin, =	207
OneVar, yhden muuttujan tilastot	122
operaattorit	
sievennysjärjestys	231
ord(), numeerinen merkkikoodi	125

P

P•Rx(), x:n suorakulmakoordinaatti	125
P•Ry(), y:n suorakulmakoordinaatti	125
paikallinen muuttuja, Local	101
paikallinen, Local	101
paloittain määritelty funktio (2-osainen)	
malli	6
paloittain määritelty funktio (N-osainen)	
malli	7
PassErr, ohita virhe	126
Pdf()	75
permutaatiot, nPr()	120
piecewise()	126
pienempi tai yhtä suuri kuin, {	209
pienin yhteinen jaettava, lcm	91
pikavalinnat, näppäimistö	229

piste	
erotus, .-	205
osamäärä, ./	206
potenssi, ^	206
summa, .+	205
tulo, .*	206
tulo, dotP()	59
poissCdf()	126
poissPdf()	127
poista	
poista, ClrErr	29
tyhjät elementit listasta	53
poistaminen	
muuttuja, DelVar	52
polaarinen	
koordinaatti, R•Pr()	137
koordinaatti, R•Pθ()	136
vektorinäyttö, 4Polaarinen	127
polyCof()	128
polyDegree()	128
polyEval(), laske polynomi	129
polyGcd()	129-130
polynomit	
laske, polyEval()	129
satunnainen, randPoly()	139
PolyRoots()	130
potenssi, ^	204
potenssiregressio, PowerReg	130-131, 143-144, 179
PowerReg, potenssiregressio	131
Prgm, määritä ohjelma	132
prodSeq()	132
product(), tulo	132
propFrac, varsinainen murtoluku	133
prosentti, %	207
pyöristä, round()	148

päivämäärien väliset päivät, <code>dbd()</code>	49
pääoman maksuerien summa	216

Q

QR-hajottaminen, <code>QR</code>	134
QR, QR-hajottaminen (tekijöihin jako)	134
QuadReg, 2. asteen regressio	134
QuartReg, 4. asteen regressio	135

R

R, radiaani	218
<code>R*Pr()</code> , napakoordinaatti	137
<code>R*Pθ()</code> , napakoordinaatti	136
radiaani, R	218
rahan aika-arvo, Korke	185
rahan aika-arvo, maksuerien lukumäärä	186
rahan aika-arvo, maksuerä	186
rahan aika-arvo, nykyarvo	186
rahan aika-arvo, Tuleva arvo	185
raja-arvo	
<code>lim()</code>	93
<code>limit()</code>	93
malli	11
rajoittava operaattori ""	223
rajoittava operaattori, laskemisjärjestys	231
<code>rand()</code> , satunnaisluku	138
<code>randBin</code> , satunnaisluku	138
<code>randInt()</code> , satunnainen kokonaisluku	138
<code>randMat()</code> , satunnaismatriisi	138
<code>randNorm()</code> , satunnainen normaalijakauma	139
<code>randPoly()</code> , satunnaispolynomi	139
<code>randSamp()</code>	139
<code>RandSeed</code> , satunnaisluvun siemenluku	140
<code>ratkaise</code> , <code>solve()</code>	163

ratkaisu, deSolve()	53
reaali, real()	140
real(), reaali	140
ref(), rivi-echelon-muoto	141
regressiot	
2. asteen, QuadReg	134
3. asteen, CubicReg	44
4. asteen, QuartReg	135
eksponentiaalinen, ExpReg	67
lineaarinen regressio, LinRegAx	94
lineaarinen regressio, LinRegBx	94-95
logaritminen, LnReg	100
Logistinen	103
logistinen, Logistic	104
mediaani-mediaani-suora, MedMed	108
MultReg	112
potenssiregressio, PowerReg	130-131, 143-144, 179
sinimuotoinen, SinReg	162
remain(), jäännös	142
Request	143
RequestStr	144
return, Return	145
Return, return	145
right(), oikea	145
right, right()	31, 62, 88, 145, 191
ristitulo, crossP()	40
rivi-echelon-muoto, ref()	141
rk23(), Runge Kutta function	145
rotate(), kierrä	147
round(), pyöristä	148
rowAdd(), matriisin rivien yhteenlasku	148
rowDim(), matriisin rivikoko	149
rowNorm(), matriisin rivinormi	149
rowSwap(), matriisin rivien vaihto	149
rref(), sievennetty rivi-echelon-muoto	149

ryhmät, lukitseminen ja vapauttaminen	102, 189
ryhmät, lukitustilan testaaminen	80

