



TI-*nspire*[™]

TI-Nspire[™] CAS
參考手冊

本手冊適用 TI-Nspire[™] 軟體 4.5 版。如欲取得最新版的文件，請前往 education.ti.com/go/download。

重要資訊

除伴隨程式的授權中明確陳述之外，德州儀器概不提供有關任何程式或書籍資料的明示或暗示保證，包括但不限於任何可售性和特別目的適合性的暗示保證，並且僅按「原樣」提供此等資料。無論任何情況，德州儀器皆不負責與購買或使用這類資料有關或所致的任何特殊、附屬、附帶或衍生損害賠償，且無論行動的形式，德州儀器的唯一責任不會超過程式授權中載明的金額。此外，德州儀器不承擔任何種類的賠償責任，不管是否有任何其他當事人因使用這些材料而索賠。

授權

請參見安裝於 `C:\Program Files\TI Education\<TI-Nspire™ Product Name>\license` 的完整授權。

© 2006 - 2017 Texas Instruments Incorporated

目錄

重要資訊	ii
運算式範本	1
依字母順序列表	8
A	8
B	17
C	20
D	43
E	55
F	64
G	72
I	81
L	88
M	103
N	110
O	118
P	120
Q	128
R	131
S	145
T	168
U	182
V	182
W	183
X	185
Z	186

符號	194
空元素	219
輸入數學式的快速鍵	221
EOS™(方程式作業系統)階層	223
常數和數值	225
錯誤代碼和訊息	226
警告代碼和訊息	234
Texas Instruments 支持与服务	236
索引	237

運算式範本

運算式範本讓您能以標準的數學符號輕鬆輸入數學式。插入範本時，範本會出現在輸入線上，並在可輸入要素的位置顯示小方塊。游標會顯示您可輸入的要素。

使用方向鍵或按 **tab** 讓游標移到每個要素的位置，然後輸入數值或運算式。按 **enter** 或 **ctrl enter** 開始對運算式求值。

分數範本

ctrl **÷** 鍵



附註：另請參考 /(除)，頁碼：196。

範例：

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

指數範本

^ 鍵



附註：輸入第一個值、按 **^**，然後輸入指數。如果要讓游標回到基線，請按向右鍵(▶)。

附註：另請參考 ^(次方)，頁碼：197。

範例：

$$2^3 \qquad 8$$

平方根範本

ctrl **x²** 鍵



附註：另請參考 √()(平方根)，頁碼：207。

範例：

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

N 次方根範本

ctrl **^** 鍵



附註：另請參考 root()，頁碼：142。

範例：

N 次方根範本

ctrl ^ 鍵

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

e 指數範本

e^x 鍵

e

自然指數 e 提升到某乘冪

附註：另請參考 $e^{\wedge}()$ ，頁碼：55。

範例：

$$e^1 \quad e$$
$$e^1. \quad 2.71828182846$$

對數範本

ctrl 10^x 鍵

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

計算指定基底的對數。如果是預設基底 10，則可省略基底。

附註：另請參考 $\log()$ ，頁碼：99。

範例：

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

分段函數範本(2 段)

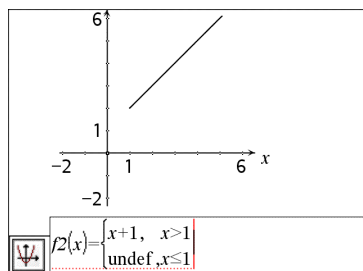
目錄 > \log_{\square}^{\square}

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

讓您建立二段函數的運算式和條件。- 如果要增加條件，請按一下範本並重複範本程序。

附註：另請參考 $\text{piecewise}()$ ，頁碼：122。

範例：



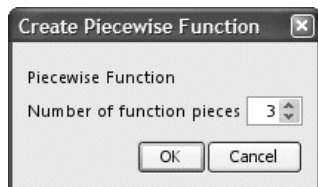
分段函數範本(N段)

目錄 > 

讓您建立 N -段函數的運算式和條件。
系統要求您輸入 N 。


範例:

請參考分段函數範本(2段)的範例。



附註: 另請參考 `piecewise()`, 頁碼: 122。

2 個聯立方程式範本

目錄 > 



建立兩個聯立方程式。如果要在現有聯立方程式中加入一列,請按一下範本並重複範本程序。

附註: 另請參考 `system()`, 頁碼: 167。

範例:

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

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

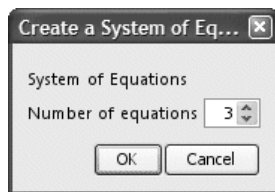
N 個聯立方程式範本

目錄 > 

讓您建立 N 個聯立方程式。系統要求您輸入 N 。


範例:

請參考聯立方程式(2個方程式)範本的範例。



附註: 另請參考 `system()`, 頁碼: 167。

絕對值範本


目錄 > 



附註: 另請參考 `abs()`, 頁碼: 8。

範例:

絕對值範本

目錄 > 

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

dd°mm'ss.ss" 範本

目錄 > 




範例:

讓您以 **dd°mm'ss.ss"** 的格式輸入角度。
dd 代表十進位度數、**mm** 代表分、**ss.ss** 代表秒。

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

矩陣範本 (2 x 2)

目錄 > 




範例:

建立 2 x 2 的矩陣。

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

矩陣範本 (1 x 2)


目錄 > 



範例:

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

矩陣範本 (2 x 1)

目錄 > 



範例:

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

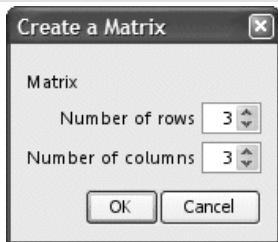
矩陣範本 (m x n)

目錄 > 

系統要求您指定列數和行數後，就會顯示範本。

範例:

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



附註：如果您建立的矩陣有大量行列，可能就要等一下才會出現矩陣。

總和範本 (Σ)

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

範例：

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

附註：另請參考 $\Sigma()$ (sumSeq)，頁碼：208。

乘積範本 (Π)

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

範例：

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

附註：另請參考 $\Pi()$ (prodSeq)，頁碼：207。

一階導數範本

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

範例：

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

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

一階導數範本可用於計算某個點的一階導數。

附註：另請參考 $d()$ (導數)，頁碼：205。

二階導數範本

目錄 > 

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

二階導數範本亦可用於計算某個點的二階導數。

附註：另請參考 **d()**(導數)，頁碼：205。

範例：

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

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

N 階導數範本

目錄 > 

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

n 階導數範本可用於計算 n 階導數。

附註：另請參考 **d()**(導數)，頁碼：205。

範例：

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

定積分範本

目錄 > 

$$\int_{\square}^{\square} \square d\square$$

附註：另請參考 **∫()** **integral()**，頁碼：194。

範例：

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

不定積分範本

目錄 > 

$$\int \square d\square$$

附註：另請參考 **∫()** **integral()**，頁碼：194。

範例：

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

極限範本

目錄 > 

$$\lim_{\square \rightarrow \square} (\square)$$

左極限請用 - 或 (-)。右極限請用 +。

範例：

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


附註：另請參考 `limit()`，頁碼：90。

依字母順序列表

名稱非字母的項目(例如 +、!、>)列在本節最後面,從第頁碼:194 頁開始。除非另行指定,否則本節中的所有範例皆是以預設的重置模式執行,並假設所有變數都是未定義。

A

abs()

目錄 > 

abs(運算式 I)⇒運算式

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

abs(列表 I)⇒列表

$$\left| 2-3 \cdot i \right|$$

abs(矩陣 I)⇒矩陣

$$\left| z \right|$$

傳回引數的絕對值。


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

附註: 另請參考絕對值範本, 頁碼: 3。

如果引數是複數, 則傳回數字的模數。

附註: 系統會把所有未定義的變數視為實變數。

amortTbl()

目錄 > 

amortTbl(NPmt,N,I,PV, [Pmt], [FV], [PpY], [CpY], [PmtAt], [四捨五入值])⇒矩陣

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

根據一組貨幣時間價值引數, 傳回攤銷表矩陣的攤銷函數。

NPmt 是要加入表中的付款期數。表格以第一期付款開始。

N,I,PV,Pmt,FV,PpY,CpY,PmtAt 的說明位於貨幣時間價值引數表(第頁碼:179 頁)。

- 如省略 Pmt, 則會以 $Pmt=tvmpmt(N,I,PV,FV,PpY,CpY,PmtAt)$ 為預設值。
- 如省略 FV, 則會以 $FV=0$ 為預設值。
- PpY、CpY、PmtAt 的預設值和貨幣時間價值函數相同。

四捨五入值用於指定四捨五入的小數位數。預設值 = 2。

所產生的矩陣行順序為：付款期數、支付利息金額、支付本金金額和結算餘額。

第 n 列顯示的餘額是付款 n 次後的餘額。

您可使用輸出矩陣當作其他攤銷函數的輸入，包括 $\Sigma\text{Int}()$ 和 $\Sigma\text{Prn}()$ (第頁碼：208 頁) 與 $\text{bal}()$ (第頁碼：17 頁)。

and

布林運算式 1 and 布林運算式 2 \Rightarrow 布林運算式

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

布林列表 1 and 布林列表 2 \Rightarrow 布林列表

布林矩陣 1 and 布林矩陣 2 \Rightarrow 布林矩陣

傳回真偽值或原始輸入的化簡形式。

整數 1 and 整數 2 \Rightarrow 整數

利用 **and** 功能個別位元比較兩個實際整數。系統內部會把兩個整數轉換成有正負號的 64 位元二進位數字。比較對應的位元時，如果兩個位元都是 1，則結果是 1，否則結果是 0。傳回的值代表位元結果，並會以基底模式為顯示依據。

您可以用任何整數做為基底。如果要輸入二進位或十六進位數字，則必須在前面分別加上 **0b** 或 **0h** 字首。如果沒有加上字首，就會把整數視為十進位 (以 10 為底)。

十六進位基底模式：

$$0\text{h}7\text{AC}36 \text{ and } 0\text{h}3\text{D}5\text{F} \quad 0\text{h}2\text{C}16$$

重要訊息：數字零，而非英文字母 O。

二進位基底模式：

$$0\text{b}100101 \text{ and } 0\text{b}100 \quad 0\text{b}100$$

十進位基底模式：

$$37 \text{ and } 0\text{b}100 \quad 4$$

附註： 您可輸入最長 64 位的二進位數字 (不含 0b 字首)。您可輸入最長 16 位的十六進位數字。

angle()

angle(運算式 1) \Rightarrow 運算式

傳回引數的角度，並將引數解譯成複數。

度數角模式：

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

附註：系統會把所有未定義的變數視為實變數。

梯度角模式：

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

弧度角模式：

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

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

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

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

angle(列表 I)⇒列表

angle(列表 I)⇒矩陣

傳回列表 I 或矩陣 I 中各元素的角度列表或矩陣，將每個元素解譯成複數，代表二維直角座標點。

ANOVA

ANOVA 列表 1, 列表 2[, 列表 3, ..., 列表 20][, 旗標]

執行單因子變異數分析，比較二至 20 個母群體的平均值。*stat.results* 變數會儲存結果摘要(請參閱第 頁碼: 162 頁)。

旗標=0 時用於資料，旗標=1 時用於統計

輸出變數	說明
stat.F	F 統計值
stat.PVal	無效假說被否定之最低意義標準
stat.df	群組自由度
stat.SS	群組的平方和
stat.MS	群組的均方
stat.dfError	誤差的自由度
stat.SSError	誤差的平方和

輸出變數	說明
stat.MSError	誤差的均方
stat.sp	合併標準差
stat.xbarlist	所輸入列表的平均值
stat.CLowerList	每個輸入列表平均值的 95% 信賴區間
stat.CUpperList	每個輸入列表平均值的 95% 信賴區間

ANOVA2way

目錄 > 

ANOVA2way 列表1,列表2[,列表3,...,列表10][,levRow]

計算雙因子變異數分析，比較二至 10 個母群體的平均值。*stat.results* 變數會儲存結果摘要(請參閱第 頁碼: 162 頁)。

LevRow=0 時用於區集

LevRow=2,3,...,*Len*-1 時用於雙因子，其中 *Len*=長度(列表1)=長度(列表2)=...=長度(列表10) 與 $Len / LevRow \in \{2,3,\dots\}$

輸出: 區集設計 (Block Design)

輸出變數	說明
stat.F	F 統計量
stat.PVal	無效假說被否定之最低意義標準
stat.df	行因子的自由度
stat.SS	行因子的平方和
stat.MS	行因子的均方
stat.FBlock	F 統計量
stat.PValBlock	無效假說被否定之最低機率
stat.dfBlock	因子的自由度
stat.SSBlock	因子的平方和
stat.MSBlock	因子的均方
stat.dfError	誤差的自由度

輸出變數	說明
stat.SSError	誤差的平方和
stat.MSError	誤差的均方
stat.s	誤差的標準差

行因子輸出

輸出變數	說明
stat.Fcol	F 統計量
stat.PValCol	行因子的機率值
stat.dfCol	行因子的自由度
stat.SSCol	行因子的平方和
stat.MSCol	行因子的均方

列因子輸出

輸出變數	說明
stat.FRow	F 統計量
stat.PValRow	列因子的機率值
stat.dfRow	列因子的自由度
stat.SSRow	列因子的平方和
stat.MSRow	列因子的均方

相互影響的輸出

輸出變數	說明
stat.FInteract	F 統計量
stat.PValInteract	相互影響的機率值
stat.dfInteract	相互影響的自由度
stat.SSInteract	相互影響的平方和
stat.MSInteract	相互影響的均方

誤差的輸出

輸出變數	說明
stat.dfError	誤差的自由度
stat.SSError	誤差的平方和
stat.MSError	誤差的均方
s	誤差的標準差

Ans

鍵

Ans⇒值

56 56

傳回最近的運算式求值的結果。

56+4 60

60+4 64

approx()

目錄 >

approx(運算式 I)⇒運算式

若可能則以包含十進位值的運算式，傳回引數的求值，不管目前的**自動或近似值**模式。

相當於輸入引數並按 。

$\text{approx}\left(\frac{1}{3}\right)$ 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(列表 I)⇒列表

approx(矩陣 I)⇒矩陣

傳回列表或矩陣，若可能則將所有元素都求值成十進位值。

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

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

▶approxFraction()

目錄 >

Expr ▶**approxFraction([Tol])**⇒運算式

列表 ▶**approxFraction([Tol])**⇒列表

矩陣 ▶**approxFraction([Tol])**⇒矩陣

使用 *Tol* 為容許值，傳回輸入量的分數值。如果省略 *Tol* 設定，則使用 5.E-14 為容許值。

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


0.8333333333333333 ▶ **approxFraction(5.E-14)**

$\frac{5}{6}$

$\{\pi, 1.5\}$ ▶ **approxFraction(5.E-14)**

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

►approxFraction()

目錄 > 

附註：如果要從電腦鍵盤插入本函數，可輸入 `@>approxFraction(...)`。

approxRational()

目錄 > 

approxRational(*Expr*[, *Tol*]) ⇒ 運算式

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

approxRational(*列表*[, *Tol*]) ⇒ 列表

$$\text{approxRational}(\{0.2, 0.33, 4.125\}, 5 \cdot 10^{-14})$$
$$\left\{ \frac{1}{5}, \frac{33}{100}, \frac{33}{8} \right\}$$

approxRational(*矩陣*[, *Tol*]) ⇒ 矩陣

使用 *Tol* 為容許值，傳回引數的分數值。如果省略 *Tol* 設定，則使用 5.E-14 為容許值。

arccos()

請參考 $\cos^{-1}()$ ，頁碼：30。

arccosh()

請參考 $\cosh^{-1}()$ ，頁碼：31。

arccot()

請參考 $\cot^{-1}()$ ，頁碼：32。

arccoth()


請參考 $\coth^{-1}()$ ，頁碼：33。

arccsc()

請參考 $\csc^{-1}()$ ，頁碼：36。

arcsch()


請參考 $\operatorname{csch}^{-1}()$ ，頁碼：36。

arcLen()目錄 > **arcLen**(運算式 I , Var , 起點, 終點) \Rightarrow 運算式傳回 運算式 I 從起點到終點的弧長 (對 Var 變數)。

計算弧長時, 將以用函數模式定義的積分計算。

arcLen(列表 I , Var , 起點, 終點) \Rightarrow 列表傳回列表 I 中每個元素從起點到終點 (對於 Var) 的弧長列表。

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

arcsec()請參考 $\sec^{-1}()$, 頁碼: 145。**arcsech()**請參考 $\text{sech}^{-1}()$, 頁碼: 146。**arcsin()**請參考 $\sin^{-1}()$, 頁碼: 155。**arcsinh()**請參考 $\sinh^{-1}()$, 頁碼: 156。**arctan()**請參考 $\tan^{-1}()$, 頁碼: 169。**arctanh()**請參考 $\tanh^{-1}()$, 頁碼: 170。**augment()**目錄 > **augment**(列表 1 , 列表 2) \Rightarrow 列表傳回將列表 2 接到列表 1 結尾的新列表。

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

augment()

目錄 >

augment(矩陣1, 矩陣2)⇒矩陣

傳回將矩陣2加到矩陣1的新矩陣。使用 [] 字元時，矩陣的列數必須相同，並將矩陣2加到矩陣1中成為新的行。並不會更改矩陣1或矩陣2。

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

avgRC()

目錄 >

avgRC(運算式I, Var [=值] [, 步階])⇒運算式

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

avgRC(運算式I, Var [=值] [, 列表I])⇒列表

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

avgRC(列表I, Var [=值] [, 步階])⇒列表

$$\text{avgRC}(x^2-x+2, x) \quad 2 \cdot (x-0.4995)$$

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

avgRC(矩陣I, Var [=值] [, 步階])⇒矩陣

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


傳回向前差商(平均變化率)。

運算式I可以是使用者定義的函數名稱(請參考 **Func**)。

指定值時，此值會取代之前賦值的所有變數，或目前以 [] 取代的所有該變數。

Step 代表步階值。如果省略步階，則會以 0.001 為預設值。

請注意，類似函數 **centralDiff()** 使用中央差商。

bal()目錄 > 

bal(*NPmt*,*N*,*I*,*PV*, [*Pmt*], [*FV*], [*PpY*], [*CpY*], [*PmtAt*], [四捨五入值])⇒值

bal(*NPmt*,*amortTable*)⇒值

攤銷函數可計算在某次付款後的每期餘額時間表。

N,*I*,*PV*,*Pmt*,*FV*,*PpY*,*CpY*,*PmtAt* 的說明位於貨幣時間價值引數表(第頁碼: 179頁)。

NPmt 代表付款期數, 您就是要計算該期後的資料。

N,*I*,*PV*,*Pmt*,*FV*,*PpY*,*CpY*,*PmtAt* 的說明位於貨幣時間價值引數表(第頁碼: 179頁)。

- 如省略 *Pmt*, 則會以 *Pmt*=*tvmPmt* (*N*,*I*,*PV*,*FV*,*PpY*,*CpY*,*PmtAt*) 為預設值。
- 如省略 *FV*, 則會以 *FV*=0 為預設值。
- *PpY*、*CpY*、*PmtAt* 的預設值和貨幣時間價值函數相同。

