

TI-Nspire™ CX CAS Sovelluksen käsikirja

Tärkeitä tietoja

Texas Instruments ei myönnä minkäänlaista nimenomaista tai oletettua takuuta mukaan lukien rajoituksetta oletetut takuut soveltuvuudesta kaupankäynnin kohteeksi tai sopivuudesta tiettyyn tarkoitukseen koskien ohjelmia tai kirjallista aineistoa, jotka annetaan saataville "sellaisina kuin ne ovat". Texas Instruments ei ole missään tapauksessa vastuussa näiden aineistojen hankinnasta tai käytöstä aiheutuvista erityisistä, rinnakkaisista, satunnaisista tai välillisistä vahingoista. Kannemuodosta riippumatta Texas Instrumentsin vastuu rajoittuu vain ja ainoastaan kyseisen tuotteen hankintahintaan. Texas Instruments ei myöskään ole velvoitettu vastaamaan minkäänlaisiin vaatimuksiin johtuen näiden materiaalien käytöstä muiden osapuolten toimesta.

© 2024 Texas Instruments Incorporated

Todelliset tuotteet saattavat erota hieman mukana tulevista kuvista.

Sisällys

Lausekemallit	1
Luettelo aakkosjärjestyksessä	7
A	7
B	15
C	20
D	36
E	45
F	53
G	60
I	71
L	78
M	94
N	102
O	110
P	113
Q	119
R	123
S	138
T	156
U	168
V	169
W	170
X	172
Z	173
Symbolit	179
TI-Nspire™ CX II – Piirtokomennot	199
Kuvaajien ohjelmointi	199
Kuvaajanäyttö	199
Oletusnäkyvä ja asetukset	200
Kuvaajanäytön virheviestit	201
Virheelliset komennot kuvaajatilassa	201
C	203
D	204
F	207
G	209
P	210
J	212
I	214

Tyhjät elementit	215
Matemaattisten lausekkeiden syöttäminen pikavalintojen avulla	217
EOS-järjestelmän (yhtälökäyttöjärjestelmä) hierarkia	218
TI-Nspire CX II – TI-Basic-ohjelmointiominaisuudet	220
Automaattinen sisennys Ohjelmointieditorissa	220
TI-Basicin virheviestien parannukset	220
Vakiot ja arvot	223
Virhekoodit ja viestit	224
Varoituskoodit ja -viestit	231
Yleistä	233
Indeksi	234

Lausekemallit

Lausekemallien avulla voit syöttää matemaattisia lausekkeita normaalissa matemaattisessa muodossa. Lisätessäsi mallin se näkyy syöterivillä siten, että elementtien syöttökohtissa 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 180.

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 181.

Esimerkki:

$$2^3 = 8$$

Neliöjuurimalli

ctrl **x²** painikkeet



Huomaa: Katso myös $\sqrt{}$ (neliöjuuri), sivu 188.

Esimerkki:

N:s juuri -malli

ctrl **^** painikkeet



Huomaa: Katso myös root(), sivu 134.

Esimerkki:

e eksponenttimalli

e^x painikkeet

e^{\square}

Esimerkki:

e -kantainen eksponenttifunktio korotettuna potenssiin

Huomaa: Katso myös $e^{\wedge}()$, sivu 45.

Logaritmimalli

ctrl 10^x painike

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

Esimerkki:

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

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

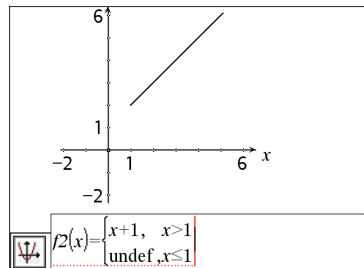
Huomaa: Katso myös $\log()$, sivu 90.

Paloittain määritellyn funktion malli (2-osainen)

Katalogi > $\left\{ \begin{array}{l} \square \\ \square \end{array} \right.$

$\left\{ \begin{array}{l} \square \\ \square \end{array} \right.$

Esimerkki:



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

Huomaa: Katso myös $\text{piecewise}()$, sivu 114.

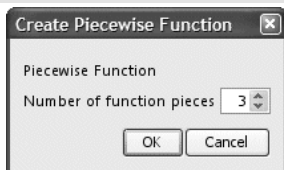
Paloittain määritellyn funktion malli (N-osainen)

Katalogi > $\left\{ \begin{array}{l} \square \\ \square \\ \square \end{array} \right.$

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

Esimerkki:

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



Huomaa: Katso myös `piecewise()`, sivu 114.

Yhtälöparin malli



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

Huomaa: Katso myös `system()`, sivu 156.

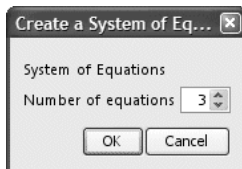
Esimerkki:

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

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

N-osaisen yhtälöryhmän malli

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



Huomaa: Katso myös `system()`, sivu 156.

Esimerkki:

Katso yhtälöparin (2 yhtälöä) mallin esimerkki.

Itseisarvon malli



Huomaa: Katso myös `abs()`, sivu 7.

Esimerkki:

Itseisarvon malli

Katalogi > 

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

dd°mm'ss.ss'' -malli

Katalogi > 

0°00'

Esimerkki:

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

Matriisimalli (2 x 2)

Katalogi > 

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

Esimerkki:

Luo 2 x 2 -matriisin.

Matriisimalli (1 x 2)

Katalogi > 

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

Esimerkki:

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

Matriisimalli (2 x 1)

Katalogi > 

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

Esimerkki:

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

Matriisimalli (m x n)

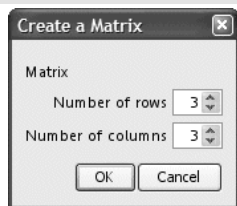
Katalogi > 

Malli tulee näkyviin määritettyäsi rivien ja sarakkeiden lukumäärän syöttöruutuun.

Esimerkki:

Matriisimalli (m x n)

Katalogi > 



$$\text{diag} \begin{pmatrix} 4 & 2 & 6 \\ 1 & 2 & 3 \\ 5 & 7 & 9 \end{pmatrix} \quad [4 \ 2 \ 9]$$

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

Summan malli (Σ)

Katalogi > 

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

Esimerkki:

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

Huomaa: Katso myös $\Sigma()$ (**sumSeq**), sivu 189.

Tulon malli (Π)

Katalogi > 

$$\prod_{i=0}^{} (i)$$

Esimerkki:

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

Huomaa: Katso myös $\Pi()$ (**prodSeq**), sivu 189.

Ensimmäisen derivaatan malli

Katalogi > 

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

Esimerkki:

Ensimmäisen derivaatan malli

Katalogi > 

Huomaa: Katso myös **d()** (derivaatta), sivu 188.

Toisen derivaatan malli

Katalogi > 

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

Esimerkki:

Huomaa: Katso myös **d()** (derivaatta), sivu 188.

Määrätyn integraalin malli

Katalogi > 

$$\int_a^b \square dx$$

Esimerkki:

Luettelo aakkosjärjestyksessä

Komennot, joiden nimiä ei voi järjestää aakkosjärjestykseen (esimerkiksi +, ! ja >), on esitetty tämän kappaleen lopussa alkaen sivulta (sivu 179). Ellei toisin ole mainittu, kaikki tämän kappaleen esimerkit on suoritettu laskimen oletustilassa, eikä mitään muuttujia ole määritetty.

A

abs()

Katalogi > 

abs(Lista1)⇒*lista*

abs(Matriisi1)⇒*matriisi*

Laskee argumentin itseisarvon.

Huomaa: Katso myös **Itseisarvon malli**, sivu 3.

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*

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

NPmt on taulukon maksuerien lukumäärä. Taulukko alkaa ensimmäisestä maksuerästä.

N, I, PV, Pmt, FV, PpY, CpY ja *PmtAt* on kuvattu TVM-argumenttien taulukossa, sivu 166.

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

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

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

oletusarvot ovat samat kuin TVM-funktioilla.

pyör:Arvo määrittää pyöristyksessä käytettävien desimaalien määrän. Oletusarvo=2.

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

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

Voit käyttää tulosmatriisia syötteenä muissa lyhennyslaskutoimituksissa $\Sigma\text{Int}()$ ja $\Sigma\text{Prn}()$, sivu 190, sekä **bal()**, sivu 16.

and

BooleanLaus1 **and**

BooleanLaus2 \Rightarrow *Boolean*
lausekeBooleanLista1

and *BooleanLista2* \Rightarrow *Boolean*
listaBooleanMatriisi1

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.

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

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:

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

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

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

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

angle()

Laskee argumentin kulman tulkiten argumentin kompleksiluvuksi.

Astekulmatilassa:

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

Graadikulmatilassa:

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

Radiaanikulmatilassa:

$\text{angle}(\text{Lista1}) \Rightarrow \text{lista}$

$\text{angle}(\text{Matriisi1}) \Rightarrow \text{matriisi}$

Laskee listan tai matriisin *Lista1*:n tai *Matriisi1*:n elementtien kulmista tulkiten jokaisen elementin kompleksiluvuksi, joka edustaa kaksiulotteista suorakulmakoordinaattipistettä.

ANOVA


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

Suorittaa yksisuuntaisen varianssianalyysin 2-20 perusjoukon keskiarvon vertailua varten. Tulosten yhteenveto tallentuu *stat.results*-muuttuun. (Katso sivu 151.)

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
stat.SS	Ryhmiin neliöiden summa
stat.MS	Ryhmiin keskineliöt
stat.dfError	Virheiden vapausasteet
stat.SSError	Virheiden neliöiden summa
stat.MSError	Virheiden keskineliö
stat.sp	Poolattu keskihajonta
stat.xbarlist	Listojen syötteiden keskiarvo
stat.CLowerList	95 %:n luottamusvälit jokaisen syötelistan keskiarvolle
stat.CUpperList	95 %:n luottamusvälit jokaisen syötelistan keskiarvolle

ANOVA2way

Katalogi > 

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

Laskee kaksisuuntaisen varianssianalyysin 2-10 perusjoukon keskiarvojen vertaamiseksi. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 151.)

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 neliöiden summa
stat.MS	Saraketekijän keskineliöt

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

SARAKETEKIJÄN tulokset

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

RIVITEKIJÄN tulokset

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

VUOROVAIKUTUKSEN tulokset



Tulosmuuttuja	Kuvaus
stat.FInteract	F-tilasto, vuorovaikutuksen F-tilasto
stat.PValInteract	Vuorovaikutuksen todennäköisyysarvo

Tulosmuuttuja	Kuvaus
stat.dfInteract	Vuorovaikutuksen vapausasteet
stat.SSInteract	Vuorovaikutuksen neliöiden summa
stat.MSInteract	Vuorovaikutuksen keskineliöt

VIRHEIDEN tulokset

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

ans

  painikkeet

ans ⇒ *arvo*


56 56

Näyttää viimeksi sievennetyn lausekkeen tuloksen.



56+4 60

60+4 64

approx()

Katalogi > 

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

Tämä vastaa argumentin syöttämistä ja painikkeen   painamista.

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

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

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

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

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

approx(Lista1) ⇒ *lista*

approx(Matriisi1) ⇒ *matriisi*

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

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

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

► approxFraction()Katalogi > *Lista* ► **approxFraction**([*Tol*])⇒*lista*

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

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

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

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

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

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

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

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

approxRational()Katalogi > *Lista*, *tol*)⇒*lista*

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

Matriisi, *tol*)⇒*matriisi*

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

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 28.**arccosh()**Katso $\cosh^{-1}()$, sivu 29.**arccot()**Katso $\cot^{-1}()$, sivu 30.**arcoth()**Katso $\coth^{-1}()$, sivu 31.

arccsc()

Katso $\text{csc}^{-1}()$, sivu 33.

arccsch()

Katso $\text{csch}^{-1}()$, sivu 33.

arcsec()

Katso $\text{sec}^{-1}()$, sivu 138.

arcsech()

Katso $\text{sech}^{-1}()$, sivu 138.

arcsin()

Katso $\text{sin}^{-1}()$, sivu 146.

arcsinh()

Katso $\text{sinh}^{-1}()$, sivu 147.

arctan()

Katso $\text{tan}^{-1}()$, sivu 157.

arctanh()

Katso $\text{tanh}^{-1}()$, sivu 158.

augment()

Katalogi > 

augment(Lista1, Lista2)⇒lista

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

Luo uuden listan, joka on *Lista2* liitettynä *Lista1*:n loppuun.

augment()

Katalogi > 

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

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

1 2	$\rightarrow m1$	1 2
3 4		3 4
5	$\rightarrow m2$	5
6		6
augment(<i>m1</i> , <i>m2</i>)		1 2 5
		3 4 6

avgRC()

Katalogi > 

avgRC(*Laus1*, *Muutt* [=Arvo] [, *Askel*]) \Rightarrow *lauseke*

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

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

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

Laskee erotusosamäärän eteenpäin (keskimääräisen muutosnopeuden).

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

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.

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

B

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

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

Lyhennysfunktio, joka laskee määritetyn maksuerän jälkeen jäljellä olevan velkasaldon.

N, *I*, *PV*, *Pmt*, *FV*, *PpY*, *CpY* ja *PmtAt* on kuvattu TVM-argumenttien taulukossa, sivu 166.

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

N, *I*, *PV*, *Pmt*, *FV*, *PpY*, *CpY* ja *PmtAt* on kuvattu TVM-argumenttien taulukossa, sivu 166.

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

pyörArvo määrittää pyöristyksessä käytettävien desimaalien määrän. Oletusarvo=2.

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 7.

Huomaa: Katso myös $\Sigma\text{Int}()$ ja $\Sigma\text{Prn}()$, sivu 190.

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

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

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

bal(4,*tbl*) 1674.27

Kokonaisluku1 ► Base2 ⇒ *kokonaisluku*

256 ► Base2

0b100000000

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

0h1F ► Base2

0b11111

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

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

Negatiiviset luvut näytetään kahden komplementteina. Esimerkki:

-1 näkyy muodossa

0hFFFFFFFFFFFFFFFF heksadesimaalisessa kantalukutilassa

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

-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.

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

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

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

► Base10 (► Kantaluku10)

Kokonaisluku1

► Base10 ⇒ kokonaisluku

0b10011 ► Base10	19
0h1F ► Base10	31

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

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

0b *binaariluku*
 0h *heksadesimaaliluku*

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

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

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

►Base16 (►Kantaluku16)

Kokonaisluku1

►Base16⇒*kokonaisluku*

256►Base16

0h100

0b111100001111►Base16

0hFOF

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

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

0b *binaariluku*

0h *heksadesimaaliluku*

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

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

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

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

Jos syötät desimaalikonaisluvun, joka on etumerkillisen, 64 bitin binaarimuodon lukualueen ulkopuolella, laskin käyttää symmetristä modulo-operaatiota, jotta arvo saadaan oikealle alueelle. Lisätietoja, katso ►Base2, sivu 17.

binomCdf()

binomCdf(n,p)⇒*lista*

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

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

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

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

binomPdf()

binomPdf(n,p)⇒*lista*

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

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

C**ceiling()**

Laskee lähimmän kokonaisluvun, joka on \geq argumentti.

ceiling(.456)

1.

Argumentti voi olla reaali- tai kompleksiluku.

Huomaa: Katso myös **floor()**.

ceiling(Lista1) ⇒ lista

ceiling(Matriisi1) ⇒ matriisi

Laskee listan tai matriisin jokaisen elementin ylärajasta.

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

centralDiff()

centralDiff(Laus1, Muutt [=Arvo]

[,Askel]) ⇒ lauseke

centralDiff(Laus1, Muutt

[,Askel]) | Muutt=Arvo ⇒ lauseke

centralDiff(Laus1, Muutt [=Arvo]

[,Lista]) ⇒ lista

centralDiff(Lista1, Muutt [=Arvo]

[,Askel]) ⇒ lista

centralDiff(Matriisi1, Muutt [=Arvo]

[,Askel]) ⇒ matriisi

Laskee numeerisen derivaatan käyttäen keskeiserotusomäärän kaavaa.

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

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

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

Huomaa: Katso myös **avgRC()**.

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

char(38)

"&"

char(65)

"A"

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

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

Laskee χ^2 -testin tarkasteltavan matriisin *ObsMatriisi* sisältämän kaksisuuntaisen lukemataulukon arvojen välisestä assosiaatiosta. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 151.)

Lisätietoja matriiissa olevien tyhjien elementtien vaikutuksesta, katso Tyhjät elementtsivulla sivu 215.

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

 χ^2 Cdf()Katalogi > 

χ^2 Cdf(alaraja,yläraja,df)⇒*luku*, jos *alaraja* ja *yläraja* ovat lukuja, *lista*, jos *alaraja* ja *yläraja* ovat listoja

chi2Cdf(alaraja,yläraja,df)⇒*luku*, jos *alaraja* ja *yläraja* ovat lukuja, *lista*, jos *alaraja* ja *yläraja* ovat listoja

χ^2 Cdf()

Katalogi > 

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

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

Lisätietoja listassa olevien tyhjien elementtien vaikutuksesta, katso Tyhjät elementitsivulla sivu 215.

χ^2 GOF

Katalogi > 

χ^2 GOF *obsLista,expLista,df*

chi2GOF *obsLista,expLista,df*

Suorittaa testin, jolla varmistetaan, että otoksen data on tiettyä jakaumaa vastaavasta perusjoukosta. *obsList* on lukemalista, ja sen tulee sisältää kokonaislukuja. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 151.)

Lisätietoja listassa olevien tyhjien elementtien vaikutuksesta, katso Tyhjät elementitsivulla sivu 215.

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

χ^2 Pdf()

Katalogi > 

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 *XArvon* arvolla määritetyille vapausasteelle *df*.

Lisätietoja listassa olevien tyhjien elementtien vaikutuksesta, katso Tyhjät elementtsivulla sivu 215.

clearAZ

clearAZ

Poistaa kaikki yksikirjaimiset muuttujat nykyiseltä tehtäväalueelta.

Jos yksi tai useampia muuttujia on lukittu, tämä komento aiheuttaa virheilmoituksen ja poistaa vain lukitsemattomat muuttujat. Katso **unLock**, sivu 168.

ClrErr

ClrErr

Poistaa virhetilan ja nolaa järjestelmän muuttujan *errCode*.

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

Huomaa: Katso myös **PassErr**, sivu 114, ja **Try**, sivu 162.

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

Esimerkki **ClrErr**-komennosta, katso esimerkki 2 **Try**-komennon kohdalla, sivu 162.

colAugment()

Katalogi > 

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

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

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

colDim()

Katalogi > 

colDim(*Matriisi*) \Rightarrow *lauseke*

Laskee *Matriisin* sisältämien sarakkeiden lukumäärän.

Huomaa: Katso myös **rowDim()**.

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

colNorm()

Katalogi > 


colNorm(*Matriisi*) \Rightarrow *lauseke*

Laskee maksimiarvon *Matriisin* sarakkeissa olevien elementtien itseisarvojen summista.

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

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

conj()

Katalogi > 

conj(*Lista1*) \Rightarrow *lista*

conj(*Matriisi1*) \Rightarrow *matriisi*

Laskee argumentin liittokompleksiluvun.

Huomaa: Kaikkia määrittämättömiä muuttujia käsitellään reaali muuttujina.

constructMat

(

Laus,
Muutt1,*Muutt2,numRivit,numSarakkeet*)⇒*matriisi*

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

Laskee matriisiin argumentteihin perustuen.

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

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

CopyVar**CopyVar** *Muutt1, Muutt2*Define $a(x)=\frac{1}{x}$ Done**CopyVar** *Muutt1., Muutt2.*Define $b(x)=x^2$ Done

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













CopyVar *a,c: c(4)* $\frac{1}{4}$ CopyVar *b,c: c(4)* 16

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

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

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

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.

<i>aa.a</i> :=45	45																
<i>aa.b</i> :=6.78	6.78																
CopyVar <i>aa.</i> , <i>bb.</i>	Done																
getVarInfo()	<table border="1"> <tbody> <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> </tbody> </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														

corrMat()

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

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

cos()

cos(*Lista1*)⇒*lista*

Astekulmatilassa:

cos(*Lista1*) määrittää listan kaikkien *Lista1*:n sisältämien elementtien kosineista.

Graadikulmatilassa:

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


Radiaanikulmatilassa:

cos(*neliömatriisi1*)⇒*neliömatriisi*

Radiaanikulmatilassa:

Laskee *neliömatriisi1*:n matriisikosinin. Tämä ei ole sama kuin kunkin elementin kosinin laskeminen.

cos()

 painike

Kun skaalarista funktiota $f(A)$ käytetään *neliömatriisi*:een (A), tulos lasketaan algoritmeilla:

Laske A :n ominaisarvot (λ_i) ja ominaisvektorit (V_i).

neliömatriisi::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()

 painike

$\cos^{-1}(\text{Lista1}) \Rightarrow \text{lista}$

Astekulmatilassa:

$\cos^{-1}(\text{Lista1})$ laskee listan Lista1 :n jokaisen elementin käänteiskosineista.

Graadikulmatilassa:

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


Radiaanikulmatilassa:

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

$\cos^{-1}(\text{neliömatriisi1}) \Rightarrow \text{neliömatriisi}$

Radiaanikulmatilassa ja suorakulmakompleksimuodossa:

$\cos^{-1}()$

 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 diagonaloitavissa. Vastaus sisältää aina liukulukuja.

$$\cos^{-1}\begin{pmatrix} 1 & 5 & 3 \\ 4 & 2 & 1 \\ 6 & -2 & 1 \end{pmatrix}$$

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

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

$\cosh()$

Katalogi > 

$\cosh(Lista1) \Rightarrow lista$

$\cosh(Lista1)$ määrittää listan $Lista1$:n kunkin elementin hyperbolisista kosineista.

$\cosh(neliömatriisi1) \Rightarrow neliömatriisi$

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 diagonaloitavissa. Vastaus sisältää aina liukulukuja.

Astekulmatilassa:

Radiaanikulmatilassa:

$$\cosh\begin{pmatrix} 1 & 5 & 3 \\ 4 & 2 & 1 \\ 6 & -2 & 1 \end{pmatrix}$$

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

$\cosh^{-1}()$

Katalogi > 

$\cosh^{-1}(Lista1) \Rightarrow lista$

$\cosh^{-1}(Lista1)$ määrittää listan $Lista1$:n kunkin elementin käänteisistä hyperbolisista kosineista.

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

`coth-1(Lista1)` ⇒ lista

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

count()

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

Laskee elementtien kokonaismäärän argumenteille, jotka sieventyvät numeroarvoiksi.

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

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

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

Tyhjiä elementtejä ei huomioida. Lisätietoja tyhjihistä elementeistä, katso sivu 215.

countif()

`countif(Lista,Kriteerit)` ⇒ arvo

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

Kriteeri voi olla:

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

`countif({1,3,"abc",undef,3,1},3)` 2

Laskee niiden elementtien lukumäärän, jotka ovat yhtä kuin 3.

`countif({"abc","def","abc",3},"def")` 1

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

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

Laskee lukumäärään 1:n ja 3:n.

countif()

Katalogi > 

lukumäärään vain ne *Listan* elementit, jotka ovat alle 5.

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

Listassa olevia tyhjiä elementtejä ei huomioida. Lisätietoja tyhjiistä elementeistä, katso sivu 215.

Huomaa: Katso myös **sumIf()**, sivu 155, ja **frequency()**, sivu 58.

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

cPolyRoots(*Kertoinlista*) ⇒ lista

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

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

Huomaa: Katso myös **polyRoots()**, sivu 116.

crossP()

Katalogi > 

crossP(*Listat1*, *Listat2*) ⇒ lista

Määrittää listan *Listat1*:n ja *Listat2*:n ristitulosta.

Listat1:n ja *Listat2*:n on oltava samankokoiset, ja koon on oltava joko 2 tai 3.

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

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

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

crossP()

Katalogi > 

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

$\text{csc}(Lista1) \Rightarrow lista$

Astekulmatilassa:

Graadikulmatilassa:

Radiaanikulmatilassa:

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

 painike

$\text{csc}^{-1}(Lista1) \Rightarrow lista$

Astekulmatilassa:

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

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

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

Graadikulmatilassa:

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

Radiaanikulmatilassa:

csch()

Katalogi > 

$\text{csch}(Lista1) \Rightarrow lista$

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

Katalogi > 

$\text{csch}^{-1}(Lista1) \Rightarrow lista$

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

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

Laskee 3. asteen polynomiregression $y = a \cdot x^3 + b \cdot x^2 + c \cdot x + d$ listaista X ja Y frekvenssillä *Frekv*. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 151.)

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

X ja Y ovat riippumattomien ja riippuvien muuttujien listoja.

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

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

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

Lisätietoja listassa olevien tyhjiä elementtien vaikutuksesta, katso Tyhjet elementitsivulla sivu 215.

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	Muokatun X Listan sisältämä datapisteiden lista, jota käytetään regressiossa komentojen <i>Frekv</i> , <i>Luokkalista</i> ja <i>Sisällytä luokat</i> rajoitusten mukaisesti.

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

cumulativeSum()

Katalogi >

cumulativeSum(Lista1)⇒lista

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

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

cumulativeSum(Matriisi1)⇒matriisi

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

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

Tyhjä elementti listassa *Listal* tai matriisissa *Matriisi1* tuottaa tyhjän elementin tuloksena olevaan listaan tai matriisiin. Lisätietoja tyhjästä elementeistä, katso sivu 215.

Cycle

Katalogi >

Cycle

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 määrittysten syöttämisestä löytyvät tuotteen ohjekirjan Laskin-osiosta.

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

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

g() 5000

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.

D

dbd()

dbd(pvm1,pvm2)⇒arvo

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

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

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

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

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

Astekulmatilassa:

►DD⇒listaMatriisi1

►DD⇒matriisi

►DD

Katalogi >

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

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

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

Graadikulmatilassa:

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

Radiaanikulmatilassa:

$\{1.5\}$ ►DD	85.9437°
---------------	----------

►Decimal

Katalogi >

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)

Katalogi >

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.

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

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

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

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

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

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

Lohko voi olla joko yksi lauseke tai eri riveillä olevien lausekkeiden sarja. *Lohko* voi sisältää myös 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 38, ja **Define LibPub**, sivu 39.

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

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

Define LibPriv (Määritä LibPriv)

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

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

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

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 37, ja **Define LibPub**, sivu 39.

```
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 37, ja **Define LibPriv**, sivu 38.

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

DelVar *Muutt*.

Poistaa määritetyn muuttujan tai muuttujaryhmän muistista.

Jos yksi tai useampia muuttujia on lukittu, tämä komento aiheuttaa virheilmoituksen ja poistaa vain lukitsemattomat muuttujat. Katso **unLock**, sivu 168.

DelVar *Muutt*. poistaa kaikki *Muutt:n* jäsenet. muuttujaryhmä (kuten tilastollinen *stat.nn* tulosta tai muuttujaa, jotka on luotu funktiolla **LibShortcut()**). Piste (.) tässä **DelVar**-komenton 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"> <tbody> <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> </tbody> </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()

delVoid(*Lista1*)⇒*lista*

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

Antaa tuloksena listan, jossa on listan *Lista1* sisältö, ja kaikki tyhjät elementit on poistettu.



Lisätietoja tyhjästä elementistä, katso sivu 215.

det()

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

Laskee *neliömatriisin* determinantin.

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

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

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

diag()

diag(Lista) ⇒ matriisi

diag(rivimatriisi) ⇒ matriisi

diag(sarakematriisi) ⇒ matriisi

$$\text{diag}([2 \ 4 \ 6]) \quad \begin{bmatrix} 2 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 6 \end{bmatrix}$$

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

diag(neliömatriisi) ⇒ rivimatriisi

$$\begin{bmatrix} 4 & 6 & 8 \\ 1 & 2 & 3 \\ 5 & 7 & 9 \end{bmatrix} \quad \begin{bmatrix} 4 & 6 & 8 \\ 1 & 2 & 3 \\ 5 & 7 & 9 \end{bmatrix}$$

Laskee rivimatriisin, joka sisältää elementit *neliömatriisin* päälävistäjästä.

neliömatriisi:n on oltava neliö.

$$\text{diag}(\text{Ans}) \quad \begin{bmatrix} 4 & 2 & 9 \end{bmatrix}$$

dim()

dim(Lista) ⇒ kokonaisluku

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

Laskee *Listan* mitat.

dim(Matriisi) ⇒ lista

$$\text{dim} \left(\begin{bmatrix} 1 & -1 \\ 2 & -2 \\ 3 & 5 \end{bmatrix} \right) \quad \{3,2\}$$

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

dim()

Katalogi > 

dim(*Merkkijono*)⇒*kokonaisluku*

Laskee merkkijonon *Merkkijono* sisältämien merkkien lukumäärän.

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

Disp

Katalogi > 

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

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

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

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

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

DispAt

Katalogi > 

DispAt *int,lauske1* [,*lauske2* ...] ...

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

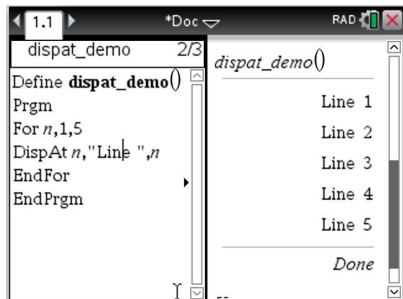
Rivinumero voidaan määritellä lausekkeeksi.

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

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

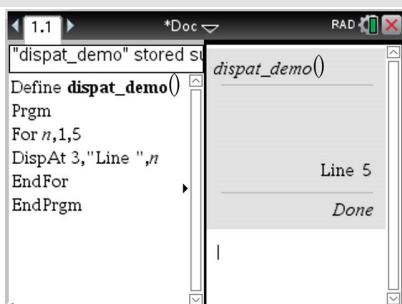
DispAt

Esimerkki



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

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



```

1.1 | *Doc | RAD | X
"disp_at_demo" stored s
Define disp_at_demo()
Prgm
For n,1,5
DispAt 3,"Line ",n
EndFor
EndPrgm
disp_at_demo()
Line 5
Done
  
```

Havainnollistavia esimerkkejä:

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

Virhetilat:

Virheviestit DispAt-rivinumeron on oltava lukujen 1 ja 8 välillä	Kuvaus Lauseke arvioi rivinumeron välin 1–8 (mukaan lukien) ulkopuolella
Liian vähän argumentteja	Toiminnosta tai komennosta puuttuu yksi tai useampi argumentti
Ei argumentteja	Sama kuin nykyinen "syntaksivirhe" - dialogi
Liian monta argumenttia	Rajoita argumenttia. Sama virhe kuin Disp.
Virheellinen tietotyyppi	Ensimmäisen argumentin on oltava numero.
Mitätön: DispAt mitäton	"Hello World" Datatyyppivirhe on mitätoity (jos soittopyyntö on määritelty)

►DMS

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

Tulkitsee argumentin kulmana ja näyttää vastaavan DMS-luvun (DDDDDD°MM'SS.ss"). DMS-muoto (asteet, minuutit, sekunnit) on kuvattu kohdissa °, ', " sivulla sivu 194 .

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

Astekulmatilassa:

(45.371) ►DMS	$45^{\circ}22'15.6''$
$\{\{45.371,60\}\}$ ►DMS	$\{45^{\circ}22'15.6'',60^{\circ}\}$

dotP()

dotP(Lista1, Lista2)⇒lauseke


Laskee kahden listan "pistetulon".

dotP(Vektori1, Vektori2)⇒lauseke



Laskee kahden vektorin "pistetulon".

Kummankin on oltava rivivektoreita, tai kummankin on oltava sarakevektoreita.

E

 $e^{\wedge}()$  painike

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

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

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

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

Laskee **e:n** arvon korotettuna *Lista l:n* jokaisen elementin potenssiin.

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

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

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

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

eff()

Katalogi > 

$\text{eff}(\text{nimelliskorko}, CpY) \Rightarrow \text{arvo}$

$\text{eff}(5.75,12)$

5.90398

Talouseläskentatoiminto, 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 106.

eigVc()

eigVc(neliömatriisi) ⇒ matriisi

Suorakulmakompleksimuodossa:

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




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

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

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

eigVc(mI)

$$\begin{bmatrix} -0.800906 & 0.767947 & (\\ 0.484029 & 0.573804+0.052258 \cdot i & 0.5738 \\ 0.352512 & 0.262687+0.096286 \cdot i & 0.2626 \end{bmatrix}$$

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

eigVI()

eigVI(neliömatriisi) ⇒ lista

Suorakulmakompleksimuodossa:

Laskee listan reaalisien tai kompleksien *neliömatriisien* ominaisarvoista.

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

eigVI(mI)

$$\{-4.40941, 2.20471+0.763006 \cdot i, 2.20471-0 \cdot i\}$$

eigV()**Katalogi** > 

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

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

Else**Katso If, sivu 71.****Elseif****Katalogi** > **If BooleanLaus1 Then***Lohko1***Elseif BooleanLaus2 Then***Lohko2*

⋮

Elseif BooleanLausN Then*LohkoN***Endif**

⋮

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

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

Return 5

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

Return 3

EndIf

EndFunc

*Done***EndFor****Katso For, sivu 56.****EndFunc****Katso Func, sivu 60.****EndIf****Katso If, sivu 71.**

euler ()

Katalogi > 

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

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

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

Käyttää Eulerin menetelmää
järjestelmän ratkaisuun

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

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

Differentiaaliyhtälö:

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

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

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

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

Yhtälöryhmä:

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

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

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

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

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

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

eval()

Laitevalikko

eval(*Expr*) ⇒ *string*

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

Aseta RGB-ledin sininen väri puolelle intensiteetille.

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

Palauta sininen väri OFF-tilaan.

Argumentin *Expr* on sievennyttävä reaalityyppi.

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

Done

eval()-argumentin on sievennyttävä reaalityyppi.

```
Send "SET LED eval("4") TO ON"
```

"Error: Invalid data type"

Ohjelmoi punainen väri voimistumaan

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

Suorita ohjelma.