S

samanaikaiset yhtälöt, <code>simult()</code>	158
sarja, <code>series()</code>	153
satunnaisluvut	
matriisi, <code>randMat()</code>	138
norm, <code>randNorm()</code>	139
polynomi, <code>randPoly()</code>	139
siemenluku, <code>RandSeed</code>	140
satunnaisotos	139
<code>sec⁻¹()</code> , käänteissekantti	150
<code>sec()</code> , sekantti	150
<code>sech⁻¹()</code> , käänteinen hyperbolinen sekantti	151
<code>sech()</code> , hyperbolinen sekantti	151
sekaluvut, käyttäen funktiota <code>propFrac()</code>	133
sekuntimuoto, "	219
sekvenssi, <code>seq()</code>	151
<code>seq()</code> , sekvenssi	151
<code>seqGen()</code>	152
<code>seqn()</code>	153
<code>sequence, seq()</code>	152-153
<code>series()</code> , sarja	153
<code>setMode()</code> , aseta tila	155
<code>shift()</code> , siirrä	156
sievennetty rivi-echelon-muoto, <code>ref()</code>	149
sievennysjärjestys	231
<code>sign()</code> , etumerkki	158
siirry, Goto	83
silmukan loppu, <code>EndLoop</code>	105
silmukka, <code>Loop</code>	105
<code>simult()</code> , samanaikaiset yhtälöt	158
<code>sin⁻¹()</code> , arkussini	160
<code>sin()</code> , sini	159

$\sinh^{-1}()$, hyperbolinen arkussini	161
$\sinh()$, hyperbolinen sini	161
sini	
näytä kulman	159
sini, $\sin()$	159
siniregressio, SinReg	162
SinReg, siniregressio	162
solve(), ratkaise	163
SortA, lajittele nousevaan järjestykseen	166
SortD, lajittele laskevaan järjestykseen	167
sqrt(), neliöjuuri	168
stat.results	169
stat.values	170
stdDevPop(), perusjoukon keskihajonta	170
stdDevSamp(), otoksen keskihajonta	171
Stop-komento	171
string(), lauseke merkkijonoksi	172
strings	
right, right()	31, 62, 88, 145, 191
studentint-todennäköisyysjakauma, tCdf()	178
studentint-todennäköisyystiheys, tPdf()	182
subMat(), alimatriisi	172, 174
sum(), yhteenlasku	172
sumlf()	173
summa (G)	
malli	9
summa, S()	215
sumSeq()	174
suorakulmavektorinäyttö, ▶Rect	140
suurempi kuin, >	209
suurempi tai yhtä suuri kuin, 	210
suurin yhteinen jakaja, gcd()	77
syöte, Input	86

T

t-testi, tTest	184
T, transponoi	174
tai (Boolean), tai	123
tai, Boolean operaattori	123
tallentaminen	
symboli, &	224-225
talouslaskentafunktiot, tvnFV()	185
talouslaskentafunktiot, tvml()	185
talouslaskentafunktiot, tvnN()	186
talouslaskentafunktiot, tvnPmt()	186
talouslaskentafunktiot, tvnPV()	186
tan ⁻¹ (), arkustangenti	175
tan(), tangenti	174
tangentLine()	176
tangenti, tan()	174
tangenttisuora, tangentLine()	176
tanh ⁻¹ (), hyperbolinen areatangenti	177
tanh(), hyperbolinen tangenti	176
taylor(), Taylorin polynomi	177
Taylorin polynomi, taylor()	177
tCdf(), studentint-todennäköisyysjakauma	178
tCollect(), trigonometrinen keruu	178
tekijä, factor()	68
Teksti-komento	179
Test_2S, 2 otoksen F-testi	76
testaa tyhjät, isVoid()	91
testaa tyhjät, testi	91
tExpand(), trigonometrinen laajennus	178
tila-asetukset, getMode()	80
tilastot	
kahden muuttujan tulokset, TwoVar	187
kertoma, !	211
keskiarvo, mean()	107