四捨五入值用於指定四捨五入的小數位數。預設值 = 2。

bal(*NPmt*,*amortTable*) 可根據 *amortTable* 攤銷表計算付款第 *NPmt* 期後的餘額。*amortTable* 引數必須是 **amortTbl()** 一節所描述的矩陣形式(第頁碼: 8頁)。


附註: 另請參考 **ΣInt()** 與 **ΣPrn()**, 頁碼: 208。

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

<i>tbl</i> := amortTbl (6,6,5.75,5000,,12,12)	
--	--

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

bal (4, <i>tbl</i>)	1674.27
-----------------------------	---------

►Base2目錄 > 

整數 *I* ►**Base2**⇒整數

256► Base2	0b10000000
-------------------	------------

附註: 如果要從電腦鍵盤插入本運算子, 可輸入 @>**Base2**。

0h1F► Base2	0b11111
--------------------	---------

把整數 I 轉換成二進位數字。二進位或十六進位數字前面一定分別有 `0b` 或 `0h` 字首。數字零，而非英文字母 `O`，後面跟著 `b` 或 `h`。

`0b` 二進位數字

`0h` 十六進位數字

二進位數字最長可到 64 位。十六進位數字最長可到 16 位。

如果沒有加上字首，就會把整數 I 視為十進位(以 10 為底)。無論使用何種基底模式，都以二進位制顯示結果。

負數以「2 的補數」形式顯示。例如：

-1 顯示為

`0hFFFFFFFFFFFFFFFF` (十六進位基底模式)

`0b111...111`(64 個 1) (二進位基底模式)

-2^{63} 顯示為

`0h8000000000000000` (十六進位基底模式)

`0b100...000`(63 個零) (二進位基底模式)

如果您輸入的十進位整數，超出具正負號 64 位元二進位格式的範圍，系統會以對稱模數運算法使數值落入適當範圍。請看以下超出範圍的數值範例。

2^{63} 變成 -2^{63} ，並顯示為

`0h8000000000000000` (十六進位基底模式)

`0b100...000`(63 個零) (二進位基底模式)

2^{64} 變成 0，並顯示為

`0h0` (十六進位基底模式)

`0b0` (二進位基底模式)

▶Base2

$-2^{63} - 1$ 變成 $2^{63} - 1$ ，並顯示為

0h7FFFFFFFFFFFFFFF (十六進位基底模式)

0b111...111(64 個 1) (二進位基底模式)

▶Base10

整數 I ▶Base10 ⇒ 整數

0b10011 ▶Base10 19

附註：如果要從電腦鍵盤插入本運算子，可輸入 @▶Base10。

0h1F ▶Base10 31

把整數 I 轉換成十進位(以 10 為底)數字。輸入二進位或十六進位數字時，前面一定分別要有 0b 或 0h 字首。

0b 二進位數字

0h 十六進位數字

數字零，而非英文字母 O，後面跟著 b 或 h。

二進位數字最長可到 64 位。十六進位數字最長可到 16 位。

如果沒有加上字首，就會把整數 I 視為十進位。無論使用何種基底模式，都以十進位制顯示結果。

▶Base16

整數 I ▶Base16 ⇒ 整數

256 ▶Base16 0h100

附註：如果要從電腦鍵盤插入本運算子，可輸入 @▶Base16。

0b111100001111 ▶Base16 0hFOF

把整數 I 轉換成十六進位數字。二進位或十六進位數字前面一定分別有 0b 或 0h 字首。

0b 二進位數字

0h 十六進位數字

數字零，而非英文字母 O，後面跟著 b 或 h。

二進位數字最長可到 64 位。十六進位數字最長可到 16 位。

如果沒有加上字首，就會把整數 I 視為十進位(以 10 為底)。無論使用何種基底模式，都以十六進位制顯示結果。

如果您輸入的十進位整數太大，超出具正負號 64 位元二進位格式的範圍，系統會以對稱模數運算法使數值落入適當範圍。如需詳細資訊，請參閱

►Base2, 頁碼: 17。

binomCdf()

binomCdf(n,p)⇒列表

binomCdf(n,p , 下限, 上限)⇒數字(若下限和上限是數字)或列表(若下限和上限是列表)

binomCdf(n,p , 上限)for $P(0 \leq X \leq \text{上限})$ ⇒數字(若上限是數字)或列表(若上限是列表)

計算 n 次試驗和每次試驗成功機率 p 的離散二項分布累積機率。

對於 $P(X \leq \text{上限})$, 請設定 下限=0

binomPdf()

binomPdf(n,p)⇒列表

binomPdf($n,p,XVal$)⇒數字(若 $XVal$ 是數字)或列表(若 $XVal$ 是列表)

計算 n 次試驗和每次試驗成功機率 p 的離散二項分布機率。

C

ceiling() 無條件進入法()

ceiling(運算式 I)⇒整數

ceiling(.456)

1.

傳回大於 \geq 引數且最接近的整數。

引數可以是實數也可以是複數。

附註：另請參考 floor()。

ceiling(列表 I) ⇒ 列表

ceiling(矩陣 I) ⇒ 矩陣

傳回各元素無條件進入後的列表或矩陣。

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

centralDiff()

centralDiff(運算式 I , Var [= 值], [步階]) ⇒ 運算式

centralDiff(運算式 I , Var [, 步階]) | Var = 值 ⇒ 運算式

centralDiff(運算式 I , Var [= 值], [列表]) ⇒ 列表

centralDiff(列表 I , Var [= 值], [步階]) ⇒ 列表

centralDiff(矩陣 I , Var [= 值], [步階]) ⇒ 矩陣

使用中央差商公式傳回數值導數。

指定值時，此值會取代之前賦值的所有變數，或目前以 [] 取代的所有該變數。

Step 代表步階值。如果省略步階，則會以 0.001 為預設值。

使用列表 I 或矩陣 I 時，運算作業會對應到列表中的值或矩陣元素。

附註：另請參考 avgRC() 與 d()。

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

cFactor(運算式 I, Var) ⇒ 運算式

cFactor(列表 I, Var) ⇒ 列表

cFactor(矩陣 I, Var) ⇒ 矩陣

cFactor(運算式 I): 傳回對所有變數除以公分母進行因式分解的運算式 I。

對運算式 I 進行因式分解時，即使可能導入新的非實數，仍會儘量朝線性有理因式進行。如果您想進行對多個變數的因式分解，即可使用這個選項。

cFactor(運算式 I, Var): 傳回對 Var 進行因式分解的運算式 I。

對運算式 I 進行因式分解時，即使可能導入無理常數或在其他變數中無理的子運算式，仍會儘量朝在 Var 中線性的因式進行。

因式和其中的項以 Var 為主變數進行排序。系統將匯集各因式中 Var 的類似乘幂。如果您只需要對 Var 進行因式分解，而且您可接受其他變數中有無理式，以便增加對 Var 的因式分解，請加入該變數。可能會對其他變數進行偶發因式分解。



如果在自動或近似值模式中設定為自動，則加入 Var 時，也能在無法以內建函數來明確表達簡潔的無理係數時，以浮點係數計算近似值。即使只有一個變數，加入 Var 也可能有更完整的因式分解。

附註: 另請參考 factor()。

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

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

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


若要看到完整結果，請按  與  移動游標。

char(Integer) ⇒ 字元

傳回計算機字元集中屬於 Integer 編號字元的字元字串。Integer 的有效範圍是 0–65535。

$\text{char}(38)$	"&"
$\text{char}(65)$	"A"

charPoly()

目錄 > 

charPoly(方陣,Var)⇒多項式

charPoly(方陣,Expr)⇒多項式

charPoly(方陣1,矩陣2)⇒多項式

傳回方陣的特徵多項式。 $n \times n$ 矩陣 A 的特徵多項式(以 $p_A(\lambda)$ 代表)定義如下:


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

I 代表 $n \times n$ 單位矩陣。

方陣1 和方陣2 的維度必須相同。

$m := \begin{bmatrix} 1 & 3 & 0 \\ 2 & -1 & 0 \\ -2 & 2 & 5 \end{bmatrix}$	$\begin{bmatrix} 1 & 3 & 0 \\ 2 & -1 & 0 \\ -2 & 2 & 5 \end{bmatrix}$
charPoly(m,x)	$-x^3+5 \cdot x^2+7 \cdot x-35$
charPoly(m,x ² +1)	$-x^6+2 \cdot x^4+14 \cdot x^2-24$
charPoly(m,m)	0

χ^2 2way

目錄 > 

χ^2 2way obsMatrix

chi22way obsMatrix


計算 χ^2 檢定, 以用於 *obsMatrix* 觀察矩陣中計數的雙向交叉表之結合。

stat.results 變數會儲存結果摘要(請參閱第頁碼: 162 頁)。

如要了解矩陣中有空元素時的影響, 請參考“空元素”, 頁碼: 219。

輸出變數	說明
stat. χ^2	卡方檢定統計: $\text{sum}(\text{觀察} - \text{預期})^2 / \text{預期}$
stat.PVal	無效假說被否定之最低意義標準
stat.df	卡方檢定統計的自由度
stat.ExpMat	預期基本計數表的矩陣, 假定無效假說
stat.CompMat	基本卡方統計貢獻的矩陣

χ^2 Cdf()

目錄 > 

χ^2 Cdf(下限,上限,df)⇒數字(若下限和上限是數字)或列表(若下限和上限是列表)

chi2Cdf(下限,上限,df)⇒數字(若下限和上限是數字)或列表(若下限和上限是列表)

針對指定自由度 df 的下限和上限之間，計算 χ^2 分布機率。

對於 $P(X \leq \text{上限})$ ，請設定 下限=0

如要了解列表中有空元素時的影響，請參考 $\text{\textcircled{R}}$ 空元素 $\text{\textcircled{R}}$ ，頁碼：219。

χ^2 GOF 觀察列表, 預期列表, df

chi2GOF 觀察列表, 預期列表, df

執行檢定，確認資料樣本的確來自符合特定分布的母群體。觀察列表是計數的列表，必須包含整數。stat.results 變數會儲存結果摘要(請參閱第 頁碼：162 頁)。

如要了解列表中有空元素時的影響，請參考 $\text{\textcircled{R}}$ 空元素 $\text{\textcircled{R}}$ ，頁碼：219。

輸出變數	說明
stat. χ^2	卡方檢定統計: $\text{sum}(\text{觀察} - \text{預期})^2 / \text{預期}$
stat.PVal	無效假說被否定之最低意義標準
stat.df	卡方檢定統計的自由度
stat.ComplList	基本卡方統計貢獻

χ^2 Pdf($XVal, df$) \Rightarrow 數字(若 $XVal$ 是數字) 或列表(若 $XVal$ 是列表)

chi2Pdf($XVal, df$) \Rightarrow 數字(若 $XVal$ 是數字) 或列表(若 $XVal$ 是列表)

針對指定自由度 df ，計算 χ^2 分布在指定 $XVal$ 值的機率密度函數(pdf)。

如要了解列表中有空元素時的影響，請參考 $\text{\textcircled{R}}$ 空元素 $\text{\textcircled{R}}$ ，頁碼：219。

ClearAZ

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

清除目前問題空間中的所有單字元變數。

如果一個以上的變數已遭鎖定，本指令會顯示錯誤訊息，並只刪除未鎖定的變數。請參考 **unLock**，頁碼：182。

ClrErr

ClrErr

若需 **ClrErr** 的參考範例，請參考第頁碼：176 頁 **Try** 指令下的例 2。

清除錯誤狀態，並將系統變數 *errCode* 設定成零。

Try...Else...EndTry 段的 **Else** 子句應該使用 **ClrErr** 或 **PassErr**。若要處理或忽略錯誤，請使用 **ClrErr**。若不知如何處理錯誤，請用 **PassErr** 送到下一個錯誤處理程式。若已經沒有待處理的 **Try...Else...EndTry** 錯誤處理程式，則錯誤對話方塊將顯示為正常。

附註：另請參考第頁碼：121 頁的 **PassErr** 和第頁碼：176 頁的 **Try**。

輸入範例的注意事項：關於輸入多行程式和函數定義的說明，請參閱產品手冊中的「計算工具」章節。

colAugment()

colAugment(矩陣1, 矩陣2)⇒矩陣

傳回將矩陣2加到矩陣1的新矩陣。矩陣的行數必須相同，並將矩陣2加到矩陣1中成為新的列。並不會更改矩陣1或矩陣2。

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

colDim()

colDim(矩陣)⇒運算式

傳回矩陣包含的行數。

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

附註：另請參考 **rowDim()**。

colNorm(矩陣)⇒運算式

傳回矩陣中各行元素取絕對值後加總之最大值。

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

附註：不得有未定義的矩陣元素。另請參考 rowNorm()。

comDenom()

comDenom(運算式 I[,Var])⇒運算式

comDenom(列表 I[,Var])⇒列表

comDenom(矩陣 I[,Var])⇒矩陣

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

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

comDenom(運算式 I): 傳回完全展開分子除以完全展開分母的簡化比例。

comDenom(運算式 I,Var): 傳回依 Var 展開的分子與分母簡化比例。項和其中的因式以 Var 為主變數進行排序。系統將匯集 Var 的類似乘幕。系統可能對匯集的係數進行偶發因式分解。若與省略 Var 相比，此功能通常能節省時間、記憶體和螢幕空間，並使運算式更容易懂。此函數也能讓根據運算結果進行的後續運算作業更快，而且比較不容易用完記憶體。

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

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

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

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

若 Var 未出現在運算式 I 中，則

comDenom(運算式 I,Var): 傳回分子與分母皆未展開的簡化比例。這類結果通常可節省更多時間、記憶體和螢幕空間。這類部份因式分解的結果也能讓根據運算結果進行的後續運算作業快得多，而且更不容易用完記憶體。

即使沒有分母，用 comden 函數進行部份因式分解的速度也通常較快(若 factor() 太慢或是會用完記憶體)。

Define comden(exprn)=comDenom(exprn,abc)
Done


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

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

$$1234 \cdot x \cdot (x \cdot y + 2) \cdot (y^2-1)$$

提示：輸入此 comden() 函數定義，平常即可以此函數當作 comDenom() 與 factor() 以外的其他選擇。

completeSquare()

目錄 > 

completeSquare(ExprOrEqn, Var)⇒運算式或方程式

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

completeSquare(ExprOrEqn, Var^Power)⇒運算式或方程式

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

completeSquare(ExprOrEqn, Var1, Var2 [...])⇒運算式或方程式

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

completeSquare(ExprOrEqn, {Var1, Var2 [...]})⇒運算式或方程式

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

將形式 $a\cdot x^2+b\cdot x+c$ 的二次多項式運算式轉換成形式 $a\cdot(x-h)^2+k$

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

- 或 -

將形式 $a\cdot x^2+b\cdot x+c=d$ 的二次方程式轉換成形式 $a\cdot(x-h)^2=k$


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

第一個引數必須是相對於第二個引數而言為標準形式的二次運算式或方程式。

第二個引數必須是單一單變量項或提升至有理數次方的單一單變量項，例如， x 、 y^2 或 $z^{(1/3)}$ 。

第三及第四個語法為完成相關變數 $Var1, Var2$ [...] 的平方。

conj()

目錄 > 

conj(運算式 I)⇒運算式

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

conj(列表 I)⇒列表

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

conj(矩陣 I)⇒矩陣

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

傳回引數的共軛複數。

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

附註：系統會把所有未定義的變數視為實變數。

constructMat**(Expr, Var1, Var2, numRows, numCols)** ⇒ 矩陣

根據引數傳回矩陣。

Expr 代表以 *Var1* 與 *Var2* 為變數的運算式。所產生矩陣中的各元素，是由對 *Var1* 與 *Var2* 求值的每個增量值之 *Expr* 而來。

Var1 是自動由第 1 列到第 *numRows* 列增量而成。*Var2* 是在每一列中從第 1 行到第 *numCols* 行增量而成。

constructMat($\frac{1}{i+j}, i, j, 3, 4$)	$\frac{1}{2}$	$\frac{1}{3}$	$\frac{1}{4}$	$\frac{1}{5}$
	$\frac{1}{3}$	$\frac{1}{4}$	$\frac{1}{5}$	$\frac{1}{6}$
	$\frac{1}{4}$	$\frac{1}{5}$	$\frac{1}{6}$	$\frac{1}{7}$
	$\frac{1}{5}$	$\frac{1}{6}$	$\frac{1}{7}$	$\frac{1}{8}$

CopyVar**CopyVar** *Var1, Var2***CopyVar** *Var1, Var2*.