```
fadein()
```

Done

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

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

iostr.SendAns
iostr.GetAns
iostr.GetStrAns

Huomio: Katso myös **Get** (sivu 62), **GetStr** (sivu 68), ja **Send** (sivu 139).

Exit

Funktion listaus:

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

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

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

```
Define g() $\Rightarrow$ Func
  Local temp,i
  0  $\rightarrow$  temp
  For i,1,100,1
    temp+i  $\rightarrow$  temp
  If temp>20 Then
    Exit
  EndIf
  EndFor
EndFunc
```

$g()$ 21

exp() **painike**

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

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

exp(Lista1) \Rightarrow lista

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

exp(neliömatriisi1) \Rightarrow neliömatriisi

Laskee *neliömatriisi1*:n matriisiekspONENTIN. Tämä ei ole sama kuin laskettaessa **e** korotettuna kunkin elementin mukaiseen potenssiin. Laskentamenetelmä on kuvattu kohdassa **cos()**.

neliömatriisi1: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

expr()Katalogi > 

expr(Merkkijono) \Rightarrow lauseke

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

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

Laskee eksponentiaalisen regression $y = a \cdot (b)^x$ listoista *X* ja *Y* frekvenssillä *Frekv*. Tulosten yhteenvedo tallentuu *stat.results*-muuttujaan. (Katso sivu 151.)

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

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

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

Luokka on luokkakoodien lista vastaavalle *X*- ja *Y*-datalle.

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

Lisätietoja listassa olevien tyhjen elementtien vaikutuksesta, katso Tyhjet elementitsivulla sivu 215.

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

Tulosmuuttuja	Kuvaus
stat.ResidTrans	Muunnettujen tietojen lineaariseen sovitukseen liittyvät jäännökset
stat.XReg	Muokatun <i>X Lista</i> :n sisältämä datapisteiden lista, jota käytetään regressiossa komentojen <i>Frekv</i> , <i>Luokkalista</i> ja <i>Sisällytä luokat</i> rajoitusten mukaisesti
stat.YReg	Muokatun <i>Y Lista</i> :n sisältämä datapisteiden lista, jota käytetään regressiossa komentojen <i>Frekv</i> , <i>Luokkalista</i> ja <i>Sisällytä luokat</i> rajoitusten mukaisesti
stat.FreqReg	Komentoja <i>stat.XReg</i> ja <i>stat.YReg</i> vastaava frekvenssilista

F



factor()

Katalogi > 

factor(rationaaliluku) laskee rationaaliluvun, joka on jaettu jaottomiin tekijöihin. Sekalukujen kohdalla laskenta-aika pitenee eksponentiaalisesti toiseksi suurimman tekijän sisältämien numeroiden määrän suhteen. Esimerkiksi 30-numeroisen kokonaisluvun tekijöihin jakaminen voi kestää pitempään kuin vuorokauden ja 100-numeroisen luvun pitempään kuin vuosisadan.

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

Pysäytä laskenta käsin,

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

Jos haluat pelkästään määrittää, onko jokin luku jaoton, käytä sen sijaan komentoa **isPrime()**. Se on paljon nopeampi, erityisesti jos *rationaaliluku* ei ole jaoton, ja jos toiseksi suurimmassa tekijässä on enemmän kuin viisi numeroa.

F Cdf

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

FCdf

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

Laskee F-jakauman
todennäköisyyden *alarajan* ja *ylärajan*
välillä määritellylle *dfOsoittajalle*
(vapausaste) ja *dfNimittäjälle*.

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

Fill

matriisiMuuttujan on oltava valmiiksi
olemassa.

1 2	\rightarrow <i>amatrix</i>	1 2
3 4		3 4

Fill 1.01,*amatrix* Done

<i>amatrix</i>	1.01 1.01
	1.01 1.01

listaMuuttujan on oltava valmiiksi
olemassa.

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

Fill 1.01,*alist* Done

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

FiveNumSummary

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

Antaa lyhennetyn version 1 muuttujan
tilastoista listalle *X*. Tulosten yhteenveto
tallentuu *stat.results*-muuttujaan. (Katso
sivu 151.)

X edustaa datan sisältävää listaa.

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

Luokka on numeeristen luokkakoodien lista vastaaville X :n arvoille.

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

Tyhjä elementti jossakin listassa X , *Frekv* tai *Luokka* saa aikaan, että kaikkien listojen vastaava elementti on tyhjä. Lisätietoja tyhjistä elementeistä, katso sivu 215.

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()

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

$$\text{floor}(-2.14) \quad -3.$$

Argumentti voi olla reaali- tai kompleksiluku.

floor(Lista1) ⇒ lista

floor(Matriisi1) ⇒ matriisi

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

Määrittää listan tai matriisin jokaisen elementin alarajasta.

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

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

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

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

Muutt ei saa olla järjestelmän muuttuja.

Askel voi olla positiivinen tai negatiivinen. Oletusarvo on 1.

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

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

Define $g()$ =Func	Done
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	
$g()$	5050

format()

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 säättyy kolmella kerrolliseksi, ja desimaalipiste siirtyy 0, 1 tai 2 numeroa oikealle.

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"

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

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

fPart()

fPart(*Lausl*) \Rightarrow lauseke

fPart(*Lista1*) \Rightarrow lista

fPart(*Matriisi1*) \Rightarrow matriisi

$fPart(-1.234)$	-0.234
-----------------	--------

$fPart(\{1, -2.3, 7.003\})$	$\{0, -0.3, 0.003\}$
-----------------------------	----------------------

Laskee argumentin murtolukuosan.

Kun kyseessä on lista tai matriisi, laskee elementtien murtolukuosat.

Argumentti voi olla reaali- tai kompleksiluku.

FPdf()

FPdf(*XArvo*, *dfOsoitt*, *dfNimitt*) \Rightarrow luku, jos *XArvo* on luku, *lista*, jos *XArvo* on lista

Laskee F-jakauman todennäköisyyden *XArvon* kohdalle määritetyille *dfOsoittajalle* (vapausasteet) ja *dfNimittäjälle*.

freqTable►list()

freqTable►list

(*Lista1*, *freqv*, *Kokonaisluku*, *Lista*) \Rightarrow lista

$freqTable►list(\{1, 2, 3, 4\}, \{1, 4, 3, 1\})$	$\{1, 2, 2, 2, 3, 3, 3, 4\}$
--	------------------------------

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

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

Listal voi olla mikä tahansa kelvollinen lista.

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

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

Tyhjiä elementtejä ei huomioida. Lisätietoja tyhjästä elementistä, katso sivu 215.

frequency()

`frequency(Listal, lokerotListal) => lista`

Luo listan, joka sisältää *Listal*:n elementtien lukumäärät. Lukumäärät perustuvat alueisiin (lokeroihin), jotka määritetään kohtaan *lokerotListal*.

Jos *lokerotListal* 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 *lokerotListal*.

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

Vastauksen selitys:

2 *Datalistan* elementtiä on ≤ 2.5

4 *Datalistan* elementtiä on > 2.5 ja ≤ 4.5

3 *Datalistan* elementtiä on > 4.5

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

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

Niitä *Listal*:n elementtejä, joita ei voi "lokeroida", ei huomioida. Tyhjiä elementtejä ei myöskään huomioida. Lisätietoja tyhjistä elementeistä, katso sivu 215.

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

Huomaa: Katso myös **countlf()**, sivu 31.

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 yhteenvedo tallentuu *stat.results*-muuttujaan. (Katso sivu 151.)

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 tyhjiä elementtien vaikutuksesta, katso Tyhjät elementtisivulla sivu 215.

Tulosmuuttuja	Kuvaus
stat.F	Laskettu F -tilasto datasekvenssille
stat.PVal	Alin merkitsevyytaso, jolla nollahypoteesi voidaan hylätä
stat.dfNumer	osoittajan vapausasteet = $n1-1$
stat.dfDenom	nimittäjän vapausasteet = $n2-1$
stat.sx1, stat.sx2	Otoksen keskihajonnat <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.

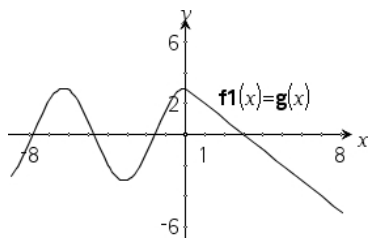
Lohko voi olla yksi lauseke tai sarja lausekkeita, jotka on erotettu toisistaan kaksoispisteellä (:), tai sarja eri riveillä olevia lausekkeita. Funktio voi käyttää **Return**-ohjetta tietyn vastauksen laskemiseen.

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

Määritä paloittain määrittely funktio:

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

Funktion $g(x)$ kuvaajan piirtämisen tulos



G

gcd()

Katalogi >

$\text{gcd}(\text{Arvo1}, \text{Arvo2}) \Rightarrow$ lauseke

$\text{gcd}(18,33)$

3

gcd()

Katalogi > 

Laskee kahden argumentin suurimman yhteisen jakajan. Kahden murtoluvun **gcd** on niiden osoittajien **gcd** jaettuna nimittäjien **lcm**:llä.

Auto or Approximate (Automaattinen tai likimääräinen) -tilassa murtoluvun liukulukujen **gcd** on 1.0.

gcd(Lista1, Lista2)⇒*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 onnistumistodennäköisyyllä *p*.

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

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

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

Arvo täytyy kysyä:

- Etukäteen **Send "READ ..."** - komennolla.
— tai —
- Upottamalla **"READ ..."** kysy valinnaisena *kehotemerkkijono*argumenttina. Tämä menetelmä antaa sinun käyttää yksittäistä komentoa kysyäksesi arvoa ja hakeaksesi sen.

Tapahtuu implisiittinen yksinkertaistus. Esimerkiksi vastaanotettu merkkijono "123" tulkitaan numeeriseksi arvoksi. Säilyttääksesi merkkijonon käytä toimintoa **GetStr** toiminnon **Get** sijaan.

Jos sisällytät valinnaisen argumentin *statusVar*, sille määrätään arvo toimenpiteen onnistumisen perusteella. Arvo nolla merkitsee, ettei tietoa ole vastaanotettu.

Järjestyksessä toisessa syntaksissa *func* ()-argumentti sallii ohjelman tallentaa vastaanotetun merkkijonon funktiomääritelmänä. Tämä syntaksi toimii ikään kuin ohjelma suorittaisi komennon:

Määrittele *func*(*arg1,* ...*argn*) = vastaanotettu merkkijono

Sen jälkeen ohjelma voi käyttää määriteltyä funktiota *func*().

Huomio: Komentoa **Get** voi käyttää käyttäjän määrittelemän ohjelman

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

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

Upota READ-kysely komentoon **Get**.

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

sisällä, mutta ei funktion sisällä.

Huomio: Katso myös **GetStr**, sivu 68 ja **Send** sivu 139.

getDenom()

Katalogi > 

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

getKey()

Katalogi > 

getKey([0|1]) ⇒ **returnString**

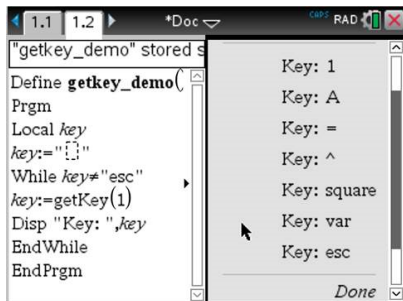
Kuvaus: **getKey()** – mahdollistaa TI-Basic -ohjelmalle näppäimistön – kannettava laite, pöytätietokone ja emulaattori pöytätietokoneella.

Esimerkki:

- alas painettu näppäin := **getKey()** palauttaa näppäimen tai tyhjän merkkijonon, jos mitään näppäintä ei ole painettu. Tämä komento palaa välittömästi.
- alas painettu näppäin := **getKey(1)** odottaa, kunnes jotakin näppäintä painetaan. Tämä komento pysäyttää ohjelman suorituksen, kunnes jotakin näppäintä on painettu.

`getKey()`

Esimerkki:



Näppäimen pitäminen painettuna:

Kannettava laite/Emulaattorinäppäin	Pöytätietokone	Palauta arvo
Esc	Esc	"esc"
Kosketuslevy – Yläpainallus	N/A	"ylös"
Päällä	N/A	"päävalikko"
Scratchapps	N/A	muistilehtiö

Kannettava laite/Emulaattorinäppäin	Pöytätietokone	Palauta arvo
Kosketuslevy – Vasen painallus	N/A	"vasen"
Kosketuslevy – Keskipainallus	N/A	"keskus"
Kosketuslevy – Oikea painallus	N/A	"oikea"
Doc	N/A	"doc"
Tabulaattori	Tabulaattori	"Tabulaattori"
Kosketuslevy – Alapainallus	Nuoli alaspäin	"alas"
Päävalikko	N/A	"päävalikko"
Ctrl	Ctrl	ei palautusta
Vaihto	Vaihto	ei palautusta
Var	N/A	"var"
Poista	N/A	"poista"
=	=	"="
Trigonometria	N/A	Trigonometria
0:sta 9:ään	0-9	"0" ... 9
Sapluunat	N/A	"sapluuna"
Lista	N/A	"lista"
^	^	"^"
X ²	N/A	"neliö"
/ (jakonäppäin)	/	"/"
* (kertonäppäin)	*	"*"
e ^x	N/A	"eksponentti"
10 ^x	N/A	"10voima"
+	+	"+"
-	-	"_"
(("("
))	")"

Kannettava laite/Emulaattorinäppäin	Pöytätietokone	Palauta arvo
.	.	"."
(-)	N/A	"-" (negatiivinen merkki)
Syötä	Syötä	"syötä"
ee	N/A	"E" (tieteellinen merkintä E)
a – z	a–z	alfa = kirjainmerkki painettu (pieni kirjain) ("a" – "z")
vaihto a–z	vaihto a–z	alfa = kirjainmerkki painettu "A" - "Z"
		Huomaa: Ctrl-vaihto lukitsee isot kirjaimet
?!	N/A	"?!"
pii	N/A	"pii"
Lippu	N/A	ei palautusta
,	,	","
Palautus	N/A	Palautus
Välilyönti	Välilyönti	Välilyönti
Ei pääsyä	Näppäimet erikoismerkeille, kuten @,!,^, etc.	Kirjainmerkki on palautettu
N/A	Toimintonäppäimet	Kirjainmerkkejä ei ole palautettu
N/A	Erityiset näytön kontrollinäppäimet	Kirjainmerkkejä ei ole palautettu
Ei pääsyä	Muita näytön näppäimiä, jotka eivät ole käytettävissä laskimessa, kun getkey() odottaa näppäimen painallusta. ({, },, ;, ...)	Sama kirjainmerkki, jonka saat Notesista (ei matemaattisessa kentässä)

Huomaa: On tärkeää huomata, että ohjelman **getKey ()** läsnäolo muuttaa ohjelmaa tiettyjen tapahtumien käsittelyä. Joitakin näistä kuvataan jäljempänä.

Lopeta ohjelma ja käsittele tapahtuma – aivan kuin jos käyttäjä poistuisi ohjelmasta painamalla **ON** -näppäintä

"Tuki " jäljempänä tarkoittaa – Järjestelmä toimii odotetusti – ohjelma jatkuu.

Tapahtuma	laite	Työpöytä – TI-Nspire™ Ohjelmisto opiskelijoille
Pikakysely	Lopeta ohjelma, käsittele tapahtuma	Sama kuin kannettava laite (TI-Nspire™ Student Software, TI-Nspire™ Navigator™ NC Teacher Software – vain)
Etätiedoston hallinta (Sisältää lähetyksen "Exit Press 2 Test" -tiedoston toisesta kämmenlaitteesta tai työpöydän kannettavasta laitteesta)	Lopeta ohjelma, käsittele tapahtuma	Sama kuin kannettava laite. (TI-Nspire™ Student Software, TI-Nspire™ Navigator™ NC Teacher Software-vain)
Lopeta oppitunti	Lopeta ohjelma, käsittele tapahtuma	Tuki (TI-Nspire™ Student Software, TI-Nspire™ Navigator™ NC Teacher Software-vain)

Tapahtuma	Laite	Työpöytä – TI-Nspire™ Kaikki versiot
TI-Innovator™ Hub Yhdistä / katkaise	Tuki – onnistuu komennolla TI-Innovator™ Hub. Kun olet lopettanut ohjelman, TI-Innovator™ Hub työskentelee edelleen kämmenlaitteen kanssa.	Sama kuin kannettava laite

getLangInfo()

Katalogi > 

getLangInfo() ⇒ *merkkijono*

getLangInfo()

"en"

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

englanti = "en"
 tanska = "da"
 saksa = "de"
 suomi = "fi"
 ranska = "fr"
 italia = "it"
 hollanti = "nl"
 flaami = "nl_BE"
 norja = "no"
 portugali = "pt"
 espanja = "es"
 ruotsi = "sv"

getLockInfo()

getLockInfo(*Muutt*)⇒*arvo*

Määrittää muuttujan *Muutt* nykyisen lukittu/lukitsematon-tilan.

arvo =0: *Muutt* on lukitsematon tai sitä ei ole olemassa.

arvo =1: *Muutt* on lukittu eikä sitä voi muuttaa tai poistaa.

Katso **Lock**, sivu 89, ja **unLock**, sivu 168.

<i>a</i> :=65	65
Lock <i>a</i>	Done
getLockInfo(<i>a</i>)	1
<i>a</i> :=75	"Error: Variable is locked."
DelVar <i>a</i>	"Error: Variable is locked."
Unlock <i>a</i>	Done
<i>a</i> :=75	75
DelVar <i>a</i>	Done

getMode()

getMode(*TilanNimiKokonaisluku*)⇒*arvo*

getMode(0)⇒*lista*

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

getMode(0) 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 141.

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
Vektorimuoto	6	1=Suorakulma, 2=Syylinteri, 3=Pallo
Kantaluku	7	1=Desimaali, 2=Heksagonaalinen, 3=Binaarinen

getNum()

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

GetStr

GetStr [*kehote*merkkijono,] *var*[, *statusVar*] Katso esimerkit kohdasta **Get**.

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

Ohjelmointikomento: Toimii samalla tavalla kuin **Get**-komento, mutta vastaanotettu arvo tulkitaan aina merkkijonoksi. **Get**-komento kuitenkin tulkitsee vastauksen lausekkeeksi, jollei sitä merkitä lainausmerkkien ("") sisään.

Huomio: Katso myös **Get**, sivu 62 ja **Send** sivu 139.

getType()

Katalogi > 

getType(*var*)⇒*merkkijono*

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

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

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

getVarInfo()

Katalogi > 

getVarInfo()⇒*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".

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

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

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

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

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

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

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

Goto

Goto tunnusnimi

Siirtää ohjauksen tunnuksen *tunnusnimi*.

tunnusnimi on määritettävä samassa funktiossa käyttäen **Lbl**-ohjetta.

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

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

Laus 1 ► Grad ⇒ lauseke

Astekulmatilassa:

Muuttaa *Laus 1*:n graadikulmaan. $(1.5) \blacktriangleright \text{Grad}$ $(1.66667)^9$

Huomaa: Voit syöttää tämän operaattorin tietokoneen näppäimistöltä kirjoittamalla $\text{e} \blacktriangleright \text{Grad}$.

Radiaanikulmatilassa:

 $(1.5) \blacktriangleright \text{Grad}$ $(95.493)^9$

/

identity()

Luettelo > 

identity(kokonaisluku) ⇒ matriisi

identity(4)	1	0	0	0
	0	1	0	0
	0	0	1	0
	0	0	0	1

Laskee identiteettimatriisin, jonka koko on kokonaisluku.

Kokonaisluvun on oltava positiivinen kokonaisluku.

Jos

Luettelo > 

Jos BooleanLaus
Ilmaisut

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

Jos BooleanLaus Niin
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 (:) merkki.

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

Jos *BooleanLaus*, **niin**

Lohko1

Tai

Lohko2

EndIf

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

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

Lohko1 ja *Lohko2* voivat olla yksi lauseke.

Jos *BooleanLaus1*, **niin**

Lohko1

Jos taas *BooleanLaus2*, **niin**

Lohko2

:

Jos taas *BooleanLausN*, **niin**

LohkoN

EndIf

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

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

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

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

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

ifFn()

ifFn(*BooleanLaus*,*Arvo_Jos_tosi* [,*Arvo_Jos_epätosi* [,*Arvo_Jos_tuntematon*]])
 \Rightarrow lauseke, lista tai matriisi

Laskee *BooleanLaus* (jokaiselle *BooleanLaus*) elementille) ja antaa tuloksen noudattaen seuraavia sääntöjä:

- *BooleanLaus* voi testata yksittäisen arvon, listan tai matriisin.
- Jos jokin *BooleanLaus* -elementti on tosi, laskee vastaavan elementin lausekkeesta *Arvo_Jos_tosi*.
- Jos jokin *BooleanLaus* -elementti on epätosi, laskee vastaavan elementin

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

1:n testiarvo on alle 2.5, joten sen vastaava *Arvo_Jos_tosi* -elementti arvolle 5 kopioidaan vastausten listaan.

2:n testiarvo on alle 2.5, joten sitä vastaava *Arvo_Jos_tosi*-elementti arvolle 6 kopioidaan vastausten listaan.

lausekkeesta *Arvo Jos epätosi*. Jos jätät pois lausekkeen *Arvo Jos epätosi*, laskee määrittelemättömäksi.

- Jos *BooleanLaus* elementti ei ole tosi eikä epätosi, laskee vastaavan elementin *Arvo If unknown*. Jos jätät pois *Arvo If unknown*, laskee määrittelemättömäksi.
- Jos funktion **josFn()** toinen, kolmas tai neljäs argumentti on yksi lauseke, Boolean testiä sovelletaan jokaiseen sijaintiin Boolean lausekkeessa *BooleanLaus*.

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

3:n testi-arvo ei ole alle 2.5, joten sitä vastaava *Arvo Jos epätosi*-elementti 10 kopioidaan vastausten listaan.

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

Arvo Jos tosi on yksittäinen arvo ja vastaa mitä tahansa valittua sijaintia.

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

Arvoa *Arvo Jos epätosi* ei ole määritelty. Käytetään merkintää EIMäär

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

Yksi elementti valittu lausekkeesta *Arvo Jos tosi*. Yksi elementti valittu lausekkeesta *Arvo Jos epätosi*.

imag()

Laskee argumentin imaginaarisen osan.

imag(Lista1) ⇒ lista

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

Laskee listan alkutekijöiden imaginaarisista osista.

imag(Matriisi1) ⇒ matriisi

Laskee matriisin alkutekijöiden imaginaarisista osista.

Epäsuora operaattori

inString()

Luettelo > 

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

<code>inString("Hello there", "the")</code>	7
<code>inString("ABCEFG", "D")</code>	0

Laskee merkin paikan merkkijonossa *srcMerkkijono*, jossa merkkijonon *alaMerkkijono* ensimmäinen esiintyminen alkaa.

Alku, jos se sisältyy, määrää merkin paikan siinä merkkijonossa *srcMerkkijono*, josta haku alkaa. Oletusarvo = 1 (*srcMerkkijonon* ensimmäinen merkki).

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

int()

Luettelo > 

int(*Lista1*) ⇒ *lista*

int(*Matriisi1*) ⇒ *matriisi*

<code>int(-2.5)</code>	-3.
<code>int([-1.234 0 0.37])</code>	<code>[-2. 0 0.]</code>

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

Argumentti voi olla reaali- tai kompleksiluku.

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

intDiv()

Luettelo > 

intDiv(*Luku1*, *Luku2*) ⇒ *kokonaisluku*

intDiv(*Lista1*, *Lista2*) ⇒ *lista*

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

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

Laskee lausekkeen *Luku1* ÷ *Luku2*) etumerkillisen kokonaislukuosan.

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

interpoloi ()

Luettelo > 

Interpoloi(*xArvo*, *xList*, *yList*, *yPrimeList*) ⇒ *lista*

Tällä toiminnolla suoritetaan seuraavaa:




Kun ilmoitetaan *xList*, *yList=f(xList)* ja *yPrimeList=f'(xList)* jollekin tuntemattomalle funktiolle *f*, käytetään kuutiointerpolanttia funktion *f* määrittelemiseksi arvolla *xArvo*.

Oletetaan, että *xList* on monotonisesti kasvavien tai laskevien numeroiden lista, mutta tämän toiminnon tuloksena saattaa olla arvo, vaikka se ei olisikaan sitä. Tämä toiminto käy läpi listan *xList* etsien väliä [*xList*[*i*], *xList*[*i*+1]], joka sisältää arvon *xValue*. Jos se löytää tällaisen välin, se palauttaa interpoloidun arvon funktiolle *f(xValue)*; muuten se antaa tuloksen **määrittelemätön**.

Sarjojen *xList*, *yList* ja *yPrimeList* on oltava samansuuruiset ≥ 2 ja sisällettävä lausekkeita, jotka sievenyvät luvuiksi.

Differentiaaliyhtälö:
 $y' = -3 \cdot y + 6 \cdot t + 5$ ja $y(0) = 5$

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

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

Käytä interpolate()-funktioita laskeaksesi funktion arvot *x*-arvolistalle:

```
xvalueList:=seq(i,i,0,10,0.5)
{0,0.5,1,1.5,2,2.5,3,3.5,4,4.5,5,5.5,6,6.5,7,7.5,8,8.5,9,9.5,10}
xlist:=mat▶list(rk[1])
{0.,1.,2.,3.,4.,5.,6.,7.,8.,9.,10.}
ylist:=mat▶list(rk[2])
{5.,3.19499,5.00394,6.99957,9.00593,10.9978}
yprimeList:=-3*y+6*t+5|y=ylist and t=xlist
{-10.,1.41503,1.98819,2.00129,1.98221,2.00606}
interpolate(xvalueList,xlist,ylist,yprimeList)
{5.,2.67062,3.19499,4.02782,5.00394,6.00011}
```

invχ²()

Luettelo > 

invχ²(*Ala*,*df*)

invChi2(*Ala*,*df*)

Laskee käänteisen kumulatiivisen χ^2 (chi-neliö) -todennäköisyysfunktion, joka on määritelty vapauden asteella, *df* annettulle käyrän alapuoliselle alalle *Ala*.

invF()

Luettelo > 

invF(*Ala*,*dfOsoitt*,*dfNimitt*)

vakioF(*Area*,*dfOsoitt*,*dfNimitt*)

Laskee käänteisen kumulatiivisen F-jakaumafunktion, jolle on määritelty *dfOsoitt* ja *dfNimitt*, annettulle käyrän alapuoliselle alueelle *Ala*.

invBinom()

Luettelo > 

invBinom

(CumulativeProb, NumTrials, Prob, OutputForm) ⇒ asteikkomuoto tai matriisi

Käänteinen binomi. Johtuen kokeiden (NumTrials) ja kunkin kokeen todennäköisyydestä onnistua (Prob), tämä toiminto laskee onnistumisten minimimäärän, ksiten, että arvo k on suurempi tai yhtä suuri kuin kumuloituva todennäköisyys (CumulativeProb).

OutputForm=0, näyttää tuloksen asteikkomuodossa (oletus).

OutputForm=1, näyttää tuloksen matriisina.

Esimerkki: Mary ja Kevin pelaavat noppapeliä. Maryn on arvattava, miten monta kertaa numero 6 enintään esiintyy 30 heittoa kohti. Jos numero 6 esiintyy yhtä monta kertaa tai vähemmän, Mary voittaa. Lisäksi, mitä pienempi on hänen arvaamansa määrä, sitä suuremmat ovat hänen voittonsa. Mikä on pienin määrä, jonka Mary voi arvata, jos hän haluaa voittamisen todennäköisyyden olevan suurempi kuin 77 %?

invBinom(0.77,30, $\frac{1}{6}$)	6				
invBinom(0.77,30, $\frac{1}{6}$,1)	<table border="1"><tr><td>5</td><td>0.616447</td></tr><tr><td>6</td><td>0.776537</td></tr></table>	5	0.616447	6	0.776537
5	0.616447				
6	0.776537				

invBinomN()

Luettelo > 

invBinomN(CumulativeProb, Prob, NumSuccess, OutputForm) ⇒ asteikkomuoto tai matriisi

N:n suhteen käänteisesti binominen. Johtuen menestyksen todennäköisyydestä kussakin kokeessa (Prob), onnistumisten määrä (NumSuccess), tämä funktio laskee kokeiden vähimmäismäärän N siten, että arvo, N , on vähemmän tai yhtä suuri kuin annettu kumulatiivinen todennäköisyys (CumulativeProb).

OutputForm=0, näyttää tuloksen asteikkomuodossa (oletus).

OutputForm=1, näyttää tuloksen matriisina.

Esimerkki: Monique harjoittelee koriin heittoa koripallossa. Hän tietää kokemuksesta, että hänen mahdollisuutensa tehdä kori on 70 %. Hän suunnittelee harjoittelevansa, kunnes hän on tehnyt 50 koria. Kuinka montaa koria hänen on yritettävä varmistaakseen, että todennäköisyys tehdä ainakin 50 koria on enemmän kuin 0,99?

invBinomN(0.01,0.7,49)	86				
invBinomN(0.01,0.7,49,1)	<table border="1"><tr><td>85</td><td>0.010451</td></tr><tr><td>86</td><td>0.00709</td></tr></table>	85	0.010451	86	0.00709
85	0.010451				
86	0.00709				

invNorm()

Luettelo > 

invNorm(Ala[,μ[,σ]])

invNorm()

Luettelo >

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

invt()

Luettelo >

invt(*Ala,df*)

Laskee käänteisen kumulatiivisen student *t*-todennäköisyysfunktion, jonka määräävät vapausaste, *df* ja annettu alue *Ala* käyrän alapuolella.

iPart()

Luettelo >

iPart(*Luku*) \Rightarrow kokonaisluku**iPart**(*List1*) \Rightarrow lista**iPart**(*Matriisi1*) \Rightarrow matriisi

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

Laskee argumentin kokonaisosan.

Laskee sarjoille ja matriiseille kunkin elementin kokonaisosan.

Argumentti voi olla reaali- tai kompleksiluku.

irr()

Luettelo >

irr(*CF0, CFLista* [, *CFFrekv*]) \Rightarrow arvo

Talouseläämäntoiminto, joka laskee investoinnin sisäisen korkokannan.

CF0 on kassavirta alussa aikana 0; arvon on oltava kokonaisluku.

CFLista on lista kassavirtamääristä alun kassavirran *CF0* jälkeen.

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

CFFrekv on valinnainen lista, jossa kukin elementti määrää esiintymisfrekvenssin ryhmitetyille (peräkkäiselle) kassavirtamäärälle, joka on *CFFrekv*:n vastaava alkutekijä. Oletusarvo on 1; jos syötät arvoja, niiden on oltava positiivisia kokonaislukuja ja < 10 000.

Huomaa: Katso myös *mirr()*, sivu 98.

isPrime()

isPrime(Luku) ⇒ *Boolean vakiolauseke*

Laskee totuusarvon tosi tai epätosi ilmaistakseen, onko *luku* kokonaisluku ≥ 2 , joka on tasan jaollinen vain itsellään ja ykkösellä (1)

Jos *Luku* on pitempi kuin 306 numeroa, eikä siinä ole tekijöitä ≤ 1021 , kaava *isPrime(Luku)* näyttää virheilmoituksen.

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

<i>isPrime(5)</i>	true
<i>isPrime(6)</i>	false

Funktio, jolla etsitään seuraava jaoton luku määrätyn luvun jälkeen_

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

isVoid()

isVoid(Muutt) ⇒ *Boolean vakiolauseke*
isVoid(Laus) ⇒ *Boolean vakiolauseke*
isVoid(Lista) ⇒ *lista Boolean vakiolausekkeista*

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

Lisätietoja tyhjästä elementeistä, katso sivu 215.

<i>a:=_</i>	-
<i>isVoid(a)</i>	true
<i>isVoid({1,_,3})</i>	{false,true,false}

L

Lbl tunnusnimi

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

Ohjeella **Siirry tunnusnimi** voit siirtää ohjauksen kyseistä tunnusta välittömästi seuraavaan ohjaukseen.

tunnusnimellä on samat nimeämissäännöt kuin muuttujan nimellä.

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

Define $g()$ =Func

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

lcm()Katalogi > **lcm(Luku1, Luku2)**⇒*lauseke***lcm(Lista1, Lista2)**⇒*lista***lcm(Matriisi1, Matriisi2)**⇒*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.

 $lcm(6,9)$

18

$$lcm\left(\left\{\frac{1}{3}, -14, 16\right\}, \left\{\frac{2}{15}, 7, 5\right\}\right) \quad \left\{\frac{2}{3}, 14, 80\right\}$$
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(Lista1[, Num])⇒*lista* $left(\{1, 3, 2, 4\}, 3)$ $\{1, 3, 2\}$

Määrittää vasemmanpuoleisimmat *Num*-elementit, jotka sisältyvät listaan *Lista1*.

Jos jätät pois komennon *Num*, määrittää kaiken listasta *Lista1*.

left(*Vertailu*) \Rightarrow *lauseke*

Laskee yhtälön tai epäyhtälön vasemman puolen.

```
left(x<3) x
```

libShortcut()

libShortcut(*KirjNimiMerkkijono*,
PikavalNimiMerkkijono
[, *KirjYksitLippu*]) \Rightarrow *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 26.

Muuttujaryhmän poistaminen, katso **DelVar** sivulla sivu 40.

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

```
getVarInfo("linalg2")
  [clearmat "FUNC" "LibPub "
   gauss1  "PRGM" "LibPriv "
   gauss2  "FUNC"  "LibPub " ]
```

```
libShortcut("linalg2","la")
  {1a.clearmat,1a.gauss2}
```

```
libShortcut("linalg2","la",1)
  {1a.clearmat,1a.gauss1,1a.gauss2}
```

LinRegBx

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

Laskee lineaarisen regressioy = a+b · x-listoista *X* ja *Y* frekvenssillä *Frekv*. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 151.)

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

X ja Y ovat riippumattomien ja riippuvien muuttujien listoja.

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

Luokka on luokkakoodien lista vastaavalle X - ja Y -datalle..

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

Lisätietoja listassa olevien tyhjiä elementtien vaikutuksesta, katso Tyhjat elementtsivulla sivu 215.

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

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

Laskee lineaarisen regression $y = m \cdot x + b$ listoista X ja Y frekvenssillä *Frekv.* Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 151.)

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

X ja Y ovat riippumattomien ja riippuvien muuttujien listoja.

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

Luokka on luokkakoodien lista vastaavalle X - ja Y -datalle.

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

Lisätietoja listassa olevien tyhjiä elementtien vaikutuksesta, katso Tyhjat elementtsivulla sivu 215.

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 <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

LinRegtIntervals $X, Y, F[, 0, CTaso]]]$

Kulmakerroin. Laskee tason C luottamusvälin kulmakertoimelle.

LinRegtIntervals $X, Y, F[, 1, Xarvo[, CTaso]]]$

Vaste. Laskee ennustetun y :n arvon, tason C ennustevälin yhdelle havainnolle ja tason C luottamusvälin keskiarvovasteelle.

Tulosten yhteenveto tallentuu stat.results-muuttujaan. (Katso sivu sivu 151.)

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 Tyhjä elementtisivulla sivu 215.

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

Vain Kulmakerroin-tyyppi

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

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

Vain Vaste-tyyppi

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

LinRegtTest

Katalogi > 

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

Laskee lineaarisen regression X - ja Y -listoista ja suorittaa t -testin kulmakertoimen β ja korrelaatiokertoimen ρ arvosta yhtälölle $y = \alpha + \beta x$. Testaa nollahypoteesia $H_0: \beta = 0$ (vastaavasti, $\rho = 0$) johonkin kolmesta vaihtoehdoisesta hypoteesista.

Kaikkien listojen on oltava samankokoisia.

X ja Y ovat riippumattomien ja riippuvien muuttujien listoja.

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

Hypot on valinnainen arvo, joka määrittää yhden kolmesta hypoteesista, johon nollahypoteesia ($H_0: \beta = \rho = 0$) testataan.

Kun $H_a: \beta \neq 0$ ja $\rho \neq 0$ (oletus), aseta $Hypot=0$

Kun $H_a: \beta < 0$ ja $\rho < 0$, aseta $Hypot < 0$

Kun $H_a: \beta > 0$ ja $\rho > 0$, aseta $Hypot > 0$

Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu sivu 151.)

Lisätietoja listassa olevien tyhjen elementtien vaikutuksesta, katso Tyhjät elementtsivulla sivu 215.

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

linSolve()

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

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

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

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

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

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

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

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

linSolve(*Lineaariyht1 ja Lineaariyht2*

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

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

Laskee ratkaisulistan muuttujille Muutt1 , Muutt2 , ...

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

Esimerkiksi yhtälön $\text{linSolve}(x=1 \text{ and } x=2, x)$ sieventäminen antaa tuloksena virheilmoituksen Argumenttivirhe.

 $\Delta\text{List}()$

$\Delta\text{List}(\text{Lista1}) \Rightarrow \text{lista}$

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

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

Määrittää listan, joka sisältää Lista1 :n peräkkäisten elementtien väliset erotukset. Jokainen Lista1 :n elementti vähennetään Lista1 :n seuraavasta elementistä. Tuloksena oleva lista on aina yhden elementin lyhyempi kuin alkuperäinen Lista1 .

list▶mat()

$\text{list}\blacktriangleright\text{mat}(\text{Lista } [, \text{elementtiäRivillä}]) \Rightarrow \text{matriisi}$

$\text{list}\blacktriangleright\text{mat}(\{1, 2, 3\})$	$\begin{bmatrix} 1 & 2 & 3 \end{bmatrix}$
$\text{list}\blacktriangleright\text{mat}(\{1, 2, 3, 4, 5\}, 2)$	$\begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 0 \end{bmatrix}$

Laskee matriisin, joka on täytetty rivi riviltä Listan elementeillä.

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

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

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

ln()

ctrl e^x painikkeet

ln(*Lista*) ⇒ *lista*

ln(2.) 0.693147

Määrittää argumentin luonnollisen logaritmin.

Jos kompleksilukumuodon tila on Reaali:

Jos kyseessä on lista, määrittää elementtien luonnolliset logaritmit.

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

ln(*neliomatriisi*) ⇒ *neliomatriisi*

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

Jos kompleksilukumuodon tila on Suorakulma:
Radiaanikulmatilassa ja suorakulmakompleksimuodossa:

ln $\begin{pmatrix} 1 & 5 & 3 \\ 4 & 2 & 1 \\ 6 & -2 & 1 \end{pmatrix}$

1.83145+1.73485 <i>i</i>	0.009193-1.49086
0.448761-0.725533 <i>i</i>	1.06491+0.623491 <i>i</i>
-0.266891-2.08316 <i>i</i>	1.12436+1.79018 <i>i</i>

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

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

LnReg

Katalogi > 

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

Laskee logaritmisien regression $y = a + b \cdot \ln(x)$ listoista *X* ja *Y* frekvenssillä *Frekv*. Tulosten yhteenveto tallentuu *stat.results*-muuttuun. (Katso sivu 151.)

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

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

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

Luokka on luokkakoodien lista vastaavalle X - ja Y -datalle.

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

Lisätietoja listassa olevien tyhjen elementtien vaikutuksesta, katso Tyhjät elementitsivulla sivu 215.

Tulosmuuttuja	Kuvaus
stat.RegEqn	Regressioyhtälö: $a+b \cdot \ln(x)$
stat.a, stat.b	Regressiokertoimet
stat.r ²	Muunnettujen tietojen lineaarimäärittelyn kerroin
stat.r	Muunnettujen tietojen korrelaatiokerroin ($\ln(x)$, y)
stat.Resid	Logaritmimalliin liittyvät jäännökset
stat.ResidTrans	Muunnettujen tietojen lineaariseen sovitukseen liittyvät jäännökset
stat.XReg	Muokatus <i>Y Lista</i> :n sisältämä datapisteiden lista, jota käytetään regressiossa komentojen <i>Frekv</i> , <i>Luokkalista</i> ja <i>Sisällytä luokat rajoitusten mukaisesti</i>
stat.YReg	Muokatus <i>Y Lista</i> :n sisältämä datapisteiden lista, jota käytetään regressiossa komentojen <i>Frekv</i> , <i>Luokkalista</i> ja <i>Sisällytä luokat rajoitusten mukaisesti</i>
stat.FreqReg	Komentoja <i>stat.XReg</i> ja <i>stat.YReg</i> vastaava frekvenssilista

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

Määrittää määritetyt *muuttujat* paikallisina muuttujina. Nämä muuttujat ovat olemassa vain funktion sievennyksen aikana, ja ne poistetaan, kun funktion suoritus päättyy.

Huomaa: Paikalliset muuttujat säästävät muistia, koska ne ovat olemassa vain väliaikaisesti. Lisäksi ne eivät häiritse mitään olemassa olevia globaalien muuttujien arvoja. Paikallisia muuttujia on käytettävä **For**-silmukoissa sekä tallennettaessa arvoja väliaikaisesti monirivisessä funktiossa, sillä globaalien muuttujien modifiointeja ei sallita funktiossa.

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

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

Lock**Lock***Muutt1*[, *Muutt2*] [, *Muutt3*] ...
Lock*Muutt*.

Lukitsee määritetyt muuttujat tai muuttujaryhmän. Lukittuja muuttujia ei voi muokata eikä poistaa.

Et voi lukita tai vapauttaa järjestelmän muuttujaa *Ans*, etkä voi lukita järjestelmän muuttujaryhmiä *stat*. tai *tvM*.

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

Katso **unLock**, sivu 168, ja **getLockInfo()**, sivu 67.

<i>a</i> :=65	65
Lock <i>a</i>	<i>Done</i>
getLockInfo(<i>a</i>)	1
<i>a</i> :=75	"Error: Variable is locked."
DelVar <i>a</i>	"Error: Variable is locked."
Unlock <i>a</i>	<i>Done</i>
<i>a</i> :=75	75
DelVar <i>a</i>	<i>Done</i>

Jos kompleksilukumuodon tila on Reali:

Jos kompleksilukumuodon tila on Suorakulma:

Huomaa: Katso myös **Logaritmimalli**, sivu 2.

Jos toinen argumentti jätetään pois, kantalukuna käytetään lukua 10.

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

Jos kantalukuargumentti jätetään pois, kantalukuna käytetään lukua 10.

Radiaanikulmatilassa ja suorakulmakompleksimuodossa:

$$\log_{10} \begin{pmatrix} 1 & 5 & 3 \\ 4 & 2 & 1 \\ 6 & -2 & 1 \end{pmatrix}$$

0.795387+0.753438 <i>i</i>	0.003993-0.6474 <i>i</i>
0.194895-0.315095 <i>i</i>	0.462485+0.2707 <i>i</i>
-0.115909-0.904706 <i>i</i>	0.488304+0.7774 <i>i</i>

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

Logistic

Katalogi > 

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

Laskee logistisen regression $y = (c/(1+a \cdot e^{-bx}))$ listoista *X* ja *Y* frekvenssillä *Frekv*.

Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 151.)

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

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

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

Luokka on luokkakoodien lista vastaavalle *X*- ja *Y*-datalle.

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

Lisätietoja listassa olevien tyhjen elementtien vaikutuksesta, katso Tyhjät elementtsivulla sivu 215.

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

LogisticD

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

Laskee logistisen regression $y = (c/(1+a \cdot e^{-bx})+d)$ listoista *X* ja *Y* frekvenssillä *Frekv* käyttäen tiettyä *Iteraatioiden* määrää. Tulosten yhteenvedo tallentuu *stat.results*-muuttujaan. (Katso sivu 151.)

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

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

Iteraatiot on valinnainen arvo, joka määrittää ratkaisun yrityskertojen enimmäismäärän. Mikäli se jätetään pois, käytetään arvoa 64. Suuremmilla arvoilla saadaan tyypillisesti parempi tarkkuus, mutta suoritus aika on pitempi ja päin vastoin.

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

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

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

Lisätietoja listassa olevien tyhjiä elementtien vaikutuksesta, katso Tyhjät elementtisivulla sivu 215.

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	Muokatun X 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	Muokatun Y 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

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
```

	<i>Done</i>
rollcount()	16
rollcount()	3

LU

LU *Matriisi*, *lMatriisi*, *uMatriisi*, *pMatriisi*[*Tol*]

Laskee Doolittlen LU (ala-ylä)-dekomponoinnin reaali- tai kompleksimatriisista. Alakolmiomatriisi tallentuu muuttujaan *lMatriisi*, yläkolmiomatriisi muuttujaan *uMatriisi* ja permutaatiomatriisi (joka kuvaa laskennan aikana tehdyt rivien vaihdot) muuttujaan *pMatriisi*.

$$lMatriisi \cdot uMatriisi = pMatriisi \cdot matriisi$$

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

- Jos käytät painikkeita tai **Automaattinen tai likimääräinen** -tilan valintaa Approximate (Likimääräinen), laskut suoritetaan liukulukuaritmetiikalla.