keskihajonta, stdDev()	170-171, 189
kombinaatiot, nCr()	115
mediaani, median()	108
permutaatiot, nPr()	120
satunnainen normaalijakauma, randNorm()	139
siemenluku, RandSeed	140
variassi, variance()	190
yhden muuttujan tilastot, OneVar	122
tilat	
asetus, setMode()	155
TInterval, t-luottamusväli	180
TInterval_2Samp, kahdenotoksen t-luottamusvälin	180
ΔtmpCnv() [tmpCnv]	181
tmpCnv()	181
todennäköisyysfunktio, normPdf()	119
toinen derivaatta	
malli	10
tPdf(), studentint-todennäköisyysfunktio	182
trace()	182
transponoi, T	174
trigonometrinen keruu, tCollect()	178
trigonometrinen laajennus, tExpand()	178
Try, virheenkäsitelykomento	183
tTest, t-testi	184
tTest_2Samp, kahden otoksen t-testi	184
tulo (P)	
malli	10
tulo, P()	214
tulo, product()	132
tulokset, tilastot	169
tulosarvot, tilastot	170
tunnus, Lbl	91
TVM-argumentit	186
TVM-funktioiden argumentit	186
tvmFV()	185

tvml()	185
tvmN()	186
tvmPmt()	186
tvmPV()	186
TwoVar, kahden muuttujan tulokset	187
tyhjät elementit	227
tyhjät elementit, poista	53
täsmällinen, exact()	64

U

unitV(), yksikkövektori	189
unlock, vapauta muuttuja tai muuttujaryhmä	189
uusi	
lista, newList()	116
matriisi, newMat()	116

V

vaihto, shift()	156
vakio	
funktiossa solve()	164
vakiot	
funktiossa cSolve()	43
funktiossa cZeros()	48
funktiossa deSolve()	53
funktiossa solve()	165
pikavalinnat	229
varianssi, variance()	190

W

warnCodes(), Warning codes	191
varoituskoodit ja -viestit	240
varPop()	189
varSamp(), otoksen varianssi	190

varsinainen murtoluku, propFrac	133
vasen, left()	92
vastaus	
näytä e:n arvolla	64
näytä kulman kosini	33
näytä kulman sini	159
vastaus (viimeinen), ans	17
vektorit	
näyttö vektorina sylinterikoordinaatistossa, *Cylind	46
pistetulo, dotP()	59
ristitulo, crossP()	40
yksikkö, unitV()	189
when(), kun	191
while, While	192
While, while	192
virheet ja viianmääritys	
ohita virhe, PassErr	126
poista virhe, ClrErr	29
vähennyslasku, N	201

X

x:n suorakulmakoordinaatti, P►Rx()	125
x ² , neliö	205
XNOR	211
xor, Boolean eksklusiivinen tai (or)	192

Y

y:n suorakulmakoordinaatti, P►Ry()	125
yhden muuttujan tilastot, OneVar	122
yhteenlasku, +	201
yhteenlasku, sum()	172
yhteinen nimittäjä, comDenom()	30
Yhtälökäyttöjärjestelmä (EOS)	231

yhtälöpari (2 yhtälöä)	
malli	7
yhtälöryhmä (N yhtälöä)	
malli	8
yksikkövektori, unitV()	189
yksiköt	
muunna	221
yläraja, ceiling()	25, 40

Z

zeroes(), nollakohdat	193
zInterval, z-luottamusväli	195
zInterval_1Prop, yhden osuuden z-luottamusväli	196
zInterval_2Prop, kahden osuuden z-luottamusväli	196
zInterval_2Samp, kahden näytteen z-luottamusväli	197
zTest	198
zTest_1Prop, yhden osuuden z-testi	198
zTest_2Prop, kahden osuuden z-testi	199
zTest_2Samp, kahden otoksen z-testi	199

X

χ^2 Cdf()	28
χ^2 GOF	28
χ^2 Pdf()	29