CopyVar *Var1, Var2* 會把 *Var1* 變數的值複製到 *Var2* 變數，必要時會建立 *Var2*。*Var1* 變數必須有值。













若 *Var1* 是既有使用者定義函數的名稱，則將該函數的定義複製到 *Var2* 函數。您必須定義 *Var1* 函數。

Var1 必須符合變數命名規則，或必須是能化簡成符合規則之變數名稱的間接取值運算式。

CopyVar *Var1, Var2*. 可把 *Var1* 變數群組的所有成員複製到 *Var2*. 群組，必要時會建立 *Var2*。

Var1. 必須是既有變數群組的名稱(例如 *stat.mn* 統計結果)，或是用 **LibShortcut()** 函數建立的變數。若 *Var2*. 已經存在，本指令就會取代兩個群組的所有共同成員，並加入還不存在的成員。若一個以上的 *Var2*. 成員已遭鎖定，則 *Var2*. 的所有成員都會保留原樣。

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

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

corrMat()**corrMat**(列表1, 列表2[, ..., 列表20])

計算擴增矩陣 [列表1, 列表2, ..., 列表20] 的相關矩陣。

▶cos

Expr ▶cos

附註：如果要從電腦鍵盤插入本運算子，可輸入 @>cos。

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

以餘弦表示 Expr。這是顯示方式轉換運算子，只能用於輸入線末尾。

▶cos 會約化以下所有乘幂 sin(...) modulo 1-cos(...) ^2 讓 cos(...) 剩餘乘幂的指數都在 (0, 2) 範圍內。因此若且唯若 sin(...) 出現在指定運算式且是偶數乘幂時，結果才會沒有 sin(...)。

附註：度數角或梯度角模式不支援本轉換運算子。使用前請確認是否已將角度模式設成弧度角，而且 Expr 中並無明確參照度數角或梯度角。

cos()

cos(運算式 I) ⇒ 運算式

度數角模式：

cos(列表 I) ⇒ 列表

$$\frac{\cos\left(\frac{\pi}{4}\right)}{\cos(45)} = \frac{\frac{\sqrt{2}}{2}}{\frac{\sqrt{2}}{2}}$$

cos(運算式 I) 可將引數的餘弦傳回為運算式。

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

cos(列表 I)：傳回列表 I 中各元素之餘弦的列表。

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

附註：系統會根據目前的角度模式設定，將引數解譯為度數角、梯度角或弧度角。您可使用 °、G 或 R 來暫時覆寫角度模式。

梯度角模式：

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

弧度角模式：

$\cos\left(\frac{\pi}{4}\right)$	$\frac{\sqrt{2}}{2}$
$\cos(45^\circ)$	$\frac{\sqrt{2}}{2}$

cos(方陣 I) ⇒ 方陣

傳回方陣 I 的矩陣餘弦。這和計算各元素的餘弦不同。

純量函數 f(A) 對方陣 I (A) 進行運算時，計算結果時所用的算法如下：

計算 A 的特徵值 (λ_i) 和特徵向量 (V_i)。

方陣 I 必需可對角化。另外不能有尚未賦值的符號變數。

建立矩陣：

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

則 $A = X B X^{-1}$ 且 $f(A) = X f(B) X^{-1}$ 。例如 $\cos(A) = X \cos(B) X^{-1}$ ，其中：

$\cos(B) =$

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

執行所有計算作業時，都使用浮點運算法。

弧度角模式：

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

cos⁻¹()

cos⁻¹(運算式 I) ⇒ 運算式

度數角模式：

cos⁻¹(列表 I) ⇒ 列表

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

cos⁻¹(運算式 I) 可將餘弦是運算式 1 的角度傳回為運算式。

梯度角模式：

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

$\cos^{-1}()$

 鍵

$\cos^{-1}(\text{列表 } I)$: 傳回列表 I 各元素之反餘弦的列表。

附註: 系統會根據目前的角度模式設定, 將結果傳回為度數角、梯度角或弧度角。

附註: 如果要從鍵盤插入本函數, 可輸入 $\arccos(\dots)$ 。

$\cos^{-1}(\text{方陣 } I) \Rightarrow \text{方陣}$

傳回方陣 I 的矩陣反餘弦。這和計算各元素的反餘弦不同。若要了解計算方式, 請參考 $\cos()$ 。

方陣 I 必需可對角化。結果一定會包含浮點數。

弧度角模式:


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

弧度角模式與直角複數格式:

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

若要看到完整結果, 請按 \blacktriangle , 然後使用 \blacktriangleleft 與 \blacktriangleright 移動游標。

$\cosh()$

 目錄 > 

$\cosh(\text{運算式 } I) \Rightarrow \text{運算式}$

$\cosh(\text{列表 } I) \Rightarrow \text{列表}$

$\cosh(\text{運算式 } I)$ 可將引數的雙曲餘弦傳回為運算式。

$\cosh(\text{列表 } I)$: 傳回列表 I 各元素之雙曲餘弦的列表。

$\cosh(\text{方陣 } I) \Rightarrow \text{方陣}$

傳回方陣 I 的矩陣雙曲餘弦。這和計算各元素的雙曲餘弦不同。若要了解計算方式, 請參考 $\cos()$ 。

方陣 I 必需可對角化。結果一定會包含浮點數。


度數角模式:

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

弧度角模式:

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

$\cosh^{-1}()$

 目錄 > 

$\cosh^{-1}(\text{運算式 } I) \Rightarrow \text{運算式}$

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

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

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

cosh⁻¹(運算式 I) 可將引數的反雙曲餘弦傳回為運算式。

cosh⁻¹(列表 I): 傳回列表 I 各元素之反雙曲餘弦的列表。

附註: 如果要從鍵盤插入本函數, 可輸入 arccosh(...)

cosh⁻¹(方陣 I) ⇒ 方陣

傳回方陣 I 的矩陣反雙曲餘弦。這和計算各元素的反雙曲餘弦不同。若要了解計算方式, 請參考 cos()。

方陣 I 必需可對角化。結果一定會包含浮點數。

弧度角模式與直角複數格式:

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

$$\begin{bmatrix} 2.52503+1.73485\cdot i & -0.009241-1.49086\cdot i \\ 0.486969-0.725533\cdot i & 1.66262+0.623491\cdot i \\ -0.322354-2.08316\cdot i & 1.26707+1.79018\cdot i \end{bmatrix}$$

若要看到完整結果, 請按 \blacktriangle , 然後使用 \blacktriangleleft 與 \blacktriangleright 移動游標。

cot()

 鍵

cot(運算式 I) ⇒ 運算式

度數角模式:

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

cot(列表 I) ⇒ 列表

傳回 運算式 I 的餘切, 或傳回列表 I 中各元素之餘切的列表。

梯度角模式:

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

附註: 系統會根據目前的角度模式設定, 將引數解譯為度數角、梯度角或弧度角。您可使用 °、G 或 R 來暫時覆寫角度模式。

弧度角模式:

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

cot⁻¹()

 鍵

cot⁻¹(運算式 I) ⇒ 運算式

度數角模式:

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

cot⁻¹(列表 I) ⇒ 列表

cot⁻¹()

 鍵

傳回餘切是 運算式 I 的角度，或傳回包含列表 I 中各元素之反餘切的列表。

附註：系統會根據目前的角度模式設定，將結果傳回為度數角、梯度角或弧度角。

附註：如果要從鍵盤插入本函數，可輸入 **arccot(...)**。


梯度角模式：

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

弧度角模式：

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

coth()

目錄 > 

coth(運算式 I) ⇒ 運算式


$$\frac{\text{coth}(1.2)}{\quad\quad\quad} = 1.19954$$

coth(列表 I) ⇒ 列表

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

傳回 運算式 I 的雙曲餘切，或傳回列表 I 中各元素之雙曲餘切的列表。

coth⁻¹()

目錄 > 

coth⁻¹(運算式 I) ⇒ 運算式

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


coth⁻¹(列表 I) ⇒ 列表

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

傳回 運算式 I 的反雙曲餘切，或傳回包含列表 I 中各元素之反雙曲餘切的列表。

附註：如果要從鍵盤插入本函數，可輸入 **arccoth(...)**。

count()

目錄 > 

count(值 I 或列表 I [, 值 2 或列表 2 [...]])
⇒ 值

$$\frac{\text{count}(2,4,6)}{\quad\quad\quad} = 3$$

$$\frac{\text{count}(\{2,4,6\})}{\quad\quad\quad} = 3$$

傳回引數 (會求值成數值) 中所有元素的累積個數。

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

每個引數都可以是運算式、值、列表或矩陣。您可混合資料類型，並使用不同長度的引數。

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

系統會對列表、矩陣或儲存格範圍的各元素求值，判斷是否應該加入個數之計算。

您可在 Lists & Spreadsheet 應用程式中用儲存格範圍取代任何引數。

空元素會被忽略。如需空元素的詳細資訊，請參考第 頁碼：219 頁。

在最後一例中只會計入 $1/2$ 和 $3+4*i$ 。系統並不會把剩下的引數（假設未定義 x ）求值成數值。

countif()

countif(列表, 準則) ⇒ 值

傳回列表中符合指定準則之所有元素的累積個數。

準則可以有以下形式：

- 值、運算式、字串。例如 **3** 代表只計算列表中可化簡成 3 這個值的元素個數。
- 包含 ? 符號當作各元素預留位置的布林運算式。例如 **?<5** 代表只計算列表中 small 於 5 的元素個數。

您可在 Lists & Spreadsheet 應用程式中用儲存格範圍取代列表。

列表中的空元素會遭到忽略。如需空元素的詳細資訊，請參考第 頁碼：219 頁。

附註：另請參考第 頁碼：166 頁的 **sumif()** 和第 頁碼：70 頁的 **frequency()**。

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

計算等於 3 的元素之個數。

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

計算等於「def」的元素之個數。

countif({x^2,x^-1,1,x,x^2},x) 1

計算等於 x 的元素之個數；本例假設您未定義 x 變數。

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

計算 1 和 3 的個數。

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

計算 3、5、7 的個數。

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

計算 1、3、7、9 的個數。

cPolyRoots()

目錄 >

cPolyRoots(Poly,Var)⇒列表

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

cPolyRoots(係數表)⇒列表

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

第一個語法 **cPolyRoots(Poly,Var)** 可傳回 *Poly* 多項式的複數根列表(對 *Var*)。

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

Poly 必須是一個變數的多項式。

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

第二個語法 **cPolyRoots(係數表)** 可傳回係數表中係數的複數根列表。

附註: 另請參考 **polyRoots()**, 頁碼: 125。

crossP()

目錄 >

crossP(列表1,列表2)⇒列表

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

將列表1和列表2的向量外積傳回為列表。

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

列表1和列表2的維數必須相同,且必須是2維或3維。

crossP(向量1,向量2)⇒向量

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

所傳回的列向量或行向量(依引數而定),是向量1和向量2的向量外積(叉積)。

向量1和向量2必須同為列向量或同為行向量。兩個向量的維數必須相同,且必須是2維或3維。

csc()

鍵

csc(運算式1)⇒運算式

度數角模式:

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

csc(列表1)⇒列表

梯度角模式:

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

傳回運算式1的餘割,或傳回包含列表1中各元素之餘割的列表。

弧度角模式:

csc()

trig 鍵

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

csc⁻¹()

trig 鍵

csc⁻¹(運算式 I) ⇒ 運算式

度數角模式:

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

csc⁻¹(列表 I) ⇒ 列表傳回餘割是 *運算式 I* 的角度，或傳回包含 *列表 I* 中各元素之反餘割的列表。


梯度角模式:

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

附註: 系統會根據目前的角度模式設定，將結果傳回為度數角、梯度角或弧度角。

弧度角模式:

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

附註: 如果要從鍵盤插入本函數，可輸入 **arccsc(...)**。**csch()**目錄 > **csch(運算式 I) ⇒ 運算式**

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

csch(列表 I) ⇒ 列表傳回 *運算式 I* 的雙曲餘割，或傳回 *列表 I* 中各元素之雙曲餘割的列表。

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

csch⁻¹()目錄 > **csch⁻¹(運算式 I) ⇒ 運算式**

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

csch⁻¹(列表 I) ⇒ 列表傳回 *運算式 I* 的雙曲餘割，或傳回 *列表 I* 中各元素之雙曲餘割的列表。

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

附註: 如果要從鍵盤插入本函數，可輸入 **arccsch(...)**。

cSolve()

cSolve(運算式, Var) ⇒ 布林運算式

cSolve(運算式, Var=猜測值) ⇒ 布林運算式

cSolve(不等式, Var) ⇒ 布林運算式

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

傳回 *Var* 的方程式或不等式之可能複數解。目的是得出所有可能的實數解和非實數解。即使方程式是實數，**cSolve()** 仍允許非實數之答案在複數表示方式下之實數格式。

雖然系統處理結尾不是底線 () 的所有未定義變數時，會將其視為實變數，但 **cSolve()** 仍可解出複數解的多項式方程式。

即使目前的值域是實數，**cSolve()** 仍會在解題時將值域暫時設定成複數。在複數值域中，有奇分母的分數幕會使用主枝，而非實數分枝。因此從 **solve()** 到方程式並與這類分數幕有關的解，不一定是 **cSolve()** 這些解的子集合。

cSolve() 從真值之符號運算法開始。**cSolve()** 必要時也會用迭代近似複數多項式的因式分解。

附註：另請參考 **cZeros()**、**solve()**、**zeros()**。

附註：若方程式是含 **abs()**、**angle()**、**conj()**、**real()** 或 **imag()** 這類函數的非多項式，則應將底線字元 (按 **ctrl** **↵**) 加到 *Var* 末尾。依預設，系統將變數視為實數值。

若您使用 *var_* 格式，則系統會將變數視為複數。

您也應該以 *var_* 格式，用於方程式中可能有非實數值的其他變數。否則可能會出現意外結果。

cSolve(Eqn1 and Eqn2 [and...], VarOrGuess1, VarOrGuess2 [, ...]) ⇒ 布林運算式

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

準確到小數點後第二位的顯示位數模式：

exact(cSolve($x^5+4\cdot x^4+5\cdot x^3-6\cdot x-3=0,x$)))	
$x\cdot(x^4+4\cdot x^3+5\cdot x^2-6)=3$	
cSolve(Ans,x)	
$x=-1.11+1.07\cdot i$ or $x=-1.11-1.07\cdot i$ or $x=-2.9$	

若要看到完整結果，請按 **▲**，然後使用 **◀** 與 **▶** 移動游標。

cSolve(conj(z_)=1+i,z_)	$z_=-1-i$
-------------------------	-----------

cSolve(聯立方程式, *VarOrGuess1*,
VarOrGuess2 [, ...])⇒布林運算式

傳回聯立代數方程式的可能複數解。
每個 *VarOrGuess* 代表應求解的變數。

您也可以指定或猜測一個變數的初始
值。所有 *VarOrGuess* 的格式必須如
下：

變數

- 或 -

變數 = 實數或非實數

舉例來說, x 和 $x=3+i$ 都有效。

若所有方程式都是多項式, 且若您 [並未]指定或猜測任何初始值, 則 **cSolve()** 會以 Grobner/Buchberger 詞彙的消去法, 試圖求得**所有**複數解。

複數解可包括實數解和非實數解, 如右例所示。

聯立多項式方程式可包含無具體值的額外變數(但表現出稍後可代入的指定數值)。

您也可以加入方程式中未出現的求解變數。從這些解能看出求解系可包含 ck 格式的任意常數, 其中 k 是從 1 到 255 的整數下標。

以多項式系統而言, 求解變數的列出順序對於計算時間或記憶體用量可能影響甚大。若您最初的選擇用完記憶體或耗費太多時間, 請重排方程式或 *VarOrGuess* 列表中的變數。

附註: 以下範例使用底線字元(按 **ctrl** **⏏**), 以便將變數視為複數。

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

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

若要看到完整結果, 請按 **▲**, 然後使用 **◀**與**▶**移動游標。


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

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

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

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

cSolve()

目錄 > 

若您不加入任何猜測值，且任何方程式都不是任何變數的多項式，但所有方程式都是所有求解變數的線性方程式，則 **cSolve()** 會以高斯消去法，試圖求得所有的解。




若某聯立方程式既非所有變數的多項式，亦非求解變數的線性方程式，則 **cSolve()** 利用近似迭代法最多只能求得一個解。為此，求解變數的數量必須等於方程式數量，且方程式中的其他變數必須化簡成數字。

若要求出非實數解，通常需要非實數猜測值。為了收斂，猜測值必須接近解值。


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

$$\text{cSolve}\left(e^{z_{-}}=w_{-} \text{ and } w_{-}=z_{-}^2, \{w_{-}, z_{-}\}\right)$$
$$w_{-}=0.494866 \text{ and } z_{-}=0.703467$$

$$\text{cSolve}\left(e^{z_{-}}=w_{-} \text{ and } w_{-}=z_{-}^2, \{w_{-}, z_{-}=1+i\}\right)$$
$$w_{-}=0.149606+4.8919i \text{ and } z_{-}=1.58805+1i$$

若要看到完整結果，請按 ，然後使用  與  移動游標。

CubicReg

目錄 > 

CubicReg *X*, *Y*, [*Freq*] [, 類別, 包含]]

計算 *X* 列表與 *Y* 列表的 $y = a \cdot x^3 + b \cdot x^2 + c \cdot x + d$ 三次多項式迴歸 (頻率為 *Freq*)。 *stat.results* 變數會儲存結果摘要 (請參閱第 頁碼: 162 頁)。



所有列表的維數都必須相同，包含除外。

X、*Y* 是自變數和因變數列表。

Freq 是頻率值列表 (非必要)。*Freq* 的每個元素，可用於指定各 *X*、*Y* 對應資料點的出現頻率。預設值是 1。所有元素都必須是 ≥ 0 的整數。


類別是 *X*、*Y* 對應資料的類別代號列表。

包含是一個以上類別代號的列表。只有本列表中類別代號所對應的資料項目，才會包含在計算作業中。

如要了解列表中有空元素時的影響，請參考  空元素 ，頁碼: 219。

輸出變數	說明
stat.RegEqn	迴歸方程式: $a \cdot x^3 + b \cdot x^2 + c \cdot x + d$
stat.a、stat.b、 stat.c、stat.d	迴歸係數
stat.R ²	判定係數
stat.Resid	迴歸殘差
stat.XReg	實際用於迴歸的已修改 X 列表中的資料點列表(根據 <i>Freq</i> 、類別列表、包含類別的限制)
stat.YReg	實際用於迴歸的已修改 Y 列表中的資料點列表(根據 <i>Freq</i> 、類別列表、包含類別的限制)
stat.FreqReg	<i>stat.XReg</i> 與 <i>stat.YReg</i> 對應的頻率列表

cumulativeSum()

目錄 > 

cumulativeSum(列表 I) ⇒ 列表

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

傳回列表 I (從 1 元素開始) 中各元素的累積總和列表。


cumulativeSum(矩陣 I) ⇒ 矩陣

1 2	→ mI	1 2
3 4		3 4
5 6		5 6
$\text{cumulativeSum}(mI)$		1 2
		4 6
		9 12

傳回矩陣 I 中各元素的累積總和矩陣。每個元素都是該行從上到下的累積總和。

列表 I 或矩陣 I 若有空元素，則會在得出的列表或矩陣中產生空元素。如需空元素的詳細資訊，請參考第 頁碼：219 頁。

Cycle

目錄 > 

Cycle

以下函數列出從 1 到 100 但略過 50 的整數總和。

立刻把控制權轉到目前迴圈的下一輪迴圈 (**For**、**While**、**Loop**)。

Cycle 不可用於三種迴圈結構以外 (**For**、**While**、**Loop**)。

輸入範例的注意事項：關於輸入多行程式和函數定義的說明，請參閱產品手冊中的「計算工具」章節。

Define $g()$ =Func	Done
Local $temp,i$	
$0 \rightarrow temp$	
For $i,1,100,1$	
If $i=50$	
Cycle	
$temp+i \rightarrow temp$	
EndFor	
Return $temp$	
EndFunc	
<hr/>	
$g()$	5000

►Cylind

向量 ►Cylind

附註：如果要從電腦鍵盤插入本運算子，可輸入 @>Cylind。

以圓柱座標形式 $[r, \angle\theta, z]$ 顯示列向量或行向量。

向量必須剛好包含三個元素，可以是行向量也可以是列向量。

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



cZeros()

cZeros(Expr, Var)⇒列表

傳回可以使 $Expr=0$ 的 Var 之可能實數解和非實數解列表。cZeros() 為完成作業，必須計算：

exp$list(cSolve(Expr=0,Var),Var)$。否則 cZeros() 會與 zeros() 類似。




附註：另請參考 cSolve()、solve()、zeros()。

附註：若 Expr 是含 abs()、angle()、conj()、real() 或 imag() 這類函數的非多項式，則應將底線字元(按  ) 加到 Var 末尾。依預設，系統將變數視為實數值。若您使用 var_ 格式，則系統會將變數視為複數。