<table border="1"> <tr><td>6</td><td>12</td><td>18</td></tr> <tr><td>5</td><td>14</td><td>31</td></tr> <tr><td>3</td><td>8</td><td>18</td></tr> </table>	6	12	18	5	14	31	3	8	18	→ <i>m1</i>	<table border="1"> <tr><td>6</td><td>12</td><td>18</td></tr> <tr><td>5</td><td>14</td><td>31</td></tr> <tr><td>3</td><td>8</td><td>18</td></tr> </table>	6	12	18	5	14	31	3	8	18
6	12	18																		
5	14	31																		
3	8	18																		
6	12	18																		
5	14	31																		
3	8	18																		
LU <i>m1</i> , <i>lower</i> , <i>upper</i> , <i>perm</i>		<i>Done</i>																		
<i>lower</i>		<table border="1"> <tr><td>1</td><td>0</td><td>0</td></tr> <tr><td>5</td><td>1</td><td>0</td></tr> <tr><td>6</td><td></td><td></td></tr> <tr><td>1</td><td>1</td><td>1</td></tr> <tr><td>2</td><td>2</td><td>1</td></tr> </table>	1	0	0	5	1	0	6			1	1	1	2	2	1			
1	0	0																		
5	1	0																		
6																				
1	1	1																		
2	2	1																		
<i>upper</i>		<table border="1"> <tr><td>6</td><td>12</td><td>18</td></tr> <tr><td>0</td><td>4</td><td>16</td></tr> <tr><td>0</td><td>0</td><td>1</td></tr> </table>	6	12	18	0	4	16	0	0	1									
6	12	18																		
0	4	16																		
0	0	1																		
<i>perm</i>		<table border="1"> <tr><td>1</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>1</td></tr> </table>	1	0	0	0	1	0	0	0	1									
1	0	0																		
0	1	0																		
0	0	1																		

- Jos *Tol* jätetään pois tai sitä ei käytetä, oletusarvoinen toleranssi lasketaan seuraavasti:
 $5E-14 \cdot \max(\dim(\text{Matriisi})) \cdot \text{riviNorm}(\text{Matriisi})$

LU-dekomponointialgoritmi käyttää osittaista rivien vaihtoa (pivoting).

M

mat▶list()

 Katalogi > 

mat▶list(Matriisi)⇒lista

Luo listan, joka on täytetty *Matriisin* elementeillä. Elementit kopioidaan *Matriisista* rivi riviltä.

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

mat▶list ([1 2 3])	{1,2,3}
$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix} \rightarrow m1$	$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}$
mat▶list (m1)	{1,2,3,4,5,6}

max()

 Katalogi > 

max(Lista1, Lista2)⇒lista

max(Matriisi1, Matriisi2)⇒matriisi

Laskee kahden argumentin maksimiarvon. Jos argumentteina on kaksi listaa tai matriiseja, laskee listan tai matriisin, joka sisältää vastaavien elementtien kunkin parin maksimiarvon.

max(Lista)⇒lauseke

Laskee *lista*:n maksimielementin.

max(Matriisi1)⇒matriisi

Luo rivivektorin, joka sisältää *Matriisi1*:n jokaisen sarakkeen maksimielementin.

Tyhjiä elementtejä ei huomioida.

Lisätietoja tyhjistä elementeistä, katso sivu 215.

Huomaa: Katso myös **min()**.

max (2,3,1,4)	2.3
max ({1,2},{-4,3})	{1,3}

max ({0,1,-7,1.3,0.5})	1.3
-------------------------------	-----

max ($\begin{bmatrix} 1 & -3 & 7 \\ -4 & 0 & 0.3 \end{bmatrix}$)	[1 0 7]
---	---------

mean()Katalogi > **mean(Lista[, frekvLista])**⇒*lauseke*Laskee *Listan* sisältämien elementtien keskiarvon.Jokainen *frekvListan* elementti näyttää *Listan* vastaavien elementtien peräkkäisten esiintymien lukumäärän.**mean(MatriisiI[, frekvMatriisi])**⇒*matriisi*Luo rivivektorin kaikkien *MatriisiI*:n sarakkeiden keskiarvoista.Jokainen *frekvMatriisin* elementti näyttää *MatriisiI*:n vastaavien elementtien peräkkäisten esiintymien lukumäärän.

Tyhjiä elementtejä ei huomioida. Lisätietoja tyhjistä elementeistä, katso sivu 215.

$\text{mean}(\{0.2, 0.1, -0.3, 0.4\})$	0.26
$\text{mean}(\{1, 2, 3\}, \{3, 2, 1\})$	$\frac{5}{3}$

Suorakulmavektorimuodossa:

$\text{mean} \begin{pmatrix} 0.2 & 0 \\ -1 & 3 \\ 0.4 & -0.5 \end{pmatrix}$	$[-0.133333 \quad 0.833333]$
$\text{mean} \begin{pmatrix} 1 & 0 \\ 5 & 0 \\ -1 & 3 \\ 2 & -1 \\ 5 & 2 \end{pmatrix}$	$\begin{bmatrix} -2 & 5 \\ 15 & 6 \end{bmatrix}$
$\text{mean} \begin{pmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{pmatrix} \begin{bmatrix} 5 & 3 \\ 4 & 1 \\ 6 & 2 \end{bmatrix}$	$\begin{bmatrix} 47 & 11 \\ 15 & 3 \end{bmatrix}$

median()Katalogi > **median(Lista[, frekvLista])**⇒*lauseke*Laskee *Listan* elementtien mediaanin.Jokainen *frekvListan* elementti näyttää *Listan* vastaavien elementtien peräkkäisten esiintymien lukumäärän.**median(MatriisiI[, frekvMatriisi])**⇒*matriisi*Luo rivivektorin, joka sisältää *MatriisiI*:n sarakkeiden mediaanit.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

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

$\text{median} \begin{pmatrix} 0.2 & 0 \\ 1 & -0.3 \\ 0.4 & -0.5 \end{pmatrix}$	$[0.4 \quad -0.3]$
---	--------------------

elementtejä ei huomioida. Lisätietoja tyhjästä elementeistä, katso sivu 215.

MedMed

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

Laskee mediaani-mediaani-suorany = $(m \cdot x + b)$ listoista X ja Y frekvenssillä $Frekv$. Tulosten yhteenveto tallentuu *stat.results*-muuttuun. (Katso sivu 151.)

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

X ja Y ovat riippumattomien ja riippuvien muuttujien listoja.

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

Luokka on luokkakoodien lista vastaavalle X - ja Y -datalle.

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

Lisätietoja listassa olevien tyhjen elementtien vaikutuksesta, katso Tyhjet elementitsivulla sivu 215.

Tulosmuuttuja	Kuvaus
stat.RegEqn	Mediaani-mediaani-suoran yhtälö: $m \cdot x + b$
stat.m, stat.b	Mallin kertoimet
stat.Resid	Mediaani-mediaani-suoran jäännökset
stat.XReg	Muokatun Y Listan sisältämä datapisteiden lista, jota käytetään regressiossa komentojen <i>Frekv</i> , <i>Luokkalista</i> ja <i>Sisällytä</i> luokat rajoitusten mukaisesti

Tulosmuuttuja	Kuvaus
stat.YReg	Muokatun <i>Y Lista</i> :n sisältämä datapisteiden lista, jota käytetään regressiossa komentojen <i>Frekv</i> , <i>Luokkalista</i> ja <i>Sisälytät 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(Lista1, Lista2) ⇒ lista

min(Matriisi1, Matriisi2) ⇒ matriisi

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

Laskee kahden argumentin minimiarvon. Jos argumentteina on kaksi listaa tai matriisia, laskee listan tai matriisin, joka sisältää vastaavien elementtien kunkin parin minimiarvon.

$\text{min}(\text{Lista}) \Rightarrow \text{lauseke}$

$$\text{min}(\{0,1,-7,1.3,0.5\}) \quad -7$$

Laskee Listan minimielementin.

$\text{min}(\text{Matriisi}) \Rightarrow \text{matriisi}$

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

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

Huomaa: Katso myös **max()**.

mirr

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

$$\begin{aligned} \text{list1} &:= \{6000, -8000, 2000, -3000\} \\ &\quad \{6000, -8000, 2000, -3000\} \\ \text{list2} &:= \{2, 2, 2, 1\} \\ &\quad \{2, 2, 2, 1\} \\ \text{mirr}(4.65, 12, 5000, \text{list1}, \text{list2}) &\quad 13.41608607 \end{aligned}$$

Talouseläntötoiminto, joka laskee investoinnin modifioidun sisäisen korkokannan.

tal.arvo on kassavirtamääristä maksettava korkoprosentti.

uud.invest.arvo on korkokanta, jolla kassavirrat investoidaan uudelleen.

CF0 on alkuperäinen kassavirta aikana 0; arvon on oltava kokonaisluku.

CFLista on lista kassavirtamääristä ensimmäisen kassavirran *CF0* jälkeen.

CFFrekv on valinnainen lista, jossa jokainen elementti määrittää esiintymisfrekvenssin ryhmitetyille (peräkkäiselle) kassavirtamäärille, joka on *CFListan* vastaava elementti.

Oletusarvo on 1; jos syötät arvoja, niiden on oltava positiivisia kokonaislukuja < 10,000.

Huomaa: Katso myös **irr()**, sivu 77.

mod()

Katalogi > 

mod(Lista1, Lista2) ⇒ lista

mod(Matriisi1, Matriisi2) ⇒ matriisi

Laskee ensimmäisen argumentin modulo toinen argumentti identtisten yhtälöiden määrittelyn mukaisesti:

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

$$\text{mod}(x,y) = x - y \text{ floor}(x/y)$$

Kun toinen argumentti on ei-nolla, vastaus on periodinen tässä argumentissa. Vastaus on joko nolla tai samanmerkinen kuin toinen argumentti.

Jos argumentteina on kaksi listaa tai matriisia, laskee listan tai matriisin, joka sisältää vastaavien elementtien kunkin parin modulon (jakojäännöksen).

Huomaa: Katso myös **remain()**, sivu 129

$\text{mod}(7,0)$	7
$\text{mod}(7,3)$	1
$\text{mod}(-7,3)$	2
$\text{mod}(7,-3)$	-2
$\text{mod}(-7,-3)$	-1
$\text{mod}(\{12,-14,16\},\{9,7,-5\})$	$\{3,0,-4\}$

mRow()

Katalogi > 

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

mRowAdd()

Katalogi > 

Luo kopion *Matriisi1*:stä, jossa jokainen rivin *Indeksi2* elementti *Matriisi1*:ssä on korvattu seuraavasti:

MultReg

Katalogi > 

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

Laskee listan *Y* moninkertaisen lineaarisen regression listojen *X1*, *X2*, ..., *X10* suhteen. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 151.)

Kaikkien listojen on oltava samankokoisia.

Lisätietoja listassa olevien tyhjiin elementtien vaikutuksesta, katso Tyhjät elementitsivulla sivu 215.

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 määrittelyn kerroin
stat.yLista	\hat{y} Lista = $b_0+b_1 \cdot x_1+ \dots$
stat.Resid	Regressioyhtälön jäännökset

MultRegIntervals

MultRegIntervals $Y, X1[,X2[,X3, \dots$
 $[,X10]]], XArvoLista[,CTaso]$

Laskee ennustetun y :n arvon, tason C ennustevälin yhdelle havainnolle ja tason C luottamusvälin keskiarvovasteelle.

Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 151.)

Kaikkien listojen on oltava samankokoisia.

Lisätietoja listassa olevien tyhjiin elementtien vaikutuksesta, katso Tyhjät elementitsivulla sivu 215.

Tulosmuuttuja	Kuvaus
stat.RegEqn	Regressioyhtälö: $b_0+b_1 \cdot x_1+b_2 \cdot x_2+ \dots$
stat.y	Pisteen A arvio: $\hat{y} = b_0 + b_1 \cdot x_1 + \dots$ <i>XArvoListalle</i>
stat.dfError	Virheen vapausasteet
stat.CLower, stat.CUpper	Keskiarvovasteen luottamusväli
stat.ME	Luottamusvälin virhemarginaali
stat.SE	Keskiarvovasteen keskivirhe
stat.LowerPred, stat.UpperrPred	Yhden havainnon ennusteväli

Tulosmuuttuja	Kuvaus
stat.MEPred	Ennustevälin virhemarginaali
stat.SEPred	Ennusteen keskivirhe
stat.bList	Regressiokertoimien lista, {b0,b1,b3,...}
stat.Resid	Regressioyhtälön jäännökset

MultRegTests

Katalogi > 

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

Moninkertaisen lineaarisen regression testi laskee lineaarisen regression tietystä datasta ja määrittää kertoimille globaalin F -testin tilastot sekä t -testin tilastot.

Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu sivu 151.)

Lisätietoja listassa olevien tyhjen elementtien vaikutuksesta, katso Tyhjat elementtsivulla sivu 215.

Tulokset

Tulosmuuttuja	Kuvaus
stat.RegEqn	Regressioyhtälö: $b_0+b_1 \cdot x_1+b_2 \cdot x_2+ \dots$
stat.F	Globaalin F -testin tilasto
stat.PVal	Globaaliin F -tilastoon liittyvä P-arvo
stat.R ²	Moninkertaisen määrittelyn kerroin
stat.AdjR ²	Moninkertaisen määrittelyn säädetty kerroin
stat.s	Virheen keskijajonta
stat.DW	Durbin-Watsonin tilasto; käytetään määrittettäessä, esiintyykö mallissa ensimmäisen asteen automaattista korrelaatiota
stat.dfReg	Regression vapausasteet
stat.SSReg	Regression neliöiden summa
stat.MSReg	Regression keskineliö
stat.dfError	Virheen vapausasteet
stat.SSError	Virheen neliöiden summa

Tulosmuuttuja	Kuvaus
stat.MSError	Virheen keskineliö
stat.bList	{b0,b1,...} Kertoimien lista
stat.tList	t-tilastojen lista, yksi kullekin bListan sisältämälle kertoimelle
stat.PList	P-arvojen lista kullekin t-tilastolle
stat.SEList	Keskivirheiden lista bListan sisältämille kertoimille
stat.yLista	\hat{y} Lista = $b_0 + b_1 \cdot x_1 + \dots$
stat.Resid	Regressioyhtälön jäännökset
stat.sResid	Standardoidut jäännökset; saadaan jakamalla jäännös keskihajonnalla
stat.CookDist	Cookin etäisyys; jäännökseen ja tuottosuhteeseen perustuvan havainnon vaikutus
stat.Leverage	Miten kaukana riippumattoman muuttujan arvot ovat niiden keskiarvoista

N

nand (ei-ja)

ctrl **=** **näppäimet**

BooleanLaus1 nand BooleanLaus2 antaa vastauksena *Boolean lausekkeen BooleanList1*

nand BooleanList2 antaa vastauksena *Boolean listan BooleanMatriisi1*

nand BooleanMatriisi2 antaa vastauksena *Boolean matriisin*

Antaa vastauksena loogisen **and** operaation negaation kahdesta argumentista. Antaa vastauksena totuusarvon tosi, epätosi tai yhtälön sievennetyn muodon.

Listojen ja matriisien kohdalla vastauksena on vertailuja elementti elementiltä.

Kokonaisluku1 nand Kokonaisluku2 ⇒ *kokonaisluku*

$x \geq 3$ and $x \geq 4$	$x \geq 4$
$x \geq 3$ nand $x \geq 4$	$x < 4$

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

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

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

nCr()

Katalogi >

nCr(Lista1, Lista2)⇒lista

$$\text{nCr}(\{5,4,3\},\{2,4,2\}) \quad \{10,1,3\}$$

Laskee listan kombinaatioista kahden listan sisältämiin vastaaviin elementtipareihin perustuen. Argumenttien on oltava samankokoisia listoja.

nCr(Matriisi1, Matriisi2)⇒matriisi

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

Laskee matriisin kombinaatioista kahden matriisin sisältämiin vastaaviin elementtipareihin perustuen. Argumenttien on oltava samankokoisia matriiseja.

nDerivative()

Katalogi >

nDerivative(Laus1, Muutt=Arvo [,Aste])⇒arvo

$$\text{nDerivative}(|x|, x=1) \quad 1$$

$$\text{nDerivative}(|x|, x)=0 \quad \text{undef}$$

nDerivative(Laus1, Muutt[,Aste]) | Muutt=Arvo⇒arvo

$$\text{nDerivative}(\sqrt{x-1}, x)=1 \quad \text{undef}$$

Laskee numeerisen derivaatan käyttäen automaattisia derivointimenetelmiä.

nDerivative()

Katalogi > 

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

Derivaatan Asteen on oltava **1** tai **2**.

newList()

Katalogi > 

newList(*numElementit*) \Rightarrow *lista*

newList(4)	{0,0,0,0}
------------	-----------

Antaa tuloksena listan, jonka koko on *numElementit*. Jokainen elementti on nolla.

newMat()

Katalogi > 

newMat(*numRivit*,
numSarakkeet) \Rightarrow *matriisi*

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

Antaa tuloksena nollamatriisin, jonka koko on *numRivit* ja *numSarakkeet*.

nfMax()

Katalogi > 

nfMax(*Laus*, *Muutt*) \Rightarrow *arvo*

nfMax(*Laus*, *Muutt*, *alaraja*) \Rightarrow *arvo*

nfMax(*Laus*, *Muutt*, *alaraja*,
yläraja) \Rightarrow *arvo*

nfMax(*Laus*, *Muutt*) *alaraja* \leq *Muutt*
*\leq**yläraja* \Rightarrow *arvo*

$\text{nfMax}(x^2 - 2 \cdot x - 1, x)$	-1.
$\text{nfMax}(0.5 \cdot x^3 - x - 2, x, -5, 5)$	5.

Laskee muuttujan *Muutt* numeerisen ehdotusarvon, jossa lausekkeen *Laus* paikallinen maksimi esiintyy.

Jos ilmoitat *alarajan* ja *ylärajan*, funktio etsii suljetulta väliltä [*alaraja*,*yläraja*] paikallista maksimia.

nfMin()

Katalogi >

nfMin(Laus, Muutt)⇒arvo
nfMin(Laus, Muutt, alaraja)⇒arvo
nfMin(Laus, Muutt, alaraja, yläraja)⇒arvo
nfMin(Laus, Muutt) alaraja≤Muutt
≤yläraja⇒arvo

$\text{nfMin}(x^2+2\cdot x+5,x)$	-1.
$\text{nfMin}(0.5\cdot x^3-x-2,x,-5,5)$	-5.

Laskee muuttujan *Muutt* numeerisen ehdotusarvon, jossa lausekkeen *Laus* paikallinen minimi esiintyy.

Jos ilmoitat *alarajan* ja *ylärajan*, funktio etsii suljetulta väliltä [*alaraja*,*yläraja*] paikallista minimia.

nInt()

Katalogi >

nInt(Laus1, Muutt, Ala, Ylä)⇒lauseke

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

Jos integroitava funktio *Laus1* ei sisällä muita muuttujia kuin *Muutt*, ja jos *Ala* ja *Ylä* ovat vakioita, positiivinen ∞ tai negatiivinen ∞, tällöin **nInt()** laskee likiarvon lausekkeesta $\int(Laus1, Var, Ala, Ylä)$. 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.

Sijoiat **nInt()**-komentoa sisäkkäin, jos haluat suorittaa moninkertaisen numeerisen integroinnin. Integroinnin raja-arvot voivat riippua niiden ulkopuolella olevista integrointimuuttujista.

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

nom(*efektiivinenKorko*, *CpY*) \Rightarrow arvo

nom(5.90398,12)



5.75

Talouselaskentatoiminto, joka muuntaa efektiivisen vuosikoron *efektiivinenKorko* nimelliskoroksi, kun *CpY* määritetään korkojaksojen lukumääräksi vuodessa.

efektiivinenKorko on oltava reaaliluku, ja *CpY*:n on oltava reaaliluku > 0 .

Huomaa: Katso myös **eff()**, sivu 45.

nor (eikä)

  näppäimet

BooleanLaus1 **nor** *BooleanLaus2* antaa vastauksena *Boolean* lausekkeen *BooleanList1*

$x \geq 3$ or $x \geq 4$

$x \geq 3$

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

3 or 4

7

nor *Kokonaisluku2* \Rightarrow *kokonaisluku*

3 nor 4

-8

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.

{1,2,3} or {3,2,1}

{3,2,3}

{1,2,3} nor {3,2,1}

{-4,-3,-4}

nor (eikä)

ctrl [= näppäimet

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*) \Rightarrow *lauseke*

norm(*Vektori*) \Rightarrow *lauseke*

Laskee Frobeniusin normin.

normCdf()

Katalogi > 

normCdf(*alaraja*,*yläraja*[, μ],[σ]) \Rightarrow *luku*, jos *yläraja* ja *alaraja* ovat lukuja, *lista*, jos *alaraja* ja *yläraja* ovat listoja

Laskee normaalijakauman todennäköisyyden *alarajan* ja *ylärajan* välillä määritetyille μ :lle (oletus=0) ja σ :lle (oletus=1).

normPdf()

Katalogi > 

normPdf(*XArvo*[, μ],[σ]) \Rightarrow *luku*, jos *XArvo* on luku, *lista*, jos *XArvo* on lista

Laskee normaalijakauman pistetodennäköisyysfunktion määritetyssä *XArvossa* määritetyille μ :lle ja σ :lle.

not (ei)

Katalogi > 

not *BooleanLaus* \Rightarrow *Boolean lauseke*

Määrittää totuusarvon tosi, epätosi tai argumentin sievennetyn muodon.

not *Kokonaisluku1* \Rightarrow *kokonaisluku*

Heksadesimaalisessa kantalukutilassa:

npv(*Korkoprosentti*,*CF0*,*CFLista*
[,*CFFrekv*])

Talouseläntötoiminto, joka laskee nettonykyarvon; tulevien ja poistuvien kassavirtojen nykyisten arvojen summan. Jos npv:n tulos on positiivinen, investointi on kannattava.

Korkoprosentti on prosentti, jolla kassavirtoja (rahan kustannusta) vähennetään yhden jakson aikana.

CF0 on alkuperäinen kassavirta aikana 0; arvon on oltava kokonaisluku.

CFLista on lista kassavirtamääristä ensimmäisen kassavirran *CF0* jälkeen.

CFFrekv on lista, jossa jokainen elementti määrittää esiintymisfrekvenssin ryhmitetyille (peräkkäiselle) kassavirtamäärille, joka on *CFListan* vastaava elementti. Oletusarvo on 1; jos syötät arvoja, niiden on oltava positiivisia kokonaislukuja < 10,000.

$list1 := \{6000, -8000, 2000, -3000\}$	$\{6000, -8000, 2000, -3000\}$
$list2 := \{2, 2, 2, 1\}$	$\{2, 2, 2, 1\}$
$npv(10, 5000, list1, list2)$	4769.91

nSolve()

nSolve(*Yhtälö*,*Muutt*[=*Arvaus*]) \Rightarrow *luku*
tai virhe_merkkijono

$nSolve(x^2 + 5 \cdot x - 25 = 9, x)$	3.84429
---------------------------------------	---------

nSolve(*Yhtälö*,*Muutt*[=*Arvaus*],*alaraja*)
 \Rightarrow *luku* *tai virhe_merkkijono*

$nSolve(x^2 = 4, x = 1)$	-2.
--------------------------	-----

$nSolve(x^2 = 4, x = 1)$	2.
--------------------------	----

nSolve(*Yhtälö*,*Muutt*
[=*Arvaus*],*alaraja*,*yläraja*) \Rightarrow *luku* *tai*
virhe_merkkijono

nSolve(*Yhtälö*,*Muutt*[=*Arvaus*]) |
alaraja \leq *Muutt* \leq *yläraja*
 \Rightarrow *luku* *tai virhe_merkkijono*

Etsii iteratiivisesti yhtä likimääräistä numeerista ratkaisua *Yhtälön* yhdelle muuttujalle. Määritä muuttuja seuraavasti:

Huomaa: Jos ratkaisuja on useita, voit yrittää löytää tietyn ratkaisun käyttämällä apuna arvausta.

muuttuja

– tai –

*muuttuja = reaalityluku*Esimerkiksi x kelpaa ja samoin $x=3$.

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).

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

O**OneVar****OneVar** [1,]X[,][Frekv][,Luokka,Sisällytä]]**OneVar** [n,]X1,X2[X3[,...[,X20]]]

Laskee yhden muuttujan tilaston enintään 20 listasta. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 151.)

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

X -argumentit ovat datalistoja.

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

Luokka on numeeristen luokkakoodien lista vastaaville X :n arvoille.

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

Tyhjä elementti jossakin listassa X , *Frekv* tai *Luokka* saa aikaan, että kaikkien listojen vastaava elementti on tyhjä. Tyhjä elementti jossakin listassa $X1-X20$ saa aikaan, että kaikkien listojen vastaava elementti on tyhjä. Lisätietoja tyhjistä elementeistä, katso sivu 215.

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)

BooleanLaus1 or *BooleanLaus2* antaa vastauksena *Boolean* lausekkeen *BooleanList1* or *BooleanList2* antaa vastauksena *Boolean* listan *BooleanMatriisi1* or *BooleanMatriisi2* antaa vastauksena *Boolean* matriisin

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

Define $g(x)$ =Func	Done
If $x \leq 0$ or $x \geq 5$	
Goto end	
Return $x \cdot 3$	
Lbl end	
EndFunc	
$g(3)$	9
$g(0)$	A function did not return a value

Antaa vastauksen tosi, jos jompikumpi tai molemmat lausekkeet ovat tosia. Antaa vastauksen epätosi, jos kumpikin lauseke on epätosi.

Huomaa: Katso **xor**.

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

Kokonaisluku1

or*Kokonaisluku2*⇒*kokonaisluku*

Vertaa kahta reaalikokonaislukua bitti bitiltä or-operaation avulla. Sisäisesti kumpikin kokonaisluku muunnetaan etumerkilliseksi, 64 bitin binaariluvuksi. Kun vastaavia bittejä verrataan, tulos on 1, jos jompikumpi bitti on 1; tulos on 0 vain, jos kumpikin bitti on 0. Laskettu arvo edustaa bittituloksia, ja se näkyy kantelukutilan mukaisesti.

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

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

Jos syötät desimaalikokonaisluvun, joka on etumerkillisen, 64 bitin binaarimuodon lukualueen ulkopuolella, laskin käyttää symmetristä modulo-operaatiota, jotta arvo saadaan oikealle alueelle. Lisätietoja, katso ►**Base2**, sivu 17.

Huomaa: Katso **xor**.

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 l*) \Rightarrow *lista*

Määrittää merkkijonon *Merkkijono* ensimmäisen merkin numerokoodin tai luo listan jokaisen listaelementin ensimmäisistä merkeistä.

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

P**P ▶ Rx()**Katalogi > **P ▶ Rx**(*rLaus*, θ *Laus*) \Rightarrow *lauseke*

Radiaanikulmatilassa:

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 °, G tai r .

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

P ▶ Ry()Katalogi > **P ▶ Ry**(*rLista*, θ *Lista*) \Rightarrow *lista*

Radiaanikulmatilassa:

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.

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

PassErr

Ohittaa virheen siirtyen seuraavalle tasolle.

Jos järjestelmän muuttuja *errCode* on nolla, **PassErr** ei tee mitään.

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

Huomaa: Katso myös **ClrErr**, sivu 24, ja **Try**, sivu 162.

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

Esimerkki **PassErr**-komennosta, katso esimerkki 2 **Try**-komennon kohdalla, sivu 162.

piecewise()

piecewise(*Laus1* [, *Ehto1* [, *Laus2* [, *Ehto2* [, ...]]]])

Laatii määritelmät paloittain määritellylle funktiolle listan muodossa. Voit luoda paloittain määriteltyjen funktioiden määrittäjiä myös mallin avulla.

Huomaa: Katso myös **Paloittain määritellyn funktion malli**, sivu 2.

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

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

$\textit{poissPdf}(\lambda, XArvo) \Rightarrow \textit{luku}$, jos *XArvo* on luku, *lista*, jos *XArvo* on lista

Laskee todennäköisyyden diskreetille Poissonin jakaumalle, jolla on määritetty keskiarvo λ .

Vektori ► Polar

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

Näyttää *vektorin* polaarisessa muodossa [$r \angle \theta$]. Vektorin on oltava kooltaan 2, ja se voi olla rivi tai sarake.

Huomaa: ► Polar on näyttömuodon ohje, ei muunnosfunktio. Voit käyttää komentoa ainoastaan syöterivin lopussa, eikä se päivitä *ans*:n arvoa.

Huomaa: Katso myös ► Rect, sivu 126.

kompleksiArvo ►Polaarinen

Näyttää *kompleksiVektorin* polaarisessa muodossa.

- Astekulmatilassa vastauksena on ($r \angle \theta$).
- Radiaanikulmatilassa vastauksena on $re^{i\theta}$.

Radiaanikulmatilassa:

Graadikulmatilassa:

(4*i*)►Polar

(4∠ 100.)

Astekulmatilassa:

kompleksiArvolla voi olla mikä tahansa kompleksilukumuoto. Syöte $rei\theta$ aiheuttaa kuitenkin virheen astekulmatilassa.

Huomaa: Polaarissa syötteessä ($r\angle\theta$) on käytettävä sulkeita.

polyEval()

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.

polyRoots()

polyRoots(Poly, Muutt) ⇒ lista

polyRoots(Kertoinlista) ⇒ lista

Ensimmäinen syntaksi, **polyRoots(Poly, Muutt)**, laskee polynomin *Poly* reaalityyppisten juurten listan muuttujan *Muutt* suhteen. Jos reaalityyppistä ei ole, tuloksena on tyhjä lista: { }.

Toinen syntaksi, **polyRoots(Kertoinlista)**, askee reaalityyppisten juurten listan kertoimille, jotka sisältyvät *Kertoinlistaan*.

Huomaa: Katso myös **cPolyRoots()**, sivu 32.

PowerReg

PowerReg X, Y [, Frekv] [, Luokka, Sisällytä]]

Laskee potenssiregression $y = (a \cdot x^b)$ listoista *X* ja *Y* frekvenssillä *Frekv*. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 151.)

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 tyhjiin elementtien vaikutuksesta, katso Tyhjät elementtisivulla sivu 215.

Tulosmuuttuja	Kuvaus
stat.RegEqn	Regressioyhtälö: $a \cdot (x)^b$
stat.a, stat.b	Regressiokertoimet
stat.r ²	Muunnettujen tietojen lineaarimäärittelyn kerroin
stat.r	Muunnettujen tietojen korrelaatiokerroin ($\ln(x)$, $\ln(y)$)
stat.Resid	Potenssimalliin liittyvät jäännökset
stat.ResidTrans	Muunnettujen tietojen lineaariseen sovitukseen liittyvät jäännökset
stat.XReg	Muokatun 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	Muokatun Y 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

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 Π(), sivu 189.

Product (Π)

Katso Π(), sivu 189.

product()

Katalogi > 

product(Lista[, Alku[, loppu]])⇒lauseke

Laskee *Listan* sisältämien elementtien tulon. *Alku* ja *Loppu* ovat valinnaisia. Ne määrittävät elementtien alueen.

product(Matriisi[, Alku[, loppu]])⇒matriisi

Laskee rivivektorin, joka sisältää *Matriisi*:n sarakkeiden elementtien tulot. *Alku* ja *Loppu* ovat valinnaisia. Ne määrittävät rivialueen.

```
product  $\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix}$  [28 80 162]
```

```
product  $\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix}, 1, 2$  [4 10 18]
```

Tyhjiä elementtejä ei huomioida.
Lisätietoja tyhjistä elementeistä, katso sivu 215.

propFrac()

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.

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

QR *Matriisi, qMatNimi, rMatNimi[, Tol]*

Liukuluku (9.) muuttujassa m1 aiheuttaa sen, että vastaukset lasketaan liukulukumuodossa.

Suorittaa Householderin QR-dekomponoinnin reaali- tai kompleksilukumatriisista. Tuloksena olevat Q- ja R-matriisit tallentuvat määritettyihin *MatNimi*-muuttujiin. Q-matriisi on jakamaton. R-matriisi on yläkolmionmatriisi.

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

- Jos käytät painikkeita   tai **Automaattinen tai likimääräinen**-tilan valintaa Approximate (Likimääräinen), laskut suoritetaan liukulukuaritmetiikalla.
- Jos *Tol* jätetään pois tai sitä ei käytetä, oletusarvoinen toleranssi lasketaan seuraavasti:
 $5E-14 \cdot \max(\dim(\text{Matriisi})) \cdot \text{rowNorm}(\text{Matriisi})$

QR-hajotus lasketaan numeerisesti Householderin transformaatioiden avulla. Symbolinen ratkaisu lasketaan Gram-Schmidtin menetelmällä. Matriisin *qMatNimi* sarakkeet ovat ne ortonormitetut perusvektorit, jotka sijoittuvat *matriisin* määrittämään tilaan.

$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix} \rightarrow mI$	$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$
QR <i>mI,qm,rm</i>	Done
<i>qm</i>	$\begin{bmatrix} 0.123091 & 0.904534 & 0.408248 \\ 0.492366 & 0.301511 & -0.816497 \\ 0.86164 & -0.301511 & 0.408248 \end{bmatrix}$
<i>rm</i>	$\begin{bmatrix} 8.12404 & 9.60114 & 11.0782 \\ 0. & 0.904534 & 1.80907 \\ 0. & 0. & 0. \end{bmatrix}$

$\begin{bmatrix} m & n \\ o & p \end{bmatrix} \rightarrow mI$	$\begin{bmatrix} m & n \\ o & p \end{bmatrix}$
QR <i>mI,qm,rm</i>	Done
<i>qm</i>	$\begin{bmatrix} \frac{m}{\sqrt{m^2+o^2}} & \frac{-\text{sign}(m \cdot p - n \cdot o) \cdot o}{\sqrt{m^2+o^2}} \\ \frac{o}{\sqrt{m^2+o^2}} & \frac{m \cdot \text{sign}(m \cdot p - n \cdot o)}{\sqrt{m^2+o^2}} \end{bmatrix}$
<i>rm</i>	$\begin{bmatrix} \sqrt{m^2+o^2} & \frac{m \cdot n + o \cdot p}{\sqrt{m^2+o^2}} \\ 0 & \frac{ m \cdot p - n \cdot o }{\sqrt{m^2+o^2}} \end{bmatrix}$

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

Laskee 2. asteen polynomiregression $y = a \cdot x^2 + b \cdot x + c$ listaosta X ja Y frekvenssillä *Frekv*. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 151.)

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

X ja Y ovat riippumattomien ja riippuvien muuttujien listoja.

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

Luokka on luokkakoodien lista vastaavalle X - ja Y -datalle..

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

Lisätietoja listassa olevien tyhjen elementtien vaikutuksesta, katso Tyhjät elementtsivulla sivu 215.

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

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

Laskee 4. asteen polynomiregression $y = a \cdot x^4 + b \cdot x^3 + c \cdot x^2 + d \cdot x + e$ listoista X ja Y frekvenssillä *Frekv*. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu sivu 151.)

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

X ja Y ovat riippumattomien ja riippuvien muuttujien listoja.

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

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

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

Lisätietoja listassa olevien tyhjien elementtien vaikutuksesta, katso Tyhjä elementtsivulla sivu 215.

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	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>

Tulosmuuttuja	Kuvaus
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 <i>frekvenssilista</i>

R

R ► Pθ()

Luettelo > 

R ► Pθ (*xLista*, *yLista*) ⇒ *lista*

R ► Pθ (*xMatriisi*, *yMatriisi*) ⇒ *matriisi*

Laskee yhtäpitävän θ -koordinaatin argumenttiparille (*x*,*y*).

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

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

Astekulmatilassa:

Graadikulmatilassa:

Radiaanikulmatilassa:

R ► Pr()

Luettelo > 

R ► Pr (*xLista*, *yLista*) ⇒ *lista*

R ► Pr (*xMatriisi*, *yMatriisi*) ⇒ *matriisi*

Laskee yhtäpitävän r -koordinaatin argumenttiparille (*x*,*y*).

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

Radiaanikulmatilassa:

► Rad

Luettelo > 

Muuntaa argumentin radiaanikulmayksikköön.

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

Astekulmatilassa:

$(1.5) \blacktriangleright \text{Rad}$ $(0.02618)^r$

Graadikulmatilassa:

$(1.5) \blacktriangleright \text{Rad}$ $(0.023562)^r$ **rand()****rand()** ⇒ lauseke**rand(Kokeiden lkm)** ⇒ lista**rand()** laskee satunnaisen arvon välillä 0 ja 1.**rand(Kokeiden lkm)** laskee listan, joka sisältää *Kokeiden lkm* satunnaista arvoa välillä 0 ja 1.

Aseta satunnainen siemenluku.

RandSeed 1147

Done

rand(2)

{0.158206,0.717917}

randBin()**randBin(*n*, *p*)** ⇒ lauseke**randBin(*n*, *p*, *Kokeiden lkm*)** ⇒ lista**randBin(*n*, *p*)** laskee satunnaisen reaaliluvun määrätystä binomijakaumasta.**randBin(*n*, *p*, *Kokeiden lkm*)** laskee listan, joka sisältää *Kokeiden lkm* satunnaista reaalilukua määrätystä binomijakaumasta.**randInt()****randInt****(
Muutt.alaraja
,Muutt.yläaraja)** ⇒
lauseke**randInt****(
Muutt.alaraja
,Muutt.yläaraja
,Kokeiden lkm)** ⇒
lista

randInt()

Luettelo > 

randInt

(
Muutt.alaraja
,Muutt.yläraja) laskee
satunnaisen
kokonaisluvun
rajojen *Muutt.alaraja*
ja *Muutt.yläraja*
määrämällä
kokonaislukuvälillä.

randInt

(
Muutt.alaraja
,Muutt.yläraja
,Kokeiden lkm)
laskee listan, joka
sisältää *Kokeiden lkm*
satunnaista
kokonaislukua
määrätyllä välillä.

randMat()

Luettelo > 

**randMat(numRivit, numSarakeet) ⇒
matriisi**

Laskee välillä -9 ja 9 määrätylle mitalle
kokonaislukumatriisin.

Kummankin argumentin on
sievennyttävä kokonaisluvuksi.

RandSeed 1147	Done									
randMat(3,3)	<table border="1"><tr><td>8</td><td>-3</td><td>6</td></tr><tr><td>-2</td><td>3</td><td>-6</td></tr><tr><td>0</td><td>4</td><td>-6</td></tr></table>	8	-3	6	-2	3	-6	0	4	-6
8	-3	6								
-2	3	-6								
0	4	-6								

Huomaa: Tämän matriisin arvot muuttuvat
aina, kun painat .

randNorm()

Luettelo > 

randNorm(μ , σ) ⇒ lauseke
randNorm(μ , σ , *Kokeiden lkm*) ⇒ lista

randNorm(μ , σ) laskee desimaaliluvun
määräystä normaalijakaumasta. Luku
voi olla mikä tahansa reaalityyppinen, mutta
se keskittyy voimakkaasti välille $[\mu-3\sigma$,
 $\mu+3\sigma]$.

randNorm(μ , σ , *Kokeiden lkm*) listan
(list), joka sisältää *Kokeiden lkm*
desimaalilukua määrätystä
normaalijakaumasta.

RandSeed 1147	Done
randNorm(0,1)	0.492541
randNorm(3,4.5)	-3.54356

randPoly()

Luettelo > 

randPoly(*Muutt*, *Aste*) ⇒ *lauseke*

Laskee polynomin muuttujasta *Muutt* määrättyssä järjestyksessä *Aste*.

Kertoimet ovat satunnaisia kokonaislukuja välillä -9 ja 9 . Johtava kerroin ei ole 0 .

Asteen on oltava $0-99$.