準確到小數點後三位的顯示位數模式：

$$cZeros(x^5+4x^4+5x^3-6x-3,x)$$

$$\{-1.1138+1.07314 \cdot i, -1.1138-1.07314 \cdot i, 2. \}$$

若要看到完整結果，請按 ，然後使用  與  移動游標。

$$cZeros(conj(z_)-1-i,z_)$$

$$\{1-i\}$$

您也應該以 *var* 格式，用於 *Expr* 中可能有非實數值的其他變數。否則可能會出現意外結果。

cZeros{*運算式1*, *運算式2* [, ...] },
{*VarOrGuess1*, *VarOrGuess2* [, ...] } ⇒ 矩陣

傳回運算式同時是零的可能位置。每個 *VarOrGuess* 用於指定您尋找的未知數值。

您也可以指定或猜測一個變數的初始值。所有 *VarOrGuess* 的格式必須如下：

變數

- 或 -



變數 = 實數或非實數

舉例來說，*x* 和 *x=3+i* 都有效。

若所有運算式都是多項式，且您並未指定或猜測任何初始值，則 **cZeros()** 會以 Gröbner/Buchberger 詞彙的消去法，試圖求得所有複零點。

複數零點解可包括實數和非實數零點解，如右例所示。

所產生矩陣的每一列都代表一個零點，其元素順序和 *VarOrGuess* 列表相同。為擷取某列字串，請依 [row] 製作矩陣的索引。

附註： 以下範例使用底線字元 *_* (按  )，以便將變數視為複數。

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

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

擷取第 2 列：

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

聯立多項式可包含無具體值的額外變數(但表現出稍後可代入的指定數值)。

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

0	0
$-\frac{\sqrt{1-4c_-}}{4}$	$-\frac{\sqrt{1-4c_-}}{2}$
4	2
$-\frac{\sqrt{1-4c_+}}{4}$	$\frac{\sqrt{1-4c_+}}{2}$
4	2

您也可以加入運算式中未出現的未知變數。從這些零點解能看出零點系可包含 ck 格式的任意常數, 其中 k 是從 1 到 255 的整數下標。

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

0	0	$c4$
$\frac{1}{2} - \frac{\sqrt{3}}{2}i$	$\frac{1}{2} + \frac{\sqrt{3}}{2}i$	$c4$
2	2	$c4$
$\frac{1}{2} + \frac{\sqrt{3}}{2}i$	$\frac{1}{2} - \frac{\sqrt{3}}{2}i$	$c4$
2	2	$c4$

以多項式系統而言, 未知數的列出順序對於計算時間或記憶體用量可能影響甚大。若您最初的選擇用完記憶體或耗費太多時間, 請重排運算式或 *VarOrGuess* 列表中的變數。

若您不加入任何猜測值, 且任何運算式都不是任何變數的多項式, 但所有運算式都是所有未知數的一次式, 則 **cZeros()** 會以高斯消去法, 試圖求得所有零點解。

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

$\frac{e^{w_-+i}}{2}$	$\frac{e^{w_-i}}{2}$
-----------------------	----------------------

若某聯立方程式既非所有變數的多項式, 亦非未知數的線性方程式, 則 **cZeros()** 利用近似迭代法最多只能求得一個零點。為此, 未知數的數量必須等於運算式數量, 且運算式中的其他變數必須化簡成數字。

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

[0.494866 -0.703467]

若要求出非實數零點解, 通常需要非實數猜測值。為收斂, 猜測值必須接近零點解。

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

[0.149606+4.8919i 1.58805+1.54022i]

D

dbd()

dbd(日期1, 日期2)⇒值

利用實際天數法傳回日期1和日期2之間的天數。

日期1與日期2可以是標準日曆日期範圍內的數字, 也可以是數字列表。若日期1和日期2都是列表, 則長度必須相同。

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

日期1 和日期2 必須在 1950 年到 2049 年範圍內。

您可輸入兩種格式的日期。小數點的位置可區分兩種日期格式。

MM.DDYY(美國常見格式)

DDMM.YY(歐洲常見格式)

►DD

運算式1 ►DD⇒值

度數角模式：

列表1 ►DD⇒列表

$$\overline{(1.5^\circ)\blacktriangleright DD} \quad 1.5^\circ$$

矩陣1 ►DD⇒矩陣

$$\overline{(45^\circ 22' 14.3'')\blacktriangleright DD} \quad 45.3706^\circ$$

附註：如果要從電腦鍵盤插入本運算子，可輸入 @>DD。

$$\overline{\{\{45^\circ 22' 14.3'', 60^\circ 0' 0''\}\}\blacktriangleright DD} \quad \{45.3706^\circ, 60^\circ\}$$

針對以度數表達的引數，傳回十進位相等值。系統根據角度模式設定，將引數解譯為梯度角、弧度角或度數角的數字、列表或矩陣。

梯度角模式：

$$\overline{1\blacktriangleright DD} \quad \frac{9}{10}$$

弧度角模式：

$$\overline{(1.5)\blacktriangleright DD} \quad 85.9437^\circ$$

►Decimal

運算式1 ►Decimal⇒運算式

$$\overline{\frac{1}{3}\blacktriangleright Decimal} \quad 0.333333$$

列表1 ►Decimal⇒運算式

矩陣1 ►Decimal⇒運算式

附註：如果要從電腦鍵盤插入本運算子，可輸入 @>Decimal。

以十進位形式顯示引數。本運算子只能用於輸入線的末尾。

Define Var = 運算式**Define Function(Param1, Param2, ...)= 運算式**

定義變數 *Var* 或使用者定義的函數。

Param1 這類參數是將引數傳給函數時所用的預留位置。呼叫使用者定義的函數時，必須提供對應到參數的引數 (例如值或變數)。呼叫時，函數會利用所提供的引數對運算式求值。

Var 和函數不能用系統變數或內建函數或指令的名稱。

附註： 這個形式的 **Define** 與執行以下運算式相同：**運算式** → **函數** (*Param1, Param2*)。

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

區段

EndFunc**Define Program(Param1, Param2, ...)=****Prgm**

區段

EndPrgm

使用者定義的函數或程式，可利用這個形式執行整個區段的多條語句。

區段可以只是一個語句，也可以是分成多行的一連串語句。區段也可以包括運算式和指令 (例如 **If**、**Then**、**Else**、**For**)。

輸入範例的注意事項： 關於輸入多行程式和函數定義的說明，請參閱產品手冊中的「計算工具」章節。

附註： 另請參考第 頁碼：45 頁的 **Define LibPriv** 和第 頁碼：46 頁的 **Define LibPub**。

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

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

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

Define LibPriv**Define LibPriv Var = 運算式**

Define LibPriv 函數(*Param1*, *Param2*, ...)=
運算式

Define LibPriv 函數(*Param1*, *Param2*, ...)=
Func
區段
EndFunc

Define LibPriv 程式(參數1, 參數2, ...)=
Prgm
區段
EndPrgm

功能與 **Define** 相同，但只定義自訂的資料庫變數、函數或程式。自訂函數和程式並不會出現在目錄中。

附註：另請參考第 頁碼：45 頁的 **Define** 和第 頁碼：46 頁的 **Define LibPub**。

Define LibPub *Var* = 運算式

Define LibPub 函數(*Param1*, *Param2*, ...)=
運算式

Define LibPub 函數(*Param1*, *Param2*, ...)=
Func
區段
EndFunc


Define LibPub 程式(參數1, 參數2, ...)=
Prgm
區段
EndPrgm

功能與 **Define** 相同，但只定義公用的資料庫變數、函數或程式。儲存並重新整理資料庫後，公用函數和程式就會出現在目錄中。

附註：另請參考第 頁碼：45 頁的 **Define** 和第 頁碼：45 頁的 **Define LibPriv**。

deltaTmpCnv()

DelVar

目錄 > 

DelVar 變數1[, 變數2][, 變數3] ...

$2 \rightarrow a$ 2

DelVar *Var*.

$(a+2)^2$ 16

刪除記憶體中的指定變數或變數群組。

DelVar *a* Done

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

如果一個以上的變數已遭鎖定, 本指令會顯示錯誤訊息, 並只刪除未鎖定的變數。請參考 **unLock**, 頁碼: 182。

DelVar *Var*. 會刪除 *Var*. 變數群組(例如 *stat.mn* 統計結果, 或以 **LibShortcut()** 函數建立的變數)的所有成員。這個形式中 **DelVar** 指令的點(.)用途為只刪除變數群組; 簡單變數 *Var* 則不受影響。

aa.a:=45 45

aa.b:=5.67 5.67

aa.c:=78.9 78.9

getVarInfo()	<i>aa.a</i>	"NUM"	"0"
	<i>aa.b</i>	"NUM"	"0"
	<i>aa.c</i>	"NUM"	"0"

DelVar *aa.* Done

getVarInfo() "NONE"

delVoid()

目錄 > 

delVoid(列表1)⇒列表

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

傳回列表1內容的列表, 但移除所有空元素。


如需空元素的詳細資訊, 請參考第 頁碼: 219 頁。

derivative()

請參考 **d()**, 頁碼: 205。

deSolve(一階或二階ODE, Var, 因變數)
⇒ 通解

所傳回的方程式，可顯函數或隱函數指定之一階或二階常微分方程式 (ODE) 的通解。在 ODE 中：

- 使用上標符號 (按 ) 可指定因變數對自變數的一階導數。
- 使用兩個上標符號可指定對應的二階導數。

上標符號只用於 **deSolve()** 中的導數。其他情形請用 **d()**。

一階方程式的通解，包含 ck 格式的任意常數，其中 k 是從 1 到 255 的整數下標。二階方程式的解包含兩個這類常數。

若要将隱解轉成一個以上的等值顯解，請將 **solve()** 套用到隱解。

與課本或手算結果比較時請注意，不同算法會在不同階段導入任意常數，因此可能求出不同的通解。

deSolve(一階ODE and 初始條件, Var, 因變數) ⇒ 特解

傳回滿足一階ODE 和 初始條件的特解。本指令通常比求出通解、代換初值、求解任意常數，然後將該值代入通解等步驟來得簡單。

初始條件是格式如下的方程式：

因變數 (初始獨立值) = 初始相依值

初始獨立值和初始相依值可以是沒有已儲存值的變數，例如 **x0** 和 **y0**。隱微分法可協助確認隱解。

$$\begin{array}{l} \text{deSolve}(y''+2\cdot y'+y=x^2,x,y) \\ y=(c3\cdot x+c4)\cdot e^{-x}+x^2-4\cdot x+6 \\ \text{right(Ans)}\rightarrow \text{temp} \quad (c3\cdot x+c4)\cdot e^{-x}+x^2-4\cdot x+6 \\ \frac{d^2}{dx^2}(\text{temp})+2\cdot \frac{d}{dx}(\text{temp})+\text{temp}-x^2 \quad 0 \\ \text{DelVar temp} \quad \text{Done} \end{array}$$

$$\begin{array}{l} \text{deSolve}(y'=(\cos(y))^2\cdot x,x,y) \quad \tan(y)=\frac{x^2}{2}+c4 \\ \text{solve(Ans,y)} \\ y=\tan^{-1}\left(\frac{x^2+2\cdot c4}{2}\right)+n3\cdot \pi \\ \text{Ans}|c4=c-1 \text{ and } n3=0 \\ y=\tan^{-1}\left(\frac{x^2+2\cdot (c-1)}{2}\right) \end{array}$$

$$\begin{array}{l} \sin(y)=(y\cdot e^x+\cos(y))\cdot y' \rightarrow \text{ode} \\ \sin(y)=(e^x\cdot y+\cos(y))\cdot y' \\ \text{deSolve(ode and } y(0)=0,x,y) \rightarrow \text{soln} \\ \frac{-2\cdot \sin(y)+y^2}{2}=(e^x-1)\cdot e^{-x}\cdot \sin(y) \\ \text{soln}|x=0 \text{ and } y=0 \quad \text{true} \\ \text{ode}|y'=\text{impDif(soln,x,y)} \quad \text{true} \\ \text{DelVar ode,soln} \quad \text{Done} \end{array}$$

deSolve(二階ODE and 初始條件1 and 初始條件2, Var, 因變數) ⇒ 特解

所傳回的特解可滿足二階ODE, 並有因變數的特定值以及在某個點的一階導數。

初始條件1的格式如下:

因變數(初始獨立值) = 初始相依值

初始條件2的格式如下:

因變數(初始獨立值) = 初始一階導數

deSolve(二階ODE and 邊界條件1 and 邊界條件2, Var, 因變數) ⇒ 特解

傳回滿足二階ODE 而且在兩個點有指定值的特解。

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

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

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

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

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

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

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

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

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

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

det()

det(方陣[, 容許值]) ⇒ 運算式

傳回方陣的行列式值。

若任何矩陣元素的絕對值小於容許值, 則亦可將此元素視為零。只有在矩陣中包含浮點數項目, 而且不包含尚未賦值的任何符號變數時, 才會使用此容許值。其他時候都會忽略容許值。

- 若您使用 **ctrl** **enter** 或將自動或近似值模式設定成近似值, 則系統會利用浮點運算法執行計算作業。
- 若指令省略或未使用容許值, 則預設容許值的計算方式如下:

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

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

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

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

$$\begin{bmatrix} 1.E20 & 1 \\ 0 & 1 \end{bmatrix} \rightarrow \text{mat1} \quad \begin{bmatrix} 1.E20 & 1 \\ 0 & 1 \end{bmatrix}$$

$$\det(\text{mat1}) = 0$$

$$\det(\text{mat1}, .1) = 1.E20$$

diag()

目錄 >

diag(列表)⇒矩陣

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

diag(列陣)⇒矩陣

diag(行陣)⇒矩陣

所傳回的矩陣，是由引數列表或矩陣主對角線上的值所構成。

diag(方陣)⇒列陣

所傳回的列陣，包含方陣主對角線上的元素。

方陣必需是方陣。

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

dim()

目錄 >

dim(列表)⇒整數

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

傳回列表的維度。

dim(矩陣)⇒列表

以兩行列表 {列, 行} 傳回矩陣維度。

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

dim(String)⇒整數

傳回 String 字元字串包含的字元數。

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

Disp

目錄 >

Disp 運算式或字串 1[, 運算式或字串 2] ...

顯示 *Calculator* 歷史記錄中的引數。畫面會連續顯示引數，並以短空格作區分。

主要用於程式與函數，以顯示計算過程。

輸入範例的注意事項：關於輸入多行程式和函數定義的說明，請參閱產品手冊中的「計算工具」章節。

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

DispAtint, 運算式 1[, 運算式 2 ...]

...

DispAt 可讓您指定特定運算式或字串顯示在螢幕上的哪一行。

您可使用運算式來指定行號。

請注意，行號不代表整個螢幕，而是緊接在指令/程式以後的區域。

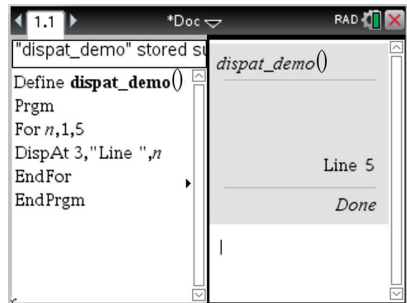
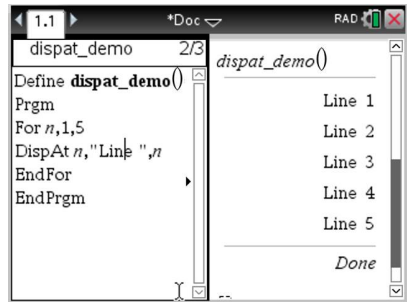
這個指令可讓程式產生類似儀表板的輸出畫面，將運算式的值或感應器的讀數更新在同一行中。

同一程式可同時使用 **DispAt** 和 **Disp**。

注意:最高行號設定為 8，因為此數值等於計算機螢幕的全螢幕行數，只要確定行中不含 2D 數學運算式即可。實際的行數必須根據螢幕所顯示的資訊內容而定。

DispAt

範例



示範範例:

Define z()=	輸出
Prgm	z()
For n,1,3	疊代 1:
DispAt 1,\"N: \",n	行 1:N:1
Disp \"Hello\"	行 2:Hello
EndFor	
EndPrgm	疊代 2:
	行 1:N:2
	行 2:Hello
	行 3:Hello
	疊代 3:
	行 1:N:3

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

錯誤狀況:

錯誤訊息	說明
DispAt 行號須介於 1 到 8 之間	運算式會計算超出 1 到 8 (含) 範圍的行號
引數太少	函數或指令缺少一個以上的引數。
沒有引數	與目前的 語法錯誤 對話方塊相同
引數太多	限制引數。錯誤與 Disp 相同。
資料類型無效	第一個引數必須是數字。
無效:DispAt 無效	"Hello World" 資料類型錯誤會在無效時發出 (如果已定義回調的話)
轉換運算子:DispAt 2_ft @>_m, "Hello World"	CAS: 發出資料類型錯誤 (如果已定義回調的話) 數字: 系統會計算轉換,若結果為有效的引數,DispAt 會在結果行上列出字串。

▶DMS

Expr ▶DMS

度數角模式:

列表 ▶DMS

 $\{45.371\}$ ▶DMS 45°22'15.6"

矩陣 ▶DMS

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

附註: 如果要從電腦鍵盤插入本運算子,可輸入 @>DMS。

將引數解譯為角度，並顯示等值的 DMS(DDDDD°MM'SS.ss") 數字。請參考第 頁碼: 212 頁的 °、'、" 說明 DMS(度、分、秒) 格式。

附註: 若用於弧度角模式，則 ►DMS 可將弧度角轉成度數角。若輸入內容後面跟著度數符號 °，則不會轉換。►DMS 只能用於輸入線的末尾。

domain()

domain(運算式 I, Var)⇒運算式

根據 Var 傳回運算式 I 的域。

domain() 可用於檢查函數的域。它必須是真實且有限的域。

受電腦代數化簡和解題系統演算法缺陷的影響，該函數具有局限性。

不管其為可顯函數還是顯示在使用者定義的變數和函數中，一些函數都不能用作 **domain()** 的引數。下例中的運算式不能化簡，就是因為 f() 是不允許的函數。

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

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

dominantTerm(運算式 I , Var [, 點]) ⇒ 運算式

dominantTerm(運算式 I , Var [, 點]) | $Var > 點$ ⇒ 運算式

dominantTerm(運算式 I , Var [, 點]) | $Var < 點$ ⇒ 運算式

傳回幕級數的主要項，代表繞著點展開的運算式 I 。 $Var = 點$ 附近在量方面成長最快的，就是主要項。($Var - 點$) 求出的乘幕可以是負指數，也可以是分數指數。這個乘幕的係數可包括 ($Var - 點$) 的對數，以及受制於 ($Var - 點$) 所有乘幕(指數符號相同)之 Var 的其他函數。

點的預設值是 0。點可以是 ∞ 也可以是 $-\infty$ 。若是這兩種情形，則主要項是擁有 Var 最大指數的項，而非 Var 的最小項。

dominantTerm(...) 在無法求解時會傳回 **dominantTerm(...)**，例如針對 $z=0$ 的 $\sin(1/z)$ 、 $z=0$ 的 $e^{-1/z}$ ，或 $z = \infty$ 或 $-\infty$ 的 e^z 這類實質奇點時。

若級數或其中一個導數在點處有跳躍不連續性，則結果可能包含 **sign(...)** 形式的子運算式，或實數展開變數的 **abs(...)**，或複數展開變數(以 [] 結尾)的 $(-1)^{\text{floor}(\dots \text{angle}(\dots))}$ 。若只需將主要項用於點其中一側的值，請在 **dominantTerm(...)** 後面加上 [$Var > 點$]、[$Var < 點$]、[$Var \geq 點$] 或 [$Var \leq 點$] 的其中一項，以便簡化結果。

dominantTerm() 分布在第一個引數列表和矩陣中。

dominantTerm() 的用途，是可針對另一個以 $Var \rightarrow 點$ 表示的運算式，求出漸近的最簡單運算式。級數的第一個非零項之度數並不明顯，而且您不想以互動或程式迴圈方式反覆猜測時，亦可使用 **dominantTerm()**。

附註：另請參考 **series()**，頁碼：149。

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

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

dotP()

目錄 >

dotP(列表1, 列表2)⇒運算式

傳回兩個列表的「點」積。

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

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

dotP(向量1, 向量2)⇒運算式

傳回兩個向量的「點」積。

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

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

兩者必須同為列向量或同為行向量。

E**e^()**

鍵

e^(運算式I)⇒運算式

傳回以 e 為底，運算式 I 為冪的指數值。

e^1	e
-------	---

$e^1.$	2.71828
--------	---------

e^{3^2}	e^9
-----------	-------

附註：另請參考 **e 指數範本**，頁碼：2。**附註：**按 可顯示 e^ (與按鍵盤上的 **E** 字元不同)。您可輸入 $re^{i\theta}$ 極座標式複數。但您只能在弧度角模式採用這種形式。若是在度數角或梯度角模式，則會導致區域錯誤。**e^(列表I)⇒列表**

傳回以 e 為底，列表 I 各元素為冪的指數值。

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

e^(方陣I)⇒方陣傳回方陣 I 的矩陣指數。這和計算以 e 為底、以各元素為冪的指數值不同。若要了解計算方式，請參考 **cos()**。

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

方陣 I 必需可對角化。結果一定會包含浮點數。

eff()

目錄 >

eff(nominalRate, CpY)⇒值可將 *nominalRate* 名目利率轉成實質年利率的財務函數(已知 *CpY* 是每年複利期數)。

$\text{eff}(5.75,12)$	5.90398
-----------------------	---------

nominalRate 必須是實數，*CpY* 必須是 > 0 的實數。

附註：另請參考 `nom()`，頁碼：113。

eigVc()

eigVc(方陣) \Rightarrow 矩陣

直角複數格式：

所傳回的矩陣，包含實數或複數方陣的特徵向量。結果中的每一行代表一個特徵值。請注意，特徵向量並非唯一，可隨任何常數因式改變。特徵向量需正規化，亦即若 $V = [x_1, x_2, \dots, x_n]$ ，則：

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

方陣首先要經由相似變換法予以平衡，直到列範數與行範數儘量接近同一值為止。接著方陣會簡化成上 Hessenberg 形式，並透過 Schur 因式分解法計算特徵向量。

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

eigVc(mI)

-0.800906	0.767947	(
0.484029	0.573804+0.052258 <i>i</i>	0.5738*
0.352512	0.262687+0.096286 <i>i</i>	0.2626

若要看到完整結果，請按 \blacktriangle ，然後使用 \blacktriangleleft 與 \blacktriangleright 移動游標。

eigVl()

eigVl(方陣) \Rightarrow 列表

直角複數格式模式：

傳回實數或複數方陣的特徵值列表。

方陣首先要經由相似變換法予以平衡，直到列範數與行範數儘量接近同一值為止。接著方陣會簡化成上 Hessenberg 形式，並從上 Hessenberg 矩陣計算特徵值。

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

eigVl(mI)

{	-4.40941,2.20471+0.763006 <i>i</i> ,2.20471-0.763006 <i>i</i>	}
---	---	---

若要看到完整結果，請按 \blacktriangle ，然後使用 \blacktriangleleft 與 \blacktriangleright 移動游標。

Else

請參考 `If`，頁碼：81。

If 布林運算式1 Then

區段1

Elseif 布林運算式2 Then

區段2

⋮

Elseif 布林運算式N Then

區段N

EndIf

⋮

輸入範例的注意事項：關於輸入多行程式和函數定義的說明，請參閱產品手冊中的「計算工具」章節。

```
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***EndFor**

請參考 For, 頁碼: 68。

EndFunc

請參考 Func, 頁碼: 71。

EndIf

請參考 If, 頁碼: 81。

EndLoop

請參考 Loop, 頁碼: 102。

EndPrgm

請參考 Prgm, 頁碼: 127。

EndTry

請參考 Try, 頁碼: 176。

EndWhile

請參考 While, 頁碼: 185。

euler()

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

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

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

使用尤拉方法解出方程組

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

其中, $\text{depVar}(\text{Var}0)=\text{depVar}0$, 區間為 [*Var0*, *VarMax*]. 傳回一個矩陣, 其中第一列定義 *Var* 輸出值, 且第二列定義位於對應 *Var* 值的第一個求解元素的值, 依此類推。

Expr 是在右邊定義常微分方程式 (ODE)。

SystemOfExpr 是在右邊的方程組, 定義 ODE 方程組 (對應於 *ListOfDepVars* 中的因變數順序)。

ListOfExpr 是在右邊的列表, 定義 ODE 方程組 (對應於 *ListOfDepVars* 中的因變數順序)。

Var 是自變數。

ListOfDepVars 是因變數列表。

{*Var0*, *VarMax*} 是一個雙元素列表, 表示函數從 *Var0* 積分至 *VarMax*。

ListOfDepVars0 是因變數初始值的列表。

VarStep 是非零數字, 使 **sign**(*VarStep*) = **sign**(*VarMax-Var0*), 傳回 $\text{Var}0+i \cdot \text{VarStep}$ 對所有的 $i=0,1,2,\dots$ 的解, 使 $\text{Var}0+i \cdot \text{VarStep}$ 位於 [*var0*, *VarMax*] (在 *VarMax* 可能不會有解)。

微分方程式:

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

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

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

若要看到完整結果, 請按 ▲, 然後使用 ◀與 ▶ 移動游標。

將以上的結果與使用 **deSolve**() 和 **seqGen**() 得到的 CAS 精確解作比較:

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

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

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

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

方程組:

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

其中, $y1(0)=2$ 及 $y2(0)=5$

$$\text{euler}\left(\begin{cases} y1 + 0.1 \cdot y1 \cdot y2 \\ 3 \cdot y2 - y1 \cdot y2 \end{cases}, t, \{y1, y2\}, \{0, 5\}, \{2, 5\}, 1\right)$$

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

eulerStep 是一個正整數 (預設值為 1), 定義在輸出值之間的尤拉步階數目。尤拉方法所用的實際步階大小是 $VarStep/eulerStep$ 。

eval ()

eval(Expr) ⇒ 字串

eval() 只有在 TI-Innovator™ Hub 程式設計指令 **Get**、**GetStr** 和 **Send** 的指令引數中才有效。軟體會對運算式 *Expr* 求值, 該結果會將 **eval()** 語句取代為字元字串。

引數 *Expr* 必須化簡成實數。

雖然 **eval()** 不會顯示其結果, 但您可以在執行指令後, 透過檢查以下任一項特殊變數來檢視求出的分享器指令字串。

iostr.SendAns
iostr.GetAns
iostr.GetAns

附註:另請參考 **Get** (頁碼: 73)、**GetStr** (頁碼: 79) 和 **Send** (頁碼: 146)。

分享器功能表

將 RGB LED 藍色元素的強度設為一半。

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

將藍色元素重設為「關閉」。

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

eval() 引數必須化簡成實數。

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

讓紅色元素淡入的程式

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

執行程式。

<i>fadein()</i>	Done
-----------------	------

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

exact()

目錄 >

exact(運算式 I [, 容許值])⇒運算式

$\text{exact}(0.25)$	$\frac{1}{4}$
----------------------	---------------

exact(列表 I [, 容許值])⇒列表

$\text{exact}(0.333333)$	$\frac{333333}{1000000}$
--------------------------	--------------------------

exact(矩陣 I [, 容許值])⇒矩陣

$\text{exact}(0.333333, 0.001)$	$\frac{1}{3}$
---------------------------------	---------------

以真值模式運算法，儘量傳回與引數等值的有理數。

容許值用於指定轉換容許值，預設值是 0 (零)。

$\text{exact}(3.5 \cdot x + y)$	$\frac{7 \cdot x}{2} + y$
---------------------------------	---------------------------

$\text{exact}(\{0.2, 0.33, 4.125\})$	$\left\{ \frac{1}{5}, \frac{33}{100}, \frac{33}{8} \right\}$
--------------------------------------	--

Exit

目錄 >

Exit

函數列表:

結束目前的 **For**、**While** 或 **Loop** 區段。**Exit** 不可用於三種迴圈結構以外 (**For**、**While** 或 **Loop**)。**輸入範例的注意事項:** 關於輸入多行程式和函數定義的說明，請參閱產品手冊中的「計算工具」章節。

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

g()	21
-----	----

exp

目錄 >

Expr ▶ exp以自然指數 e 表現 $Expr$ 。這是顯示方式轉換運算子，只能用於輸入線末尾。**附註:** 如果要從電腦鍵盤插入本運算子，可輸入 **@>exp**。

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

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

exp()e^x 鍵**exp(運算式 I)⇒運算式**傳回以 **e** 為底，運算式 *I* 為冪的指數值。**附註：**另請參考 **e** 指數範本，頁碼：2。您可輸入 $re^{i\theta}$ 極座標式複數。但您只能在弧度角模式採用這種形式。若是在度數角或梯度角模式，則會導致領域錯誤。**exp(列表 I)⇒列表**傳回以 **e** 為底，列表 *I* 各元素為冪的指數值。**exp(方陣 I)⇒方陣**傳回方陣 *I* 的矩陣指數。這和計算以 **e** 為底、以各元素為冪的指數值不同。若要了解計算方式，請參考 **cos()**。方陣 *I* 必需可對角化。結果一定會包含浮點數。

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

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

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

exp▶list()

目錄 >

exp▶list(Expr,Var)⇒列表檢查 *Expr* 中以 **[r]** 分隔的方程式並傳回列表，所包含元素是形式為 $Var=Expr$ 的方程式右側。若某些求解值已經嵌入 **solve()**、**cSolve()**、**fMin()**、**fMax()** 等函數的結果中，本功能可方便您擷取這些值。**附註：****zeros** 和 **cZeros()** 函數不需要 **exp▶list()**，因為這些函數會直接傳回求解值的列表。如果要從鍵盤插入本函數，可輸入 **exp@>list(...)**。

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

expand(運算式 I , Var) ⇒ 運算式

expand(列表 I , Var) ⇒ 列表

expand(矩陣 I , Var) ⇒ 矩陣

expand(運算式 I): 傳回對其所有變數展開的運算式 I 。若是多項式，則為多項式展開；若是有理式，則為部份分數式展開。

expand() 的目標是把運算式 I 轉成簡單項的和或是差。反之，**factor()** 的目標是把運算式 I 轉成簡單因式的積或商。

expand(運算式 I , Var): 傳回對 Var 展開的運算式 I 。系統將匯集 Var 的類似乘幕。各項及其因式以 Var 為主變數進行排序。系統可能對匯集的係數進行偶發因式分解或展開。若與省略 Var 相比，此功能通常能節省時間、記憶體和螢幕空間，並使運算式更容易懂。

即使只有一個變數，使用 Var 也可能用於部份分數式展開的分母因式分解結果更完整。

提示：若是有理式，則 **propFrac()** 是 **expand()** 以外速度較快且較不極端的其他選擇。

附註：另請參考 **comDenom()** 中分子與分母皆展開的形式。

expand(運算式 I , Var) 也會分配對數和分數幕而忽略 Var 。若是分配更徹底的對數和分數幕，可能需要不等式約束條件，才能確保某些因式不是負值。

expand(運算式 I , Var) 也會分配絕對值、**sign()** 和指數，而忽略 Var 。

附註：另請參考 **tExpand()** 一節的三角形內角和以及多角度展開。

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

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

$$\frac{1}{x-1}-\frac{1}{x}-\frac{1}{y-1}-\frac{1}{y}$$

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

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

$$\frac{1}{y-1}-\frac{1}{y}-\frac{1}{x\cdot(x-1)}$$

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

$$\frac{1}{x-1}-\frac{1}{x}-\frac{1}{y\cdot(y-1)}$$

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

$$\frac{2\cdot x}{x^2-2}+x+1$$

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

$$\frac{1}{x-\sqrt{2}}+\frac{1}{x+\sqrt{2}}+x+1$$

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

$$\frac{\ln(2\cdot x\cdot y)+\sqrt{2}\cdot x\cdot y}{\ln(x\cdot y)+\sqrt{2}\cdot\sqrt{x\cdot y}+\ln(2)}$$

$$\text{expand}(Ans)|y\geq 0$$

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

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

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

expr(字串)⇒運算式

將字串包含的字元字串傳回為運算式，並且立刻執行。

expr("1+2+x^2+x")	x^2+x+3
expr("expand((1+x)^2)")	$x^2+2\cdot x+1$
"Define cube(x)=x^3" → funcstr	"Define cube(x)=x^3"
expr(funcstr)	Done
cube(2)	8

ExpReg X, Y [, [Freq] [, 類別, 包含]]

計算 X 列表與 Y 列表的 $y = a \cdot (b)^x$ 指數迴歸(頻率為 *Freq*)。stat.results 變數會儲存結果摘要(請參閱第 頁碼: 162 頁)。

所有列表的維數都必須相同，包含除外。

X 、 Y 是自變數和因變數列表。

Freq 是頻率值列表(非必要)。*Freq* 的每個元素，可用於指定各 X 、 Y 對應資料點的出現頻率。預設值是 1。所有元素都必須是 ≥ 0 的整數。

類別是 X 、 Y 對應資料的類別代號列表。

包含是一個以上類別代號的列表。只有本列表中類別代號所對應的資料項目，才會包含在計算作業中。


如要了解列表中有空元素時的影響，請參考°空元素°，頁碼: 219。

輸出變數	說明
stat.RegEqn	迴歸方程式: $a \cdot (b)^x$
stat.a°Bstat.b	迴歸係數
stat.r2	所轉換資料之線性判定的係數
stat.r	所轉換資料 (x , $\ln(y)$) 的相關係數
stat.Resid	與指數模型相關的殘差
stat.ResidTrans	與所轉換資料之線性擬合相關的殘差

輸出變數	說明
stat.XReg	實際用於迴歸的已修改 X 列表中的資料點列表(根據 <i>Freq</i> 、類別列表、包含類別的限制)
stat.YReg	實際用於迴歸的已修改 Y 列表中的資料點列表(根據 <i>Freq</i> 、類別列表、包含類別的限制)
stat.FreqReg	<i>stat.XReg</i> 與 <i>stat.YReg</i> 對應的頻率列表

F

factor()

目錄 > 

factor(運算式 I , Var) ⇒ 運算式

factor(列表 I , Var) ⇒ 列表

factor(矩陣 I , Var) ⇒ 矩陣

factor(運算式 I): 傳回對所有變數除以公分母進行因式分解的運算式 I 。

對運算式 I 進行因式分解時，會儘量朝線性有理因式進行，而不導入新的非實數子運算式。如果您想進行對多個變數的因式分解，即可使用這個選項。

factor(運算式 I , Var): 傳回對 Var 變數進行因式分解的運算式 I 。

對運算式 I 進行因式分解時，即使可能導入無理常數或在其他變數中無理的子運算式，仍會儘量朝在 Var 中線性的實數因式進行。

因式和其中的項以 Var 為主變數進行排序。系統將匯集各因式中 Var 的類似乘幂。如果只需要對 Var 進行因式分解，而且您可接受其他變數中有無理式，以便增加對 Var 的因式分解，請加入該變數。可能會對其他變數進行偶發因式分解。

如果在自動或近似值模式中設定為自動，則加入 Var 時，能在無法以內建函數來明確表達簡潔的無理係數時，以浮點係數計算近似值。即使只有一個變數，加入 Var 也可能有更完整的因式分解。

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

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

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

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

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

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

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

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

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

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


附註：另請參考 **comDenom()** 說明的部份快速因式分解法(若 **factor()** 不夠快或是會用完記憶體)。

附註：另請參考 **cFactor()**，以了解如何從因式分解到複數之係數，以求出線性因式。

factor(有理數)：傳回已因式分解成質數的有理數。若是合成數，則計算時間會隨著第二大因式的位數而大幅增加。例如對 30 位整數進行因式分解可能要花超過一天，對 100 位數進行因式分解可能要花超過一世紀。

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

若要手動停止計算，

- **計算機：**按住  鍵並重複按 **.** 鍵。
- **Windows®：**按住 **F12** 鍵並重複按 **Enter** 鍵。
- **Macintosh®：**按住 **F5** 鍵並重複按 **Enter** 鍵。
- **iPad®：**應用程式顯示提示。您可以繼續等待或取消。

若您只想知道某數字是否為質數，請改用 **isPrime()**。此法速度快得多，若有理數不是質數，或如果第二大因數超過五位數，則能省更多時間。

F Cdf()

F Cdf(下限, 上限, dfNumer, dfDenom) ⇒ 數字(若下限和上限是數字)或列表(若下限和上限是列表)

F Cdf(下限, 上限, dfNumer, dfDenom) ⇒ 數字(若下限和上限是數字)或列表(若下限和上限是列表)

針對所指定 *dfNumer*(自由度)和 *dfDenom* 的下限和上限之間，計算 **F** 分布機率。

對於 $P(X \leq)$ ，請設定 下限=0。

Fill Expr, 矩陣變數⇒矩陣

將矩陣變數中的各元素替換成 *Expr*。

矩陣變數必須已經存在。

1 2	→ <i>amatrix</i>	1 2
3 4		3 4

Fill 1.01, <i>amatrix</i>	Done
---------------------------	------

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

Fill Expr, 列表變數⇒列表

將列表變數中的各元素替換成 *Expr*。

列表變數必須已經存在。

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

Fill 1.01, <i>alist</i>	Done
-------------------------	------

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

FiveNumSummary**FiveNumSummary** *X*,[*Freq*],[類別,包含]]

提供 *X* 列表的簡化版單元統計。

stat.results 變數會儲存結果摘要(請參閱第頁碼: 162 頁)。

X 代表包含資料的列表。

Freq 是頻率值列表(非必要)。*Freq* 的每個元素,可用於指定各 *X*、*Y* 對應資料點的出現頻率。預設值是 1。

類別是 *X* 對應資料的數字類別代號列表。

包含是一個以上類別代號的列表。只有本列表中類別代號所對應的資料項目,才會包含在計算作業中。

X、*Freq*、類別的任何列表中若有空元素,則所有這些列表的對應元素就會是空元素。如需空元素的詳細資訊,請參考第頁碼: 219 頁。

輸出變數	說明
stat.MinX	x 值的最小值。
stat.Q ₁ X	x 的第一四分位數。
stat.MedianX	x 的中位數。
stat.Q ₃ X	x 的第三四分位數。
stat.MaxX	x 值的最大值。

floor()

目錄 >

floor(運算式 I)⇒整數

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

傳回小於或等於引數的最大整數。本功能與 **int()** 相同。

引數可以是實數也可以是複數。

floor(列表 I)⇒列表

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

floor(矩陣 I)⇒矩陣

$\text{floor}\begin{pmatrix} 1.2 & 3.4 \\ 2.5 & 4.8 \end{pmatrix}$	$\begin{bmatrix} 1. & 3. \\ 2. & 4. \end{bmatrix}$
--	--

傳回各元素無條件捨去值的列表或矩陣。

附註：另請參考 **ceiling()** 與 **int()**。

fMax()

目錄 >

fMax(Expr, Var)⇒布林運算式

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

fMax(Expr, Var, 下限)

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

fMax(Expr, Var, 下限, 上限)**fMax(Expr, Var) | 下限 ≤ Var ≤ 上限**

傳回用於指定 *Var* 候選值的布林運算式，以求出 *Expr* 的最大值，或找出最小上限。

您可利用約束 (**|**) 運算子限制求解區間，或指定其他限制。

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

如果在**自動或近似值**模式中設定為近似值，則 **fMax()** 會反覆搜尋近似的局部極大值。這種方式通常較快。若您用 **|** 運算子限制為搜尋較小的區間，使得其中剛好只有一個局部極大值，則速度更是快。

附註：另請參考 **fMin()** 與 **max()**。

fMin()

目錄 >

fMin(Expr, Var)⇒布林運算式

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

fMin(Expr, Var, 下限)

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

fMin(Expr, Var, 下限, 上限)**fMin(Expr, Var) | 下限 ≤ Var ≤ 上限**

fMin()

傳回用於指定 *Var* 候選值的布林運算式，以求出 *Expr* 的最小值，或找出最大下限。

您可利用約束 (`[]`) 運算子限制求解區間，或指定其他限制。

如果在自動或近似值模式中設定為近似值，則 **fMin()** 會反覆搜尋近似的局部極小值。這種方式通常較快。若您用 `[]` 運算子限制為搜尋較小的區間，使得其中剛好只有一個局部極小值，則速度更是快。

附註：另請參考 **fMax()** 與 **min()**。

For

For *Var*, 低, 高 [, 步階]

區段

EndFor

針對 *Var* 各個值反覆執行區段中的語句，順序為由低到高，步階則是增加單位。

Var 不得是系統變數。

步階值可正可負。預設值是 1。

區段可以只是一個語句，也可以是由 `[]` 字元分隔的一連串語句。

輸入範例的注意事項：關於輸入多行程式和函數定義的說明，請參閱產品手冊中的「計算工具」章節。

Define <i>g()</i> =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</i> → <i>tempsum</i>	
EndFor	
EndFunc	
<i>g()</i>	5050

format()

format(Expr[, 格式字串]) ⇒ 字串

根據格式範本將 *Expr* 傳回為字元字串。

Expr 必須化簡成數字。

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"

格式字串的格式必須如下： $\text{F}[n]$ 、 $\text{S}[n]$ 、 $\text{E}[n]$ 、 $\text{G}[n][c]$ ， $[\]$ 代表非必要部份。

$\text{F}[n]$ ：固定格式。 n 代表小數點後要顯示的位數。

$\text{S}[n]$ ：科學格式。 n 代表小數點後要顯示的位數。

$\text{E}[n]$ ：工程格式。 n 代表第一個有效位數後要顯示的位數。指數調整成 3 的倍數，且小數點往右移 0、1 或 2 位。

$\text{G}[n][c]$ ：與固定格式相同，但也將數基點左邊的數字分為 3 個一組。 c 用於指定分組字元，預設為逗號。若 c 是句點，則將數基點顯示為逗號。

$[\text{Rc}]$ ：以上任何指定符號都可加上 Rc 數基點標示的下標，其中 c 是單一字元，用於指定取代數基點的符號。

fPart()

fPart(運算式 I) \Rightarrow 運算式

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

fPart(列表 I) \Rightarrow 列表

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

fPart(矩陣 I) \Rightarrow 矩陣

傳回引數的分數部份。

若是列表或矩陣，則傳回各元素的分數部份。

引數可以是實數也可以是複數。

Fpdf()

Fpdf($XVal$, $dfNumer$, $dfDenom$) \Rightarrow 數字(若 $XVal$ 是數字)或列表(若 $XVal$ 是列表)

針對指定 $dfNumer$ (自由度)和 $dfDenom$ ，計算 $XVal$ 的 **F** 分布機率。

freqTable►list(列表I, 頻率整數表)⇒列表

所傳回列表包含列表I中的元素，且依頻率整數表中的頻率展開這些元素。本函數可用於製作 Data & Statistics 應用程式的次數表。

列表I可以是任何有效列表。

頻率整數表和列表I的維數必須相同，而且只能包含非負整數的元素。各元素可用於指定列表I對應元素在結果列表中的重複次數。值如果是零，代表排除列表I的對應元素。

附註：如果要從電腦鍵盤插入本函數，可輸入 **freqTable@>list(...)**。

空元素會被忽略。如需空元素的詳細資訊，請參考第 頁碼：219 頁。

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

frequency()

frequency(列表I, binsList)⇒列表

傳回列表I中各元素出現次數的列表。次數是以您在 binsList 中定義的範圍 (bins) 為準。

若 binsList 是 {b(1), b(2), ..., b(n)}，指定範圍是 {?≤b(1), b(1)<?≤b(2), ..., b(n-1)<?≤b(n), b(n)>?}。所求出的列表是超過 binsList 單一元素長度。

結果中的各元素，皆與列表I中在該組距範圍內的元素數目相同。若以 **countif()** 函數表示，則結果是 {countif(list, ?≤b(1)), countif(list, b(1)<?≤b(2)), ..., countif(list, b(n-1)<?≤b(n)), countif(list, b(n)>?)}。

系統會略過列表I無法放在組距內的元素。空元素也會遭到忽略。如需空元素的詳細資訊，請參考第 頁碼：219 頁。

您可在 Lists & Spreadsheet 應用程式中用儲存格範圍取代兩個引數。

附註：另請參考 **countif()**，頁碼：34。