RandSeed 1147

Done

randPoly(x,5)	$-2 \cdot x^5 + 3 \cdot x^4 - 6 \cdot x^3 + 4 \cdot x - 6$
---------------	--

randSample()

Luettelo > 

randSample(*List*, *Kokeiden lkm*[, *noRepl*])

⇒ *lista*

Laskee satunnaisesta otoksesta listan, joka koostuu valinnoista *Kokeiden lkm* listasta *List* valinnaisena otoksen korvaaminen muulla (*noRepl*=0), tai ilman otoksen korvaamista (*noRepl*=1). Oletusarvona on otoksen korvaaminen

RandSeed

Luettelo > 

RandSeed *Luku*

Jos *Luku* = 0, tämä asettaa siemenluvut satunnaislukugeneraattorin tehdasasetuksiin. Jos *Luku* ≠ 0, sitä käytetään luomaan kaksi siemenlukua, jotka tallennetaan järjestelmän muuttujiin siemen1 ja siemen2

RandSeed 1147

Done

rand()	0.158206
--------	----------

real()

Luettelo > 

Laskee argumentin reaali osan.

real(*List1*) ⇒ *lista*

Laskee kaikkien alkutekijöiden reaali osat

real(*Matriisi1*) ⇒ *matriisi*

Laskee kaikkien alkutekijöiden reaali osat.

► Rect

Luettelo > 

Vectr ► **Rect**

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

Näyttää *Vektorin* suorakulmamuodossa $[x, y, z]$ Vektorin on oltava kooltaan 2 tai 3, ja se voi olla rivi tai sarake.

Huomaa: ►Rect on näyttömuodon ohje, ei muunnosfunktio. Voit käyttää komentoa ainoastaan syöterivin lopussa, eikä se päivitä arvoa *ans*.

Huomaa: Katso myös ►Polar, sivu 115.

kompleksiArvo ►Rect

Näyttää kompleksisen arvon *complexValues* suorakulmaisessa muodossa $a+bi$. Arvolla *complexValue* voi olla mikä tahansa kompleksinen muoto. Syöte $re^{i\theta}$ aiheuttaa kuitenkin virheen astekulmatilassa.

Huomaa: Polaarisisä syötteessä $(r \angle \theta)$ on käytettävä sulkeita.

Radiaanikulmatilassa:

Graadikulmatilassa:

$((1 \angle 100)) \blacktriangleright \text{Rect}$ i

Astekulmatilassa:



Huomaa: Näppäilläksesi merkin \angle valitse se Luettelon symboliliuettelosta.

ref()

$\text{ref}(\text{Matriisi}[, \text{Tol}]) \Rightarrow \text{matriisi}$

Laskee matriisin *Matriisi* l rivi-echelonmuodossa.

Valinnaisesti kaikkia matriisin elementtejä käsitellään nollana, jos niiden itseisarvo on pienempi kuin *Tol*. Tätä toleranssia käytetään vain, jos matriisissa on liukulukusyötteitä eikä se sisällä symbolisia muuttujia, joille ei ole määrätty arvoa. Muussa tapauksessa komentoa *Tol* ei huomioida.

- Jos käytät painikkeita   tai asetat tilan **Auto or Approximate**, laskenta suoritetaan liukulukuaritmetiikalla.
- Jos *Tol* jätetään pois tai sitä ei käytetä, oletusarvoinen toleranssi lasketaan

$$\text{ref} \left(\begin{pmatrix} -2 & -2 & 0 & -6 \\ 1 & -1 & 9 & -9 \\ -5 & 2 & 4 & -4 \end{pmatrix} \right) \begin{vmatrix} 1 & -2 & -4 & 4 \\ & 5 & 5 & 5 \\ 0 & 1 & 4 & 11 \\ & & 7 & 7 \\ 0 & 0 & 1 & -62 \\ & & & 71 \end{vmatrix}$$

seuraavasti:

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

Vältä määrittelemättömiä alkutekijöitä matriisissa *MatriisiI*. Tuloksena saattaa olla odottamattomia vastauksia.

Jos esimerkiksi *a* on määrittelemätön seuraavassa lausekkeessa, näkyviin tulee varoitusviesti, ja vastaus näyttää seuraavalta:

$$\text{ref} \left(\begin{bmatrix} a & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \right) \quad \begin{bmatrix} 1 & \frac{1}{a} & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Varoitus näkyy, koska yleistetty alkutekijä $1/a$ ei olisi mahdollinen lausekkeelle $a=0$.

Voit välttää tämän tallentamalla etukäteen arvon *a*:lle tai käyttämällä rajoittavaa ("|")-operaattoria arvon korvaamiseksi, kuten on esitetty seuraavassa esimerkissä.

$$\text{ref} \left(\begin{bmatrix} a & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \right) | a=0 \quad \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}$$

Huomaa: Katso myös **rref()**, page 137.

RefreshProbeVars

RefreshProbeVars

Päiset mittausarvoon kaikista TI-Basic-ohjelmaan kytketyistä antureista.

StatusVar
Value

Tila

statusVar
 =0

Normaali (jatka ohjelmalla)

Esimerkki


Lämmön määrittely (=

Prgm

© Tarkista, onko järjestelmä valmiina

RefreshProbeVars-tila

Jos tila=0, niin

StatusVar Value	Tila	Käyttötila
<i>statusVar</i> =1	Vernier DataQuest™ on tiedonkeruutilassa. Huomaa: Vernier DataQuest™ -sovelluksen on oltava metrisessä tilassa, jotta tämä komento toimisi. 	"valmis" Muuttujalle n,1,50 RefreshProbeVars-tila lämpötila:=mittari.lämpötila Disp "Lämpötila: ",lämpötila Jos lämpötila>30, niin Disp "Liian kuuma" LopetaJos © Odota 1 sekunti otoksien välillä Odota 1 LopetaKoska Tai Disp "Ei valmis". Yritä myöhemmin uudelleen" EndIf EndPrgm
<i>statusVar</i> =2	Vernier DataQuest™ -sovellusta ei ole käynnistetty.	
<i>statusVar</i> =3	The Vernier DataQuest™-sovellus on käynnistetty, mutta et ole yhdistänyt mitään antureita.	Huomaa: Tätä voi käyttää myös TI-Innovator™ _i -laitteen keskiössä.

remain()

Luettelo > 

remain(Lista1, Lista2) ⇒ lista
remain(Matriisi1, Matriisi2) ⇒ matriisi

Laskee ensimmäisen argumentin jäännöksen toisen argumentin suhteen seuraavien identtisten yhtälöiden määritelmien mukaisesti:

remain(x,0) x
 remain(x,y) $x - y \cdot \text{iPart}(x/y)$

remain(7,0)	7
remain(7,3)	1
remain(-7,3)	-1
remain(7,-3)	1
remain(-7,-3)	-1
remain({12,-14,16},{9,7,-5})	{3,0,1}

Tämän seurauksena huomaa, että `remain(-x,y) = remain(x,y)`. Vastaus on joko nolla tai samanmerkkinen kuin ensimmäinen argumentti.

$$\text{remain} \left(\begin{pmatrix} 9 & -7 \\ 6 & 4 \end{pmatrix}, \begin{pmatrix} 4 & 3 \\ 4 & -3 \end{pmatrix} \right) \quad \begin{pmatrix} 1 & -1 \\ 2 & 1 \end{pmatrix}$$

Huomaa: Katso myös `mod()`, sivu 99.

Pyydä

Pyydä *promptString*, *muutt*
[, *NäytäLippu* [, *statusVar*]]

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

Ohjelmointikomento: Keskeyttää ohjelman ja näyttää valintaruudun, jossa on viesti *kehotemerkkijono* sekä syöttöruutu käyttäjän antamaa vastausta varten.

Kun käyttäjä kirjoittaa vastauksen ja klikkaa **OK**-painiketta, syöttöruudun sisältö määrätään muuttujalle *var*.

Jos käyttäjä klikkaa **Cancel**, ohjelma etenee hyväksymättä mitään syötteitä. Ohjelma käyttää muuttujan *muutt* aikaisempaa arvoa, jos *var* on jo määritelty.

Valinnainen *NäytäLippu* -argumentti voi olla mikä tahansa lauseke.

- Jos *NäytäLippu* jätetään pois, tai jos se sievennyy arvoon **1**, kehoteviesti ja käyttäjän vastaus näkyvät laskimen historiassa.
- Jos *NäytäLippu* sievennyy arvoon **0**, kehote ja vastaus eivät näy historiassa.

Valinnainen *statusVar* argumentti laskee ohjelmalle tavan määrittellä, kuinka käyttäjä on sulkenut valintaikkunan. Huomaa, että *statusVar* vaatii *NäytäLippu* -argumentin.

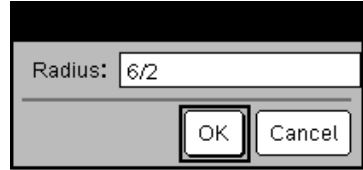
- Jos käyttäjä on klikannut **OK** tai painanut **Enter** tai **Ctrl+Enter**, muuttuja *statusVar* asetetaan arvoon **1**.

Määrittele ohjelma:

```
Määrittele pyydä_demo()=Ohjelma
Pyydä "Säde: ",r
Disp "Ala = ",pi*r^2
EndPrgm
```

Aja ohjelma ja kirjoita vastaus:

Pyydä_demo()



Vastaus **OK**-näppäimen painamisen jälkeen:

```
Säde: 6/2
Ala = 28.2743
```

Määrittele ohjelma:

```
Määrittele polynomi()=Ohjelma
Pyyntö "Syötä polynomi x:ään:",p(x)
Disp "Reaalilukujuurat
ovat:",polyRoots(p(x),x)
EndPrgm
```

Aja ohjelma ja kirjoita vastaus:

- Muuten muuttujastatus`Var` asetetaan arvoon `0`.



`func()`-argumentin avulla ohjelma voi tallentaa käyttäjän vastauksen funktion määritelmäksi. Tämä syntaksi toimii ikään kuin käyttäjä suorittaisi komennon:

Määrittele `func(arg1, ...argn) = käyttäjän vastaus`

Sen jälkeen ohjelma voi käyttää määriteltyä funktiota `func()`. Komennon `promptString` pitäisi opastaa käyttäjää syöttämään sopiva `käyttäjän vastaus`, joka täydentää funktion määritelmän.

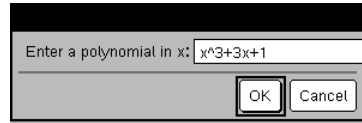
Huomaa: Voit käyttää Pyydä komento käyttäjän määrittelemässä ohjelmassa, mutta ei funktiossa.

Pysäytä ohjelma, joka sisältää komennon **Request** loputtoman silmukan sisällä:

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

Huomaa: Katso myös **RequestStr**, page 131.

polynomi()



Tulos, kun x^3+3x+1 on syötetty ja valittu **OK**:

Reaalilukujuuret ovat: `{-0.322185}`

RequestStr

RequestStr *kehotemerkkijono, muutt[, NäytäLippu]*

Määrittele ohjelma:

```
Määrittele requestStr_demo()=Ohjelma
RequestStr "Oma nimesi:",nimi,0
Disp "Vastauksessa on ",dim(nimi),"
merkkiä."
EndPrgm
```



Aja ohjelma ja kirjoita vastaus:

```
requestStr_demo()
```

Ohjelmointikomento: Toimii samalla tavalla kuin **Request**-komennon ensimmäinen syntaksi paitsi, että käyttäjän vastaus tulkitaan aina merkkijonoksi. Vastakohtaisesti **Request**-komento tulkitsee vastauksen lausekkeeksi, jollei käyttäjä merkitse sitä lainausmerkkien ("") sisään.

Huomaa: Voit käyttää **RequestStr** -komentoa käyttäjän määrittelemässä ohjelmassa, mutta ei funktiossa.

Pysäytä ohjelma, joka sisältää **RequestStr** komento äärettömässä silmukassa:

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

Huomaa: Katso myös **Pyydä**, page 130.



Vastaus **OK**-näppäimen painamisen jälkeen (Huomaa, että jos *NäytäLippu*-argumentti on **0**, kehote ja vastaus eivät näy historiassa):

```
requestStr_demo()
```

Vastauksessa on 5 merkkiä.

Return [Laus]

Laskee lausekkeen *Laus* funktion tuloksena Käytetään lohkon **Func...EndFunc** sisällä.

Huomaa: Käytä **Return** ilman argumenttia lohossa **Prgm...EndPrgm** poistuaksesi ohjelmasta.

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

```
Define factorial (nn)=
Func
Local answer,counter
1 → answer
For counter,1,nn
answer· counter → answer
EndFor
Return answer|
EndFunc
```

factorial (3)

6

right()Luettelo > **oikea**(*ListI*, *Num*) ⇒ *lista* $\text{right}(\{1,3,-2,4\},3)$ $\{3,-2,4\}$ Laskee oikeanpuoleisimmat *Num*-alkutekijät, jotka sisältyvät listaan *ListI*.Jos jätät pois komennon *Num*, laskee kaiken listasta *ListI*.**right**(*lähdemerkkijono*[, *Num*]) ⇒ *merkkijono* $\text{right}(\text{"Hello"},2)$ "lo"Laskee oikeanpuoleisimmat *Num*-merkit, jotka sisältyvät merkkijonoon *lähdemerkkijono*Jos jätät pois komennon *Num*, laskee kaiken *lähdemerkkijonosta*.**right**(*Vertailu*) ⇒ *lauseke* $\text{right}(x<3)$ 3

Laskee yhtälön tai epäyhtälön oikean puolen.

rk23 ()Luettelo > **rk23**(*Laus*, *Var*, *depVar*, {*Var0*, *VarMax*}, *depVar0*, *VarStep* [, *diftol*]) ⇒ *matriisi*

Differentsiaalilyhtälö:

 $y' = 0.001 \cdot y \cdot (100 - y)$ ja $y(0) = 10$ $\text{rk23}(0.001 \cdot y \cdot (100 - y), t, y, \{0, 100\}, 10, 1)$

0.	1.	2.	3.	4.
10.	10.9367	11.9493	13.042	14.2

rk23(*SystemOfExpr*, *Var*, *ListOfDepVars*, {*Var0*, *VarMax*}, *ListOfDepVars0*, *VarStep*[, *diftol*]) ⇒ *matriisic*

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

rk23(*ListOfExpr*, *Var*, *ListOfDepVars*, {*Var0*, *VarMax*}, *ListOfDepVars0*, *VarStep*[, *diftol*]) ⇒ *matriisi*Sama yhtälö, jossa *diftol* on asetettu 1.E-6

Käyttää Runge-Kutta-menetelmää järjestelmän ratkaisuun

 $\frac{d \text{depVar}}{d \text{Var}} = \text{Expr}(\text{Var}, \text{depVar})$ $\text{rk23}(0.001 \cdot y \cdot (100 - y), t, y, \{0, 100\}, 10, 1, 1.E-6)$

0.	1.	2.	3.	4.
10.	10.9367	11.9495	13.0423	14.2189

funktioilla *depVar*(*Var0*)=*depVar0* välillä [*Var0*,*VarMax*]. Laskee tulokseksi matriisin, jonka ensimmäinen rivi määrittelee *Var* tulosarvot muuttujan *VarStep* määrittelemällä tavalla. Toinen rivi määrää ensimmäisen ratkaisukomponentin arvon vastaavissa *Var*-arvoissa, jne.

Yhtälöryhmä:

$$\begin{cases} y1' = -y1 + 0.1 \cdot y1 \cdot y2 \\ y2' = 3 \cdot y2 - y1 \cdot y2 \end{cases}$$
kun $y1(0) = 2$ and $y2(0) = 5$

rk23 ()

Luettelo > 

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

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

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

Var on riippumaton muuttuja.

ListOfDepVars on riippuvaisten muuttujien luettelo.

{*Var0*, *VarMax*} on kahden elementin lista, joka määrää funktion integroimaan muuttujasta *Var0* to *VarMax*.

ListOfDepVars0 on lista riippuvaisten muuttujien alkuarvoja.

Jos *VarStep* on arvoltaan nolasta eroava numero: merkki(*VarStep*) = merkki (*VarMax-Var0*) ja ratkaisut lasketaan muuttujalla *Var0+i*VarStep* kaikille $i=0,1,2,\dots$ niin, että *Var0+i*VarStep* on alueella [*muutt0*,*VarMax*] (muuttujalla *VarMax* ei ehkä ole ratkaisuarvoa).

jos *VarStep* sievenee nolaksi, ratkaisut lasketaan "Runge-Kutta" -muuttujan *Muutt* arvoilla.

diffol on virhetoleranssi (oletuksena 0.001).

$$\text{rk23} \left(\begin{array}{l} -y1+0.1 \cdot y1 \cdot y2 \\ 3 \cdot y2 - y1 \cdot y2 \end{array}, t, \{y1, y2\}, \{0, 5\}, \{2, 5\}, 1 \right)$$

0.	1.	2.	3.	4.
2.	1.94103	4.78694	3.25253	1.82848
5.	16.8311	12.3133	3.51112	6.27245

root()

Luettelo > 

Huomaa: Katso myös **N:s juuri -malli**, sivu 1.

rotate()

Luettelo > 

rotate(Kokonaisluku1[,KiertojenLkm]) ⇒
kokonaisluku

Binaarisessa kantalukutilassa:

rotate()

Luettelo > 

Jos *KiertojenLkm* on positiivinen, kierto on vasemmalle. Jos *KiertojenLkm* on negatiivinen, kierto on oikealle. Oletusarvo on -1 (kierrä oikealle yksi merkki).

round()

Luettelo > 

Laskee argumentin pyöristettynä jättäen desimaalipisteen jälkeen määrätyn määrän numeroita.

$$\text{round}(1.234567,3) \quad 1.235$$

numeroiden on oltava kokonaisluku välillä $0-12$. Jos *numeroita* ei sisällytetä, laskee argumentin pyöristettynä 12 merkitykselliseen numeroon.

Huomaa: Näytettävät numerot -tila saattaa vaikuttaa näyttöön.

round(Lista1[, numerot]) \Rightarrow lista

Pyöristää alkutekijöiden listan jättäen desimaalipisteen jälkeen määrätyn määrän numeroita.

$$\text{round}(\{\pi, \sqrt{2}, \ln(2)\}, 4) \\ \{3.1416, 1.4142, 0.6931\}$$

round(Matriisi1[, numerot]) \Rightarrow matriisi

Pyöristää alkutekijöiden matriisin jättäen desimaalipisteen jälkeen määrätyn määrän numeroita.

$$\text{round}\left(\begin{bmatrix} \ln(5) & \ln(3) \\ \pi & e^1 \end{bmatrix}, 1\right) \quad \begin{bmatrix} 1.6 & 1.1 \\ 3.1 & 2.7 \end{bmatrix}$$

rowAdd()

Luettelo > 

rowAdd(Matriisi1, rIndeksi1, rIndeksi2) \Rightarrow matriisi

Laskee matriisiin *Matriisi1* kopion rivin *rIndeksi2* korvattuna rivien summalla *rIndeksi1* ja *rIndeksi2*.

rowDim()

Luettelo > 

rowDim(Matriisi) \Rightarrow lauseke

Laskee *Matriisin* sisältämien rivien lukumäärän.

Huomaa: Katso myös **colDim()**, sivu 25.

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix} \rightarrow m1 \quad \begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix} \\ \text{rowDim}(m1) \quad 3$$

rowNorm()

Luettelo >

rowNorm(Matriisi) ⇒ lauseke

Laskee *Matriisin* riveillä olevien elementtien itseisarvojen summien maksimin.

$$\text{rowNorm} \left(\begin{bmatrix} -5 & 6 & -7 \\ 3 & 4 & 9 \\ 9 & -9 & -7 \end{bmatrix} \right) \quad 25$$

Huomaa: Kaikkien matriisien alkutekijöiden on sievennyttävä luvuiksi. Katso myös **colNorm()**, sivu 25.

rowSwap()

Luettelo >

rowSwap(Matriisi1, rIndeksi1, rIndeksi2) ⇒ matriisi

Laskee matriisiin *Matriisi1*, jossa rivien *rIndeksi1* ja *rIndeksi2* paikkoja on vaihdettu.

$$\begin{array}{l} \begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix} \rightarrow \text{mat} \\ \text{rowSwap}(\text{mat}, 1, 3) \end{array} \quad \begin{array}{l} \begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix} \\ \begin{bmatrix} 5 & 6 \\ 3 & 4 \\ 1 & 2 \end{bmatrix} \end{array}$$

rref()

Luettelo >

rref(Matriisi1[, Tol]) ⇒ matriisi

Laskee matriisiin *Matriisi1* pelkistetyn rivi-echelon-muodon.

$$\text{rref} \left(\begin{bmatrix} -2 & -2 & 0 & -6 \\ 1 & -1 & 9 & -9 \\ -5 & 2 & 4 & -4 \end{bmatrix} \right) \quad \begin{bmatrix} 1 & 0 & 0 & \frac{66}{71} \\ 0 & 1 & 0 & \frac{147}{71} \\ 0 & 0 & 1 & \frac{-62}{71} \end{bmatrix}$$

Valinnaisesti kaikkia matriisin elementtejä käsitellään nollana, jos niiden itseisarvo on pienempi kuin *Tol*. Tätä toleranssia käytetään vain, jos matriisissa on liukulukusyötteitä eikä se sisällä symbolisia muuttujia, joille ei ole määritelty arvoa. Muussa tapauksessa kommentoa *Tol* ei huomioida.


$$\text{rref} \left(\begin{bmatrix} a & b \\ c & d \end{bmatrix} \right) \quad \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

- Jos käytät painikkeita tai **Auto or Approximate** -tilan valintaa Approximate (Likimääräinen), laskenta suoritetaan liukulukuaritmetiikalla.
- Jos *Tol* jätetään pois tai sitä ei käytetä, oletusarvoinen toleranssi lasketaan seuraavasti:
 $5E-14 \cdot \max(\dim(\text{Matriisi1}))$
 $\bullet \text{rowNorm}(\text{Matriisi1})$

Huomaa: Katso myös `ref()`, page 127.

S

sec()

 -painike

`sec(Lista1)` ⇒ lista

Astekulmatilassa:

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

sec⁻¹() -painike

`sec-1(Lista1)` ⇒ lista

Astekulmatilassa:

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

Graadikulmatilassa:

Radiaanikulmatilassa:

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

sech()

Katalogi > 

`sech(Lista1)` ⇒ lista

sech⁻¹()Katalogi > 

`sech-1(Lista1)` ⇒ lista

Radiaanikulma- ja suorakulmakompleksitilassa:

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

Send *lausTaiMerkkijono1* [, *lausTaiMerkkijono2*] ...

Ohjelmointikomento: Lähettää yhden tai useamman TI-Innovator™ Hub komennon liitettyyn laitteeseen.

*lausTaiMerkkijono*n täytyy olla validi TI-Innovator™ Hub Komento. Tavallisesti *lausTaiMerkkijono* sisältää "SET ..."komento laitteen tai "READ ..."hallitsemiseksi komento tiedon kysymiseksi.

Argumentit lähetetään laitteeseen peräkkäisinä.

Huomio: Komentoa **Send** voi käyttää käyttäjän määrittelemän ohjelman sisällä mutta ei funktion sisällä.

Huomio: Katso myös **Get** (sivu 62), **GetStr** (sivu 68), ja **eval()** (sivu 49).

Esimerkki: Sytytä sisäänrakennetun RGB-ledin sininen väri 0,5 sekunniksi.

```
Send "SET COLOR.BLUE ON TIME .5"
Done
```

Esimerkki: Kysy laitteen sisäänrakennetun valotasoaunturin tämänhetkinen arvo. **Get**-komento noutaa arvon ja sijoittaa sen muuttujaan *lightval*.

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

Esimerkki: Lähetä laskettu taajuus laitteen sisäänrakennettuun kaiuttimeen. Käytä erikoismuuttujaa *iostr.SendAns* näyttääksesi laitekomennon ilmaus arvioituna.

```
n:=50 50
m:=4 4
Send "SET SOUND eval(m·n)" Done
iostr.SendAns "SET SOUND 200"
```

seq() (sekv)

seq(Laus, Muutt, Matala, Korkea[, Askel]) ⇒ lista



Lisää muuttujan *Var* arvoa arvosta *Low* arvoon *High* välillä *Step*, laskee *Expr*, ja antaa tulokset luettelona. Muuttujan *Var* alkuperäinen sisältö on edelleen tallessa funktion **seq()** suorittamisen jälkeen.

Oletusarvo *Step* = 1.

Katalogi >

$\text{seq}\left(n^2, n, 1, 6\right)$	$\{1, 4, 9, 16, 25, 36\}$
$\text{seq}\left(\frac{1}{n}, n, 1, 10, 2\right)$	$\left\{1, \frac{1}{3}, \frac{1}{5}, \frac{1}{7}, \frac{1}{9}\right\}$
$\text{sum}\left(\text{seq}\left(\frac{1}{n^2}, n, 1, 10, 1\right)\right)$	$\frac{1968329}{1270080}$

Huom: Vastauksen pakottaminen likimääräiseksi:

Kämmenlaite: Paina  .

Windows®: Paina **Ctrl+Enter**.

Macintosh®: Paina **⌘+Enter**.

iPad®: Pidä **enter** ja valitse .

$\text{sum}\left(\text{seq}\left(\frac{1}{n^2}, n, 1, 10, 1\right)\right)$	1.54977
--	---------

seqGen()

Katalogi > 

seqGen(*Expr*, *Var*, *depVar*, {*Var0*, *VarMax*}, [*ListOfInitTerms*], [*VarStep*, *CeilingValue*])) ⇒ *lista*

Luo termiluettelon sekvenssille *depVar* (*Var*)=*Expr* seuraavasti: Lisää riippumattoman muuttujan *Var* arvoa arvosta *Var0* arvoon *VarMax* portailla *VarStep*, laskee *depVar*(*Var*) muuttujan *Var* vastaaville arvoille käyttäen *Expr*-kaavaa ja *ListOfInitTerms* -luetteloa, ja antaa tulokset luettelona

seqGen(*ListOrSystemOfExpr*, *Var*, [*ListOfDepVars*, {*Var0*, *VarMax*}], [*MatrixOfInitTerms*], [*VarStep*, *CeilingValue*])) ⇒ *matrix*

Luo termimatriisin sekvenssien *ListOfDepVars* (*Var*)=*ListOrSystemOfExpr* ryhmälle (tai listalle) seuraavasti: Lisää riippumattoman muuttujan *Var* arvoa arvosta *Var0* arvoon *VarMax* portailla *VarStep*, laskee *ListOfDepVars*(*Var*) muuttujan *Var* vastaaville arvoille käyttäen *Expr*-kaavaa ja *MatrixOfInitTerms* -luetteloa, ja antaa tulokset matriisina.

Muuttujan *Var* alkuperäinen sisältö on edelleen 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:

$$\text{seqGen}\left(\frac{u(n-1)+1}{n}, n, u, \{2,5\}, \{3\}\right)$$

$$\left\{3, \frac{4}{3}, \frac{7}{12}, \frac{19}{60}\right\}$$

Kahden sekvenssin ryhmä:

$$\text{seqGen}\left(\left\{\frac{1}{n}, \frac{u(n-1)}{2} + u(n-1)\right\}, n, \{u1, u2\}, \{1,5\}, \left[-\right]_2\right)$$

$$\begin{bmatrix} 1 & \frac{1}{2} & \frac{1}{3} & \frac{1}{4} & \frac{1}{5} \\ 2 & 2 & \frac{3}{2} & \frac{13}{12} & \frac{19}{24} \end{bmatrix}$$

Huomaa: Muuttujaa Tyhjä () yllä olevassa aloitustermimatriisissa käytetään ilmoittamaan, että $u1(n):n$ aloitustermi on laskettu käyttämällä täsmällistä sekvenssikaavaa $u1(n)=1/n$.

seqn()

Katalogi > 

seqn(*Expr*(*u*, *n*, [*ListOfInitTerms*], *nMax*), [*CeilingValue*])) ⇒ *list*

Luo sekvenssin 6 ensimmäistä termiä $u(n) = u(n-1)/2$, kun $u(1)=2$.

$$\text{seqn}\left(\frac{u(n-1)}{n}, \{2\}, 6\right)$$

$$\left\{2, 1, \frac{1}{3}, \frac{1}{12}, \frac{1}{60}, \frac{1}{360}\right\}$$

Luo termiluettelon sekvenssille $u(n) = \text{Expr}(u, n)$ seuraavasti: Lisää muuttujan n arvoa arvosta 1 arvoon $nMax$ 1 välein, laskee lausekkeen $u(n)$ muuttujan n vastaaville arvoille käyttäen $\text{Expr}(u, n)$ -kaavaa ja ListOfInitTerms -luetteloa ja antaa tulokset luettelona.

$$\text{seqn}\left(\frac{1}{n^2}, 6\right) \quad \left\{1, \frac{1}{4}, \frac{1}{9}, \frac{1}{16}, \frac{1}{25}, \frac{1}{36}\right\}$$

seqn($\text{Expr}(n$ [, $nMax$ [, $\text{CeilingValue}]]) \Rightarrow list$

Luo termiluettelon ei-rekursiiviselle sekvenssille $u(n) = \text{Expr}(u, n)$ seuraavasti: Lisää muuttujan n arvoa arvosta 1 arvoon $nMax$ 1 välein, laskee lausekkeen $u(n)$ muuttujan n vastaaville arvoille käyttäen $\text{Expr}(n)$ -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 $nstep=1$

setMode()

setMode($tilaNimiKokonaisluku$, $asetusKokonaisluku$) $\Rightarrow kokonaisluku$
setMode($lista$) $\Rightarrow kokonaislukulista$

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.

Näytä π :n likiarvo käyttäen Näytettävät numerot -tilan oletusasetusta ja näytä sen jälkeen π asetuksella Kiinteä2. Tarkista, että oletusarvo palautuu ohjelman suorittamisen jälkeen.

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 67.

Huomaa: Nykyiset tila-asetukset siirtyvät haettuihin alarutiineihin. Jos jokin alarutiini muuttaa tila-asetusta, tilamuutos häviää, kun ohjaus palautuu hakurutiiniin.

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

Tilan nimi	Tilaa vastaava kokonaisluku	Asetuksia vastaavat kokonaisluvut
Näytettävät numerot	1	1=Liukuva, 2=Liukuva1, 3=Liukuva2, 4=Liukuva3, 5=Liukuva4, 6=Liukuva5, 7=Liukuva6, 8=Liukuva7, 9=Liukuva8, 10=Liukuva9, 11=Liukuva10, 12=Liukuva11, 13=Liukuva12, 14=Kiinteä0, 15=Kiinteä1, 16=Kiinteä2, 17=Kiinteä3, 18=Kiinteä4, 19=Kiinteä5, 20=Kiinteä6, 21=Kiinteä7, 22=Kiinteä8, 23=Kiinteä9, 24=Kiinteä10, 25=Kiinteä11, 26=Kiinteä12
Kulma	2	1=Radiaani, 2=Aste, 3=Graadi
EkspONENTTIMUOTO	3	1=Normaali, 2=Kymmenpotenssi, 3=Tekninen

Tilan nimi	Tilaa vastaava kokonaisluku	Asetuksia vastaavat kokonaisluvut
Reaali- tai kompleksiluku	4	1=Reaali, 2=Suorakulma, 3=Polaarinen
Automaattinen tai likimääräinen.	5	1=Automaattinen, 2=Likimääräinen
Vektorimuoto	6	1=Suorakulma, 2=Sylinteri, 3=Pallo
Kantaluku	7	1=Desimaali, 2=Heksagonaalinen, 3=Binaarinen

shift()

Katalogi > 

shift(*Kokonaisluku1* [*,SiirtojenLkm*]) \Rightarrow *kokonaisluku*

Siirtää binaarisen kokonaisluvun bittejä. Voit syöttää *Kokonaisluku1*:n minä tahansa kantalukuna; se muunnetaan automaattisesti etumerkilliseen 64 bitin binaarimuotoon. Jos *Kokonaisluku1* on liian suuri tälle muodolle, symmetrinen modulo-operaatio sovittaa sen alueelle sopivaksi. Lisätietoja, katso ▶**Base2**, sivu 17.

Jos *SiirtojenLkm* on positiivinen, siirto tapahtuu vasemmalle. Jos *SiirtojenLkm* on negatiivinen, siirto tapahtuu oikealle. Oletusarvo on -1 (siirrä yksi bitti oikealle).

Oikealle tapahtuvassa siirrossa oikeanpuoleisin bitti pudotetaan, ja 0 ja 1 lisätään vastaamaan vasemmanpuoleista bittiiä. Vasemmalle tapahtuvassa siirrossa vasemmanpuoleisin bitti pudotetaan, ja 0 lisätään vastaamaan oikeanpuoleisinta bittiiä.

Esimerkki siirrosta oikealle:

Jokainen bitti siirtyy oikealle.

0b000000000000111101011000011010

Lisää 0:n, jos vasemmanpuoleisin bitti on 0, tai 1:n, jos vasemmanpuoleisin bitti on 1.

tuottaa:

Binaarisessa kantalukutilassa:

shift(0b1111010110000110101)	
	0b111101011000011010
shift(256,1)	0b1000000000

Heksadesimaalisessa kantalukutilassa:

shift(0h78E)	0h3C7
shift(0h78E,-2)	0h1E3
shift(0h78E,2)	0h1E38

Tärkeää: Binaariluvun edelle tulee aina merkitä etumerkki 0b ja heksadesimaaliluvun edelle 0h (nolla, ei 0-kirjain).

0b00000000000000111101011000011010

Vastaus näkyy kantaluutilan mukaisesti.
Alkunollia ei näytetä.

shift(Lista l [,SiirtojenLkm])⇒*lista*

Luo *Lista l*:n kopion, jota on siirretty oikealle tai vasemmalle *SiirtojenLkm*:n elementtien määrittämän määrän. Ei muuta *Lista l*: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(Merkkijono l [,SiirtojenLkm])⇒*merkkijono*

Luo *Merkkijono l*:n kopion, jota on siirretty oikealle tai vasemmalle *SiirtojenLkm*:n merkkien määrittämän määrän. Ei muuta *Merkkijono l*: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 kantaluutilassa:

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)

sign(Lista l)⇒*lista*

sign(Matriisi l)⇒*matriisi*

Jos kompleksilukutila on Reaali:

sign(0) edustaa kompleksialueen yksikköpiiriä.

Kun kyseessä on lista tai matriisi, määrittää kaikkien elementtien etumerkit.

simult(kerroinMatriisi, vakioVektori[, tol]) ⇒ matriisi

Määrittää sarakevektorin, joka sisältää lineaarisen yhtälöryhmän ratkaisut.

Huomaa: Katso myös **linSolve()**, sivu 85.

kerroinMatriisin on oltava neliömatriisi, joka sisältää yhtälöiden kertoimet.

vakioVektorissa on oltava sama rivimäärä (sama koko) kuin *kerroinMatriisissa* ja sen tulee sisältää vakiot.

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

- Jos asetat **Automaattinen tai likimääräinen** -tilan valintaan Approximate (Likimääräinen), laskut suoritetaan liukulukuaritmetiikalla.
- Jos *Tol* jätetään pois tai sitä ei käytetä, oletusarvoinen toleranssi lasketaan seuraavasti:
 $5E-14 \cdot \max(\dim(\textit{kerroinMatriisi}) \cdot \textit{rowNorm}(\textit{kerroinMatriisi}))$

simult(kerroinMatriisi, vakioMatriisi[, tol]) ⇒ matriisi

Ratkaisee lineaarisia yhtälöryhmiä, joissa jokaisessa ryhmässä on samat yhtälöiden kertoimet mutta eri vakiot.

Jokaisen *vakioMatriisin* sarakkeen tulee sisältää jonkin yhtälöryhmän vakiot. Jokainen tulosmatriisin sarake sisältää vastaavan ryhmän ratkaisun.

Ratkaise yhtälöstä x ja y:

$$x + 2y = 1$$

$$3x + 4y = -1$$

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

Ratkaisu on x=-3 and y=2.

Ratkaise:

$$ax + by = 1$$

$$cx + dy = 2$$

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.

sin()



$\sin(Lista1) \Rightarrow lista$

Astekulmatilassa:

$\sin(Lista1)$ määrittää listan, joka sisältää $Lista1$:n kaikkien elementin sinit.

Graadikulmatilassa:

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

Radiaanikulmatilassa:

$\sin(neliomatriisi1) \Rightarrow neliomatriisi$

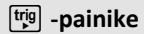
Radiaanikulmatilassa:

Määrittää $neliomatriisi1$:n matriisin sinin. Tämä ei ole sama kuin kunkin elementin sinin laskeminen. Laskentamenetelmä on kuvattu kohdassa $\cos()$.

$$\sin \begin{pmatrix} 1 & 5 & 3 \\ 4 & 2 & 1 \\ 6 & -2 & 1 \end{pmatrix} = \begin{bmatrix} 0.9424 & -0.04542 & -0.031999 \\ -0.045492 & 0.949254 & -0.020274 \\ -0.048739 & -0.00523 & 0.961051 \end{bmatrix}$$

$neliomatriisi1$:n on oltava diagonalisoitavissa. Vastaus sisältää aina liukulukuja.

$\sin^{-1}()$



$\sin^{-1}(Lista1) \Rightarrow lista$

Astekulmatilassa:

$\sin^{-1}(Lista1)$ määrittää listan, joka sisältää $Lista1$:n kaikkien elementtien käänteissinit.

Graadikulmatilassa:

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

Radiaanikulmatilassa:

Huomaa: Voit syöttää tämän funktion näppäimistöltä kirjoittamalla $\arcsin(\dots)$.

$\sin^{-1}(neliomatriisi1) \Rightarrow neliomatriisi$

Radiaanikulmatilassa ja suorakulmakompleksimuodossa:

Laskee $neliomatriisi1$:n matriisin käänteissinin. Tämä ei ole sama kuin kunkin elementin käänteissinin laskeminen. Laskentamenetelmä on kuvattu kohdassa $\cos()$.

$$\sin^{-1} \begin{pmatrix} 1 & 5 \\ 4 & 2 \end{pmatrix} = \begin{bmatrix} -0.174533 - 0.12198 \cdot i & 1.74533 - 2.35591 \cdot i \\ 1.39626 - 1.88473 \cdot i & 0.174533 - 0.593162 \cdot i \end{bmatrix}$$

$neliomatriisi1$:n on oltava diagonalisoitavissa. Vastaus sisältää aina liukulukuja.

sinh()Katalogi > **sinh(Lista1)**⇒*lista*

sinh(1.2) 1.50946

sinh (Lista1) määrittää listan, joka sisältää *Lista1*:n jokaisen elementin hyperbolisen sinin.

sinh({0,1,2,3}) {0,1.50946,10.0179}

sinh(neliömatriisi1)⇒*neliömatriisi*

Radiaanikulmatilassa:

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

$$\sinh \begin{pmatrix} 1 & 5 & 3 \\ 4 & 2 & 1 \\ 6 & -2 & 1 \end{pmatrix} \begin{bmatrix} 360.954 & 305.708 & 239.604 \\ 352.912 & 233.495 & 193.564 \\ 298.632 & 154.599 & 140.251 \end{bmatrix}$$

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

sinh⁻¹()Katalogi > **sinh⁻¹(Lista1)**⇒*lista*

sinh⁻¹(Lista1) määrittää listan, joka sisältää *Lista1*:n jokaisen elementin käänteiset hyperboliset sinin.