```
datalist={1,2,e,3,π,4,5,6,"hello",7}
          {1,2,2.71828,3,3.14159,4,5,6,"hello",7}
frequency(datalist,{2.5,4.5})           {2,4,3}
```

解釋結果：

2 個元素 (Datalist) ≤2.5

4 個元素 (Datalist) >2.5 且 ≤4.5

3 個元素 (Datalist) >4.5

「hello」元素是字串，無法放在既定的任何組距中。

FTest_2Samp 列表1,列表2[,Freq1[,Freq2
[,Hypoth]]]

FTest_2Samp 列表1,列表2[,Freq1[,Freq2
[,Hypoth]]]

(輸入資料列表)

FTest_2Samp $sx1, n1, sx2, n2$ [,Hypoth]

FTest_2Samp $sx1, n1, sx2, n2$ [,Hypoth]

(輸入統計摘要)

執行二樣本 F 檢定。stat.results 變數會儲存結果摘要(請參閱第 頁碼: 162 頁)。

$H_a: \sigma_1 > \sigma_2$ 時, 設定 Hypoth>0

$H_a: \sigma_1 \neq \sigma_2$ (預設值) 時, 設定 Hypoth =0

$H_a: \sigma_1 < \sigma_2$ 時, 設定 Hypoth<0

如要了解列表中有空元素時的影響, 請參考[®]空元素[®], 頁碼: 219。

輸出變數	說明
stat.F	計算資料序列的 \hat{U} 統計
stat.PVal	無效假說被否定之最低意義標準
stat.dfNumer	分子自由度 = $n1-1$
stat.dfDenom	分母自由度 = $n2-1$
stat.sx1、stat.sx2	列表 1 和列表 2 中資料序列的標準差樣本
stat.x1_bar stat.x2_bar	列表 1 和列表 2 中資料序列的平均值樣本
stat.n1、stat.n2	樣本大小

Func

定義分段函數:

區段

EndFunc

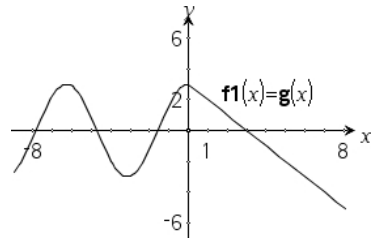
製作使用者自行定義函數的範本。

區段可以只是一個語句、由 [] 字元分隔的一連串語句，也可以是分成多行的一連串語句。本函數可利用 **Return** 指令傳回特定結果。

輸入範例的注意事項：關於輸入多行程式和函數定義的說明，請參閱產品手冊中的「計算工具」章節。

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

$g(x)$ 繪圖結果



G

gcd()

$\text{gcd}(\text{數字}1, \text{數字}2) \Rightarrow$ 運算式

$\text{gcd}(18,33)$ 3

傳回兩個引數的最大公因數。兩個分數的 **gcd**，是分子的 **gcd** 除以分母的 **lcm**。

在自動或近似值模式中，分數浮點數的 **gcd** 是 1.0。

$\text{gcd}(\text{列表}1, \text{列表}2) \Rightarrow$ 列表

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

傳回列表 1 和列表 2 中對應元素的最大公因數。

$\text{gcd}(\text{矩陣}1, \text{矩陣}2) \Rightarrow$ 矩陣

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

傳回矩陣 1 和矩陣 2 中對應元素的最大公因數。

geomCdf()

$\text{geomCdf}(p, \text{下限}, \text{上限}) \Rightarrow$ 數字 (若下限和上限是數字) 或列表 (若下限和上限是列表)

geomCdf(*p*, *上限*) for $P(1 \leq X \leq \text{上限}) \Rightarrow$ 數字 (若 *上限* 是數字) 或 列表 (若 *上限* 是列表)

計算從下限到上限的累積幾何機率 (指定成功機率是 *p*)。

對於 $P(X \leq \text{上限})$, 請設定 *下限* = 1。

geomPdf()

geomPdf(*p*, *XVal*) \Rightarrow 數字 (若 *XVal* 是數字) 或 列表 (若 *XVal* 是列表)

針對離散幾何分布計算 *XVal* 的機率, 亦即第一次成功的試驗次數 (指定成功機率是 *p*)。

Get

分享器功能表

Get[*promptString*], *var* [, *statusVar*]

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

程式設計指令: 從已連接的 TI-Innovator™ Hub 擷取數值, 並將該數值指定給變數 *var*。

必須要求數值:

- 預先透過傳送 "READ ..." 指令要求。
—或—
- 嵌入 "READ ..." 要求作為選用的 *promptString* 引數。此方法可讓您使用單一指令要求及擷取數值。

執行隱簡化。例如, 收到的字串 [123] 會解譯為數值。如要保留字串, 請使用 **GetStr** 而不是 **Get**。

若加入選用引數 *statusVar*, 則將操作成功與否指派為其值。零值表示未收到任何資料。

在第二個語法中, *func*() 引數可讓程式把收到的字串儲存成函數定義。這個語法的功能如同程式執行以下指令:

例如: 要求分享器內建亮度等級感應器目前的值。使用 **Get** 擷取數值, 並將數值指定給變數 *lightval*。

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

在 **Get** 指令中嵌入 READ 要求。

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

Define $func(arg1, \dots, argn) =$ 收到的字串

然後，程式就能使用已定義函數 $func()$ 。

附註：您可以在使用者自行定義的程式中使用 **Get** 指令，但無法在函數中使用。

附註：另請參考 **GetStr**，頁碼：79 和 **Send**，頁碼：146。

getDenom()

目錄 >

getDenom(運算式 I) ⇒ 運算式

將引數轉成有簡化公分母的運算式，然後傳回其分母。

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

getKey()

目錄 >

getKey([0|1]) ⇒ returnString

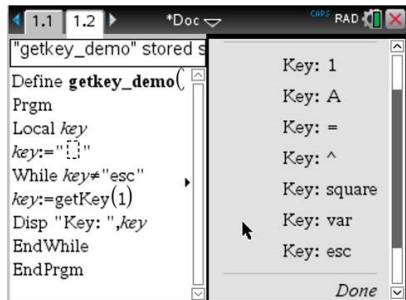
說明： $getKey()$ 可讓 TI-Basic 程式擷取計算機、桌面及桌面模擬器的鍵盤輸入內容。

範例：

- 若使用者沒有按下任何鍵， $keypressed := getKey()$ 會傳回一個鍵或空字串。這個呼叫指令會立即傳回結果。
- $keypressed := getKey(1)$ 會等到使用者按下一個鍵，才顯示結果。在使用者按下任何鍵之前，這個呼叫指令都會暫停執行程式。

getKey()

範例：



按鍵處理方式：

計算機/模擬器按鍵	桌面	傳回數值
Esc	Esc	"esc"
觸控板 - 點擊頂部	N/A	"up"
On	N/A	"home"
Scratchapps	N/A	"scratchpad"
觸控板 - 點擊左邊	N/A	"left"
觸控板 - 點擊中央	N/A	"center"
觸控板 - 點擊右邊	N/A	"right"
Doc	N/A	"doc"
Tab	Tab	"tab"
觸控板 - 點擊底部	向下鍵	"down"
功能表	N/A	"menu"
Ctrl	Ctrl	無傳回結果
Shift	Shift	無傳回結果
Var	N/A	"var"
Del	N/A	"del"
=	=	"="
trig	N/A	"trig"
0 到 9	0-9	"0" ... "9"
範本	N/A	"template"
目錄	N/A	"cat"
^	^	"^"
X^2	N/A	"square"
/(除號鍵)	/	"/"
*(乘號鍵)	*	"*"
e^x	N/A	"exp"
10^x	N/A	"10power"
+	+	"+"
-	-	"_"

計算機/模擬器按鍵	桌面	傳回數值
(("("
))	")"
.	.	"."
(-)	N/A	"-" (負號)
Enter	Enter	"enter"
ee	N/A	"E" (科學記號 E)
a - z	a-z	字母 = 按下的字母 (小寫) ("a" - "z")
shift a-z	shift a-z	字母 = 按下的字母 "A" - "Z"
		注意 :ctrl-shift 會鎖定大寫
?!	N/A	"?!"
pi	N/A	"pi"
旗幟	N/A	無傳回結果
,	,	","
回傳	N/A	"return"
空格	空格	" " (空格)
無法存取	特殊字元按鍵, 例如 @、!、^ 等。	傳回字元
N/A	功能鍵	無傳回字元
N/A	特殊桌面控制鍵	無傳回字元
無法存取	getKey() 等待使用者按下按鍵時, 計算工具上沒有的其他桌面按鍵。({, }, ;, ...)	與在筆記 (並非數學方塊) 中取得的字元相同

注意:請務必注意, 如在程式中使用 `getKey()`, 系統處理特定事件的方式就會改變。以下說明幾個這類案例。


終止程式並處理事件 - 如同使用者按下 **ON** 鍵以中斷程式

下列的 **"Support"** 代表程式繼續執行, 系統也如預期中運作。

事件	裝置	桌面 - TI-Nspire™ 學生軟體
簡易投票	終止程式，處理事件	與計算機相同 (僅限 TI-Nspire™ Student Software, TI-Nspire™ Navigator™ NC Teacher Software)
遠端檔案管理 (包括從另一部計算機或桌面計算機傳送 Exit Press 2 Test 檔案)	終止程式，處理事件	與計算機相同 (僅限 TI-Nspire™ Student Software, TI-Nspire™ Navigator™ NC Teacher Software)
結束課程	終止程式，處理事件	支援 (僅限 TI-Nspire™ Student Software, TI-Nspire™ Navigator™ NC Teacher Software)

事件	裝置	桌面 - TI-Nspire™ 所有版本
TI-Innovator™ Hub 連線/ 中斷連線	支援 - 可成功發送指令給 TI-Innovator™ Hub。在您離開程式後，TI-Innovator™ Hub 仍會與計算機共同運作。	與計算機相同

getLangInfo()

目錄 > 

getLangInfo() ⇒ 字串

getLangInfo()

"en"

傳回目前所使用語言簡稱的對應字串。例如您可用於程式或函數中，以決定目前的語言。

英文 = [en]

丹麥文 = [da]

德文 = [de]

芬蘭文 = [fi]

法文 = [fr]

義大利文 = [it]

荷蘭文 = [nl]

比利時荷蘭文 = [nl_BE]

挪威文 = [no]

葡萄牙文 = [pt]

西班牙文 = [es]

瑞典文 = [sv]

getLockInfo(*Var*)⇒值

傳回 *Var* 目前的狀態是已鎖定還是已解開鎖定。

值 =0: *Var* 已解開鎖定或不存在。

值 =1: *Var* 已遭鎖定且無法加以修改或刪除。

請參考第 98 頁的 **Lock** 和第 182 頁的 **unLock**。

<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(模式名稱整數)⇒值

getMode(0)⇒列表

getMode(模式名稱整數)所傳回的值，代表模式名稱整數模式的目前設定。

getMode(0): 傳回包含成對數值的列表。每對數字由代表該模式整數和設定整數組成。

如需模式列表及模式設定資訊，請參考下表。


若您以 **getMode(0)** → 變數儲存設定，則可在函數或程式中使用 **setMode(變數)**，以便在只執行該函數或程式期間暫時還原設定。請參考 **setMode()**，頁碼: 150。

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

模式名稱	模式整數	設定整數
顯示位數	1	1=浮點數、2=浮點數1、3=浮點數2、4=浮點數3、5=浮點數4、6=浮點數5、7=浮點數6、8=浮點數7、9=浮點數8、10=浮點數9、11=浮點數10、12=浮點數11、13=浮點數12、14=固定0、15=固定1、16=固定2、17=固定3、18=固定4、19=固定5、20=固定6、21=固定7、22=固定8、23=固定9、24=固定10、25=固定11、26=固定12
角度	2	1=弧度角、2=度數角、3=梯度角
指數格式	3	1=正常、2=科學、3=工程
實數或複數	4	1=實數、2=直角座標、3=極座標

模式名稱	模式整數	設定整數
自動或近似值	5	1=自動、2=近似值、3=精確值
向量格式	6	1=直角座標、2=圓柱座標、3=球面座標
基底	7	1=十進位、2=十六進位、3=二進位
單位制	8	1=國際單位制、2=英制

getNum()

目錄 > 

getNum(運算式 I)⇒運算式

將引數轉成有簡化公分母的運算式，然後傳回其分子。

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

GetStr

分享器功能表

GetStr[promptString,] var[, statusVar]


例如，另請參考 **Get**。

GetStr[promptString,] func(arg1, ...argn)
[, statusVar]

程式設計指令：與 **Get** 指令原理相同，不過本指令一律將收到的值解譯為字串。相反地，除非使用者把回答放在引號 ("") 中，否則 **Get** 指令會將回答解譯成運算式。

附註：另請參考 **Get**，頁碼：73 和 **Send**，頁碼：146。

getType()

目錄 > 

getType(var)⇒字串

傳回一個字串，說明變數 *var* 的資料類型。

如果未定義 *var*，則會傳回字串 [NONE]。

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

getVarInfo()**getVarInfo()** ⇒ 矩陣@ \mathbb{C} 字串**getVarInfo(LibNameString)** ⇒ 矩陣@ \mathbb{C} 字串

getVarInfo() 可針對目前問題中定義的所有變數與資料庫物件，傳回相關資訊的矩陣(變數名稱、類型、資料庫存取能力，以及已鎖定/已解開鎖定狀態)。

若尚未定義變數，則 **getVarInfo()** 會傳回「無」字串。

getVarInfo(LibNameString) 可針對 *LibNameString* 資料庫中定義的所有資料庫物件，傳回相關資訊的矩陣。*LibNameString* 必須是字串(以英文引號括住的文字)或字串變數。

若 *LibNameString* 資料庫不存在，則會出現錯誤。

請注意左例，*vs* 變數的賦值就是 **getVarInfo()** 的結果。若要顯示 *vs* 的第 2 列或第 3 列，則會傳回「列表或矩陣無效」錯誤，因為這些列中至少有一個元素(例如 *b* 變數)的求值結果是矩陣。

利用 *Ans* 重新對 **getVarInfo()** 結果求值時，也可能出現本錯誤。

方程式組之所以出現以上錯誤，是因為目前版本的軟體，不支援矩陣元素既可是矩陣又能是列表的廣義矩陣結構。

getVarInfo()	"NONE"												
Define $x=5$	Done												
Lock x	Done												
Define LibPriv $y=\{1,2,3\}$	Done												
Define LibPub $z(x)=3 \cdot x^2 - x$	Done												
getVarInfo()	<table border="1"><tr><td>x</td><td>"NUM"</td><td>"{"</td><td>1</td></tr><tr><td>y</td><td>"LIST"</td><td>"LibPriv"</td><td>0</td></tr><tr><td>z</td><td>"FUNC"</td><td>"LibPub"</td><td>0</td></tr></table>	x	"NUM"	"{"	1	y	"LIST"	"LibPriv"	0	z	"FUNC"	"LibPub"	0
x	"NUM"	"{"	1										
y	"LIST"	"LibPriv"	0										
z	"FUNC"	"LibPub"	0										

getVarInfo(tmp3)
"Error: Argument must be a string"

getVarInfo("tmp3")
[volcvl2 "NONE" "LibPub" 0]

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

Goto *labelName*

把控制權轉到 *labelName* 標籤。

您必須用 **Lbl** 指令在同一個函數中定義 *labelName*。

輸入範例的注意事項：關於輸入多行程式和函數定義的說明，請參閱產品手冊中的「計算工具」章節。

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

▶Grad

運算式1 ▶ Grad⇒運算式

把 **運算式1** 轉成梯度角測量單位。

附註：如果要從電腦鍵盤插入本運算子，可輸入 **@>Grad**。

度數角模式：

(1.5)▶Grad	(1.66667) ^g
------------	------------------------

弧度角模式：

(1.5)▶Grad	(95.493) ^g
------------	-----------------------

/

identity()**identity(整數) ⇒ 矩陣**

傳回以整數為維度的單位矩陣。

整數必須是正整數。

identity(4)	$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
-------------	--

If**If** 布林運算式
語句**If** 布林運算式 **Then**
區段**EndIf**

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

若布林運算式為真，則在繼續執行程式以前，執行單一語句 *Statement* 或整個區段語句 *Block*。

若布林運算式為假，則繼續執行程式，但不執行單一語句或整個區段語句。

區段可以只是一個語句，也可以是由「`]`」字元分隔的一系列語句。

輸入範例的注意事項：關於輸入多行程式和函數定義的說明，請參閱產品手冊中的「計算工具」章節。

If 布林運算式 Then
 區段1

Else
 區段2

Endif

若布林運算式為真，則執行區段1並跳過區段2。

若布林運算式為假，則跳過區段1但執行區段2。

區段1和區段2可以是單一語句。

If 布林運算式1 Then
 區段1

Elseif 布林運算式2 Then
 區段2

:

Elseif 布林運算式N Then
 區段N

Endif

允許有分枝。若布林運算式1為真，則執行區段1。若布林運算式1為假，則計算布林運算式2運算式，依此類推。