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


sinh⁻¹(neliömatriisi1)⇒*neliömatriisi*

Radiaanikulmatilassa:

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

$$\sinh^{-1} \begin{pmatrix} 1 & 5 & 3 \\ 4 & 2 & 1 \\ 6 & -2 & 1 \end{pmatrix} \begin{bmatrix} 0.041751 & 2.15557 & 1.1582 \\ 1.46382 & 0.926568 & 0.112557 \\ 2.75079 & -1.5283 & 0.57268 \end{bmatrix}$$

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

SinRegKatalogi > 

SinReg *X*, *Y* [, [*Iteraatiot*] , [*Jakso*] [, *Luokka*, *Sisällytä*]]

Laskee siniregression listoista X ja Y .
Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 151.)

Kaikkien listojen on oltava samankokoisia
Sisällytä-listaa lukuunottamatta.

X ja Y ovat riippumattomien ja riippuvien muuttujien listoja.

Iteraatiot on arvo, joka määrittää ratkaisun yrityskertojen (1-16) enimmäismäärän. Mikäli sitä ei määritetä, oletuksena käytetään arvoa 8. Suuremmilla arvoilla saadaan tyypillisesti parempi tarkkuus, mutta suoritus aika on pitempi ja päin vastoin.

Jakso määrittää arvioidun jakson. Mikäli sitä ei käytetä, X :n arvojen välisen eron tulisi olla sama ja arvojen tulisi olla peräkkäisessä järjestyksessä. Jos määrität *Jakson*, x :n arvojen väliset erot voivat olla erisuuria.

Luokka on luokkakoodien lista vastaavalle X - ja Y -datalle..

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

SinReg:n tulos esitetään aina radiaaneina riippumatta kulmatilan asetuksesta.

Lisätietoja listassa olevien tyhjen elementtien vaikutuksesta, katso Tyhjet elementtsivulla sivu 215.

Tulosmuuttuja	Kuvaus
stat.RegEqn	Regressioyhtälö: $a \cdot \sin(bx+c)+d$
stat.a, stat.b, stat.c, stat.d	Regressiokertoimet
stat.Resid	Regressioyhtälön jäännökset
stat.XReg	Muokatun Y Listan sisältämä datapisteiden lista, jota käytetään regressiossa komentojen <i>Frekv</i> , <i>Luokkalista</i> ja <i>Sisällytä luokat rajoitusten mukaisesti</i>

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

SortA

Katalogi > 

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 siirtyvät alas. Lisätietoja tyhjästä elementeistä, katso sivu 215.

$\{2,1,4,3\} \rightarrow list1$	$\{2,1,4,3\}$
SortA <i>list1</i>	Done
<i>list1</i>	$\{1,2,3,4\}$
$\{4,3,2,1\} \rightarrow list2$	$\{4,3,2,1\}$
SortA <i>list2,list1</i>	Done
<i>list2</i>	$\{1,2,3,4\}$
<i>list1</i>	$\{4,3,2,1\}$

SortD

Katalogi > 

SortD *Lista1* [, *Lista2*] [, *Lista3*] ...
SortD *Vektori1* [, *Vektori2*] [, *Vektori3*] ...

Muuten samanlainen kuin **SortA** paitsi, että **SortD** lajittelee elementit laskevaan järjestykseen.

Ensimmäisen argumentin sisältämät tyhjät elementit siirtyvät alas. Lisätietoja tyhjästä elementeistä, katso sivu 215.

$\{2,1,4,3\} \rightarrow list1$	$\{2,1,4,3\}$
$\{1,2,3,4\} \rightarrow list2$	$\{1,2,3,4\}$
SortD <i>list1,list2</i>	Done
<i>list1</i>	$\{4,3,2,1\}$
<i>list2</i>	$\{3,4,1,2\}$

►Sphere

Katalogi > 

Vektori ►Sphere

Huom: Vastauksen pakottaminen likimääräiseksi:

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

Näyttää rivi- tai sarakevektorin pallonmuotoisena [ρ $\angle\theta$ $\angle\phi$].

Vektorin on oltava kooltaan 3, ja se voi olla rivi- tai sarakevektori.

Huomaa: ►Sphere on näyttömuodon ohje, ei muunnosfunktio. Voit käyttää komentoa ainoastaan syöterivin lopussa.

Kämmenlaite: Paina .

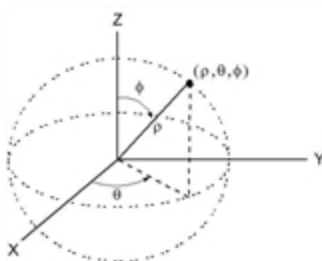
Windows®: Paina **Ctrl+Enter**.

Macintosh®: Paina **⌘+Enter**.

iPad®: Pidä **enter** ja valitse .

```
[ 1 2 3 ]►Sphere
[ 3.74166  ∟1.10715  ∟0.640522 ]
```

```
( [ 2  ∟π  3 ] )►Sphere
[ 3.60555  ∟0.785398  ∟0.588003 ]
```



sqrt()

sqrt(Lista1)⇒lista

Laskee argumentin neliöjuuren.

Kun kyseessä on lista, laskee kaikkien Lista1:n elementtien neliöjuuret.

Huomaa: Katso myös **Neliöjuurimalli**, sivu 1.

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 tilastoanalyseissä käytettävillä muuttujilla. Joissakin tapauksissa tästä voi olla seurauksena virhetilanne. Tilastoanalyseissä käytettävät muuttujanimet on esitetty alla olevassa taulukossa.

 $xlist:=\{1,2,3,4,5\}$ $\{1,2,3,4,5\}$
 $ylist:=\{4,8,11,14,17\}$ $\{4,8,11,14,17\}$
LinRegMx $xlist,ylist,1: stat.results$

"Title"	"Linear Regression (mx+b)"
"RegEqn"	"m*x+b"
"m"	3.2
"b"	1.2
"r ² "	0.996109
"r"	0.998053
"Resid"	"{...}"

<i>stat.values</i>	"Linear Regression (mx+b)"
	"m*x+b"
	3.2
	1.2
	0.996109
	0.998053
	"{-0.4,0.4,0.2,0.,-0.2}"

stat.a	stat.dfDenom	stat.MedianY	stat.Q3X	stat.SSBlock
stat.AdjR ²	stat.dfBlock	stat.MEPred	stat.Q3Y	stat.SSCol
stat.b	stat.dfCol	stat.MinX	stat.r	stat.SSX
stat.b0	stat.dfError	stat.MinY	stat.r ²	stat.SSY
stat.b1	stat.dfInteract	stat.MS	stat.RegEqn	stat.SSError
stat.b2	stat.dfReg	stat.MSBlock	stat.Resid	stat.SSInteract
stat.b3	stat.dfNumer	stat.MSCol	stat.ResidTrans	stat.SSReg
stat.b4	stat.dfRow	stat.MSError	stat.σ _x	stat.SSRow
stat.b5	stat.DW	stat.MSInteract	stat.σ _y	stat.tList
stat.b6	stat.e	stat.MSReg	stat.σ _{x1}	stat.UpperPred
stat.b7	stat.ExpMatrix	stat.MSRow	stat.σ _{x2}	stat.UpperVal
stat.b8	stat.F	stat.n	stat.Σ _x	stat.X̄
stat.b9	stat.FBlock	Stat. \hat{p}	stat.Σ _x ²	stat.X̄ ₁
stat.b10	stat.Fcol	stat. \hat{p} ₁	stat.Σ _{xy}	stat.X̄ ₂
stat.bList	stat.FInteract	stat. \hat{p} ₂	stat.Σ _y	stat.X̄Diff
stat.χ ²	stat.FreqReg	stat. \hat{p} Diff	stat.Σ _y ²	stat.X̄List
stat.c	stat.Frow	stat.PList	stat.s	stat.XReg
stat.CLower	stat.Leverage	stat.PVal	stat.SE	stat.XVal
stat.CLowerList	stat.LowerPred	stat.PValBlock	stat.SEList	stat.XValList
stat.CompList	stat.LowerVal	stat.PValCol	stat.SEPred	stat.ȳ
stat.CompMatrix	stat.m	stat.PValInteract	stat.sResid	stat.ŷ
stat.CookDist	stat.MaxX	stat.PValRow	stat.SESlope	stat.ŷList

stat.CUpper	stat.MaxY	stat.Q1X	stat.sp	stat.YReg
stat.CUpperList	stat.ME	stat.Q1Y	stat.SS	
stat.d	stat.MedianX			

Huomaa: Aina kun Listat & Taulukot -sovellus laskee tilastolaskujen vastauksia, se kopioi "stat." -ryhmän muuttujat "stat#." -ryhmään, jossa # on luku, jota lisätään automaattisesti. Tällä tavoin voit säilyttää aikaisemmat tulokset suorittaessasi useita laskutoimituksia.

stat.values

Katalogi > 

stat.values

Katso esimerkki kohdassa
stat.results.

Näyttää matriisin viimeksi 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

Radiaanikulma- ja automaattisissa tiloissa:

Laskee *Lista*:n sisältämien elementtien perusjoukon keskihajonnan.

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

Huomaa: *Listassa* tulee olla vähintään kaksi elementtiä. Tyhjiä elementtejä ei huomioida. Lisätietoja tyhjistä elementeistä, katso sivu 215.

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 tyhjistä elementeistä, katso sivu 215.

stDevSamp(*List*[, *frekvList*]) \Rightarrow *lauseke*

Laskee *List*an sisältämien elementtien otoksen keskihajonnan.

Jokainen *frekvList*an elementti näyttää *List*an vastaavien elementtien peräkkäisten esiintymien lukumäärän.

Huomaa: *List*assa tulee olla vähintään kaksi elementtiä. Tyhjiä elementtejä ei huomioida. Lisätietoja tyhjistä elementeistä, katso sivu 215.

stDevSamp(*MatriisiI*[, *frekvMatriisi*]) \Rightarrow *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 tyhjistä elementeistä, katso sivu 215.

StopKatalogi > **Stop**

Ohjelmointikomento: Pysäyttää ohjelman.

Stop ei ole sallittu funktioissa.

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

$i:=0$	0
Define $prog1()$ =Prgm	Done
For $i,1,10,1$	
If $i=5$	
Stop	
EndFor	
EndPrgm	
$prog1()$	Done
i	5

Store

Katso → (tallenna), sivu 197.

string() (merkkijono)Katalogi > 

string(Laus) ⇒ merkkijono

Sieventää lausekkeen *Laus* ja antaa vastauksen merkkijonona.

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.

$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix} \rightarrow m1$	$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$
$subMat(m1,2,1,3,2)$	$\begin{bmatrix} 4 & 5 \\ 7 & 8 \end{bmatrix}$
$subMat(m1,2,2)$	$\begin{bmatrix} 5 & 6 \\ 8 & 9 \end{bmatrix}$

Sum (Sigma)Katso $\Sigma()$, sivu 189.

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 215.

sum(MatriisiI[, Alku[, Loppu]]) ⇒ matriisi

Laskee rivivektorin, joka sisältää *MatriisiI*:n sarakkeiden elementtien summat.

Alku ja *Loppu* ovat valinnaisia. Ne määrittävät rivialueen.

Mikä tahansa tyhjä argumentti tuottaa tyhjän vastauksen. *MatriisiI*:ssä olevia tyhjiä elementtejä ei huomioida. Lisätietoja tyhjistä elementeistä, katso sivu 215.

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]

sumlf(Lista, Kriteerit[, SummaLista]) ⇒ arvo

Laskee kaikkien niiden *Listan* sisältämien elementtien kumuloituneen summan, jotka vastaavat määritettyjä kriteereitä *Kriteerit*. Voit halutessasi antaa kumuloivat elementit määrittämällä vaihtoehdoisen listan, *summaLista*.

*List*a voi olla lauseke, lista tai matriisi. *SummaListalla*, mikäli se määritetään, on oltava samat mitat kuin *Listalla*.

Kriteeri voi olla:

- Arvo, lauseke tai merkkijono. Esimerkiksi **34** kumuloi vain niitä *Listan* elementtejä, jotka sieventyvät arvoon 34.
- Boolean lauseke, joka sisältää symbolin ? kunkin elementin paikanpitäjänä.

sumIf()

Katalogi > 

Esimerkiksi lauseke $\sum_{i=1}^{10} i$ kumului vain niitä *Listan* elementtejä, jotka ovat alle 10.

Kun jokin *Listan* elementti vastaa kriteereitä *Kriteerit*, elementti lisätään kumuloituvaan summaan. Jos sisällytät funktioon *summaListan*, summaan lisätäänkin sen sijaan vastaava *summaListan* elementti.

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

Tyhjiä elementtejä ei huomioida. Lisätietoja tyhjistä elementeistä, katso sivu 215.

Huomaa: Katso myös *countIf()*, sivu 31.

sumSeq()

Katso $\Sigma()$, sivu 189.

system()

Katalogi > 

Laskee yhtälöryhmän listaksi muotoiltuna. Voit luoda yhtälöryhmän myös mallin avulla.

$$\text{solve}\left(\begin{cases} x+y=0 \\ x-y=8 \end{cases}, x, y\right) \quad x=4 \text{ and } y=-4$$

T

T (transponoi)


Katalogi > 

Matriisi $I^T \Rightarrow$ *matriisi*

Laskee *Matriisi* I :n transponoidun liittokompleksimatriisin.


Huomaa: Voit syöttää tämän operaattorin tietokoneen näppäimistöltä kirjoittamalla @t.

tan()

 -painike

Astekulmatilassa:

$\tan(Lista1) \Rightarrow lista$

tan() -painike

tan(Lista1) määrittää *Lista1*:n kaikkien elementtien tangenttien listan.

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

tan(neliömatriisi1) ⇒ *neliömatriisi*

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


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

Graadikulmatilassa:

Radiaanikulmatilassa:

Radiaanikulmatilassa:

$$\tan \left(\begin{bmatrix} 1 & 5 & 3 \\ 4 & 2 & 1 \\ 6 & -2 & 1 \end{bmatrix} \right) = \begin{bmatrix} -28.2912 & 26.0887 & 11.1142 \\ 12.1171 & -7.83536 & -5.48138 \\ 36.8181 & -32.8063 & -10.4594 \end{bmatrix}$$

tan⁻¹() -painike

tan⁻¹(Lista1) ⇒ *lista*

tan⁻¹(Lista1) luo listan *Lista1*: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(neliömatriisi1) ⇒ *neliömatriisi*

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

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

Astekulmatilassa:

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

Graadikulmatilassa:

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

Radiaanikulmatilassa:

$$\tan^{-1}(\{0,0.2,0.5\}) = \{0,0.197396,0.463648\}$$

Radiaanikulmatilassa:

$$\tan^{-1} \left(\begin{bmatrix} 1 & 5 & 3 \\ 4 & 2 & 1 \\ 6 & -2 & 1 \end{bmatrix} \right) = \begin{bmatrix} -0.083658 & 1.26629 & 0.62263 \\ 0.748539 & 0.630015 & -0.070012 \\ 1.68608 & -1.18244 & 0.455126 \end{bmatrix}$$

tanh(Lista1) ⇒ lista

tanh(Lista1) luo listan *Lista1*:n jokaisen elementin hyperbolisista tangenteista.

tanh(neliömatriisi1) ⇒ neliömatriisi

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

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

Radiaanikulmatilassa:

$$\text{tanh} \begin{pmatrix} 1 & 5 & 3 \\ 4 & 2 & 1 \\ 6 & -2 & 1 \end{pmatrix} = \begin{bmatrix} -0.097966 & 0.933436 & 0.425972 \\ 0.488147 & 0.538881 & -0.129382 \\ 1.28295 & -1.03425 & 0.428817 \end{bmatrix}$$

tanh⁻¹()

tanh⁻¹(Lista1) ⇒ lista

Suorakulmakompleksimuodossa:

tanh⁻¹(Lista1) luo listan *Lista1*:n jokaisen elementin käänteisistä hyperbolisista tangenteista.

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




tanh⁻¹(neliömatriisi1) ⇒ neliömatriisi

Radiaanikulmatilassa ja suorakulmakompleksimuodossa:

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

$$\text{tanh}^{-1} \begin{pmatrix} 1 & 5 & 3 \\ 4 & 2 & 1 \\ 6 & -2 & 1 \end{pmatrix} = \begin{bmatrix} -0.099353+0.164058 \cdot i & 0.267834-1.4908 \\ -0.087596-0.725533 \cdot i & 0.479679-0.94730 \\ 0.511463-2.08316 \cdot i & -0.878563+1.7901 \end{bmatrix}$$

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

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

tCdf()

tCdf(alaraja,yläraja,df) ⇒ luku, jos *alaraja* ja *yläraja* ovat lukuja, *lista*, jos *alaraja* ja

yläraja ovat listoja

Laskee Studentin t -todennäköisyysjakauman *alarajan* ja *ylärajan* välillä määritetyille vapausasteelle df .

Text

Text*kehotemerkkijono*[, *Näytälippu*]

Ohjelmointikomento: Keskeyttää ohjelman ja näyttää merkkijonon *kehotemerkkijono* valintaruudussa.

Kun käyttäjä valitsee **OK**-näppäimen, ohjelman suoritus jatkuu.



Valinnainen *lippu*-argumentti voi olla mikä tahansa lauseke.

- Jos *NäytäLippu* jätetään pois, tai jos se sieventyy arvoksi **1**, tekstimuotoinen viesti lisätään laskimen historiaan.
- Jos *NäytäLippu* sieventyy arvoon **0**, tekstimuotoista viestiä ei lisätä historiaan.

Jos ohjelma vaatii käyttäjän kirjoittaman vastauksen, katso **Request**, sivu **130**, tai **RequestStr**, sivu **131**.

Huomaa: Tätä komentoa voi käyttää käyttäjän määrittämän ohjelman sisällä mutta ei funktion sisällä.

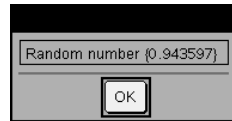
Määritä ohjelma, joka keskeytyy ja näyttää kunkin viidestä satunnaisluvusta valintaruudussa.

Paina mallin Prgm...EndPrgm jokaisen rivin lopussa näppäintä  näppäimen  sijaan. Tietokoneen näppäimistöllä **Alt**-näppäintä pidetään alhaalla ja painetaan **Enter**.

```
Define text_demo()=Prgm
  For i,1,5
    strinfo:="Random number " &
    string(rand(i))
    Text strinfo
  EndFor
EndPrgm
```

Suorita ohjelma:
text_demo()

Esimerkki yhdestä valintaruudusta:



Then

TInterval *Lista[,Frekv[,CTaso]]*

(Datalistan syöte)

TInterval $\bar{x}, Sx, n[,CTaso]$

(Yhteenvetotilaston syöte)

Laskee t -luottamusvälin. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 151.)

Lisätietoja listassa olevien tyhjien elementtien vaikutuksesta, katso Tyhjät elementtsivulla sivu 215.

Tulosmuuttuja	Kuvaus
stat.CLower, stat.CUpper	Tuntemattoman perusjoukon keskiarvon luottamusväli
stat. \bar{x}	Satunnaisesta normaalijakaumasta olevan datasekvenssin otoksen keskiarvo
stat.ME	Virhemarginaali
stat.df	Vapausasteet
stat. σ_x	Otoksen keskihajonta
stat.n	Otoksen keskiarvon sisältävän datasekvenssin pituus

TInterval_2Samp

TInterval_2Samp *Lista1,Lista2[,Frekv1
[,Frekv2[,CTaso[,Poolaus]]]]*

(Datalistan syöte)

TInterval_2Samp $\bar{x}1, Sx1, n1, \bar{x}2, Sx2, n2$
[,CTaso[,Poolaus]]

(Yhteenvetotilaston syöte)

Laskee kahden otoksen t -luottamusvälin. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 151.)

Poolaus=1 poolaa varianssit; *Poolaus=0* ei poolaa variansseja.

Lisätietoja listassa olevien tyhjen elementtien vaikutuksesta, katso Tyhjät elementtisivulla sivu 215.

Tulosmuuttuja	Kuvaus
stat.CLower, stat.CUpper	Luottamusväli, joka sisältää jakauman luottamusvälin todennäköisyyden
stat. $\bar{x}1$ - $\bar{x}2$	Satunnaisesta normaalijakaumasta olevien datasekvenssien otosten keskiarvot
stat.ME	Virhemarginaali
stat.df	Vapausasteet
stat. $\bar{x}1$, stat. $\bar{x}2$	Satunnaisesta normaalijakaumasta olevien datasekvenssien otosten keskiarvot
stat. $\sigma x1$, stat. $\sigma x2$	Listat 1:n ja Listat 2:n otosten keskihajonnat
stat.n1, stat.n2	Otosten lukumäärä datasekvensseissä
stat.sp	Poolattu keskihajonta. Laskettu, kun Poolaus = KYLLÄ

tPdf()

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

Laskee todennäköisyysfunktio (pdf) Studentin t -jakaumalle määritetyllä $x:n$ arvolla ja määritetyillä vapausasteilla df .

trace()

Laskee *neliömatriisin* jäljityksen (päälävistäjän kaikkien elementtien summan).

```
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 224.

lohko1 ja *lohko2* voivat olla joko yksi lauseke tai sarja lausekkeita, jotka on erotettu toisistaan kaksoispisteellä (:).

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

Esimerkki 2

Jos halaut nähdä kommentojen **Try**, **ClrErr** ja **PassErr** toiminnan, syötä oikealla näkyvä *eigenvals()*-ohjelma. Suorita ohjelma suorittamalla kukin seuraavista lausekkeista.

$$\text{eigenvals}\left(\begin{bmatrix} -3 \\ -41 \\ 5 \end{bmatrix}, [-1 \ 2 \ -3.1]\right)$$

Huomaa: Katso myös **ClrErr**, sivu 24, ja **PassErr**, sivu 114.

```
Define progI()=Prgm
  Try
  z:=z+1
  Disp "z incremented."
  Else
  Disp "Sorry, z undefined."
  EndTry
EndPrgm
```

Done

```
z:=1:progI()
-----
z incremented.
```

Done

```
DelVar z:progI()
-----
Sorry, z undefined.
```

Done

Define *eigenvals(a,b)*=Prgm

© Program *eigenvals(A,B)* displays eigenvalues of A·B

Try

Disp "A= ",a

Disp "B= ",b

Disp " "

Disp "Eigenvalues of A·B are:","eigVl(a*b)

Else

If *errCode*=230 Then

Disp "Error: Product of A·B must be a square matrix"

ClrErr

Else

PassErr

EndIf

EndTry

tTest**tTest** $\mu_0, \text{Lista}, [\text{Frekv}], [\text{Hypot}]$

(Datalistan syöte)

tTest $\mu_0, \bar{x}, s_x, n, [\text{Hypot}]$

(Yhteenvetotilaston syöte)

Testaa hypoteesia yhden tuntemattoman perusjoukon keskiarvoon μ , kun perusjoukon keskihajontaa σ ei tunneta. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 151.)

Testaa $H_0: \mu = \mu_0$, jonkin seuraavan vaihtoehdon suhteen:

Kun $H_a: \mu < \mu_0$, aseta *Hypot*<0

Kun $H_a: \mu \neq \mu_0$ (oletus), aseta *Hypot*=0

Kun $H_a: \mu > \mu_0$, aseta *Hypot*>0

Lisätietoja listassa olevien tyhjen elementtien vaikutuksesta, katso Tyhjät elementtivulla sivu 215.

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 *Lista1, Lista2, [Frekv1], [Frekv2], [Hypot], [Poolaus]]]*

(Datalistan syöte)

tTest_2Samp $\bar{x}1, sx1, n1, \bar{x}2, sx2, n2, Hypot$
[,Poolaus]

(Yhteenvetotilaston syöte)

Suorittaa kahden otoksen *t*-testin. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 151.)Testaa $H_0: \mu_1 = \mu_2$, jonkin seuraavan vaihtoehdon suhteen:Kun $H_a: \mu_1 < \mu_2$, aseta *Hypot*<0Kun $H_a: \mu_1 \neq \mu_2$ (oletus), aseta *Hypot*=0Kun $H_a: \mu_1 > \mu_2$, aseta *Hypot*>0*Poolaus*=1 poolaa varianssit*Poolaus*=0 ei poolaa variansseja

Lisätietoja listassa olevien tyhjien elementtien vaikutuksesta, katso Tyhjät elementitsivulla sivu 215.

Tulosmuuttuja	Kuvaus
stat.t	Keskiarvojen erotukselle laskettu vakio-ohjearvo
stat.PVal	Alin merkitsevyytaso, jolla nollahypoteesi voidaan hylätä
stat.df	t-tilaston vapausasteet
stat. $\bar{x}1$, stat. $\bar{x}2$	<i>Lista 1</i> :n ja <i>Lista 2</i> :n sisältämien datasekvenssien otosten keskiarvot
stat.sx1, stat.sx2	<i>Lista 1</i> :n ja <i>Lista 2</i> :n sisältämien datasekvenssien otosten keskihajonnat
stat.n1, stat.n2	Otosten koko
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

Talouslaskentafunktio, joka laskee rahan tulevan arvon.

tvmFV()Katalogi > 

Huomaa: TVM-funktioissa käytetyt argumentit on kuvattu TVM-argumenttien taulukossa, sivu 166. Katso myös **amortTbl()**, sivu 7.

tvmI()Katalogi > 

tvmI($N, PV, Pmt, FV, [PpY], [CpY], [PmtAt]$) $\Rightarrow arvo$

tvmI(240,100000,-1000,0,12,12)	10.5241
--------------------------------	---------

Talouselaskentafunktio, joka laskee vuosikoron.

Huomaa: TVM-funktioissa käytetyt argumentit on kuvattu TVM-argumenttien taulukossa, sivu 166. Katso myös **amortTbl()**, sivu 7.

tvmN()Katalogi > 

tvmN($I, PV, Pmt, FV, [PpY], [CpY], [PmtAt]$) $\Rightarrow arvo$

tvmN(5,0,-500,77641,12,12)	120.
----------------------------	------

Talouselaskentafunktio, joka laskee maksuerien lukumäärän.

Huomaa: TVM-funktioissa käytetyt argumentit on kuvattu TVM-argumenttien taulukossa, sivu 166. Katso myös **amortTbl()**, sivu 7.

tvmPmt()Katalogi > 

tvmPmt($N, I, PV, FV, [PpY], [CpY], [PmtAt]$) $\Rightarrow arvo$

tvmPmt(60,4,30000,0,12,12)	-552.496
----------------------------	----------

Talouselaskentafunktio, joka laskee jokaisen maksuerän määrän.

Huomaa: TVM-funktioissa käytetyt argumentit on kuvattu TVM-argumenttien taulukossa, sivu 166. Katso myös **amortTbl()**, sivu 7.

$tvmPV(N, I, Pmt, FV, [PpY], [CpY], [PmtAt]) \Rightarrow arvo$

$tvmPV(48, 4, -500, 30000, 12, 12)$ -3426.7

Talouselaskentafunktio, joka laskee nykyarvon.

Huomaa: TVM-funktioissa käytetyt argumentit on kuvattu TVM-argumenttien taulukossa, sivu 166. Katso myös **amortTbl()**, sivu 7.

TVM-argumentti*	Kuvaus	Datatyyppi
N	Maksuerien lukumäärä	reaaliluku
I	Vuosikorko	reaaliluku
PV	Nykyarvo	reaaliluku
Pmt	Maksun määrä	reaaliluku
FV	Tuleva arvo	reaaliluku
PpY	Maksuerien määrä vuodessa, oletusarvo=1	kokonaisluku > 0
CpY	Korkojaksoja vuodessa, oletusarvo=1	kokonaisluku > 0
$PmtAt$	Erääntyvän maksun määrä kunkin jakson lopussa tai alussa, oletusarvo=loppu	kokonaisluku (0=loppu, 1=alku)

* Nämä rahan aika-arvon argumenttien nimet ovat samat kuin TVM-muuttujien nimet (kuten **tvm.pv** ja **tvm.pmt**), joita käytetään *Laskin*-sovelluksen talouselaskentatoiminnossa. Talouselaskentafunktioiden argumenttien arvot tai vastaukset eivät kuitenkaan tallennu TVM-muuttujiin.

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

Laskee kahden muuttujan tilastot. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 151.)

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

X ja Y ovat riippumattomien ja riippuvien muuttujien listoja.

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

Luokka on luokkakoodien lista vastaavalle X - ja Y -datalle.

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

Tyhjä elementti jossakin listassa X , *Frekv* tai *Luokka* saa aikaan, että kaikkien listojen vastaava elementti on tyhjä. Tyhjä elementti jossakin listassa $X1$ - $X20$ saa aikaan, että kaikkien listojen vastaava elementti on tyhjä. Lisätietoja tyhjistä elementeistä, katso sivu 215.

Tulosmuuttuja	Kuvaus
stat. \bar{x}	$x:n$ arvojen keskiarvo
stat. x	$x:n$ arvojen summa
stat. x^2	$x^2:n$ arvojen summa
stat. s_x	$x:n$ otoksen keskihajonta
stat. σ_x	$x:n$ perusjoukon keskihajonta
stat. n	Datapisteiden lukumäärä
stat. \bar{y}	$y:n$ arvojen keskiarvo
stat. Σy	$y:n$ arvojen summa
stat. Σy^2	$y^2:n$ arvojen summa
stat. s_y	$y:n$ otoksen keskihajonta
stat. σ_y	$y:n$ perusjoukon keskihajonta
stat. Σxy	$x \cdot y$ -arvojen summa
stat. r	Korrelaatiokerroin
stat. MinX	$x:n$ arvojen minimi

Tulosmuuttuja	Kuvaus
stat.Q ₁ X	x:n ensimmäinen neljännes
stat.MedianX	x:n mediaani
stat.Q ₃ X	x:n 3. neljännes
stat.MaxX	x:n arvojen maksimi
stat.MinY	y:n arvojen minimi
stat.Q ₁ Y	y:n ensimmäinen neljännes
stat.MedY	y:n mediaani
stat.Q ₃ Y	x:n kolmas neljännes
stat.MaxY	y:n arvojen maksimi
stat.Σ(x- \bar{x}) ²	x:n keskiarvon poikkeamien neliöiden summa
stat.Σ(y- \bar{y}) ²	y:n keskiarvon poikkeamien neliöiden summa

U

unitV()

Katalogi > 

yksikköV(Vektoril)⇒vektori

Laskee joko rivi- tai sarakeyksikkövektorin riippuen *Vektoril*:n muodosta.

Vektoril :n on oltava joko yksirivinen matriisi tai yksisarakeinen matriisi.

unLock

Katalogi > 

unLockMuutt1[, Muutt2] [, Muutt3] ... unLockMuutt.

Vapauttaa määritetyt muuttujat tai muuttujaryhmän. Lukittuja muuttujia ei voi muokata eikä poistaa.

Katso **Lock**, sivu 89, ja **getLockInfo()**, sivu 67.

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

varPop()Katalogi > **varPop(Lista[, frekvLista])** ⇒ lausekeLaskee *Listan* perusjoukon 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 tyhjistä elementeistä, katso sivu 215.

varSamp()Katalogi > **varSamp(Lista[, frekvLista])** ⇒ lausekeLaskee *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 tyhjistä elementeistä, katso sivu 215.

varSamp(MatriisiI[, frekvMatriisi]) ⇒ matriisiLaskee rivivektorin, joka sisältää *MatriisiI*:n kaikkien sarakkeiden otoksen varianssin.

$$\text{varSamp} \left(\begin{pmatrix} 1 & 2 & 5 \\ -3 & 0 & 1 \\ .5 & .7 & 3 \end{pmatrix} \right) \begin{bmatrix} 4.75 & 1.03 & 4 \end{bmatrix}$$

$$\text{varSamp} \left(\begin{pmatrix} -1.1 & 2.2 \\ 3.4 & 5.1 \\ -2.3 & 4.3 \end{pmatrix}, \begin{pmatrix} 6 & 3 \\ 2 & 4 \\ 5 & 1 \end{pmatrix} \right) \begin{bmatrix} 3.91731 & 2.08411 \end{bmatrix}$$

Jokainen *frekvMatriisin* elementti näyttää *Matriisi1*: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 tyhjästä elementeistä, katso sivu 215.

Huomaa: *Matriisi1*:ssä on oltava vähintään kaksi elementtiä.

W

Wait



Wait *aikaSekunteina*

Keskeyttää toiminnon *aikaSekunteina* sekunnin ajaksi.

Wait on erityisen käyttökelpoinen ohjelmassa, jossa tarvitaan lyhyttä viivettä pyydettyjen tietojen saamiseksi käyttöön.

Argumentin *aikaSekunteina* on oltava lausekkeen muodossa, joka yksinkertaistaa desimaaliarvoon välillä 0–100. Komento pyöristää tämän arvon lähimpään 0,1 sekuntiin.

Peruuttaaksesi **Wait** käynnissä olevan,

- **Kämmenlaite:** Pidä -painiketta painettuna ja paina toistuvasti -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.

Odottaaksesi 4 sekuntia:

Wait 4

Odottaaksesi 1/2 sekuntia:

Wait 0.5

Odottaaksesi 1,3 sekuntia käyttäen muuttujaa *seklukema*:

seklukema:=1.3

Wait seklukema

Tämä esimerkki sytyttää vihreän LED-valon päälle 0,5 sekunnin ajaksi ja sammuttaa sen sitten.

Send "SET GREEN 1 ON"

Wait 0.5

Send "SET GREEN 1 OFF"

Huomaa: Voit käyttää komentoa **Wait** käyttäjän määrittelemän ohjelman sisällä mutta ei funktion sisällä.

warnCodes ()

warnCodes(*Expr1*, *StatusVar*) \Rightarrow *expression*

Laskee lausekkeen *Expr1*, antaa tuloksen ja varastoi mahdollisten luotujen varoitusten koodit *StatusVar* -luettelomuuttujaan. Jos varoituksia ei ole luotu, tämä funktio kohdistaa funktiolle *StatusVar* tyhjän luettelon.

Expr1 voi olla mikä tahansa sallittu TI-Nspire™:n tai TI-Nspire™ CAS:n matemaattinen lauseke. Et voi käyttää komentoa tai tehtävää *Expr1*-lausekkeena.

StatusVar:n arvon on oltava sallittu muuttujan nimi.

Katso varoituskoodien ja niihin liittyvien viestien luettelo sivulla sivu 231.

when()

when(*Ehto*, *tosiTulos* [, *epätosiTulos*][, *tuntematonTulos*])
 \Rightarrow *lauseke*

Määrittää totuusarvon *tosiTulos*, *epätosiTulos* tai *tuntematonTulos* riippuen siitä, onko *Ehto* tosi, epätosi vai tuntematon. Antaa vastauksena syötteen, jos oikean vastauksen määrittämiseen on liian vähän argumentteja.

Jätä pois sekä *epätosiTulos* että *tuntematonTulos*, kun haluat, että lauseke määritetään vain alueella, jolla *Ehto* on tosi.

Käytä komentoa **undef** *epätosiTulos*, kun haluat määrittää lausekkeen, joka piirtyy vain jollekin välille.

$\text{when}(x < 0, x + 3) x = 5$	undef
-------------------------------------	-------

when()

Katalogi > 

when() on hyödyllinen komento rekursiivisten funktioiden määrittämisessä.

$\text{when}(n > 0, n \cdot \text{factoral}(n-1), 1) \rightarrow \text{factoral}(n)$	Done
$\text{factoral}(3)$	6
3!	6

While

Katalogi > 

While *Ehto*
Lohko
EndWhile

Suorittaa *Lohkon* sisältämät lausekkeet, mikäli *Ehto* on tosi.

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

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

Define $\text{sum_of_recip}(n)$ = Func	
Local $i, \text{tempsum}$	
$1 \rightarrow i$	
$0 \rightarrow \text{tempsum}$	
While $i \leq n$	
$\text{tempsum} + \frac{1}{i} \rightarrow \text{tempsum}$	
$i + 1 \rightarrow i$	
EndWhile	
Return tempsum	
EndFunc	
	Done
$\text{sum_of_recip}(3)$	$\frac{11}{6}$
	6

X

xor

Katalogi > 

BooleanLaus1 **xor** *BooleanLaus2* antaa vastauksena *Boolean* lausekkeen *BooleanList1* **xor** *BooleanList2* antaa vastauksena *Boolean* listan *BooleanMatriisi1* **xor** *BooleanMatriisi2* antaa vastauksena *Boolean* matriisin

true xor true	false
5 > 3 xor 3 > 5	true

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 111.

Kokonaisluku1 xor Kokonaisluku2 \Rightarrow kokonaisluku

Vertaa kahta reaalikokonaislukua bitti bitiltä xor-operaation avulla. Sisäisesti kumpikin kokonaisluku muunnetaan etumerkilliseksi, 64 bitin binaariluvuksi. Kun vastaavia bittejä verrataan, tulos on 1, jos jompikumpi bitti (mutta ei molemmat) on 1; tulos on 0, jos kumpikin bitti on 0 tai kumpikin bitti on 1. Laskettu arvo edustaa bittituloksia, ja se näkyy kantelukutilan mukaisesti.

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

Jos syötät desimaalikokonaisluvun, joka on etumerkillisen, 64 bitin binaarimuodon lukualueen ulkopuolella, laskin käyttää symmetristä modulo-operaatiota, jotta arvo saadaan oikealle alueelle. Lisätietoja, katso ►Base2, sivu 17.

Huomaa: Katso or, sivu 111.

Heksadesimaalisessa kantelukutilassa:

Tärkeää: Nolla, ei O-kirjain.

0h7AC36 xor 0h3D5F	0h79169
--------------------	---------

Binaarisessa kantelukutilassa:

0b100101 xor 0b100	0b100001
--------------------	----------

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

Z

zInterval

zInterval $\sigma, \text{Lista}, [\text{Frekv}], [\text{CTaso}]$

(Datalistan syöte)

zInterval $\sigma, \bar{x}, n, [\text{CTaso}]$

(Yhteenvetotilaston syöte)

Laskee z-luottamusvälin. Tulosten yhteenveto tallentuu *stat.results*-muuttuun. (Katso sivu 151.)

Lisätietoja listassa olevien tyhjiin elementtien vaikutuksesta, katso Tyhjät elementitsivulla sivu 215.

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

zInterval_1Prop x, n [,CTaso]

Laskee yhden osuuden z -luottamusvälin. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 151.)

x on ei-negatiivinen kokonaisluku.

Lisätietoja listassa olevien tyhjiin elementtien vaikutuksesta, katso Tyhjät elementitsivulla sivu 215.

Tulosmuuttuja	Kuvaus
stat.CLower, stat.CUpper	Luottamusväli, joka sisältää jakauman luottamusvälin todennäköisyyden
stat. \hat{p}	Laskettu onnistumisten osuus
stat.ME	Virhemarginaali
stat.n	Otosten lukumäärä datasekvenssissä

zInterval_2Prop

zInterval_2Prop $x1, n1, x2, n2$ [,CTaso]

Laskee kahden osuuden z -luottamusvälin. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 151.)

$x1$ ja $x2$ ovat ei-negatiivisia kokonaislukuja.

Lisätietoja listassa olevien tyhjiin elementtien vaikutuksesta, katso Tyhjät elementitsivulla sivu 215.

Tulosmuuttuja	Kuvaus
stat.CLower, stat.CUpper	Luottamusväli, joka sisältää jakauman luottamusvälin todennäköisyyden
stat. \hat{p} Diff	Osuuksien välinen laskettu erotus
stat.ME	Virhemarginaali
stat. $\hat{p}1$	Arvio ensimmäisen näytteen osuudesta
stat. $\hat{p}2$	Arvio toisen näytteen osuudesta
stat.n1	Otoksen koko datasekvenssissä 1
stat.n2	Otoksen koko datasekvenssissä 2

zInterval_2Samp

zInterval_2Samp $\sigma_1, \sigma_2, Lista1, Lista2$
[,Frekv1[,Frekv2[,CTaso]]]

(Datalistan syöte)

zInterval_2Samp $\sigma_1, \sigma_2, \bar{x}1, n1, \bar{x}2, n2$ [,CTaso]

(Yhteenvetotilaston syöte)

Laskee kahden näytteen z -luottamusvälin. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 151.)

Lisätietoja listassa olevien tyhjiin elementtien vaikutuksesta, katso Tyhjät elementitsivulla sivu 215.

Tulosmuuttuja	Kuvaus
stat.CLower, stat.CUpper	Luottamusväli, joka sisältää jakauman luottamusvälin todennäköisyyden

Tulosmuuttuja	Kuvaus
stat. $\bar{x}1-\bar{x}2$	Satunnaisesta normaalijakaumasta olevien datasekvenssien otosten keskiarvot
stat.ME	Virhemarginaali
stat. $\bar{x}1$, stat. $\bar{x}2$	Satunnaisesta normaalijakaumasta olevien datasekvenssien otosten keskiarvot
stat. $\sigma x1$, stat. $\sigma x2$	Listat 1:n ja Listat 2:n otosten keskihajonnat
stat.n1, stat.n2	Otosten lukumäärä datasekvensseissä
stat.r1, stat.r2	Datasekvenssien Listat 1 ja Listat 2 tunnetut perusjoukon keskihajonnat

zTest

Katalogi > 

zTest $\mu0, \sigma, Lista, [Frekv[, Hypot]]$

(Datalistan syöte)

zTest $\mu0, \sigma, \bar{x}, n[, Hypot]$

(Yhteenvetotilaston syöte)

Suorittaa z-testin frekvenssillä *frekvlista*. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 151.)

Testaa $H_0: \mu = \mu_0$, jonkin seuraavan vaihtoehdon suhteen:

Kun $H_a: \mu < \mu_0$, aseta *Hypot*<0

Kun $H_a: \mu \neq \mu_0$ (oletus), aseta *Hypot*=0

Kun $H_a: \mu > \mu_0$, aseta *Hypot*>0

Lisätietoja listassa olevien tyhjen elementtien vaikutuksesta, katso Tyhjät elementitsivulla sivu 215.

Tulosmuuttuja	Kuvaus
stat.z	$(\bar{x} - \mu_0) / (\sigma / \text{sqrt}(n))$
stat.P Value	Pienin todennäköisyys, jolla nollahypoteesi voidaan hylätä
stat. \bar{x}	Listan sisältämän datasekvenssin otoksen keskiarvo
stat.sx	Datasekvenssin otoksen keskihajonta. Lasketaan vain <i>Data</i> -syötteelle.
stat.n	Otoksen koko

zTest_1Prop $p0, x, n[, Hypot]$

Laskee yhden osuuden z -testin. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 151.)

x on ei-negatiivinen kokonaisluku.

Testaa $H_0: p = p0$ jonkin seuraavan vaihtoehdon suhteen:

Kun $H_a: p > p0$, aseta *Hypot*>0

Kun $H_a: p \neq p0$ (*oletus*), aseta *Hypot*=0

Kun $H_a: p < p0$, aseta *Hypot*<0

Lisätietoja listassa olevien tyhjien elementtien vaikutuksesta, katso Tyhjät elementtivuorolla sivu 215.

Tulosmuuttuja	Kuvaus
stat.p0	Perusjoukon osuuden hypoteesiarvo
stat.z	Osuudelle laskettu vakio-ohjearvo
stat.PVal	Alin merkitsevyytaso, jolla nollahypoteesi voidaan hylätä
stat. \hat{p}	Arvioitu otoksen osuus
stat.n	Otoksen koko

zTest_2Prop**zTest_2Prop** $x1, n1, x2, n2[, Hypot]$

Laskee kahden osuuden z -testin. Tulosten yhteenveto tallentuu *stat.results*-muuttujaan. (Katso sivu 151.)

$x1$ 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 elementtivuorolla sivu 215.

Tulosmuuttuja	Kuvaus
stat.z	Osuuksien erotukselle laskettu vakio-ohjearvo
stat.PVal	Alin merkitsevyytaso, jolla nollahypoteesi voidaan hylätä
stat. $\hat{p}1$	Arvio ensimmäisen näytteen osuudesta
stat. $\hat{p}2$	Arvio toisen näytteen osuudesta
stat. \hat{p}	Poolattu arvio otoksen osuudesta
stat.n1, stat.n2	Kokeissa 1 ja 2 otettujen otosten lukumäärä

zTest_2Samp

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 151.)

Testaa $H_0: \mu_1 = \mu_2$, jonkin seuraavan vaihtoehdon suhteen:

Kun $H_a: \mu_1 < \mu_2$, aseta *Hypot*<0

Kun $H_a: \mu_1 \neq \mu_2$ (oletus), aseta *Hypot*=0

Kun $H_a: \mu_1 > \mu_2$, *Hypot*>0

Lisätietoja listassa olevien tyhjiin elementtien vaikutuksesta, katso Tyhjät elementtsivulla sivu 215.

Tulosmuuttuja	Kuvaus
stat.z	Keskiarvojen erotukselle laskettu vakio-ohjearvo
stat.PVal	Alin merkitsevyytaso, jolla nollahypoteesi voidaan hylätä
stat. $\bar{x}1$, stat. $\bar{x}2$	<i>Lista1</i> :n ja <i>Lista2</i> :n sisältämien datasekvenssien otosten keskiarvot
stat.sx1, stat.sx2	<i>Lista1</i> :n ja <i>Lista2</i> :n sisältämien datasekvenssien otosten keskihajonnat
stat.n1, stat.n2	Otosten koko

Symbolit

+ (yhteenlasku)

painike

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.

$15 + \{10, 15, 20\}$	$\{25, 30, 35\}$
$\{10, 15, 20\} + 15$	$\{25, 30, 35\}$

Huomaa: Käytä merkintää $+$ (piste plus), kun haluat lisätä lausekkeen jokaiseen elementtiin.

$20 + \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$	$\begin{bmatrix} 21 & 2 \\ 3 & 24 \end{bmatrix}$
---	--

-(vähennyslasku)

painike

$Lista1 - Lista2 \Rightarrow lista$

$Matriisi1 - Matriisi2 \Rightarrow matriisi$

Vähentää $Lista2$:n (tai $Matriisi2$:n) jokaisen elementin $Lista1$:n (tai $Matriisi1$:n) vastaavasta elementistä ja antaa tuloksena vastaukset.

Argumenttien tulee olla mitoiltaan samanlaisia.

$15 - \{10, 15, 20\}$	$\{5, 0, -5\}$
$\{10, 15, 20\} - 15$	$\{-5, 0, 5\}$

-(vähennyslasku)

[-] painike

Huomaa: Käytä merkintää .- (piste miinus), kun haluat vähentää lausekkeen jokaisesta elementistä.

$$20 - \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \quad \begin{bmatrix} 19 & -2 \\ -3 & 16 \end{bmatrix}$$

·(kertolasku)

[x] painike

Laskee kahden argumentin tulon.

$Lista1 \cdot Lista2 \Rightarrow lista$

Luo listan, joka sisältää $Lista1$:n ja $Lista2$:n vastaavien elementtien tulot.

Listojen tulee olla mitoiltaan samanlaisia.

$Matriisi1 \cdot Matriisi2 \Rightarrow matriisi$

Laskee $Matriisi1$:n ja $Matriisi2$:n matriisitulon.

$Matriisi1$:n sarakkeiden lukumäärän on oltava sama kuin $Matriisi2$:n rivien lukumäärä.

Huomaa: Käytä merkintää .· (piste kerro), kun haluat kertoa lausekkeen jokaisella elementillä.

/ (jakolasku)

[÷] painike

Huomaa: Katso myös **Murtolukumalli**, sivu 1.

$Lista1 / Lista2 \Rightarrow lista$


Määrittää listan, joka sisältää osamäärät laskutoimituksista $Lista1$ jaettuna $Lista2$:lla.

$$\frac{\{1, 2, 3\}}{\{4, 5, 6\}} \quad \left\{ 0.25, \frac{2}{5}, \frac{1}{2} \right\}$$

Listojen tulee olla mitoiltaan samanlaisia.

Huomaa: Käytä merkintää ./ (piste jaa), kun haluat jakaa lausekkeen jokaisella elementillä.

^ (potenssi)

 painike

$List1 \wedge Lista2 \Rightarrow lista$

Laskee ensimmäisen argumentin korotettuna toisen argumentin potenssiin.

Huomaa: Katso myös **Eksponenttimalli**, sivu 1.

Jos kyseessä on lista, laskee *List1*:n elementit korotettuna *List2*:n vastaavien elementtien potenssiin.

Reaalilukujen alueella murtolukupotenssit, joilla on supistetut eksponentit ja parittomat nimittäjät, käyttävät reaalista aluetta versus pääalue kompleksitilassa.

$$\{1,2,3,4\}^{-2} \quad \left\{1, \frac{1}{4}, \frac{1}{9}, \frac{1}{16}\right\}$$

neliömatriisi1 \wedge kokonaisluku \Rightarrow matriisi

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}^2 \quad \begin{bmatrix} 7 & 10 \\ 15 & 22 \end{bmatrix}$$

Laskee *neliömatriisi1*:n korotettuna kokonaisluvun *kokonaisluku* potenssiin.

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}^{-1} \quad \begin{bmatrix} -2 & 1 \\ 3 & -1 \\ 2 & 2 \end{bmatrix}$$

neliömatriisi1:n on oltava neliömatriisi.

Jos *kokonaisluku* = -1, laskee käänteismatriisin.

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}^{-2} \quad \begin{bmatrix} 11 & -5 \\ 2 & 2 \\ -15 & 7 \\ 4 & 4 \end{bmatrix}$$

Jos *kokonaisluku* < -1, laskee käänteismatriisin korotettuna sopivaan positiiviseen potenssiin.

x² (neliö)

 painike

Laskee argumentin neliön.

$$4^2 \quad 16$$

$List1^2 \Rightarrow lista$

$$\{2,4,6\}^2 \quad \{4,16,36\}$$

Laskee listan, joka sisältää *List1*:n elementtien neliöt.

$$\begin{bmatrix} 2 & 4 & 6 \\ 3 & 5 & 7 \\ 4 & 6 & 8 \end{bmatrix}^2 \quad \begin{bmatrix} 40 & 64 & 88 \\ 49 & 79 & 109 \\ 58 & 94 & 130 \end{bmatrix}$$

*neliömatriisi1*² \Rightarrow matriisi

$$\begin{bmatrix} 2 & 4 & 6 \\ 3 & 5 & 7 \\ 4 & 6 & 8 \end{bmatrix}^{\wedge 2} \quad \begin{bmatrix} 4 & 16 & 36 \\ 9 & 25 & 49 \\ 16 & 36 & 64 \end{bmatrix}$$

x² (neliö) painike

Laskee *neliömatrissi*1:n matriisinelion. Tämä ei ole sama kuin kunkin elementin neliön laskeminen. Käytä merkintää $.^2$, kun haluat laskea jokaisen elementin neliön.

.+ (piste lisää) painikkeet

*Matriisi*1 .+ *Matriisi*2 \Rightarrow *matriisi*

*Matriisi*1 .+ *Matriisi*2 laskee matriisin, joka on *Matriisi*1:n ja *Matriisi*2:n vastaavien elementtiparien summa.

.- (piste-erotus) painikkeet

*Matriisi*1 .- *Matriisi*2 \Rightarrow *matriisi*

*Matriisi*1 .- *Matriisi*2 laskee matriisin, joka on *Matriisi*1:n ja *Matriisi*2:n vastaavien elementtiparien välinen erotus.

.· (pistetulo) painikkeet

*Matriisi*1 .· *Matriisi*2 \Rightarrow *matriisi*

*Matriisi*1 .· *Matriisi*2 laskee matriisin, joka on *Matriisi*1:n ja *Matriisi*2:n vastaavien elementtiparien tulo.

. / (pisteosamäärä) painikkeet


*Matriisi*1 . / *Matriisi*2 \Rightarrow *matriisi*

*Matriisi*1 . / *Matriisi*2 laskee matriisin, joka on *Matriisi*1:n ja *Matriisi*2:n vastaavien elementtiparien osamäärä.

.^ (pistepotenssi) painikkeet


*Matriisi*1.^ *Matriisi*2 \Rightarrow *matriisi*

.^ (pistepotenssi)

  painikkeet

Matriisi1 .^ *Matriisi2* laskee matriisin, jossa *Matriisi2*:n jokainen elementti on eksponentti *Matriisi1*:n vastaavalle elementille.

-(negaatio)

 painike

-*Listal* ⇒ *lista*

-*Matriisi1* ⇒ *matriisi*

Laskee argumentin negaation.

Kun kyseessä on lista tai matriisi, määrittää kaikkien elementtien negaatiot.

Jos argumentti on binaarinen tai heksadesimaalinen kokonaisluku, negaatio antaa kakkosen komplementin.

Binaarisessa kantalukutilassa:

```
-0b100101
0b11111111111111111111111111111111▶
```

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

% (prosentti)

  painikkeet

Listal % ⇒ *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)

 painike


Laus1 = *Laus2* ⇒ *Boolean lauseke*

Listal = *Listal2* ⇒ *Boolean lista*

Matriisi1 = *Matriisi2* ⇒ *Boolean matriisi*

Esimerkkifunktio, jossa on käytetty matematiikkatestisymboleita: =, ≠, <, ≤, >, ≥

= (on yhtä kuin)

 painike

Antaa totuusarvon tosi, jos *Laus1* määritetään olevan yhtä kuin *Laus2*.

Antaa totuusarvon epätosi, jos *Laus1* määritetään olevan ei yhtä kuin *Laus2*.

Kaikissa muissa tapauksissa vastauksena on yhtälön sievennetty muoto.

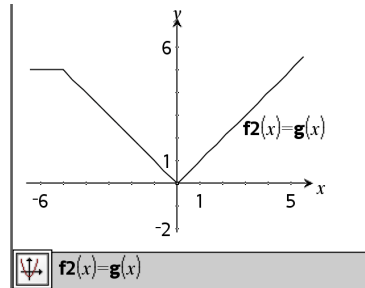
Listojen ja matriisien kohdalla vastauksena on vertailuja elementti elementiltä.

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


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

Done

Funktion $g(x)$ kuvaajan piirtämisen tulos



≠ (ei yhtä kuin)

 painikkeet

$Laus1 \neq Laus2 \Rightarrow$ Boolean lauseke

$Lista1 \neq Lista2 \Rightarrow$ Boolean lista

$Matriisi1 \neq Matriisi2 \Rightarrow$ Boolean matriisi

Antaa totuusarvon tosi, jos *Laus1* määritetään olevan ei yhtä kuin *Laus2*.



Antaa totuusarvon epätosi, jos *Laus1* määritetään olevan yhtä kuin *Laus2*.

Kaikissa muissa tapauksissa vastauksena on yhtälön sievennetty muoto.

Listojen ja matriisien kohdalla vastauksena on vertailuja elementti elementiltä.



Katso esimerkki kohdasta “=” (on yhtä kuin).

≠ (ei yhtä kuin)

  painikkeet

Huomaa: Voit syöttää tämän operaattorin näppäimistöltä kirjoittamalla /=

< (pienempi kuin)

  painikkeet

$Laus1 < Laus2 \Rightarrow \text{Boolean lauseke}$

Katso esimerkki kohdasta “=” (on yhtä kuin).

$Lista1 < Lista2 \Rightarrow \text{Boolean lista}$

$Matriisi1 < Matriisi2 \Rightarrow \text{Boolean matriisi}$



Antaa totuusarvon tosi, jos *Laus1* määritetään olevan pienempi kuin *Laus2*.

Antaa totuusarvon epätosi, jos *Laus1* määritetään olevan suurempi tai yhtä suuri kuin *Laus2*.

Kaikissa muissa tapauksissa vastauksena on yhtälön sievennetty muoto.

Listojen ja matriisien kohdalla vastauksena on vertailuja elementti elementiltä.

≤ (pienempi tai yhtä suuri kuin)

  painikkeet

$Laus1 \leq Laus2 \Rightarrow \text{Boolean lauseke}$

Katso esimerkki kohdasta “=” (on yhtä kuin).

$Lista1 \leq Lista2 \Rightarrow \text{Boolean lista}$

$Matriisi1 \leq Matriisi2 \Rightarrow \text{Boolean matriisi}$

Antaa totuusarvon tosi, jos *Laus1* määritetään olevan pienempi tai yhtä suuri kuin *Laus2*.

Antaa totuusarvon epätosi, jos *Laus1* määritetään olevan suurempi kuin *Laus2*.

Kaikissa muissa tapauksissa vastauksena on yhtälön sievennetty muoto.

Listojen ja matriisien kohdalla vastauksena on vertailuja elementti elementiltä.

Huomaa: Voit syöttää tämän operaattorin näppäimistöltä kirjoittamalla <=

> (suurempi kuin)

  painikkeet

$Laus1 > Laus2 \Rightarrow \text{Boolean lauseke}$

Katso esimerkki kohdasta “=” (on yhtä kuin).

$Lista1 > Lista2 \Rightarrow \text{Boolean lista}$

$Matriisi1 > Matriisi2 \Rightarrow \text{Boolean matriisi}$


Antaa totuusarvon tosi, jos *Laus1* määritetään olevan suurempi kuin *Laus2*.

Antaa totuusarvon epätosi, jos *Laus1* määritetään olevan pienempi tai yhtä suuri kuin *Laus2*.

Kaikissa muissa tapauksissa vastauksena on yhtälön sievennetty muoto.

Listojen ja matriisien kohdalla vastauksena on vertailuja elementti elementiltä.

≥ (suurempi tai yhtä suuri kuin)

  painikkeet

$Laus1 \geq Laus2 \Rightarrow \text{Boolean lauseke}$

Katso esimerkki kohdasta “=” (on yhtä kuin).

$Lista1 \geq Lista2 \Rightarrow \text{Boolean lista}$

$Matriisi1 \geq Matriisi2 \Rightarrow \text{Boolean matriisi}$

Antaa totuusarvon tosi, jos *Laus1* määritetään olevan suurempi tai yhtä suuri kuin *Laus2*.

Antaa totuusarvon epätosi, jos *Laus1* määritetään olevan pienempi kuin *Laus2*.

Kaikissa muissa tapauksissa vastauksena on yhtälön sievennetty muoto.

Listojen ja matriisien kohdalla vastauksena on vertailuja elementti elementiltä.

Huomaa: Voit syöttää tämän operaattorin näppäimistöltä kirjoittamalla >=

⇒ (looginen seuraus)**ctrl = näppäimet***BooleanLaus1 ⇒ BooleanLaus2* antaa vastauksena *Boolean lausekkeen*

$5 > 3$ or $3 > 5$	true
--------------------	------

BooleanList1 ⇒ BooleanList2 antaa vastauksena *Boolean listan*

$5 > 3 ⇒ 3 > 5$	false
-----------------	-------

BooleanMatriisi1 ⇒ BooleanMatriisi2 antaa vastauksena *Boolean matriisin*

3 or 4	7
--------	---

$3 ⇒ 4$	-4
---------	----

Kokonaisluku1 ⇒ Kokonaisluku2 antaa vastauksena *kokonaisluvun*

$\{1,2,3\}$ or $\{3,2,1\}$	$\{3,2,3\}$
----------------------------	-------------

$\{1,2,3\} ⇒ \{3,2,1\}$	$\{-1,-1,-3\}$
-------------------------	----------------

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 =>

⇔ (looginen kaksoisseuraus, XNOR)**ctrl = näppäimet***BooleanLaus1 ⇔ BooleanLaus2* antaa vastauksena *Boolean lausekkeen*

$5 > 3$ xor $3 > 5$	true
---------------------	------

BooleanList1 ⇔ BooleanList2 antaa vastauksena *Boolean listan*

$5 > 3 ⇔ 3 > 5$	false
-----------------	-------

BooleanMatriisi1 ⇔ BooleanMatriisi2 antaa vastauksena *Boolean matriisin*

3 xor 4	7
---------	---

$3 ⇔ 4$	-8
---------	----

Kokonaisluku1 ⇔ Kokonaisluku2 antaa vastauksena *kokonaisluvun*

$\{1,2,3\}$ xor $\{3,2,1\}$	$\{2,0,2\}$
-----------------------------	-------------

$\{1,2,3\} ⇔ \{3,2,1\}$	$\{-3,-1,-3\}$
-------------------------	----------------

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 \Leftrightarrow

! (kertoma)

?!▶ painike

Lista! ⇒ lista

5! 120

Matriisi! ⇒ matriisi

{5,4,3}! {120,24,6}

Määrittää argumentin kertoman.

$\begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}!$ $\begin{bmatrix} 1 & 2 \\ 6 & 24 \end{bmatrix}$

Jos kyseessä on lista tai matriisi, määrittää listan tai matriisin elementtien kertomista.

& (liitä)

ctrl 📖 painikkeet

Merkkijono1 & Merkkijono2 ⇒ *merkkijono*

"Hello "&"Nick" "Hello Nick"

Antaa vastauksena tekstimerkkijonon, joka on *Merkkijono2* liitettynä *Merkkijono1*:een.

d() (derivaatta)

Katalogi > 📖

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

√() (neliöjuuri)

ctrl x² painikkeet

√(*Lista!*)⇒*lista*

Laskee argumentin neliöjuuren.

Kun kyseessä on lista, laskee kaikkien *Lista!*:n elementtien neliöjuuret.

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

Huomaa: Katso myös **Neliöjuurimalli**, sivu 1.

$\Pi()$ (tulo)

Katalogi > 

$\Pi(Laus\ I, Muutt, Matala, Korkea) \Rightarrow lauseke$

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

Sieventää *Laus I*:n jokaisen *Muutt*:n arvon suhteen väliltä *Matala* - *Korkea* ja laskee vastausten tulon.

Huomaa: Katso myös **Kertolaskumalli** (Π), sivu 5.

$\Pi(Laus\ I, Muutt, Matala, Matala-1) \Rightarrow 1$

$\Pi(Laus\ I, Muutt, Matala, Korkea) \Rightarrow 1/\Pi(Laus\ I, Muutt, Korkea+1, Matala-1)$, 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.

$$\prod_{k=4}^3 (k) \quad 1$$

$$\prod_{k=4}^1 \left(\frac{1}{k}\right) \quad 6$$

$$\prod_{k=4}^1 \left(\frac{1}{k}\right) \cdot \prod_{k=2}^4 \left(\frac{1}{k}\right) \quad \frac{1}{4}$$

$\Sigma()$ (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 5.

$\Sigma(\text{Laus1, Muutt, Matala, Matala-1}) \Rightarrow 0$

$$\sum_{k=4}^3 (k) = 0$$

$\Sigma(\text{Laus1, Muutt, Matala, Korkea}) \Rightarrow -\Sigma(\text{Laus1, Muutt, Korkea+1, Matala-1})$, jos $\text{Korkea} < \text{Matala-1}$

$$\sum_{k=4}^1 (k) = -5$$

Esimerkkien yhteenlaskukaavat on otettu seuraavasta viitteestä:

Ronald L. Graham, Donald E. Knuth, and Oren Patashnik. *Concrete Mathematics: A Foundation for Computer Science*. Reading, Massachusetts: Addison-Wesley, 1994.

$$\sum_{k=4}^1 (k) + \sum_{k=2}^4 (k) = 4$$

$\Sigma\text{Int}()$

$\Sigma\text{Int}(\text{NPmt1, NPmt2, N, I, PV, [Pmt], [FV], [PpY], [CpY], [PmtAt], [pyörArvo]}) \Rightarrow \text{arvo}$

$\Sigma\text{Int}(1,3,12,4,75,20000,,,12,12) = -218.11$

$\Sigma\text{Int}(\text{NPmt1, NPmt2, amortTable}) \Rightarrow \text{arvo}$

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 166.

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

pyörArvo määrittää pyöristyksessä käytettävien desimaalien määrän. Oletusarvo=2.

<i>tbt:=amortTbl(12,12,4.75,20000,,,12,12)</i>			
0	0.	0.	20000.
1	-79.17	-1630.69	18369.3
2	-72.71	-1637.15	16732.2
3	-66.23	-1643.63	15088.5
4	-59.73	-1650.13	13438.4
5	-53.19	-1656.67	11781.7
6	-46.64	-1663.22	10118.5
7	-40.05	-1669.81	8448.7
8	-33.44	-1676.42	6772.28
9	-26.81	-1683.05	5089.23
10	-20.14	-1689.72	3399.51
11	-13.46	-1696.4	1703.11
12	-6.74	-1703.12	-0.01

$\Sigma\text{Int}(1,3,tbt) = -218.11$

$\Sigma\text{Int}(NPmt1, NPmt2, amortTable)$ laskee koron summan lyhennystaulukon *amortTable* mukaisesti. *amortTable*-argumentin on oltava matriisi, joka on kohdassa **amortTbl()** kuvatun muotoinen, katso sivu 7.

Huomaa: Katso myös $\Sigma\text{Prn}()$, jäljempänä, sekä **Bal()**, sivu 16.

 $\Sigma\text{Prn}()$

$\Sigma\text{Prn}(NPmt1, NPmt2, N, I, PV, [Pmt], [FV], [PpY], [CpY], [PmtAt], [pyörArvo]) \Rightarrow arvo$

$\Sigma\text{Prn}(1, 3, 12, 4.75, 20000, \dots, 12, 12)$ -4911.47

 ΣPrn

$(NPmt1, NPmt2, amortTable) \Rightarrow arvo$

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 166.

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

pyörArvo määrittää pyöristyksessä käytettävien desimaalien määrän. Oletusarvo=2.

$\Sigma\text{Prn}(NPmt1, NPmt2, amortTable)$ laskee pääoman summan lyhennystaulukon *amortTable* perusteella. *amortTable*-argumentin on oltava matriisi, joka on kohdassa **amortTbl()** kuvatun muotoinen, katso sivu 7.

tbl:=amortTbl(12, 12, 4.75, 20000, \dots, 12, 12)

0	0.	0.	20000.
1	-79.17	-1630.69	18369.3
2	-72.71	-1637.15	16732.2
3	-66.23	-1643.63	15088.5
4	-59.73	-1650.13	13438.4
5	-53.19	-1656.67	11781.7
6	-46.64	-1663.22	10118.5
7	-40.05	-1669.81	8448.7
8	-33.44	-1676.42	6772.28
9	-26.81	-1683.05	5089.23
10	-20.14	-1689.72	3399.51
11	-13.46	-1696.4	1703.11
12	-6.74	-1703.12	-0.01

$\Sigma\text{Prn}(1, 3, tbl)$ -4911.47

Huomaa: Katso myös Σ Int(), edellä, sekä Bal(), sivu 16.

(epäsuora operaattori) **painikkeet**# *muuttNimiMerkkijono*

Luo muuttujan xyz tai viittaa siihen.

Viittaa muuttujaan, jonka nimi on *muuttNimiMerkkijono*. Tällä operaattorilla voit luoda muuttujanimiä funktion sisältä merkkijonojen avulla.

$10 \rightarrow r$	10
"r" $\rightarrow sI$	"r"
#sI	10

Laskee arvon muuttujalle (r), jonka nimi on tallennettu muuttujaan s1.

E (kymmenpotenssimuoto) **painike***mantissaEeksponentti*

23000.	23000.
2300000000.+4.1E15	4.1E15
$3 \cdot 10^4$	30000

Syöttää luvun kymmenpotenssimuodossa. Luku tulkitaan seuraavasti: *mantissa* $\times 10^{\text{eksponentti}}$.

Vinkki: Jos haluat syöttää 10-potenssin ilman, että vastauksena on desimaaliluku, käytä komentoa $10^{\text{kokonaisluku}}$.

Huomaa: Voit syöttää tämän operaattorin tietokoneen näppäimistöltä kirjoittamalla @E. Kirjoita esimerkiksi 2.3@E4, kun haluat syöttää 2.3E4.

g (graadi) **painike***Lista l g \Rightarrow lista*

Aste-, graadi- tai radiaanikulmatilassa.

Matriisi l g \Rightarrow matriisi

Tämän funktion avulla voit määrittää graadikulman ollessasi aste- tai radiaanikulmatilassa.

g (graadi)

1 painike

Kun laskin on radiaanikulmatilassa, kertoo $Laus I$:n arvolla $\pi/200$.

Kun laskin on astekulmatilassa, kertoo $Laus I$:n arvolla $g/100$.

Graadikulmatilassa antaa vastauksena lausekkeen $Laus I$ muuttumattomana.

Huomaa: Voit syöttää tämän symbolin tietokoneen näppäimistöltä kirjoittamalla @g.

r (radiaani)

1 painike

$Lista I^r \Rightarrow lista$

Aste-, graadi- tai radiaanikulmatilassa:

$Matriisi I^r \Rightarrow matriisi$

Tämän funktion avulla voit määrittää radiaanikulman ollessasi aste- tai graadikulmatilassa.

Kun laskin on astekulmatilassa, kertoo argumentin arvolla $180/\pi$.

Radiaanikulmatilassa antaa vastauksena argumentin muuttumattomana.

Kun laskin on graadikulmatilassa, kertoo argumentin arvolla $200/\pi$.

Vinkki: Käytä komentoa r , jos haluat pakottaa funktion määritelmän yksiköksi radiaanit riippumatta tilasta, joka on käytössä funktion käytön aikana.

Huomaa: Voit syöttää tämän symbolin tietokoneen näppäimistöltä kirjoittamalla @r.

° (aste)

1 painike

$Lista I^\circ \Rightarrow lista$

Aste-, graadi- tai radiaanikulmatilassa:

$Matriisi I^\circ \Rightarrow matriisi$

Radiaanikulmatilassa:

Huom: Vastauksen pakottaminen likimääräiseksi:

° (aste)

painike

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.

Kämmenlaite: Paina .

Windows®: Paina **Ctrl+Enter**.

Macintosh®: Paina **⌘+Enter**.

iPad®: Pidä **enter** ja valitse .

$$\cos\left(\left\{0, \frac{\pi}{4}, 90^\circ, 30.12^\circ\right\}\right)$$

$$\{1, 0.707107, 0., 0.864976\}$$

°, ', " (astetta/minuuttia/sekuntia)

painikkeet

$dd^\circ mm' ss.ss'' \Rightarrow$ lauseke

dd Positiivinen tai negatiivinen luku

mm Ei-negatiivinen luku

$ss.ss$ Ei-negatiivinen luku

Laskee $dd+(mm/60)+(ss.ss/3600)$.

Tässä kantaluku-60:n syötemuodossa voit:

- Syöttää kulman asteina/minuutteina/sekunteina nykyisestä kulmatilasta riippumatta.
- Syöttää kellonajan tunteina/minuutteina/sekunteina.

Huomaa: Merkitse sekuntien $ss.ss$ perään kaksi heittomerkkiä ("), ei lainausmerkkiä (').

Astekulmatilassa:

25°13'17.5"	25.2215
25°30'	$\frac{51}{2}$

∠ (kulma)

painikkeet

[Säde, ∠θ_Kulma] ⇒ vektori
(polaarinen syöte)

[Säde, ∠θ_Kulma, Z_Koordinaatti] ⇒ vektori
(lieriömäinen syöte)

Radiaanikulmatilassa ja kun vektorimuoto on asetettu valintaan: suorakulma

sylinteri

∠ (kulma)

ctrl  painikkeet

[*Säde*,∠*θ* *Kulma*,∠*θ* *Kulma*]⇒*vektori*
(pallonmuotoinen syöte)

pallo

Laskee koordinaatit vektorina riippuen vektorimuotoilan asetuksesta: suorakulma, sylinteri tai pallo.

Huomaa: Voit syöttää tämän symbolin tietokoneen näppäimistöä kirjoittamalla @<.

(*Magnitudi* ∠ *Kulma*)⇒*kompleksiArvo*
(polaarinen syöte)

Radiaanikulmatilassa ja suorakulmakompleksimuodossa:

Syöttää kompleksilukuarvon ($r\angle\theta$) polaarisisä muodossa. *Kulma* tulkitaan nykyisen kulmatila-asetuksen mukaisesti.

_ (alaviiva edustaa tyhjää elementtiä)

KatsoTyhjät elementit, sivu 215.

10^()

Katalogi > 

10^ (*Listal*)⇒*lista*

Laskee luvun 10 korotettuna argumentin potenssiin.

Jos kyseessä on lista, laskee luvun 10 korotettuna *Listal*:n elementtien potenssiin.

10^(*neliömatrissi*)⇒*neliömatrissi*

Laskee luvun 10 korotettuna *neliömatrissi*l:n potenssiin. Tämä ei ole sama kuin kunkin elementin 10-potenssiin korottamisen laskeminen. Laskentamenetelmä on kuvattu kohdassa **cos()**.

$$10^{\begin{bmatrix} 1 & 5 & 3 \\ 4 & 2 & 1 \\ 6 & -2 & 1 \end{bmatrix}}$$
$$\begin{bmatrix} 1.14336\text{E}7 & 8.17155\text{E}6 & 6.67589\text{E}6 \\ 9.95651\text{E}6 & 7.11587\text{E}6 & 5.81342\text{E}6 \\ 7.65298\text{E}6 & 5.46952\text{E}6 & 4.46845\text{E}6 \end{bmatrix}$$

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

Listal $\wedge^{-1} \Rightarrow$ *lista*

Laskee argumentin käänteisluvun.



Jos kyseessä on lista, laskee *Listal*:n elementtien käänteisluvut.

neliömatriisi1 $\wedge^{-1} \Rightarrow$ *neliömatriisi*

Laskee *neliömatriisi1*:n käänteisluvun.

neliömatriisi1:n oltava ei-singulaarinen neliömatriisi.

| (rajoittava operaattori)

  -näppäimet

Laus | *Booleanlaus1* [**and***Booleanlaus2*]...

Laus | *Booleanlaus1* [**or***Booleanlaus2*]...

Rajoittava (“|”)-symboli toimii binaarisena operaattorina. Operaattorin | vasemmalla puolella oleva operandi on lauseke. Operaattorin | oikealla puolella oleva operandi määrittää yhden tai useampia suhteita, joiden tarkoitus on vaikuttaa lausekkeen sieventämiseen. Operaattorin | jäljessä olevat useat suhteet on yhdistettävä loogisilla operaattoreilla “**and**” tai “**or**”.


Rajoittava operaattori tarjoaa kolme perustoimintoa:

- Sijoitukset
- Välin rajoitusehdot
- Pois rajaaminen

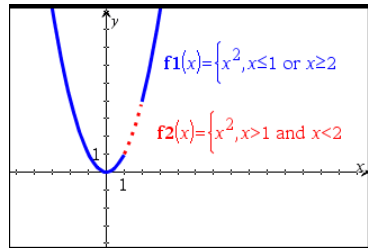
Sijoitukset ovat yhtälön muodossa, kuten $x=3$ tai $y=\sin(x)$. Tehokkainta on, kun vasen puoli on yksinkertainen muuttuja.

Laus | *Muuttuja* = *arvo* korvaa *arvon* jokaisessa *Muuttujan* esiintymiskohdassa lausekkeessa *Laus*.

| (rajoittava operaattori)

ctrl  -näppäimet

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.



Pois sulkemisessa käytetään suhteoperaattoria “ei ole yhtä kuin” (\neq tai \neq), jonka avulla jokin tietty arvo suljetaan pois.


$$\text{solve}(x^2 - 1 = 0, x) | x \neq 1 \quad x = -1$$

→ (tallenna)

ctrl var painike

Huomaa: Voit syöttää tämän operaattorin näppäimistöltä kirjoittamalla = : pikavalintana. Kirjoita esimerkiksi **pi/4 =: myvar.**

:= (määritä)

ctrl  painikkeet

Muutt := Lista

Muutt := Matriisi

Funktio(Param1,...) := Laus

*Funktio(Param1,...) := Lista Funktio
(Param1,...) := Matriisi*

© [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

 $result:=result+i^2$

EndFor

 Return $result$

EndFunc

 Done

 $g(3)$

 14
Ob, Oh**Ob** binaariluku**Oh** heksadesimaaliluku

Määrittää binaari- (Ob) tai heksadesimaaliluvun (Oh). Syöttääksesi binaari- tai heksadesimaaliluvun sinun on syötettävä etuliite Ob tai Oh riippumatta kantalukutilasta. Ilman etuliitettä lukua käsitellään desimaalilukuna (kantaluksi 10).

Tulokset näytetään kantalukutilan mukaisesti.

  painikkeet,   painikkeet

Desimaalisessa kantalukutilassa:

 $0b10+0hF+10$

 27

Binaarisessa kantalukutilassa:

 $0b10+0hF+10$

 0b11011

Heksadesimaalisessa kantalukutilassa:

 $0b10+0hF+10$

 0h1B

TI-Nspire™ CX II – Piirtokomennot

Tämä on täydentävä asiakirja TI-Nspire™-viiteoppaalle ja TI-Nspire™ CAS -viiteoppaalle. Kaikki TI-Nspire™ CX II -komennot sisällytetään ja julkaistaan TI-Nspire™-viiteoppaassa ja TI-Nspire™ CAS -viiteoppaassa.

Kuvaajien ohjelmointi

Uusia komentoja on lisätty TI-Nspire™ CX II -kämmenlaitteisiin ja TI-Nspire™-työpöytäsovellukseen kuvaajaohjelmointia varten.

TI-Nspire™ CX II -kämmenlaitteet siirtyvät tähän kuvaajatilaan, kun ne suorittavat kuvaajakomentoja, ja siirtyvät ohjelman suorittamisen jälkeen takaisin kontekstiin, jossa ohjelma suoritettiin.

Näytön yläreunassa näkyy ”Running...”, kun ohjelmaa suoritetaan. Näkyviin tulee ”Finished”, kun ohjelma on valmis. Minkä tahansa näppäimen painaminen siirtää järjestelmän pois kuvaajatilasta.

- Siirtyminen kuvaajatilaan käynnistyy automaattisesti, kun jokin Draw (kuvaajat) -komennoista kohdataan TI-Basic -ohjelman suorituksen aikana.
- Tämä siirtyminen tapahtuu vain, kun ohjelma suoritetaan laskimella; asiakirjassa tai muistiolehtiön laskimessa.
- Siirtyminen pois kuvaajatilasta tapahtuu ohjelman sulkemisen jälkeen.
- Kuvaajatila on käytettävissä vain TI-Nspire™ CX II- kämmenlaitteissa ja TI-Nspire™ CX II -kämmenlaitteiden työpöytäversion näkymässä. Tämä tarkoittaa, että se ei ole käytettävissä tietokoneen asiakirjanäkymässä työpöytäversiossa tai iOS-käyttöjärjestelmässä.
 - Jos kuvaajakomento kohdataan, kun TI-Basic -ohjelmaa suoritetaan virheellisestä kontekstista, näkyviin tulee virheviesti ja TI-Basic -ohjelma suljetaan.

Kuvaajanäyttö

Kuvaajanäytön yläosassa on otsikko, jota ei voi kirjoittaa kuvaajakomennoilla.

Kuvaajanäytön piirustusalue tyhjennetään (color = 255,255,255), kun kuvaajanäyttö alustetaan.

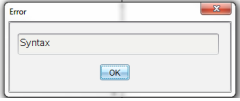
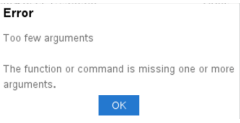
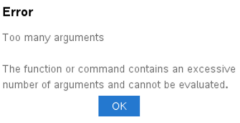
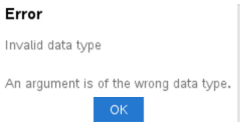
Kuvaajanäyttö	Oletus
Korkeus	212
Leveys	318
Väri	valkoinen: 255,255,255

Oletusnäkyvä ja asetukset

- Yläpalkin tilakuvakkeet (akun tila, press-to-start-tila, verkon ilmainen jne.) eivät näy, kun kuvaajaohjelma on käynnissä.
- Piirtämisen oletusväri: Musta (0,0,0)
- Kynän oletustyyli – normaali, tasainen
 - Paksuus: 1 (tasainen), 2 (normaali), 3 (paksuin)
 - Tyyli 1 (tasainen), 2 (pisteet), 3 (viivat)
- Kaikki piirtokomennot käyttävät nykyisiä väri- ja kynäasetuksia eli joko oletusarvoja tai TI-Basic-komentojen avulla asetettuja arvoja.
- Tekstin fontti on pysyvä eikä sitä voi muuttaa.
- Kaikki kuvaajanäytön piirrokset piirretään leikkausikkunassa, joka on kuvaajanäytön piirustusalueen kokoinen. Mitään piirroksia, jotka ulottuvat leikatun kuvaajanäytön piirustusalueen ulkopuolelle, ei piirretä. Virheviestiä ei näytetä.
- Kaikki piirtokomennoille määritetyt x- ja y-koordinaatit on määritetty siten, että 0,0 on kuvaajanäytön piirustusalueen vasemmassa yläkulmassa.
 - **Poikkeukset:**
 - **DrawText** käyttää koordinaatteja tekstin rajaavan ruudun vasemmassa alakulmassa.
 - **SetWindow** käyttää näytön vasenta alakulmaa.
- Kaikki komentojen parametrit voidaan antaa lausekkeina, jotka arvioidaan numeroksi, joka pyöristetään sitten lähimpään kokonaislukuun.

Kuvaajanäytön virheviestit

Jos validointi epäonnistuu, näyttöön ilmestyy virheviesti.

Virheviestit	Kuvaus	Tarkastele
Virhe Syntaksi	Jos syntaksin tarkastuksessa löytyy syntaksivirheitä, näkyviin tulee virheilmoitus, ja editori yrittää asettaa kohdistimen ensimmäisen virheen kohdalle, jotta voit korjata sen.	
Virhe Liian vähän argumentteja	Funktiosta tai komennosta puuttuu yksi tai useampia argumentteja	
Virhe Liian monta argumenttia	Funktio tai komento sisältää liian monta argumenttia eikä sitä voida arvioida.	
Virhe Virheellinen tietotyyppi	Argumentin tietotyyppi on väärä.	

Virheelliset komennot kuvaajatilassa

Jotkin komennot eivät ole sallittuja, kun ohjelma siirtyy kuvaajatilaan. Jos nämä komennot kohdataan kuvaajatilassa, näyttöön tulee virheviesti ja ohjelma suljetaan.

Komentoa ei sallita	Virheviestit
Pyyntö	Pyyntöä ei voida toteuttaa kuvaajatilassa
PyydäMerkkij	RequestStr-toimintoa ei voida toteuttaa kuvaajatilassa
Teksti	Tekstiä ei voida toteuttaa kuvaajatilassa

Komennot, jotka kirjoittavat tekstin laskimeen – **disp** ja **dispAt** – ovat tuettuja komentoja kuvaajakontekstissa. Näiden komentojen teksti lähetetään Laskin-näyttöön (ei Kuvaaja-näyttöön) ja se näkyy sen jälkeen, kun ohjelma on suljettu ja järjestelmä siirtyy takaisin Laskin-sovellukseen.

Tyhjennä

Clear $x, y, width, height$

Tyhjentää koko näytön, jos parametreja ei ole.

Jos $x, y, width$ ja $height$ on määritetty, parametrien määrittämä suorakulmio tyhjennetään.


Tyhjennä

Tyhjentää koko näytön

Clear 10,10,100,50

Tyhjentää suorakulmioalueen, jonka vasen yläkulma on (10, 10), leveys on 100 ja korkeus on 50.

DrawArc

Katalogi > 
CXII**DrawArc** *x, y, width, height, startAngle, arcAngle*

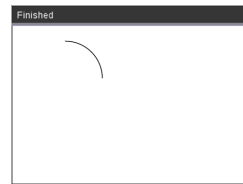
Piirrä kaari rajaavan suorakulmion sisällä annetulla aloitus- ja kaarikulmalla.

x, y: rajaavan suorakulmion vasen yläkoordinaatti*width, height*: rajaavan suorakulmion mitat

"Arc Angle" määrittää kaaren kaarevuuden.

Nämä parametrit voidaan antaa lausekkeina, jotka arvioidaan numeroksi, joka pyöristetään sitten lähimpään kokonaislukuun.

DrawArc 20,20,100,100,0,90



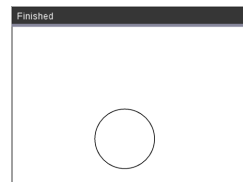
DrawArc 50,50,100,100,0,180

Katso myös: [FillArc](#)

DrawCircle

Katalogi > 
CXII**DrawCircle** *x, y, radius**x, y*: keskikohdan koordinaatti*radius*: ympyrän säde

DrawCircle 150,150,40

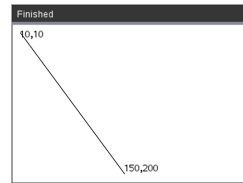
Katso myös: [FillCircle](#)

DrawLine $x1, y1, x2, y2$ Piirrä viiva pisteillä $x1, y1, x2, y2$.

Lausekkeet, jotka arvioidaan numeroksi, joka pyöristetään sitten lähimpään kokonaislukuun.

Näytön rajat: Jos määritetyt koordinaatit aiheuttavat viivan osan piirtämisen kuvaajanäytön ulkopuolelle, kyseinen viivan osa leikataan eikä virheilmoitusta näytetä.

DrawLine 10,10,150,200

**DrawPoly**

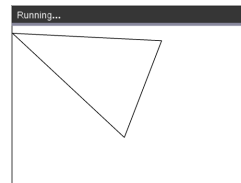
Komennoilla on kaksi varianttia:

DrawPoly $xlist, ylist$

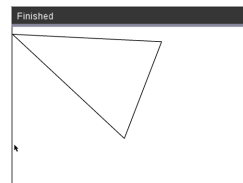
tai

DrawPoly $x1, y1, x2, y2, x3, y3...xn, yn$ **Huomaa:** DrawPoly $xlist, ylist$
Muoto yhdistää kohteet $x1, y1$ to $x2, y2, x2, y2$ to $x3, y3$ ja niin edelleen.**Huomaa:** DrawPoly $x1, y1, x2, y2, x3, y3...xn, yn$
Kohteita xn, yn EI yhdistetä automaattisesti kohteisiin $x1, y1$.Lausekkeet, jotka arvioidaan todellisten liukulukujen luetteloon $xlist, ylist$ Lausekkeet, jotka arvioidaan yksittäiseksi liukuluvuksi $x1, y1...xn, yn$ = monikulmion kärkien koordinaatit