```
Define g(x)=Func Done
  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
```

g(-4)	4
g(10)	3

ifFn()

ifFn(布林運算式,若為真的值 [,若為假的值 [,若未知的值]]) ⇒ 運算式、列表或矩陣

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

對布林運算式(或布林運算式中的各元素)求值,並根據以下規則求出結果:

- 布林運算式可檢定個別的值、列表或矩陣。
- 若布林運算式的某元素為真,則傳回若為真的值對應元素。
- 若布林運算式的某元素為假,則傳回若為偽的值對應元素。若省略若為假的值,則傳回 undef。
- 若布林運算式的某元素既非真亦非假,則傳回若未知的值對應元素。若省略若未知的值,則傳回未定義(undef)。
- 若 ifFn() 函數的第二、第三或第四個引數是單一運算式,則會將布林檢定套用到布林運算式的所有位置。

注意:若化簡的布林運算式語句是列表或矩陣,則其他列表或矩陣引數的維數都必須相同,而結果的維數也會相同。

1 的檢定值小於 2.5,所以會將

若為真的值對應元素(5)複製到結果列表。

2 的檢定值小於 2.5,所以會將

若為真的值對應元素(6)複製到結果列表。

3 的檢定值不小於 2.5,所以會將若為假的值對應元素(10)複製到結果列表。

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

若為真的值是單一值,因此與任何選取位置相同。

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

若為假的值則尚未指定。此處使用 undef。

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

已從若為真的值選取一個元素。已從若未知的值選取一個元素。

imag()

imag(運算式 I) ⇒ 運算式

傳回引數的虛部。

注意:系統會把所有未定義的變數視為實變數。另請參考 real(), page 134

imag(列表 I) ⇒ 列表

傳回元素的虛部列表。

imag(矩陣 I) ⇒ 矩陣

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


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

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

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


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

imag()

目錄 > 

傳回元素的虛部矩陣。

impDif()

目錄 > 

impDif(方程式, 變數, 因變數[, 階數])
⇒ 運算式

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


階數的預設值是 1。

計算方程式的隱微分函數，其中一個變數是利用其他變數來定義隱式。

間接

請參考 #(), 頁碼: 210.

inString()

目錄 > 

inString(來源字串, 分段字串[, 開始])
⇒ 整數


$\text{inString}(\text{"Hello there"}, \text{"the"})$	7
$\text{inString}(\text{"ABCEFG"}, \text{"D"})$	0

傳回來源字串中分段字串第一次出現的字元開始位置。

開始(若納入指令中)可用於指定在來源字串中開始搜尋的字元位置。預設值 = 1(來源字串的第一個字元)。

若來源字串不包含分段字串或開始值 > 來源字串長度, 則傳回零。

int()

目錄 > 

int(運算式) ⇒ 整數

$\text{int}(-2.5)$	-3.
--------------------	-----

int(列表 *I*) ⇒ 列表
int(矩陣 *I*) ⇒ 矩陣

$\text{int}([-1.234 \ 0 \ 0.37])$	$[-2. \ 0 \ 0.]$
-----------------------------------	------------------

傳回小於或等於引數的最大整數。本功能與 **floor()** 相同。

引數可以是實數也可以是複數。

若是列表或矩陣, 則傳回各元素的最大整數。

intDiv(數字1, 數字2) ⇒ 整數

intDiv(列表1, 列表2) ⇒ 列表

intDiv(矩陣1, 矩陣2) ⇒ 矩陣

傳回 (數字1 ÷ 數字2) 具正負號的整數部份。

若是列表和矩陣，則傳回各成對元素的 (引數1 ÷ 引數2) 具正負號之整數部份。

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

積分

請參考 `∫()`，頁碼：205。

插入 ()

插入(*x*值, *x*列表, *y*列表, *y*質數序列列表) ⇒ 列表

此函數會進行下列計算：

針對未知函數 **f**，給定 *x*列表、*y*列表=**f**(*x*列表) 且 *y*質數序列列表=**f'**(*x*列表)，使用三次迴歸插值求得函數 **f** 的 *x*值。這是假設 *x*列表是單調遞增或遞減數字的列表，但此函數可能會傳回一個值 (即使其並非如此)。此函數會搜尋 *x*列表以尋找包含 *x*值的區間 [*x*列表[*i*], *x*列表[*i*+1]]。如果此函數找到該區間，則會針對 **f**(*x*值) 傳回一個插值；否則，它會傳回 **未定義**。

*x*列表、*y*列表及 *y*質數序列列表必須屬於相等的維數 ≥ 2 並包含化簡為數字的運算式。

*x*值可以是未定義的變數、數字或數字列表。

微分方程式：

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

$rk := rk23(-3y + 6t + 5, t, \{0, 10\}, 5, 1)$
$\begin{Bmatrix} 0. & 1. & 2. & 3. & 4. \\ 5. & 3.19499 & 5.00394 & 6.99957 & 9.00593 & 10 \end{Bmatrix}$

若要看到完整結果，請按 **▲**，然後使用 **◀** 與 **▶** 移動游標。

使用 `interpolate()` 函數以計算 `xvalueList` 的函數值：

$xvalueList := seq(i, 0, 10, 0.5)$
$\{0, 0.5, 1., 1.5, 2., 2.5, 3., 3.5, 4., 4.5, 5., 5.5, 6., 6.5, \}$
$xList := mat▶list(rk[1])$
$\{0., 1., 2., 3., 4., 5., 6., 7., 8., 9., 10.\}$
$yList := mat▶list(rk[2])$
$\{5., 3.19499, 5.00394, 6.99957, 9.00593, 10.997\}$
$yprimeList := -3y + 6t + 5 y = yList \text{ 且 } t = xList$
$\{-10., 1.41503, 1.98819, 2.00129, 1.98221, 2.006\}$
$interpolate(xvalueList, xList, yList, yprimeList)$
$\{5., 2.67062, 3.19499, 4.02782, 5.00394, 6.00011\}$

invχ²()

invχ²(面積, *df*)

invChi2(面積, *df*)

針對曲線下已知面積的自由度 df ，計算指定的反累積 χ^2 (卡方) 機率函數。

invF(面積, 分子自由度, 分母自由度)

invF(面積, 分子自由度, 分母自由度)

針對曲線下已知面積的分子自由度與分母自由度，計算指定的反累積 F 分佈函數。

invBinom(累積機率, 試驗次數, 機率, 輸出形式) \Rightarrow 純量或矩陣

反二項式。針對試驗次數 ($NumTrials$) 和各試驗成功的機率 ($Prob$)，此函數會傳回最低成功次數 k ，使 k 值大於或等於特定累積機率 ($CumulativeProb$)。

若輸出形式=0，則顯示以純量顯示結果(預設)。

若輸出形式=1，則以矩陣顯示結果。

範例: Mary 和 Kevin 正在完擲骰子遊戲。Mary 必須猜出 30 次中擲出數字 6 的最高次數。若擲出數字 6 的次數恰巧等於其猜測之次數(或小於該次數)，則 Mary 獲勝。此外，若她猜測的次數越小，則贏得的獎金越大。若 Mary 希望獲勝的機率高於 77%，則她必須猜測的最低次數為多少？

invBinom(0.77,30, $\frac{1}{6}$)	6
invBinom(0.77,30, $\frac{1}{6}$,1)	$\begin{bmatrix} 5 & 0.616447 \\ 6 & 0.776537 \end{bmatrix}$

invBinomN(累積機率, 機率, 成功次數, 輸出形式) \Rightarrow 純量或矩陣

針對各試驗成功的機率 ($Prob$) 與成功次數 ($NumSuccess$)，此函數會傳回最低試驗次數 N ，使 N 值小於或等於特定累積機率 ($CumulativeProb$)。

若輸出形式=0，則顯示以純量顯示結果(預設)。

若輸出形式=1，則以矩陣顯示結果。

範例: Monique 正在練習籃網球的投籃。根據過去的經驗，她知道自己進籃的機率为 70%。她計劃不斷練習投籃，直到得到 50 分為止。她必須嘗試投籃多少次，才能確保得到至少 50 分的機率高於 0.99？

invBinomN(0.01,0.7,49)	86
invBinomN(0.01,0.7,49,1)	$\begin{bmatrix} 85 & 0.010451 \\ 86 & 0.00709 \end{bmatrix}$

invNorm(面積[, μ], σ)]

針對 μ 和 σ 指定的常態分佈曲線，計算下方已知面積的反累積常態分佈函數。

invt()

invt(面積, df)

針對曲線下已知面積的自由度 df ，計算指定的反累積學生-t 機率函數。

iPart()

iPart(數字) \Rightarrow 整數iPart(列表 I) \Rightarrow 列表iPart(矩陣 I) \Rightarrow 矩陣

傳回引數的整數部份。

若是列表和矩陣，則傳回各元素的整數部份。

引數可以是實數也可以是複數。

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

irr()

irr(CF_0 , CF 列表 [, CF 頻率]) \Rightarrow 值

可計算內部投資報酬率的財務函數。

CF_0 是時間為 0 時的初始現金流量，必須是實數。


CF 列表是初始現金流量 CF_0 之後的現金流量金額列表。

CF 頻率是選用的列表，其中各元素可用於指定群組(相鄰)現金流量金額(CF 列表對應元素)的出現頻率。預設值是 1; 若您輸入值，則必須是 $< 10,000$ 的正整數。

注意: 另請參考 mirr(), 頁碼: 106.

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

isPrime()

目錄 > 

isPrime(數字) ⇒ 布林常數運算式

傳回真假值，以表示數字是否是 ≥ 2 且只能由自己和 1 整除的整數。

若數字超過大約 306 位數，而且沒有 ≤ 1021 的因數，則 **isPrime(數字)** 會顯示錯誤訊息。

若您只想知道數字是否為質數，請改用 **isPrime()**，而非 **factor()**。此方法速度快得多，特別若是數字不為質數，且第二大因數超過五位數左右，則能省更多時間。


輸入範例的注意事項：關於輸入多行程式和函數定義的說明，請參閱產品手冊中的「計算工具」章節。

isPrime(5)	true
isPrime(6)	false

可找出指定數字後下一個質數的函數：

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

isVoid()

目錄 > 

isVoid(變數) ⇒ 布林常數運算式

isVoid(運算式) ⇒ 布林常數運算式

isVoid(列表) ⇒ 布林常數運算式列表

傳回真假值，指出引數是否是「空」資料類型。

如需空元素的詳細資訊，請參考第 頁碼：219。

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

L

Lbl

目錄 > 

Lbl 標籤名稱

在函數中定義名稱是標籤名稱的標籤。

您可利用 **Goto** 標籤名稱指令，將控制權轉給標籤後緊接的指令。

標籤名稱必須符合變數名稱的命名規則。

輸入範例的注意事項：關於輸入多行程式和函數定義的說明，請參閱產品手冊中的「計算工具」章節。

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

lcm(數字1, 數字2)⇒運算式

lcm(6,9) 18

lcm(列表1, 列表2)⇒列表

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

lcm(矩陣1, 矩陣2)⇒矩陣

傳回兩個引數的最小公倍數。兩個分數的 **lcm**，是分子的 **lcm** 除以分母的 **gcd**。分數浮點數的 **lcm** 就是浮點數的乘積。

若是兩個列表或矩陣，則傳回對應元素的最小公倍數。

left()

left(來源字串[, Num])⇒字串

left("Hello", 2) "He"

傳回來源字串中從左邊算來的 *Num* 個字元。

若省略 *Num*，則傳回整個來源字串。

left(列表I[, Num])⇒列表

left({1,3,-2,4}, 3) {1,3,-2}

傳回列表 *I* 中從左邊算來的 *Num* 個元素。

若省略 *Num*，則傳回整個列表 *I*。

left(比較)⇒運算式

left(x<3) x

傳回方程式或不等式左邊的部份。

libShortcut()

libShortcut(LibNameString, ShortcutNameString [, LibPrivFlag])⇒變數列表

本例假設有已經正常儲存並重新整理的 **linalg2** 資料庫文件，其中包含的已定義物件是 *clearmat*、*gauss1*、*gauss2*。

在目前問題中建立變數群組。此問題需包含指定資料庫文件 *libNameString* 中所有物件的參照。本功能亦可將群組成員加入 [變數] 功能表。接著即可利用各物件的 *ShortcutNameString* 叫出該物件。

設定 *LibPrivFlag*=0，排除自訂資料庫物件(預設值)

```
getVarInfo("linalg2")
┌──────────┬──────────┬──────────┬──────────┬──────────┬──────────┐
│ clearmat │ "FUNC"   │ "LibPub"  │           │           │           │
│ gauss1   │ "PRGM"   │ "LibPriv" │           │           │           │
│ gauss2   │ "FUNC"   │ "LibPub"  │           │           │           │
└──────────┴──────────┴──────────┴──────────┴──────────┴──────────┘
```

```
libShortcut("linalg2", "la")
┌──────────┬──────────┬──────────┬──────────┬──────────┬──────────┐
│ la.clearmat │ la.gauss2 │           │           │           │           │
└──────────┴──────────┴──────────┴──────────┴──────────┴──────────┘
```