```
xlist:={0,200,150,0}
ylist:={10,20,150,10}
DrawPoly xlist,ylist
```



DrawPoly 0,10,200,20,150,150,0,10



Huomaa: DrawPoly: Lisää kokomitat (leveys/korkeus), jotka vastaavat piirrettyjä viivoja.

Viivat piirretään rajaavan ruudun määrittelykoordinaatin ja mittojen ympärille siten, että piirretyn monikulmion todellinen koko on suurempi kuin leveys ja korkeus.

Katso myös: [FillPoly](#)

DrawRect

DrawRect *x, y, width, height*

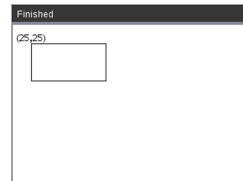
x, y: suorakulmion vasen yläkoordinaatti

width, height: suorakulmion leveys ja korkeus (suorakulmio, joka on piirretty alaspäin ja oikealle lähtökoordinaatista)

Huomaa: Viivat piirretään rajaavan ruudun määrittelykoordinaatin ja mittojen ympärille siten, että piirretyn suorakulmion todellinen koko on suurempi kuin leveys ja korkeus osoittavat.

Katso myös: [FillRect](#)

DrawRect 25,25,100,50



DrawText

DrawText *x, y, exprOrString*
[,exprOrString2]...

x, y: tekstisyötön koordinaatti


Piirtää tekstin kohteeseen *exprOrString* määritetyssä *x, y*-koordinaattien sijainnissa.

Säännöt kohteelle *exprOrString* ovat samat kuin kohteelle **Disp** – **DrawText** voi käsitellä useita argumentteja.

DrawText 50,50,"Hello World"



FillArc

Katalogi > 
CXII

FillArc *x, y, width, height, startAngle, arcAngle*

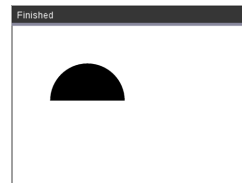
x, y: rajaavan suorakulmion vasen yläkoordinaatti

Piirrä ja täytä kaari rajaavan suorakulmion sisällä annetulla aloitus- ja kaarikulmalla.

Täytön oletusväri on musta. Täyttöväri voidaan asettaa komennolla [SetColor](#).

”Arc Angle” määrittää kaaren kaarevuuden.

FillArc 50,50,100,100,0,180



FillCircle

Katalogi > 
CXII

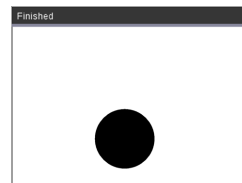
FillCircle *x, y, radius*

x, y: keskikohdan koordinaatti

Piirrä ja täytä ympyrä määritellyllä keskikohdalla ja määritetyllä säteellä.

Täytön oletusväri on musta. Täyttöväri voidaan asettaa komennolla [SetColor](#).

FillCircle 150,150,40



Täällä!

FillPoly

Katalogi > 
CXII

FillPoly *xlist, ylist*

tai

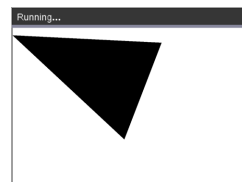
FillPoly *x1, y1, x2, y2, x3, y3...xn, yn*

Huomaa: Viivan ja värin määrittää komennot [SetColor](#) ja [SetPen](#)

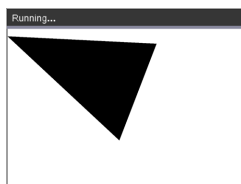
xlist:={0,200,150,0}

ylist:={10,20,150,10}

FillPoly *xlist, ylist*



FillPoly 0,10,200,20,150,150,0,10

**FillRect****FillRect** *x, y, width, height**x, y*: suorakulmion vasen yläkoordinaatti*width, height*: suorakulmion leveys ja korkeus

Piirrä ja täytä suorakulmio, jonka vasen yläkulma koordinaatissa, jonka määrittää (*x,y*).

Täytön oletusväri on musta. Täyttöväri voidaan asettaa komennolla [SetColor](#).

Huomaa: Viivan ja värin määrittää komennot [SetColor](#) ja [SetPen](#)

FillRect 25,25,100,50



G

getPlatform()

Katalogi > 
CXII

getPlatform()

getPlatform()

"dt"

Palauttaa:

"dt" työpöytäohjelmistosovelluksissa

"hh" TI-Nspire™ CX -kämmenlaitteissa

"ios" TI-Nspire™ CX iPad® -sovelluksessa

PaintBuffer

Maalaa kuvaajan puskurin näytölle

Tätä komentoa käytetään yhdessä UseBuffer-komennon kanssa näyttönopeuden lisäämiseksi, kun ohjelma luo useita kuvaajaobjekteja.

UseBuffer

Luvulle $n, 1, 10$

```
x:=randInt(0,300)
```

```
y:=randInt(0,200)
```

```
radius:=randInt(10,50)
```

```
Wait 0,5
```

```
DrawCircle x,y,radius
```

```
EndFor
```

PaintBuffer

Tämä ohjelma näyttää 10 ympyrää samanaikaisesti.

Jos UseBuffer-komento poistetaan, jokainen ympyrä näytetään sellaisena kuin se on piirretty.

Katso myös: [UseBuffer](#)

PlotXY $x, y, shape$

x, y : koordinaatit muodon muodostamiseen

$shape$: numero väliltä 1–13, joka määrittää muodon

- 1 – Täytetty ympyrä
- 2 – Tyhjä ympyrä
- 3 – Täytetty neliö
- 4 – Tyhjä neliö
- 5 – Risti
- 6 – Plusmerkkikuvio
- 7 – Ohut
- 8 – keskikokoinen piste, täytetty
- 9 – keskikokoinen piste, tyhjä
- 10 – suurempi piste, täytetty
- 11 – suurempi piste, tyhjä
- 12 – suurin piste, täytetty
- 13 – suurin piste, tyhjä

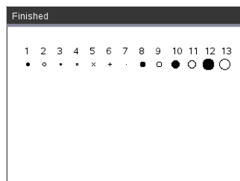
PlotXY 100,100,1

Luvulle $n, 1, 13$

DrawText 1+22*n,40,n

PlotXY 5+22*n,50,n

EndFor



SetColorKatalogi > 
CXII**SetColor**

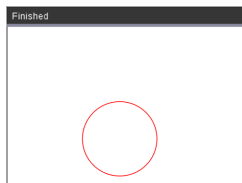
Red-arvo, Green-arvo, Blue-arvo

Kelvolliset arvot punaiselle, vihreälle ja siniselle ovat välillä 0–255

Asettaa tulevien Draw-komentojen väriin

SetColor 255,0,0

DrawCircle 150,150,100

**SetPen**Katalogi > 
CXII**SetPen**

paksuus, tyyli

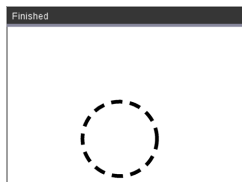
Paksuus: $1 \leq \text{paksuus} \leq 3$ | 1 on ohuin, 3 on paksuin

style: 1 = tasainen, 2 = pisteet, 3 = viivat

Asettaa tulevien Draw-komentojen kynän tyylin

SetPen 3,3

DrawCircle 150,150,50

**SetWindow**Katalogi > 
CXII**SetWindow**

xMin, xMax, yMin, yMax

Luo loogisen ikkunan, joka kartoittaa kuvaajan piirustusalueen. Kaikki parametrit vaaditaan.

Jos piirretyn objektin osa on ikkunan ulkopuolella, syöttö leikataan (ei näytetä) eikä virheilmoitusta näytetä.

SetWindow 0,160,0,120

Asettaa syöttöikkunan vasemman alakulman kohtaan 0,0. Leveys on 160 ja korkeus on 120.

DrawLine 0,0,100,100

SetWindow 0,160,0,120

SetPen 3,3

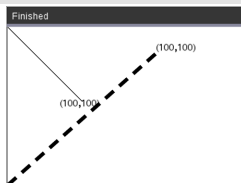
DrawLine 0,0,100,100

Jo x_{\min} on suurempi tai yhtä suuri kuin x_{\max} tai y_{\min} on suurempi tai yhtä suuri kuin y_{\max} , näyttöön tulee virheviesti.

Mitään objekteja, jotka on piirretty ennen SetWindow-komentoa, ei piirretä uudelleen uudessa kokoonpanossa.

Käytä seuraavia ikkunan parametrien nollaamiseksi oletusarvoihin:

SetWindow 0,0,0,0



UseBuffer

Piirrä kuvaaja puskuriin näytön sijaan (suorituskyvyn parantamiseksi)

Tätä komentoa käytetään yhdessä PaintBuffer-komennon kanssa näyttönopeuden lisäämiseksi, kun ohjelma luo useita kuvaajaobjekteja.

UseBuffer-komennolla kaikki kuvaajat näytetään vasta seuraavan PaintBuffer-komennon suorittamisen jälkeen.

UseBufferia täytyy käyttää vain kerran ohjelmassa, eli jokainen käytetty PaintBuffer-komento ei tarvitse vastaavaa UseBuffer-komentoa.

Katso myös: [PaintBuffer](#)

UseBuffer

```
Luvulle n,1,10
x:=randInt(0,300)
y:=randInt(0,200)
radius:=randInt(10,50)
Wait 0,5
DrawCircle x,y,radius
EndFor
PaintBuffer
```

Tämä ohjelma näyttää 10 ympyrää samanaikaisesti.



Jos UseBuffer-komento poistetaan, jokainen ympyrä näytetään sellaisena kuin se on piirretty.

Tyhjät elementit

Reaalimaailman dataa analysoitaessa käytössäsi ei aina välttämättä ole täydellistä datasarjaa. TI-Nspire™ sallii käyttää tyhjiä dataelementtejä, jolloin voit jatkaa toimintaa lähes täydellisellä datalla tarvitsematta aloittaa alusta tai hylätä epätäydellisiä tapauksia.

Esimerkki tyhjiä elementtejä sisältävästä datasta löytyy luvusta Listat & Taulukot, kohdasta Kuvaajien piirtäminen taulukko-datasta.

Funktion **delVoid()** avulla voit poistaa tyhjät elementit listasta. Funktion **isVoid()** avulla voit testata tyhjää elementtiä. Lisätietoja, katso **delVoid()**, sivu 40, ja **isVoid()**, sivu 78.

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>gcd(100,_)</code>	–
<code>3+_</code>	–
<code>{5,_,10}–{3,6,9}</code>	<code>{2,_,1}</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</code> $\begin{pmatrix} 1 & 2 \\ 3 & - \\ 5 & 6 \end{pmatrix}$	$\begin{bmatrix} 1 & 2 \\ 4 & - \\ 9 & 8 \end{bmatrix}$

Tyhjiä elementtejä sisältävät lista-argumentit

SortA ja **SortD** siirtävät kaikki ensimmäisen argumentin sisältämät tyhjät elementit viimeiseksi.

$\{5,4,3,_,1\} \rightarrow list1$	$\{5,4,3,_,1\}$
$\{5,4,3,2,1\} \rightarrow list2$	$\{5,4,3,2,1\}$
SortA <i>list1,list2</i>	Done
<i>list1</i>	$\{1,3,4,5,_\}$
<i>list2</i>	$\{1,3,4,5,2\}$

$\{1,2,3,_,5\} \rightarrow list1$	$\{1,2,3,_,5\}$
$\{1,2,3,4,5\} \rightarrow list2$	$\{1,2,3,4,5\}$
SortD <i>list1,list2</i>	Done
<i>list1</i>	$\{5,3,2,1,_\}$
<i>list2</i>	$\{5,3,2,1,4\}$

Regressioissa X- tai Y-listassa oleva tyhjä elementti aiheuttaa tyhjän elementin jäännöksen vastaavalle elementille.

$ll:=\{1,2,3,4,5\}; l2:=\{2,_,3,5,6,6\}$	$\{2,_,3,5,6,6\}$
LinRegMx <i>ll,l2</i>	Done
<i>stat.Resid</i>	$\{0.434286,_, -0.862857, -0.011429, 0.44\}$
<i>stat.XReg</i>	$\{1,_,3,4,5\}$
<i>stat.YReg</i>	$\{2,_,3,5,6,6\}$
<i>stat.FreqReg</i>	$\{1,_,1,1,1,1\}$

Poisjätetty luokka regressioissa aiheuttaa tyhjän elementin jäännöksen vastaavalle elementille.

$ll:=\{1,3,4,5\}; l2:=\{2,3,5,6,6\}$	$\{2,3,5,6,6\}$
$cat:=\{"M", "M", "F", "F"\}; incl:=\{"F"\}$	$\{"F"\}$
LinRegMx <i>ll,l2,1,cat,incl</i>	Done
<i>stat.Resid</i>	$\{_,_,0,0,0\}$
<i>stat.XReg</i>	$\{_,_,4,5\}$
<i>stat.YReg</i>	$\{_,_,5,6,6\}$
<i>stat.FreqReg</i>	$\{_,_,1,1,1\}$

Frekvenssi 0 regressioissa aiheuttaa tyhjän elementin jäännöksen vastaavalle elementille.

$ll:=\{1,3,4,5\}; l2:=\{2,3,5,6,6\}$	$\{2,3,5,6,6\}$
LinRegMx <i>ll,l2,\{1,0,1,1\}</i>	Done
<i>stat.Resid</i>	$\{0.069231,_, -0.276923, 0.207692\}$
<i>stat.XReg</i>	$\{1,_,4,5\}$
<i>stat.YReg</i>	$\{2,_,5,6,6\}$
<i>stat.FreqReg</i>	$\{1,_,1,1,1\}$

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>
$\sqrt{()}$	<code>sqrt (...)</code>
$\Sigma()$ (yhteenlaskumalli)	<code>sumSeq (...)</code>
$\Pi()$ (tulon malli)	<code>prodSeq (...)</code>
$\sin^{-1}()$, $\cos^{-1}()$, ...	<code>arcsin (...)</code> , <code>arccos (...)</code> , ...
<code>ΔList()</code>	<code>deltaList (...)</code>

Tietokoneen näppäimistöltä

Syötettävä kohde:	Pikavalinta:
i (imaginaarinen vakio)	<code>@i</code>
e (luonnollisen logaritmin kantaluku e)	<code>@e</code>
E (kymmenpotenssimuoto)	<code>@E</code>
T (transponointi)	<code>@t</code>
r (radiaanit)	<code>@r</code>
$^{\circ}$ (asteet)	<code>@d</code>
g (graadit)	<code>@g</code>
\sphericalangle (kulma)	<code>@<</code>
\blacktriangleright (muunnos)	<code>@></code>
<code>\blacktrianglerightDecimal</code> , <code>\blacktrianglerightapproxFraction</code> <code>()</code> , jne.	<code>@>Decimal</code> , <code>@>approxFraction ()</code> , jne.

EOS-järjestelmän (yhtälökäyttöjärjestelmä) hierarkia

Tässä kappaleessa kuvataan yhtälökäyttöjärjestelmä (EOS™), jota sovelletaan matematiikan ja luonnontieteiden TI-Nspire™ -oppimisteknologiassa. Luvut, muuttujat ja funktiot syötetään yksinkertaisena ja suoraviivaisena sekvenssinä. EOS™-ohjelmisto sieventää lausekkeet ja yhtälöt käyttäen sulkuryhmyksiä sekä alla kuvattuja prioriteetteja.

Laskemisjärjestys

Taso Operaattori

- 1 Kaarisulkeet (), hakasulkeet [], aaltosulkeet { }
 - 2 Epäsuora operaattori (#)
 - 3 Funktioiden komennot
 - 4 Jälkioperaattorit: astetta-minuuttia-sekuntia (°, ', "), kertoma (!), prosentti (%), radiaani (x^r), alaindeksi ([]), transponointi (T)
 - 5 Potenssiin korotus, potenssioperaattori (^)
 - 6 Negaatio (-)
 - 7 Merkkijonojen ketjutus (&)
 - 8 Kertolasku (•), jakolasku (/)
 - 9 Yhteenlasku (+), vähennyslasku (-)
 - 10 Yhtäläisyysuhteet: on yhtä kuin (=), ei ole yhtä kuin (≠ tai /=), pienempi kuin (<), pienempi tai yhtä suuri kuin (≤ tai <=), suurempi tai yhtä suuri kuin (>), suurempi tai yhtä suuri kuin (≥ tai >=)
 - 11 Looginen **not**
 - 12 Looginen **and**
 - 13 Looginen **or**
 - 14 **xor, nor, ei-ja**
 - 15 Looginen seuraus (⇒)
 - 16 Looginen kaksoisseuraus, XNOR (⇔)
 - 17 Rajoittava operaattori ("|")
 - 18 Tallenna (→)
-

Kaarisulkeet, hakasulkeet ja aaltosulkeet

Kaikki kaarisulkeiden, hakasulkeiden tai aaltosulkeiden sisällä olevat laskutoimitukset sievennetään ensin. Esimerkiksi lausekkeesta 4(1+2) EOS sieventää ensin kaarisulkeiden sisällä olevan osan, 1+2, ja kertoo tuloksen, 3, sen jälkeen luvulla 4.

Kaikki kaarisulkeiden, hakasulkeiden tai aaltosulkeiden sisällä olevat laskutoimitukset sievennetään ensin. Esimerkiksi lausekkeesta 4(1+2) EOS™-ohjelmisto sieventää ensin kaarisulkeiden sisällä olevan osan, 1+2, ja kertoo sen jälkeen tuloksen, 3, luvulla 4.

Huomaa: Koska käyttäjä voi määrittää omia funktioita TI-Nspire™ -ohjelmistossa, ohjelmisto tulkitsee muuttujan nimen, jonka perässä on sulkeissa oleva lauseke, funktion komennoiksi eikä implisiittiseksi kertolaskuksi. Esimerkiksi $a(b+c)$ on funktio a , jota sievennetään lausekkeella $b+c$. Kun haluat kertoa lausekkeen $b+c$ muuttujalla a , käytä eksplisiittistä kertolaskua: $a*(b+c)$.

Epäsuora operaattori

Epäsuora operaattori (#) muuttaa merkkijonon muuttujaksi tai funktion nimeksi. Esimerkiksi lausekkeesta #("x"&"y"&"z") luodaan muuttujanimi xyz. Epäsuora operaattori sallii myös muuttujien luomisen ja muokkaamisen ohjelman sisällä. Esimerkiksi, jos $10 \rightarrow r$ ja " r " $\rightarrow s1$, niin $\#s1=10$.

Jälkioperaattorit

Jälkioperaattorit ovat operaattoreita, jotka tulevat suoraan argumentin perään, kuten 5!, 25% tai $60^\circ 15' 45''$. Argumentit, joiden perässä on jälkioperaattori, sievennetään neljännellä prioriteettitasolla. Esimerkiksi lausekkeesta $4^3!$ sievennetään ensimmäiseksi 3! Vastauksesta, 6, tulee luvun 4 eksponentti, josta saadaan 4096.

Eksponenttilausekkeet

Eksponenttilausekkeet (^) ja potenssiin korotus elementti elementiltä (.^) sievennetään oikealta vasemmalle. Esimerkiksi lauseke 2^3^2 sievennetään samalla tavalla kuin $2^{\wedge}(3^2)$, josta saadaan 512. Tämä on eri asia kuin $(2^{\wedge}3)^2$, josta saadaan 64.

Negaatio

Negatiivinen luku syötetään painamalla ensin näppäintä $\boxed{-}$ ja syöttämällä luku sen perään. Jälkioperaattorit ja eksponenttilausekkeet suoritetaan ennen negatiota. Esimerkiksi lausekkeen $-x^2$ tulos on negatiivinen luku, ja $-9^2 = -81$. Käytä sulkeita, kun haluat määrittää neliön negatiivisesta luvusta, kuten $(-9)^2$, josta saadaan 81.

Rajoittavan operaattorin ("|") jälkeen seuraava argumentti tuottaa sarjan ehtoja, jotka vaikuttavat operaattoria edeltävän argumentin laskemiseen.

TI-Nspire CX II – TI-Basic-ohjelmointiominaisuudet

Automaattinen sisennys Ohjelmointieditorissa

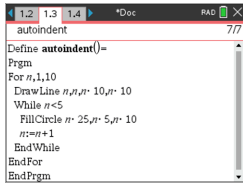
TI-Nspire™-ohjelmaeditori sientää nyt automaattisesti lauseita lohkokomennon sisällä.

Lohkokomennot ovat If/EndIf, For/EndFor, While/EndWhile, Loop/EndLoop, Try/EndTry

Editori lisää automaattisesti välilyönnit ohjelmakäskyihin lohkokomennon sisällä.

Lohkon sulkemiskomento kohdistetaan avauskomennon kanssa.

Alla oleva esimerkki näyttää automaattisen sisennyksen sisäkkäisissä lohkokomennossa.



```
autoindent 77
Define autoindent()=
Prgm
For n,1,10
DrawLine n,n,n-10,n-10
While n<5
FillCircle n-25,n-5,n-10
n:=n+1
EndWhile
EndFor
EndPrgm
```

Kopioidut ja liitetyt koodifragmentit säilyttävät alkuperäisen sisennyksen.

Ohjelmiston aikaisemmassa versiossa luodun ohjelman avaaminen säilyttää alkuperäisen sisennyksen.

TI-Basicin virheviestien parannukset

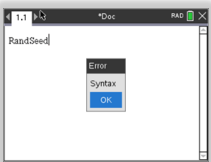
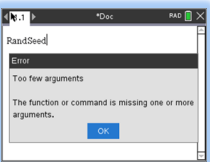
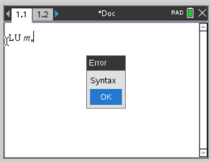
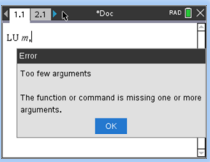
Virheet

Virhetila	Uusi viesti
Ehtokohteen virhe (If/While)	Ehdollista lausuntoa ei voitu ratkaista arvolla TRUE tai FALSE HUOMAA: Koska muutos on sijoitettu riville virheen kanssa, ei ole enää tarpeen tarkentaa, onko virhe "If"- vai "While"-kohde.
Puuttuva EndIf	Odotettiin kohdetta EndIf , mutta löydettiin toinen End-kohde
Puuttuva EndFor	Odotettiin kohdetta EndFor , mutta löydettiin toinen End-kohde
Puuttuva EndWhile	Odotettiin kohdetta EndWhile , mutta löydettiin toinen End-kohde
Puuttuva EndLoop	Odotettiin kohdetta EndLoop , mutta löydettiin toinen End-kohde
Puuttuva EndTry	Odotettiin kohdetta EndTry , mutta löydettiin

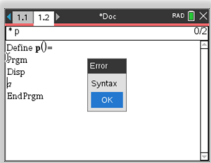
Virhetila	Uusi viesti
	toinen End-kohde
"Then" jätetty pois kohteen If <condition> jälkeen	Puuttuva If..Then
"Then" jätetty pois kohteen Elseif <condition> jälkeen	Then puuttuu kohdasta: Elseif.
Kun "Then", "Else", ja "Elseif" kohdattiin ohjauskohtien ulkopuolella	Then virheellinen kohtien ulkopuolella: If..Then..EndIf tai Try..EndTry
"Elseif" esiintyy kohdan "If..Then..EndIf" ulkopuolella	Elseif virheellinen kohdan ulkopuolella: If..Then..EndIf
"Then" esiintyy kohdan "If...EndIf" ulkopuolella	Then virheellinen kohdan ulkopuolella: If..EndIf

Syntaksivirheet

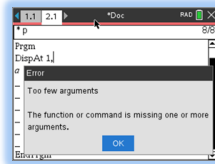
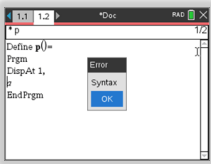
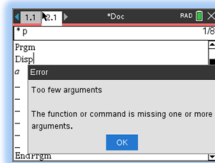
Jos tapauskomentoja, jotka odottavat yhtä tai useampaa argumenttia, kutsutaan epätäydellisellä argumenttiluettelolla, "Too few argument error" -virhe näytetään "syntax" -virheen sijaan.

Nykyinen käyttäytyminen	Uusi CX II -käyttäytyminen
	
	

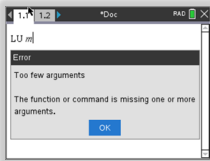
Nykyinen käyttäytyminen



Uusi CX II -käyttäytyminen



Huomaa: Kun epätäydellisen argumenttiluettelon jäljessä ei ole pilkkua, virheviesti on: "too few arguments". Tämä on sama kuin aiemmissa julkaisuissa.



Vakiot ja arvot

Seuraavassa taulukossa on lueteltu vakiot ja niiden arvot, jotka ovat käytettävissä suoritettaessa yksikköjen muunnoksia. Ne voidaan kirjoittaa manuaalisesti tai valita

Vakiot-luettelosta **Utilities> Yksikkömuunnokset** (Kannettava laite: Paina  3).

Vakio	Nimi	Arvo
_c	valon nopeus	299792458 _m/_s
_Cc	Coulombin vakio	8987551792.261 _m/_F
_Fc	Faradayn vakio	96485.33212 _coul/_mol
_g	Painovoiman kiihtyvyyys	9.80665 _m/_s ²
_Gc	Gravitaatiovakio	6.6743E-11 _m ³ /_kg/_s ²
_h	Planckin vakio	6.62607015E-34 _J _s
_k	Boltzmannin vakio	1.380649E-23 _J/_°K
_μ0	Tyhjiön permeabiliteetti	1.25663706212E-6 _N/_A ²
_μb	Bohr magnetoni	9.274009994E-24 _J _m ² /_Wb
_Me	Elektronin lepomassa	9.1093837015E-31 _kg
_Mμ	Myonin massa	1.883531627E-28 _kg
_Mn	Neutronin lepomassa	1.67492749804E-27 _kg
_Mp	Protonin lepomassa	1.67262192369E-27 _kg
_Na	Avogadron vakio	6.02214076E23 /_mol
_q	Elektronin varaus	1.602176634E-19 _coul
_Rb	Bohrin säde	5.29177210903E-11 _m
_Rc	Moolinen kaasuvakio	8.314462618 _J/_mol/_°K
_Rdb	Rydbergin vakio	10973731.568160/_m
_Re	Elektronin säde	2.8179403262E-15 _m
_u	Atomimassa	1.6605390666E-27 _kg
_Vm	moolinen tilavuus	2.241396954E-2 _m ³ /_mol
_ε0	Tyhjiön permittiivisyys (vakio)	8.8541878128E-12 _F/_m
_σ	Stefan-Boltzmannin vakio	5.670367E-8 _W/_m ² /_°K ⁴
_φ0	Magneettivuon kvantti	2.067833831E-15 _Wb

Virhekoodit ja viestit

Kun tapahtuu virhe, virhekoodi määritetään muuttujaan *errCode*. Käyttäjän määrittämällä ohjelmilla ja funktioilla voidaan tutkia *errCode*-muuttujaa virheen syyä määrittämiseksi. Esimerkki muuttujan *errCode* käytöstä on esitetty Esimerkissä 2 **Try**-komennon kohdalla, sivu 162.

Huomaa: Jotkin virhetilanteet koskevat vain *TI-Nspire™ CAS -tuotteita ja toiset taas vain TI-Nspire™-tuotteita*.

Virhekoodi Kuvaus

10 Funktio ei laskenut arvoa.

20 Kokeen tulos ei ollut TOSI eikä EPÄTOSI.

Määrittämättömiä muuttujia ei yleensä voi verrata. Esimerkiksi testi $if\ a < b$ aiheuttaa virheen, jos joko a tai b on määrittämätön suoritettaessa if -lauseketta.

30 Argumentti ei voi olla kansion nimi.

40 Argumenttinvirhe

50 Argumentin yhteensopivuusvirhe

Kahden tai useamman argumentin on oltava samaa tyyppiä.

60 Argumentin on oltava Boolean lauseke tai kokonaisluku.

70 Argumentin on oltava desimaaliluku.

90 Argumentin on oltava lista.

100 Argumentin on oltava matriisi.

130 Argumentin on oltava merkkijono.

140 Argumentin on oltava muuttujan nimi.

Tarkista, että nimi:

- ei ala numerolla
- ei sisällä välilyöntejä tai erikoismerkkejä
- ei käytä alaviivaa tai pistettä väärällä tavalla
- ei ylitä pituusrajoituksia

Lisätietoja löytyy ohjekirjan Laskin-sovellusta käsittelevästä osasta.

160 Argumentin on oltava lauseke.



165 Paristovirta ei riitä lähettämiseen tai vastaanottamiseen.

Asenna uudet paristot ennen lähettämistä tai vastaanottoa.

170 Raja

Alarajan on oltava ylärajaa pienempi hakuväliä määritettäessä.

180 Keskeytys

Näppäintä  tai  on painettu pitkän laskutoimituksen tai ohjelman suorituksen aikana.

Virhekoodi	Kuvaus
190	Kehämäärittys Tämä viesti tulee näkyviin muistin loppumisen välttämiseksi muuttujien arvojen äärettömän korvauksen aikana sievennettäessä lausekkeita. Virheen aiheuttaa esimerkiksi lauseke $a+1->a$, jossa a on määrittämätön muuttuja.
200	Määrittelyalueen lauseke ei kelpaa. Tämän virheilmoituksen voi aiheuttaa esimerkiksi lauseke $\text{solve}(3x^2-4=0,x) \mid x<0$ tai $x>5$, koska ehto on erotettu operaattorilla "or" eikä operaattorilla "and".
210	Datatyyppi ei kelpaa. Argumentin datatyyppi on väärä.
220	Riippuvainen raja-arvo
230	Dimensio Lista tai matriisi-indeksi ei kelpaa. Jos esimerkiksi lista $\{1,2,3,4\}$ on tallennettu L1:een, tällöin $L1[5]$ aiheuttaa dimensiovirheen, koska L1 sisältää vain neljä elementtiä.
235	Dimensiovirhe Listoissa ei ole riittävästi elementtejä.
240	Dimensioiden yhteensopivuusvirhe Kahdella tai useammalla argumentilla on oltava sama dimensio. Esimerkiksi lauseke $[1,2]+[1,2,3]$ aiheuttaa dimensioiden yhteensopivuusvirheen, koska matriisit sisältävät eri määrän elementtejä.
250	Jako nolalla
260	Määrittelyjoukkovirhe Argumentin on kuuluttava tiettyyn määrittelyjoukkoon. Esimerkiksi rand(0) ei kelpaa.
270	Muuttujan nimi jo käytössä
280	Else ja Elseif eivät kelpaa lohkon If..EndIf ulkopuolella
290	EndTry-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 arvoitonta 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	Indeksi alueen ulkopuolella
360	Epäsuora merkkijono ei kelpaa muuttujan nimeksi.
380	Määrittämätön Ans Edellinen laskutoimitus ei luonut Ans:n arvoa, tai aikaisempaa laskutoimitusta ei ole syötetty.

Virhekoodi	Kuvaus
390	Epäkelpo tehtävä
400	Epäkelpo tehtävän arvo
410	Virheellinen komento
430	Virheellinen nykyisille tila-asetuksille
435	Virheellinen arvaus
440	Virheellinen kertomerkitön kertolasku
	Esimerkiksi lauseke $x(x+1)$ ei kelpaa, mutta lausekkeen $x*(x+1)$ syntaksi on oikein. Tarkoituksena on välttää sekaannusta kertomerkitömiin kertolaskujen ja funktioiden kommentojen kanssa.
450	Virhe funktiossa tai nykyisessä lausekkeessa
	Vain tietyt komennot kelpaavat käyttäjän määrittämässä funktiossa.
490	Virhe Try..EndTry-lohkossa
510	Virheellinen lista tai matriisi
550	Virheellinen ulkopuolinen funktio tai ohjelma
	Jotkin komennot eivät kelpaa funktion tai ohjelman ulkopuolella. Esimerkiksi Local -komentoa ei voi käyttää, ellei se ole funktion tai ohjelman sisällä.
560	Virhe lohkojen Loop..EndLoop, For..EndFor tai While..EndWhile ulkopuolella
	Esimerkiksi Exit-komento kelpaa vain näiden silmukkalohkojen sisällä.
565	Virheellinen ulkopuolinen ohjelma
570	Polkunimi ei kelpaa
	Esimerkiksi \muutt ei kelpaa.
575	Virheellinen polaarin kompleksiluku
580	Virheellinen ohjelmaviittaus
	Ohjelmiin ei voi viitata funktioiden tai lausekkeiden sisällä, kuten $1+p(x)$, jossa p on ohjelma.
600	Kelpaamaton taulukko
605	Yksiköitä käytetty väärin
610	Virheellinen muuttujan nimi Local-lausekkeessa
620	Virheellinen muuttujan tai funktion nimi
630	Virheellinen muuttujan viittaus
640	Virheellinen vektorin syntaksi
650	Linkkisiirto
	Kahden laitteen välistä siirtoa ei ole suoritettu loppuun. Tarkista, että kaapeli on kytketty kunnolla kumpaankin laitteeseen.
665	Matriisia ei voi diagonalisoida
670	Muisti vähissä

Virhekoodi Kuvaus

1. Poista tietoja tästä asiakirjasta
 2. Tallenna ja sulje tämä asiakirja
- Jos toimenpiteet 1 ja 2 eivät auta, poista ja asenna paristot uudelleen
- 672 Resource exhaustion (Resurssit lopussa)
- 673 Resource exhaustion (Resurssit lopussa)
- 680 Puuttuva (
- 690 Puuttuva)
- 700 Puuttuva "
- 710 Puuttuva]
- 720 Puuttuva }
- 730 Syntaksista puuttuu lohkon alku tai loppu
- 740 Then puuttuu lohokosta If..EndIf
- 750 Nimi ei ole funktio eikä ohjelma
- 765 Yhtään funktiota ei ole valittu
- 780 Ratkaisua ei löydy
- 800 Ei-reaalinen tulos
- Esimerkiksi, jos ohjelmisto on Real-asetuksessa, $\sqrt{-1}$ ei kelpaa.
- Jotta voit saada kompleksilukuvastauksia, muuta reaali- tai kompleksitilan asetus valintaan RECTANGULAR (SUORAKULMA) tai POLAR (POLAARINEN).
- 830 Laskualueen ylitys
- 850 Ohjelmaa ei löydy
- Toisen ohjelman sisällä olevaa ohjelman viittausta ei löytynyt määritetystä polusta ohjelman suorittamisen aikana.
- 855 Rand-tyyppiset funktiot eivät ole sallittuja kuvaajan piirtämisen aikana
- 860 Liian syvä rekursio
- 870 Varattu nimi tai järjestelmän muuttuja
- 900 Argumenttinvirhe
- Mediaani-mediaani-mallia ei voitu käyttää datasarjaan.
- 910 Syntaksivirhe
- 920 Tekstiä ei löydy
- 930 Liian vähän argumentteja
- Funktioista tai komennosta puuttuu yksi tai useampia argumentteja.
- 940 Liian monta argumenttia

Virhekoodi	Kuvaus
	Lauseke tai yhtälö sisältää liian monta argumenttia, eikä sitä voi ratkaista.
950	Liian monta alaindeksiä
955	Liian monta määrittämätöntä muuttujaa
960	Muuttujaa ei ole määritetty
	Muuttujalle ei ole määritetty arvoa. Käytä jotakin seuraavista komennoista: <ul style="list-style-type: none"> • sto → • := • Define määrittääksesi muuttujille arvot.
965	Käyttöjärjestelmälle ei ole lisenssiä
970	Muuttuja on käytössä, joten viittaukset tai muutokset eivät ole sallittuja
980	Muuttuja on suojattu
990	Virheellinen muuttujan nimi
	Varmista, että nimi ei ylitä pituusrajoituksia
1000	Ikkunamuuttujien määrittelyalue
1010	Zoomaus
1020	Sisäinen virhe
1030	Suojatun muistin rikkomus
1040	Toimintoa ei tueta Tämä toiminto vaatii tietokonealgebrajärjestelmän Kokeile TI-Nspire™ CAS -järjestelmää.
1045	Operaattoria ei tueta. Tämä operaattori vaatii tietokonealgebrajärjestelmän. Kokeile TI-Nspire™ CAS -järjestelmää.
1050	Toimintoa ei tueta. Tämä operaattori vaatii tietokonealgebrajärjestelmän. Kokeile TI-Nspire™ CAS -järjestelmää.
1060	Syötetyn argumentin on oltava numeerinen. Vain numeerisia arvoja sisältävät syötteet sallitaan.
1070	Trig-funktion argumentti on liian suuri täsmälliseen sievennykseen
1080	Ans-muuttujan käyttöä ei tueta. Tämä sovellus ei tue Ans-muuttujan käyttöä.
1090	Funktiota ei ole määritetty. Käytä jotakin seuraavista komennoista: <ul style="list-style-type: none"> • Define • := • sto → määrittääksesi funktion.
1100	Ei-reaalinen laskutoimitus
	Esimerkiksi, jos ohjelmisto on Real-asetuksessa, $\sqrt{-1}$ ei kelpaa.

Virhekoodi	Kuvaus
	Jotta voit saada kompleksilukuvastauksia, muuta reaali- tai kompleksitilan asetus valintaan RECTANGULAR (SUORAKULMA) tai POLAR (POLAARINEN).
1110	Virheelliset rajat
1120	Ei etumerkin muutosta
1130	Argumentti ei voi olla lista tai matriisi
1140	Argumenttivorhe
	Ensimmäisen argumentin on oltava toisen argumentin sisällä oleva polynomilauseke. Jos toinen argumentti jätetään pois, ohjelmisto yrittää valita oletusarvon.
1150	Argumenttivorhe
	Ensimmäisten kahden argumentin on oltava kolmannen argumentin sisällä olevia polynomilausekkeita. Jos kolmas argumentti jätetään pois, ohjelmisto yrittää valita oletusarvon.
1160	Virheellinen kirjaston polkunimi
	Polkunimen on oltava muodossa xxx\yyy, jossa:
	<ul style="list-style-type: none"> • xxx-osassa voi olla 1-16 merkkiä. • yyy-osassa voi olla 1–15 merkkiä.
	Lisätietoja löydät ohjekirjan osasta Kirjasto.
1170	Kirjaston polkunimeä on käytetty väärin
	<ul style="list-style-type: none"> • 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:
	<ul style="list-style-type: none"> • ei sisällä pistettä • ei ala alaviivalla • ei ylitä 15 merkkiä
	Lisätietoja löydät ohjekirjan osasta Kirjasto.
1190	Kirjastodokumenttia ei löydy:
	<ul style="list-style-type: none"> • Tarkista, että kirjasto on MyLib-kansiossa. • Päivitä kirjastot.
	Lisätietoja löydät ohjekirjan osasta Kirjasto.
1200	Kirjastomuuttujaa ei löydy:
	<ul style="list-style-type: none"> • 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.

Virhekoodi Kuvaus

Tarkista, että nimi:

- ei sisällä pistettä
- ei ala alaviivalla
- ei ylitä 16 merkkiä
- ei ole varattu nimi

Lisätietoja löydät ohjekirjan osasta Kirjasto.

1220

Määrittelyjoukkovirhe:

Funktiot `tangentLine` ja `normalLine` tukevat vain reaaliarvoisia funktioita.

1230

Määrittelyjoukkovirhe.

Trigonometrysten muunnosten operaattoreita ei tueta aste- eikä graadikulmatilassa.

1250

Argumenttievirhe

Käytä lineeariyhtälöryhmää.

Esimerkki kahden lineeariyhtälön ryhmästä, jossa on muuttujat x ja y :

$$3x+7y=5$$

$$2y-5x=-1$$

1260

Argumenttievirhe:

nfMin:n tai **nfMax:n** ensimmäisen argumentin on oltava yhden muuttujan sisältämä lauseke. Se ei voi sisältää muuta arvotonta muuttujaa kuin käsittelyn kohteena olevan muuttujan.

1270

Argumenttievirhe

Derivaatan asteen on oltava 1 tai 2.

1280

Argumenttievirhe

Käytä yhden muuttujan sisältämää laajennetussa muodossa olevaa polynomia.

1290

Argumenttievirhe

Käytä yhden muuttujan sisältämää polynomia.

1300

Argumenttievirhe

Polynomin kertoimien tuloksena on oltava numeerinen arvo.

1310

Argumenttievirhe:

Yhtä tai useampaa funktion argumenttia ei voi sieventää.

1380

Argumenttievirhe:

Sisäkkäisiä kutsuja `domain()` funktioon ei sallita.

Varoituskoodit ja -viestit

Voit käyttää funktiota `warnCodes()` tallentaaksesi lausekkeen laskennan luomat varoituskoodit. Tämä taulukko luettelee jokaisen numeerisen varoituskoodin ja siihen liittyvän viestin. Esimerkki varoituskoodien tallentamisesta, katso `warnCodes()`, sivu 171.

Varoituskoodi	Viesti
10000	Laskutoimitus voi antaa vääräi ratkaisuja. Kokeile tarvittaessa graafisia menetelmiä tulosten varmistamiseksi.
10001	Yhtälön derivointi voi antaa väärän yhtälön.
10002	Kyseenalainen ratkaisu Kokeile tarvittaessa graafisia menetelmiä tulosten varmistamiseksi.
10003	Kyseenalainen tarkkuus Kokeile tarvittaessa graafisia menetelmiä tulosten varmistamiseksi.
10004	Laskutoimituksessa voi hävitä ratkaisuja. Kokeile tarvittaessa graafisia menetelmiä tulosten varmistamiseksi.
10005	cSolve voi määrittää lisää nollakohtia.
10006	Solve voi määrittää lisää nollia. Kokeile tarvittaessa graafisia menetelmiä tulosten varmistamiseksi.
10007	Lisää ratkaisuja voi olla olemassa. Yritä määrittämällä oikeat ala- ja ylärajat ja/tai arvaamalla. Esimerkkejä 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) Kokeile tarvittaessa graafisia menetelmiä tulosten varmistamiseksi.
10008	Tuloksen määrittelyjoukko voi olla pienempi kuin syötteen määrittelyjoukko.
10009	Tuloksen määrittelyjoukko voi olla suurempi kuin syötteen määrittelyjoukko.
10012	Ei-reaalinen laskutoimitus
10013	∞^0 tai undef^0 korvattu 1:llä
10014	undef^0 korvattu 1:l
10015	1^∞ tai 1^undef korvattu 1:llä

Varoituskoodi	Viesti
10016	1^undef korvattu 1:ll
10017	Laskualueen ylitys korvattu ∞ :lla tai $-\infty$:lla
10018	Laskutoimitus vaatii ja laskee 64-bittisen arvon.
10019	Muisti täynnä, sievennystä ei mahdollisesti suoriteta loppuun.
10020	Trig-funktion argumentti on liian suuri täsmälliseen sievennykseen.
10021	Syöte sisältää määrittämättömän parametrin. Tulos ei ehkä ole voimassa kaikille mahdollisille parametriarvoille.
10022	Ratkaisu on mahdollinen, kun määritetään oikeat ala- ja ylärajat.
10023	Skalaari on kerrottu tunnistematriisilla.
10024	Tulos saatu käyttäen likiarvo-aritmetiikkaa.
10025	Ekvivalenssia ei voida varmistaa EXACT-tilassa.
10026	Ehto saatetaan sivuuttaa. Määritä ehto muodossa "\" 'Muuttujan MathTestSymbol-vakio' tai näiden yhteismuoto, esim. 'x<3 and x>-12'

Yleistä

Online-tuki

education.ti.com/eguide

Valitse maasi, niin näet lisää tuotetietoja.

Ota yhteyttä TI-tukeen

education.ti.com/ti-cares

Valitse maasi, niin näet teknisiä tietoja ja muita tukiresursseja.

Huolto- ja takuutiedot

education.ti.com/warranty

Valitse maasi, niin saat tietoa takuun kestosta ja ehdoista tai tuotepalvelusta.

Rajoitettu takuu. Tämä takuu ei vaikuta lainmukaisiin oikeuksiisi.

Texas Instruments Incorporated

12500 TI Blvd.

Dallas, TX 75243

Indeksi

-			
- , vähennyslasku[*]	179	, (rajoittava operaattori)	196
!		+	
! , kertoma	188	+ , yhteenlasku	179
"		/	
" , sekuntimuoto	194	/ , jakolasku[*]	180
#		=	
# , epäsuora operaattori	192, 219	≠ , ei yhtä kuin[*]	184
%		= , on yhtä kuin	183
% , prosentti	183	>	
&		> , suurempi kuin	186
& , liitä	188	∏	
*		∏ , tulo[*]	189
* , kertolasku	180	∑	
,		∑ () , summa[*]	189
, , minuuttimuoto	194	∑Int ()	190
.		∑Prn ()	191
- , piste-erotus	182	√	
* , pistetulo	182	√ , neliöjuuri[*]	188
./ , pisteosamäärä	182	≤	
^ , pistepotenssi	182	≤ , pienempi tai yhtä suuri kuin	185
+ , pistesumma	182	≥	
:		≥ , suurempi tai yhtä suuri kuin	186
:= , määritä	197	►	
^		► , muunna graadikulmaksi[Grad]	71
^-1 , käänteisluku	196	► , asteen regressio, QuartReg	122
^ , potenssi	181	► approxFraction ()	13
		► Cylind , näytä vektorina	
		sylinderikoordinaatistossa	
		[Cylind]	36
		► DD , näytä desimaalikulmana[DD]	36

►Desimaali, näytä tulos desimaalilukuna[Desimaali]	37	2	
►DMS, näytä asteina/minuutteina/sekun teina[DMS]	44	2 otoksen F-testi	59
►Kantaluku10, näytä desimaalikonaislukuna [Kantaluku10]	18	2. asteen regressio, QuadReg	120
►Kantaluku16, näytä heksadesimaalilukuna [Kantaluku16]	19	3	
►Kantaluku2, näytä binaarisena [Kantaluku2]	17	3. asteen regressio, CubicReg	34
►Polaarinen, näytä polaarisenä vektorina[Polaarinen]	115	A	
►Rad, muunna radiaanikulmaan	123	abs(), itseisarvo	7
►Rect, näytä suorakulmavektorina	126	alaraja, floor()	55
►Sphere, näytä pallonmuotoisena vektorina[Sphere]	149	alimatriisi, subMat()	154, 156
⇒		amortTbl(), lyhennystaulukko	7, 16
⇒, looginen seuraus[*]	187, 217	and, Boolean operator	8
→		angle(), kulma	9
→, tallenna	197	ANOVA, yksisuuntainen varianssianalyysi	9
↔		ANOVA2way, kaksisuuntainen varianssianalyysi	10
⇔, looginen kaksoisseuraus[*]	187	ans, viimeinen vastaus	12
©		approx(), likimääräinen	12
©, kommentti	198	approxRational()	13
°		arccos()	13
°, astekulmamuoto[*]	193	arccosh()	13
°, astetta/minuuttia/sekuntia[*]	194	arccot()	13
0		arccoth()	13
Ob, binaarinen indikaattori	198	arccsc()	14
Oh, heksadesimaalinen indikaattori	198	arccsch()	14
1		arcsec()	14
10^(), kymmenen potenssi	195	arcsech()	14
		arcsin()	14
		arcsinh()	14
		arctan()	14
		arctanh()	14
		arkuskosini, cos ⁻¹ ()	28
		arkussini, sin ⁻¹ ()	146
		arkustangentti, tan ⁻¹ ()	157
		asetta tila, setMode()	141
		asetukset, hae nykyiset	67
		aste/minuutti/sekunti-muoto	194
		astekulmamuoto, -	193
		augment(), lisää/ketjuta	14
		avgRC(), keskimääräinen muutosnopeus	15

B		csch ⁻¹ (), käänteinen hyperbolinen kosekantti	33
binaarinen		csch(), hyperbolinen kosekantti	33
indikaattori, Ob	198	CubicReg, 3. asteen regressio	34
näytä, 4Kantaluku2	17	cumulativeSum(), kumulatiivinen summa	35
binomCdf()	20, 76	cycle, Cycle	35
binomPdf()	20	Cycle, cycle	35
Boolean operators		D	
and	8	d(), ensimmäinen derivaatta	188
Boolean operaattorit		dbd(), päivämäärien väliset päivät	36
⇒	187, 217	Define (Määritä)	37
⇔	187	Define LibPriv (Määritä LibPriv)	38
ei	107	Define LibPub (Määritä LibPub)	39
ei-ja	102	Define, määritä	37
eikä	106	deltaList()	39
tai	111	DelVar, poista muuttuja	40
xor	172	delVoid(), poista tyhjät elementit	40
C		derivaatat	
Cdf()	54	ensimmäinen derivaatta, d()	188
ceiling(), yläraja	20	numeerinen derivaatta, nDeriv()	104-105
centralDiff()	21	numeerinen derivaatta, nDerivative()	103
char(), merkkijono	22	desimaali	
clearAZ	24	kokonaislukunäyttö, 4Kantaluku10	18
ClrErr, poista virhe	24	kulmanäyttö, ►DD	36
colAugment	25	det(), matriisin determinantti	40
colDim(), matriisin sarakemitat	25	diag(), diagonaalimatriisi	41
colNorm(), matriisin sarakenormi	25	dim(), mitta	41
conj(), liittokompleksiluku	25	Disp, näytä tiedot	42, 139
constructMat(), luo matriisi	26	DispAt	42
corrMat(), korrelaatiomatriisi	27	dotP(), pistetulo	44
cos ⁻¹ , arkuskosini	28	E	
cos(), kosini	27	e eksponentti	
cosh ⁻¹ (), hyperbolinen arkuskosini	29	malli	2
cosh(), hyperbolinen kosini	29	e korotettuna potenssiin, e^()	45, 51
cot ⁻¹ (), arkuskotangenti	30	E, eksponentti	192
cot(), kotangenti	30	e^(), e korotettuna potenssiin	45
coth ⁻¹ (), hyperbolinen arkuskotangenti	31	efektiivinen korko, eff()	45
arkuskotangenti	31	eff(), muunna nimelliskorko efektiiviseksi koroksi	45
coth(), hyperbolinen kotangenti	30	ei-ja, Boolean operaattori	102
count(), laske listan kohtien lukumäärä	31	ei yhtä kuin, ≠	184
countif(), laske listan yksiköiden lukumäärä ehdollisesti	31		
cPolyRoots()	32		
crossP(), ristitulo	32		
csc ⁻¹ (), käänteiskosekantti	33		
csc(), kosekantti	33		

ei, Boolean operaattori	107	funktiot	
eigVc(), ominaisvektori	46	käyttäjän määrittämät	37
eigVI(), ominaisarvo	46	ohjelmafunktio, Func	60
eikä, Boolean operaattori	106	osa, fpart()	57
ekskluisio " " -operaattorilla	196	funktiot ja muuttujat	
eksponentiaalinen regressio, ExpReg	52	kopiointi	26
eksponentit			
malli	1		
eksponentti, E	192	G	
else if, Elseif	47	g, graadit	192
Elseif, else if	47	gcd(), suurin yhteinen jakaja	60
end		geomCdf()	61
for, EndFor	56	geomPdf()	61
funktio, EndFunc	60	Get	62, 209
ohjelma, EndPrgm	118	getDenom(), hae/laske nimittäjä ...	63
silmutta, EndLoop	93	getKey()	63
try, EndTry	162	getLangInfo(), hae/anna kielitiedot	66
while, EndWhile	172	getLockInfo(), testaa muuttujan tai	
end while, EndWhile	172	muuttujaryhmän	
EndTry, kokeilun loppu	162	lukitustilan	67
EndWhile, end while	172	getMode(), hae tila-asetukset	67
ensimmäinen derivaatta		getNum(), hae/laske luku	68
malli	5	GetStr	68
EOS (yhtälökäyttöjärjestelmä)	218	getType(), get type of variable	69
epäsuora operaattori (#)	219	getVarInfo(), hae/laske muuttujien	
epäsuora operaattori, #	192	tiedot	69
etumerkki, sign()	144	Goto, siirry	70
euler(), Euler function	48	graadimuoto, g	192
Exit, lopeta	50		
exp(), e korotettuna potenssiin	51	H	
expr(), merkkijono lausekkeeksi	51	hae/laske	
ExpReg, eksponentiaalinen regressio	52	luku, getNum()	68
		muuttujien tiedot, getVarInfo()	66, 69
F		nimittäjä, getDenom()	63
factor(), tekijä	53	heksadesimaali	
Fill, matriisiin täyttö	54	näytä, 4Kantaluku16	19
FiveNumSummary	54	heksadesimaalinen	
floor(), alaraja	55	indikaattori, Oh	198
For	56	hyperbolinen	
for, For	56	areatangenti, tanh ⁻¹ ()	158
For, for	56	arkuskosini, cosh ⁻¹ ()	29
format(), merkkijonon muoto	56	arkussini, sinh ⁻¹ ()	147
fpart(), funktion osa	57	kosini, cosh()	29
freqTable()	57	sini, sinh()	147
frequency()	58	tangenti, tanh()	158
Frobeniusin normi, norm()	107		
Func, funktio	60	I	
Func, ohjelmafunktio	60	identity(), identtinen matriisi	71
funktion loppu, EndFunc	60		

laske listan kohtien lukumäärä, count()	31	pistetulo, dotP()	44
laske listan yksiköiden lukumäärä ehdollisesti, countif()	31	ristitulo, crossP()	32
laske päivämäärien väliset päivät, dbd()	36	tulo, product()	118
laske polynomi, polyEval()	116	tyhjät elementit	215
lausekkeet		uusi, newList()	104
merkkijono lausekkeeksi, expr()	51	yhteenlasku, sum()	155
lbi, tunnus	79	ln(), luonnollinen logaritmi	87
lcm, pienin yhteinen jaettava	79	LnReg, logaritminen regressio	87
left(), vasen	79	Local, paikallinen muuttuja	89
LibPriv	38	Lock, lukitse muuttuja tai muuttujaryhmä	89
LibPub	39	Logaritmi	
libShortcut(), luo pikavalinnat kirjasto-objekteihin	80	malli	2
liitä, &	188	logaritminen regressio, LnReg	87
likimääräinen, approx()	12	logaritmit	87
lineaarinen regressio, LinRegAx	81	Logistic, logistinen regressio	90
lineaarinen regressio, LinRegBx	80, 83	LogisticD, logistinen regressio	91
LinRegBx, lineaarinen regressio	80	logistinen regressio, Logistic	90
LinRegMx, lineaarinen regressio	81	logistinen regressio, LogisticD	91
LinRegtIntervals, lineaarinen regressio	83	looginen kaksoisseuraus, ⇔	187
LinRegtTest	84	looginen seuraus, ⇒	187, 217
linSolve()	85	Loop, silmukka	93
lisää/ketjuta, augment()	14	lopeta, Exit	50
list►mat(), listasta matriisiksi	86	loppu	71
lista, laske kohtien lukumäärä	31	loppu jos, EndIf	71
lista, laske yksiköiden lukumäärä ehdollisesti	31	LU, matriisin ala-ylä-dekomponointi	93
listasta matriisiksi, list►mat()	86	lukuono	
listat		kierrä, rotate()	134
erotukset listassa, @list()	86	oikea, right()	133
erotus, @list()	86	lukuonot	
keskimmäinen merkkijono, mid()	97	oikea, right()	75
kumulatiivinen summa, cumulativeSum()	35	luo matriisi, constructMat()	26
lajittele laskevaan järjestykseen, SortD	149	luonnollinen logaritmi, ln()	87
lajittele nousevaan järjestykseen, SortA	149	lyhennystaulukko, amortTbl()	7, 16
lisää/ketjuta, augment()	14		
listasta matriisiksi, list►mat()	86	M	
maksimi, max()	94	määrätty integraali	
matriisi listaksi, mat►list()	94	malli	6
minimi, min()	97	määrittä, Define	37
		määrittäminen	
		julkinen funktio tai ohjelma	39
		yksityinen funktio tai ohjelma	38
		maksimi, max()	94
		mallit	
		e eksponentti	2
		eksponentti	1
		ensimmäinen derivaatta	5
		itseisarvo	3-4

Logaritmi	2	QR-hajottaminen, QR	119
määrätty integraali	6	rivi-echelon-muoto, ref()	127
matriisi (1 × 2)	4	rivien kertominen ja	
matriisi (2 × 1)	4	yhteenlasku, mRowAdd	
matriisi (2 × 2)	4	()	99
matriisi (m × n)	4	rivien yhteenlasku, rowAdd() ..	136
murtoluku	1	rivikoko, rowDim()	136
n:s juuri	1	rivinormi, rowNorm()	137
neliöjuuri	1	rivinvaihto, rowSwap()	137
paloittain määritelty funktio (2-		rivioperaatio, mRow()	99
osainen)	2	sarakemitat, colDim()	25
paloittain määritelty funktio (N-		sarakenormi, colNorm()	25
osainen)	2	satunnainen, randMat()	125
sum(G)	5	täyttämisen, Fill	54
toinen derivaatta	6	transponoi, T	156
tulo(P)	5	tulo, product()	118
yhtälöpari (2 yhtälöä)	3	uusi, newMat()	104
yhtälöryhmä (N yhtälöä)	3	yhteenlasku, sum()	155
matHist(), matriisi listaksi	94	max(), maksimi	94
matriisi (1 × 2)		mean(), keskiarvo	95
malli	4	mediaani-mediaani-suoran	
matriisi (2 × 1)		regressio, MedMed	96
malli	4	mediaani, median()	95
matriisi (2 × 2)		median(), mediaani	95
malli	4	MedMed, mediaani-mediaani-	
matriisi (m × n)		suoran regressio	96
malli	4	merkit	
matriisi listaksi, matHist()	94	merkkijono, char()	22
matriisit		numeerinen koodi, ord()	113
ala-ylä-dekomponointi, LU	93	merkkijono	
alimatriisi, subMat()	154, 156	mitta, dim()	41
determinantti, det()	40	pituus	41
diagonaali, diag()	41	merkkijono, char()	22
identtinen, identity()	71	merkkijonon muoto, format()	56
kumulatiivinen summa,		merkkijonon pituus	41
cumulativeSum()	35	merkkijonon sisällä, inString()	74
lisää/ketjuta, augment()	14	merkkijonot	
listasta matriisiksi, listMat() ..	86	epäsuora operaattori, #	192
maksimi, max()	94	käyttö muuttujanimien	
matriisi listaksi, matHist()	94	luomisessa	219
minimi, min()	97	keskimmäinen merkkijono, mid(
mitta, dim()	41)	97
ominaisarvo, eigVl()	46	lauseke merkkijonoksi, string()	154
ominaisvektori, eigVc()	46	liitä, &	188
piste-erotus, -	182	merkkijono lausekkeeksi, expr()	51
pisteosamäärä, ./	182	merkkijono, char()	22
pistepotenssi, ^	182	merkkikoodi, ord()	113
pistesumma, +	182	muoto, format()	56
pistetulo, .*	182	muotoilu	56

siirrä, shift()	143	näppäimistön pikavalinnat	217
sisällä, inString()	74	näytä	
vasen, left()	79	asteina/minuutteina/sekunteina,	
mid(), keskimäinen merkkijono	97	►DMS	44
min(), minimi	97	binaarisena, 4Kantaluku2	17
minimi, min()	97	desimaalikkokonaislukuna,	
minuuttimuoto,	194	4Kantaluku10	18
mirr(), modifioitu sisäinen		desimaalikulmana, ►DD	36
korkokanta	98	heksadesimaalilukuna,	
mitta, dim()	41	4Kantaluku16	19
mod(), modulo	99	pallonmuotoinen vektori,	
modifioitu sisäinen korkokanta, mirr		►Sphere	149
()	98	polaarinen vektori, 4Polaarinen	115
modulo, mod()	99	suorakulmavektorina, ►Rect	126
Moninkertaisen lineaarisen		vektorina	
regression t-testi	101	sylinderikoordinaatistos	
mRow(), matriisin rivioperaatio	99	sa, 4Cylind	36
mRowAdd(), matriisin rivien		näytä tiedot, Disp	42, 139
kertominen ja yhteenlasku	99	näyttö	
MultiReg	99	asteina/minuutteina/sekun	
MultiRegIntervals()	100	teina, ►DMS	44
MultiRegTests()	101	näyttö pallonmuotoisena vektorina,	
murtoluvut		►Sphere	149
malli	1	näyttö vektorina	
propFrac	119	sylinderikoordinaatistossa,	
muunna		►Cylind	36
4Grad	71	nCr(), kombinaatiot	103
muuntaminen		nDerivative(), numeerinen	
►Rad	123	derivaatta	103
muutoin, Else	71	negaatio, negatiivisten lukujen	
muuttaja		syöttäminen	219
nimen luominen merkkijonosta	219	neliöjuuri	
muuttajat		malli	1
paikallinen, Local	89	neliöjuuri, ‡()	150, 188
poista kaikki yksikirjaimiset	24	nettonykyarvo, npv()	109
poista, DelVar	40	newList(), uusi lista	104
muuttajat ja funktiot		newMat(), uusi matriisi	104
kopioiminen	26	nfMax(), numeerisen funktion	
muuttajat, lukitseminen ja		maksimi	104
vapauttaminen	67, 89, 168	nfMin(), numeerisen funktion	
muuttajien ja muuttajaryhmien		minimi	105
lukitseminen	89	nimelliskorko, nom()	106
muuttajien ja muuttajaryhmien		nInt(), numeerinen integraali	105
vapauttaminen	168	nom(), muuta efektiivinen korko	
		nimelliskoroksi	106
N		norm(), Frobeniusin normi	107
n:s juuri		normaalijakauman todennäköisyys,	
malli	1	normCdf()	107

normCdf ()	107	osainen)	
normPdf ()	107	malli	2
nPr () , permutaatiot	108	paloittain määritelty funktio (N- osainen)	
npv () , nettonykyarvo	109	malli	2
nSolve () , numeerinen ratkaisu	109	PassErr, ohita virhe	114
numeerinen		Pdf ()	57
derivaatta, nDeriv ()	104-105	permutaatiot, nPr ()	108
derivaatta, nDerivative ()	103	piecewise ()	114
integraali, nInt ()	105	pienempi tai yhtä suuri kuin, {	185
ratkaisu, nSolve ()	109	pienin yhteinen jaettava, lcm	79
		piirrä	204-206
		pikavalinnat, näppäimistö	217
		piste	
O		erotus, -	182
objektit		osamäärä, ./	182
luo pikavalinnat kirjastoon	80	potenssi, ^	182
Odota-komento	170	summa, +	182
ohita virhe, PassErr	114	tulo, *	182
ohjelmat		tulo, dotP ()	44
julkisen kirjaston määrittäminen	39	poissCdf ()	114
yksityisen kirjaston		poissPdf ()	115
määrittäminen	38	poista	
ohjelmat ja ohjelmointi		poista, ClrErr	24
kokeilun loppu, EndTry	162	tyhjät elementit listasta	40
lopeta ohjelma, EndPrgm	118	poistaminen	
näytä I/O-näyttö, Disp	42, 139	muuttuja, DelVar	40
poista virhe, ClrErr	24	polaarinen	
try, Try	162	koordinaatti, R►Pr ()	123
ohjelmointi		koordinaatti, R►Pθ ()	123
määritä ohjelma, Prgm	118	vektorinäyttö, 4Polaarinen	115
näytä tiedot, Disp	42, 139	polyEval () , laske polynomi	116
ohita virhe, PassErr	114	polynomit	
oikea, right ()	75, 133	laske, polyEval ()	116
ominaisarvo, eigVl ()	46	satunnainen, randPoly ()	126
ominaisvektori, eigVc ()	46	PolyRoots ()	116
on yhtä kuin, =	183	potenssi, ^	181
OneVar, yhden muuttujan tilastot	110	potenssiregressio,	
operaattorit		PowerReg	116, 130-131, 159
sievennysjärjestys	218	PowerReg, potenssiregressio	116
ord () , numeerinen merkkikoodi	113	Prgm, määritä ohjelma	118
		prodSeq ()	118
		product () , tulo	118
P		propFrac, varsinainen murtoluku	119
P►Rx () , x:n suorakulmakoordinaatti	113	prosentti, %	183
P►Ry () , y:n suorakulmakoordinaatti	113	pyöristä, round ()	136
pääoman maksuerien summa	191	PyydäMerkkij	131
paikallinen muuttuja, Local	89	Pyyntö	130
paikallinen, Local	89		
päivämäärien väliset päivät, dbd ()	36		
Palaa, return	132		
paloittain määritelty funktio (2-			

		MultiReg	99
		potenssiregressio,	
		PowerReg	116, 130-131, 159
		sinimuotoinen, SinReg	147
QR-hajottaminen, QR	119	remain(), jäännös	129
QR, QR-hajottaminen (tekijöihin jako)	119	return, Palaa	132
QuadReg, 2. asteen regressio	120	right(), oikea	133
QuartReg, 4. asteen regressio	122	right, right()	48, 171
		ristitulo, crossP()	32
		rivi-echelon-muoto, ref()	127
		rk23(), Runge Kutta -funktio	133
		rotate(), kierrä	134
		round(), pyöristä	136
		rowAdd(), matriisin rivien yhteenlasku	136
		rowDim(), matriisin rivikoko	136
		rowNorm(), matriisin rivinormi	137
		rowSwap(), matriisin rivinvaihto	137
		ref(), sievennetty rivi-echelon- muoto	137
		ryhmät, lukitseminen ja vapauttaminen	89, 168
		ryhmät, lukitustilan testaaminen ...	67
		S	
		samanaikaiset yhtälöt, simult()	145
		satunnainen	
		matriisi, randMat()	125
		normaalijakauma, randNorm() .	125
		polynomi, randPoly()	126
		siemenluku, RandSeed	126
		satunnaisotos	126
		sec ⁻¹ (), käänteissekanti	138
		sec(), sekanti	138
		sech ⁻¹ (), käänteinen hyperbolinen sekanti	138
		sech(), hyperbolinen sekanti	138
		sekaluvut, käyttäen funktiota propFrac(>)	119
		sekuntimuoto, "	194
		sekvenssi, seq()	139
		seq(), sekvenssi	139
		seqGen()	140
		seqn()	140
		sequence, seq()	140
		setMode(), aseta tila	141
		shift(), siirrä	143
		sievennetty rivi-echelon-muoto, rref	137

()		tai, Boolean operaattori	111
sievennysjärjestys	218	tallentaminen	
sign(), etumerkki	144	symboli, &	197
siirry, Goto	70	talouslaskentafunktiot, tvmlfv()	164
silmukan loppu, EndLoop	93	talouslaskentafunktiot, tvmlf()	165
silmukka, Loop	93	talouslaskentafunktiot, tvmln()	165
simult(), samanaikaiset yhtälöt	145	talouslaskentafunktiot, tvmlpmt()	165
sin ⁻¹ (), arkussini	146	talouslaskentafunktiot, tvmlpv()	166
sin(), sini	146	tan ⁻¹ (), arkustangenti	157
sinh ⁻¹ (), hyperbolinen arkussini	147	tan(), tangenti	156
sinh(), hyperbolinen sini	147	tangenti, tan()	156
sini, sin()	146	tanh ⁻¹ (), hyperbolinen areatangenti	158
siniregressio, SinReg	147	tanh(), hyperbolinen tangenti	158
SinReg, siniregressio	147	täyttö	207-208
SortA, lajittele nousevaan		tCdf(), studentint-	
järjestykseen	149	todennäköisyysjakauma	158
SortD, lajittele laskevaan		tekijä, factor()	53
järjestykseen	149	Teksti-komento	159
sqrt(), neliöjuuri	150	Test_2S, 2 otoksen F-testi	59
stat.results	151	tila-asetukset, getMode()	67
stat.values	152	tilastot	
stdDevPop(), perusjoukon		kahden muuttujan tulokset,	
keskihajonta	152	TwoVar	166
stdDevSamp(), otoksen keskihajonta	153	kertoma, !	188
Stop-komento	154	keskiarvo, mean()	95
string(), lauseke merkkijonoksi	154	keskihajonta, stdDev()	152-153, 169
strings		kombinaatiot, nCr()	103
right, right()	48, 171	mediaani, median()	95
studentint-todennäköisyysjakauma,		permutaatiot, nPr()	108
tCdf()	158	satunnainen normaalijakauma,	
studentint-todennäköisyysaste,		randNorm()	125
tPdf()	161	satunnainen siemenluku,	
subMat(), alimatriisi	154, 156	RandSeed	126
sum(), yhteenlasku	155	varianssi, variance()	169
sumlf()	155	yhden muuttujan tilastot,	
summa (G)		OneVar	110
malli	5	tilat	
summa, S()	189	asetus, setMode()	141
sumSeq()	156	TInterval, t-luottamusväli	160
suorakulmavektorinäyttö, ►Rect	126	TInterval_2Samp, kahdenotoksen t-	
suurempi kuin, >	186	luottamusvälin	160
suurempi tai yhtä suuri kuin, 	186	todennäköisyysaste, normPdf()	107
suurin yhteinen jakaja, gcd()	60	toinen derivaatta	
		malli	6
		tPdf(), studentint-	
		todennäköisyysaste	161
		trace()	161
		transponoi, T	156
		Try, virheenkäsitteilykomento	162

T

tTest, t-testi	163	virheet ja vianmäärittys	
tTest_2Samp, kahden otoksen t-testi	163	ohita virhe, PassErr	114
tulo (P)		poista virhe, ClrErr	24
malli	5	virhekoodit ja -viestit	231
tulo, P()	189		
tulo, product()	118	W	
tulokset, tilastot	151	warnCodes(), Warning codes	171
tulosarvot, tilastot	152	when(), kun	171
tunnus, Lbl	79	while, While	172
TVM-argumentit	166	While, while	172
TVM-funktioiden argumentit	166		
tvmFV()	164	X	
tvml()	165	x:n suorakulmakoordinaatti, P►Rx()	113
tvmN()	165	x ² , neliö	181
tvmPmt()	165	XNOR	187
tvmPV()	166	xor, Boolean eksklusiivinen tai (or) ...	172
TwoVar, kahden muuttujan tulokset	166		
tyhjä, testi	78	Y	
tyhjän testi, isVoid()	78	y:n suorakulmakoordinaatti, P►Ry()	113
tyhjät elementit	215	yhden muuttujan tilastot, OneVar ..	110
tyhjät elementit, poista	40	Yhtälökäyttäjärjestelmä (EOS)	218
Tyhjennä	203	yhtälöpari (2 yhtälöä)	
		malli	3
U		yhtälöryhmä (N yhtälöä)	
unitV(), yksikkövektori	168	malli	3
unLock, vapauta muuttuja tai		yhteenlasku, +	179
muuttujaryhmä	168	yhteenlasku, sum()	155
uusi		yksikkövektori, unitV()	168
lista, newList()	104	yläraja, ceiling()	20-21, 32
matriisi, newMat()	104		
		Z	
V		zInterval, z-luottamusväli	173
vähennyslasku, N	179	zInterval_1Prop, yhden osuuden z-	
vaihto, shift()	143	luottamusväli	174
varianssi, variance()	169	zInterval_2Prop, kahden osuuden z-	
varoituskoodit ja -viestit	231	luottamusväli	174
varPop()	169	zInterval_2Samp, kahden näytteen	
varSamp(), otoksen varianssi	169	z-luottamusväli	175
varsinainen murtoluku, propFrac ...	119	zTest	176
vasen, left()	79	zTest_1Prop, yhden osuuden z-testi	177
vastaus (viimeinen), ans	12	zTest_2Prop, kahden osuuden z-testi	177
vektorit		zTest_2Samp, kahden otoksen z-	
näyttö vektorina		testi	178
sylinderikoordinaattisto			
sa, ►Cylind	36		
pistetulo, dotP()	44		
ristitulo, crossP()	32		
yksikkö, unitV()	168		

Δ

Δ list(), listaerodus	86
-------------------------------------	----

X

χ^2 2way	22
χ^2 Cdf()	22
χ^2 GOF	23
χ^2 Pdf()	23