```
libShortcut("linalg2", "la", 1)
┌──────────┬──────────┬──────────┬──────────┬──────────┬──────────┐
│ la.clearmat │ la.gauss1 │ la.gauss2 │           │           │           │
└──────────┴──────────┴──────────┴──────────┴──────────┴──────────┘
```

設定 *LibPrivFlag=1*，納入自訂資料庫物件

若要複製變數群組，請參考 **CopyVar**，頁碼：28。

若要刪除變數群組，請參考 **DelVar**，頁碼：47。

limit() @CE lim()

limit(運算式 *I*, *Var*, 點 [, 方向]) ⇒ 運算式

limit(列表 *I*, *Var*, 點 [, 方向]) ⇒ 列表

limit(矩陣 *I*, *Var*, 點 [, 方向]) ⇒ 矩陣
傳回所要求的極限。

附註：另請參考**極限範本**，頁碼：6。

方向：負數=左起；正數=右起；其他=兩邊(若省略**方向**，則預設值是兩邊)。

系統一律會把正向 ∞ 極限和負向 ∞ 極限，轉成從有限側趨近的單側極限。

limit() 若無法求出唯一極限，則會傳回自己或 **undef**，視情況而定。這個情形不一定表示沒有唯一極限。**undef** 代表結果是有限或無限大的未知數，或是這類數字的整個集合。

limit() 使用羅比達規則 (L'Hopital's rule) 這類方法，因此會有無法求出的唯一極限。若 **運算式 I** 包含 **Var** 以外的未定義變數，則可能必須加以約束才能求得更簡明的結果。

四捨五入錯誤對於極限來說影響極大。計算極限時，請儘可能避免**自動或近似值**模式的近似值設定和近似數，否則可能無法求出應該是零或無限大的極限，或是有限大的非零極限。

$\lim_{x \rightarrow 5} (2 \cdot x + 3)$	13
$\lim_{x \rightarrow 0^+} \left(\frac{1}{x} \right)$	∞
$\lim_{x \rightarrow 0} \left(\frac{\sin(x)}{x} \right)$	1
$\lim_{h \rightarrow 0} \left(\frac{\sin(x+h) - \sin(x)}{h} \right)$	$\cos(x)$
$\lim_{n \rightarrow \infty} \left(\left(1 + \frac{1}{n} \right)^n \right)$	e

$\lim_{x \rightarrow \infty} (a^x)$	undef
$\lim_{x \rightarrow \infty} (a^x) a > 1$	∞
$\lim_{x \rightarrow \infty} (a^x) a > 0 \text{ and } a < 1$	0

LinRegBx

LinRegBx *X*, *Y*, [*Freq*], [類別, 包含]

計算線性回歸 $Y = A + B \cdot X$ 在列表 X 和 Y 的頻率 $Freq$ 。其結果概要存儲於 *stat.results* 的變數中。

所有列表的維數都必須相同，包含除外。

X 、 Y 是自變數和因變數列表。

$Freq$ 是頻率值列表(非必要)。 $Freq$ 的每個元素，可用於指定各 X 、 Y 對應資料點的出現頻率。預設值是 1。所有元素都必須是 ≥ 0 的整數。

類別是 X 、 Y 對應資料的類別代號列表。

包含是一個以上類別代號的列表。只有本列表中類別代號所對應的資料項目，才會包含在計算作業中。

如要了解列表中有空元素時的影響，請參考“[空元素](#)”，頁碼：219。

輸出變數	說明
stat.RegEqn	迴歸方程式: $a+b \cdot x$
stat.a、 stat.b	迴歸係數
stat.r ²	判定係數
stat.r	相關係數
stat.Resid	迴歸殘差
stat.XReg	實際用於迴歸的已修改 X 列表中的資料點列表(根據 $Freq$ 、類別列表、包含類別的限制)
stat.YReg	實際用於迴歸的已修改 Y 列表中的資料點列表(根據 $Freq$ 、類別列表、包含類別的限制)
stat.FreqReg	$stat.XReg$ 與 $stat.YReg$ 對應的頻率列表

LinRegMx $X,Y,[Freq],[類別,包含]$

計算 X 列表與 Y 列表的 $y = m \cdot x + b$ 線性迴歸(頻率為 $Freq$)。*stat.results* 變數會儲存結果摘要(請參閱第 頁碼：162 頁)。

所有列表的維數都必須相同，包含除外。

X 、 Y 是自變數和因變數列表。

$Freq$ 是頻率值列表(非必要)。 $Freq$ 的每個元素，可用於指定各 X 、 Y 對應資料點的出現頻率。預設值是 1。所有元素都必須是 ≥ 0 的整數。

類別是 X 、 Y 對應資料的類別代號列表。

包含是一個以上類別代號的列表。只有本列表中類別代號所對應的資料項目，才會包含在計算作業中。

如要了解列表中有空元素時的影響，請參考“ \emptyset 空元素”，頁碼：219。

輸出變數	說明
stat.RegEqn	迴歸方程式： $y = m \cdot x + b$
stat.m、 stat.b	迴歸係數
stat.r ²	判定係數
stat.r	相關係數
stat.Resid	迴歸殘差
stat.XReg	實際用於迴歸的已修改 X 列表中的資料點列表(根據 $Freq$ 、類別列表、包含類別的限制)
stat.YReg	實際用於迴歸的已修改 Y 列表中的資料點列表(根據 $Freq$ 、類別列表、包含類別的限制)
stat.FreqReg	$stat.XReg$ 與 $stat.YReg$ 對應的頻率列表

LinRegtIntervals

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

對於斜率。計算斜率的信賴水準信賴區間。

LinRegtIntervals $X, Y, F[, 1, Xval, CLev]$

對於反應。計算預測的 y 值、單次觀測的信賴水準預測區間，以及平均反應的信賴水準信賴區間。

stat.results 變數會儲存結果摘要(請參閱第 頁碼: 162 頁)。

所有列表的維數都必須相同。

X 、 Y 是自變數和因變數列表。

F 是頻率值列表(非必要)。 F 的每個元素, 可用於指定各 X 、 Y 對應資料點的出現頻率。預設值是 1。所有元素都必須是 ≥ 0 的整數。

如要了解列表中有空元素時的影響, 請參考“空元素”, 頁碼: 219。

輸出變數	說明
stat.RegEqn	迴歸方程式: $a+b \cdot x$
stat.a、stat.b	迴歸係數
stat.df	自由度
stat.r ²	判定係數
stat.r	相關係數
stat.Resid	迴歸殘差

僅用於斜率類型


輸出變數	說明
[stat.CLower, stat.CUpper]	斜率的信賴區間
stat.ME	信賴區間邊際誤差
stat.SESlope	斜率的標準誤差
stat.s	線的標準誤差

僅用於反應類型

輸出變數	說明
[stat.CLower, stat.CUpper]	平均反應的信賴區間
stat.ME	信賴區間邊際誤差
stat.SE	平均反應的標準誤差
[stat.LowerPred, stat.UpperPred]	單次觀測的預測區間

輸出變數	說明
stat.MEPred	預測區間邊際誤差
stat.SEPred	預測的標準誤差
stat.ŷ	$a + b \cdot XVal$

LinRegtTest

目錄 > 

LinRegtTest $X, Y, Freq, Hypoth$

計算 X 與 Y 列表的線性迴歸、 β 斜率值上的 t 檢定，以及 $y = \alpha + \beta x$ 方程式的 ρ 相關係數。本指令可根據三個對立假設檢定 $H_0: \beta = 0$ (等同於 $\rho = 0$) 虛無假設。

所有列表的維數都必須相同。

X 、 Y 是自變數和因變數列表。

$Freq$ 是頻率值列表(非必要)。 $Freq$ 的每個元素，可用於指定各 X 、 Y 對應資料點的出現頻率。預設值是 1。所有元素都必須是 ≥ 0 的整數。

$Hypoth$ 非必要值，用於指定三個對立假設中的其中一個，據以檢定虛無假設 ($H_0: \beta = \rho = 0$)。

$H_a: \beta \neq 0$ 與 $\rho \neq 0$ (預設值) 時，設定 $Hypoth = 0$

$H_a: \beta < 0$ 與 $\rho < 0$ 時，設定 $Hypoth < 0$

$H_a: \beta > 0$ 與 $\rho > 0$ 時，設定 $Hypoth > 0$


$stat.results$ 變數會儲存結果摘要(請參閱第 頁碼: 162 頁)。

如要了解列表中有空元素時的影響，請參考 \circledast 空元素 \circledast ，頁碼: 219。

輸出變數	說明
stat.RegEqn	迴歸方程式: $a + b \cdot x$
stat.t	t -統計量(顯著性檢定)
stat.PVal	無效假說被否定之最低意義標準
stat.df	自由度
stat.a、stat.b	迴歸係數

輸出變數	說明
stat.s	線的標準誤差
stat.SESlope	斜率的標準誤差
stat.r ²	判定係數
stat.r	相關係數
stat.Resid	迴歸殘差

linSolve()

目錄 > 

linSolve(*SystemOfLinearEqns*, *Var1*, *Var2*, ...) ⇒ 列表

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

linSolve(*LinearEqn1* and *LinearEqn2* and ..., *Var1*, *Var2*, ...) ⇒ 列表

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

linSolve({*LinearEqn1*, *LinearEqn2*, ...}, *Var1*, *Var2*, ...) ⇒ 列表

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

linSolve(*SystemOfLinearEqns*, {*Var1*, *Var2*, ...}) ⇒ 列表

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

linSolve(*LinearEqn1* and *LinearEqn2* and ..., {*Var1*, *Var2*, ...}) ⇒ 列表


linSolve({*LinearEqn1*, *LinearEqn2*, ...}, {*Var1*, *Var2*, ...}) ⇒ 列表

傳回 *Var1*, *Var2*, .. 變數的求解列表。

第一個引數在求值後必須是線性聯立方程式或單一線性方程式，否則會出現引數錯誤。

例如對 **linSolve**(*x=1* and *x=2*, *x*) 求值時，就會產生「引數錯誤」的結果。

ΔList()

目錄 > 


ΔList(*列表1*) ⇒ 列表

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

附註：如果要從鍵盤插入本函數，可輸入 **deltaList**(...)

傳回列表1中相鄰元素間的差分表。列表1的每個元素都會減掉在列表1中的前一個元素。所求出的列表一定會比原始列表1少一個元素。

list▶mat()

目錄 > 

list▶mat(列表[, 每列元素數])⇒矩陣
傳回逐列填入列表元素的矩陣。


elementsPerRow(若加入)可用於指定每列的元素個數。預設值是列表的元素個數(一列)。

若列表無法填滿求出的矩陣,則會補上零。

附註:如果要從電腦鍵盤插入本函數,可輸入 **list@>mat(...)**。

list▶mat {1,2,3}	[1 2 3]
list▶mat {1,2,3,4,5},2	1 2 3 4 5 0

▶ln

目錄 > 

Expr ▶ln⇒運算式

將輸入 *Expr* 轉成只包含自然對數(ln)的運算式。

附註:如果要從電腦鍵盤插入本運算子,可輸入 **@>ln**。

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

ln()

ctrl eX 鍵

ln(運算式*I*)⇒運算式

ln(列表*I*)⇒列表

傳回引數的自然對數。

若是列表,則傳回各元素的自然對數。

ln(方陣*I*)⇒方陣

傳回方陣*I*的矩陣自然對數。這和計算各元素的自然對數不同。若要了解計算方式,請參考 **cos()**。

方陣*I*必需可對角化。結果一定會包含浮點數。

ln (2.)	0.693147
----------------	----------

若複數格式模式是實數:

ln {-3,1,2,5}	"Error: Non-real calculation"
----------------------	-------------------------------

若複數格式模式是直角座標:

ln {-3,1,2,5}	{ln(3)+π·i,0.182322,ln(5)}
----------------------	----------------------------

弧度角模式與直角複數格式:

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

若要看到完整結果，請按 ▲，然後使用 ◀與▶ 移動游標。

LnReg

LnReg $X, Y, [Freq]$ [, 類別, 包含]]

計算 X 列表與 Y 列表的 $y = a+b \cdot \ln(x)$ 對數迴歸(頻率為 $Freq$)。stat.results 變數會儲存結果摘要(請參閱第 頁碼: 162 頁)。

所有列表的維數都必須相同，包含除外。

X, Y 是自變數和因變數列表。

$Freq$ 是頻率值列表(非必要)。 $Freq$ 的每個元素，可用於指定各 X, Y 對應資料點的出現頻率。預設值是 1。所有元素都必須是 ≥ 0 的整數。

類別是 X, Y 對應資料的類別代號列表。

包含是一個以上類別代號的列表。只有本列表中類別代號所對應的資料項目，才會包含在計算作業中。

如要了解列表中有空元素時的影響，請參考 °空元素°，頁碼: 219。

輸出變數	說明
stat.RegEqn	迴歸方程式: $a+b \cdot \ln(x)$
stat.a、stat.b	迴歸係數
stat.r ²	所轉換資料之線性判定的係數
stat.r	所轉換資料 ($\ln(x), y$) 的相關係數
stat.Resid	與對數模型相關的殘差
stat.ResidTrans	與所轉換資料之線性擬合相關的殘差
stat.XReg	實際用於迴歸的已修改 X 列表中的資料點列表(根據 $Freq$ 、類別列表、包含類別的限制)
stat.YReg	實際用於迴歸的已修改 Y 列表中的資料點列表(根據 $Freq$ 、類別列表、包含類別的限制)

輸出變數	說明
stat.FreqReg	stat.XReg 與 stat.YReg 對應的頻率列表

Local

目錄 >

Local Var1[, Var2] [, Var3] ...

宣告指定的 *vars* 是區域性變數。這些變數只在函數求值期間存在，一旦函數執行完成就會遭到刪除。

附註：區域性變數可節省記憶體，因為只暫時存在。另外這類變數也不會干擾既有的通用變數值。區域性變數必須用於 **For** 迴圈，以及用於暫時儲存多行函數中的值，因為您無法修改函數中的通用變數。

輸入範例的注意事項：關於輸入多行程式和函數定義的說明，請參閱產品手冊中的「計算工具」章節。

```
Define rollcount()=Func
```

```
Local i
```

```
1 → i
```

```
Loop
```

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

```
Goto end
```

```
i+1 → i
```

```
EndLoop
```

```
Lbl end
```

```
Return i
```

```
EndFunc
```

Done

rollcount()	16
-------------	----

rollcount()	3
-------------	---

Lock

目錄 >

Lock Var1[, Var2] [, Var3] ...

Lock Var.

鎖定指定變數或變數群組。您無法修改或刪除已鎖定的變數。

您無法鎖定或解開鎖定 *Ans* 系統變數，也無法鎖定 *stat.*或 *tvm* 系統變數群組。

附註：若將 **Lock** 指令用於已經解開鎖定的變數，則復原/重複功能的歷史記錄會遭到清除。

請參考第 頁碼：182 頁的 **unLock** 和第 頁碼：78 頁的 **getLockInfo()**。

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
----------	------

log()

ctrl 10x 鍵

log(運算式1[,運算式2])⇒運算式

log(列表1[,運算式2])⇒列表

傳回第一個引數以 運算式2 為底的對數值。

附註：另請參考**對數範本**，頁碼：2。

若是列表，則傳回各元素以 運算式2 為底的對數值。

如果省略第二個引數，則會以 10 為底。

log(方陣1[,Expr])⇒方陣

傳回方陣1 以 Expr 為底的對數矩陣。這和計算各元素以 Expr 為底的對數不同。若要了解計算方式，請參考 **cos()**。

方陣1 必需可對角化。結果一定會包含浮點數。

如果省略基底引數，則會以 10 為底。

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

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

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

若複數格式模式是實數：

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

若複數格式模式是直角座標：

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

弧度角模式與直角複數格式：

$$\log_{10}\begin{pmatrix} 1 & 5 & 3 \\ 4 & 2 & 1 \\ 6 & -2 & 1 \end{pmatrix}$$
$$\begin{bmatrix} 0.795387+0.753438\cdot i & 0.003993-0.6474\cdot i \\ 0.194895-0.315095\cdot i & 0.462485+0.27077\cdot i \\ -0.115909-0.904706\cdot i & 0.488304+0.7774\cdot i \end{bmatrix}$$

若要看到完整結果，請按 ▲，然後使用 ◀與 ▶ 移動游標。

logbase

目錄 >

Expr ▶logbase(運算式1)⇒運算式

將輸入運算式化簡成以 運算式1 為底的運算式。

附註：如果要從電腦鍵盤插入本運算子，可輸入 @>logbase(...)。

$$\log_3(10)-\log_5(5) \blacktriangleright \logbase(5)$$
$$\frac{\log_5\left(\frac{10}{3}\right)}{\log_5(3)}$$

Logistic

目錄 >

Logistic X, Y[, [Freq] [, 類別, 包含]]

計算 X 列表與 Y 列表的 $y = (c/(1+a \cdot e^{-bx}))$ 羅吉斯迴歸 (頻率為 *Freq*)。 *stat.results* 變數會儲存結果摘要 (請參閱第 頁碼: 162 頁)。

所有列表的維數都必須相同, 包含除外。

X 、 Y 是自變數和因變數列表。

Freq 是頻率值列表 (非必要)。*Freq* 的每個元素, 可用於指定各 X 、 Y 對應資料點的出現頻率。預設值是 1。所有元素都必須是 ≥ 0 的整數。

類別是 X 、 Y 對應資料的類別代號列表。

包含是一個以上類別代號的列表。只有本列表中類別代號所對應的資料項目, 才會包含在計算作業中。

如要了解列表中有空元素時的影響, 請參考 \emptyset 空元素 \emptyset , 頁碼: 219。

輸出變數	說明
stat.RegEqn	迴歸方程式: $c/(1+a \cdot e^{-bx})$
stat.a、stat.b、stat.c	迴歸係數
stat.Resid	迴歸殘差
stat.XReg	實際用於迴歸的已修改 X 列表中的資料點列表 (根據 <i>Freq</i> 、類別列表、包含類別的限制)
stat.YReg	實際用於迴歸的已修改 Y 列表中的資料點列表 (根據 <i>Freq</i> 、類別列表、包含類別的限制)
stat.FreqReg	<i>stat.XReg</i> 與 <i>stat.YReg</i> 對應的頻率列表

LogisticD X , Y [, [迭代], [*Freq*] [, 類別, 包含]]

利用指定迭代數, 計算 X 列表與 Y 列表的 $y = (c/(1+a \cdot e^{-bx})+d)$ 羅吉斯迴歸 (頻率為 *Freq*)。 *stat.results* 變數會儲存結果摘要 (請參閱第 頁碼: 162 頁)。

所有列表的維數都必須相同，包含除外。

X 、 Y 是自變數和因變數列表。

$Freq$ 是頻率值列表(非必要)。 $Freq$ 的每個元素，可用於指定各 X 、 Y 對應資料點的出現頻率。預設值是 1。所有元素都必須是 ≥ 0 的整數。

類別是 X 、 Y 對應資料的類別代號列表。

包含是一個以上類別代號的列表。只有本列表中類別代號所對應的資料項目，才會包含在計算作業中。

如要了解列表中有空元素時的影響，請參考“ β 空元素”，頁碼：219。

輸出變數	說明
stat.RegEqn	迴歸方程式： $c/(1+a \cdot e^{-bx})+d$
stat.a、stat.b、 stat.c、stat.d	迴歸係數
stat.Resid	迴歸殘差
stat.XReg	實際用於迴歸的已修改 X 列表中的資料點列表(根據 $Freq$ 、類別列表、包含類別的限制)
stat.YReg	實際用於迴歸的已修改 Y 列表中的資料點列表(根據 $Freq$ 、類別列表、包含類別的限制)
stat.FreqReg	$stat.XReg$ 與 $stat.YReg$ 對應的頻率列表

Loop

區段

EndLoop

反覆執行區段中的語句。請注意，迴圈會永久執行，除非在區段中執行 **Goto** 或 **Exit** 指令。

區段是由 [] 字元分隔的一系列語句。

輸入範例的注意事項：關於輸入多行程式和函數定義的說明，請參閱產品手冊中的「計算工具」章節。

```

Define rollcount()=Func
    Local i
    1 → i
    Loop
    If randInt(1,6)=randInt(1,6)
    Goto end
    i+1 → i
    EndLoop
    Lbl end
    Return i
    EndFunc
    
```

	Done
rollcount()	16
rollcount()	3

LU

LU 矩陣, l 矩陣, u 矩陣, p 矩陣, Tol

計算實數或複數矩陣的 Doolittle LU 上下分解值。 l 矩陣負責儲存下三角矩陣， u 矩陣負責儲存上三角矩陣， p 矩陣負責儲存置換矩陣(描述計算期間執行的列交換情形)。

$$l \text{ 矩陣} \cdot u \text{ 矩陣} = p \text{ 矩陣} \cdot \text{矩陣}$$


若任何矩陣元素的絕對值小於 Tol (容許值)，則亦可將此元素視為零。只有在矩陣中包含浮點數項目，而且不包含尚未賦值的任何符號變數時，才會使用此容許值。其他時候都會忽略 Tol (容許值)。

- 若您使用 **ctrl enter** 或將自動或近似值模式設定成近似值，則系統會利用浮點運算法執行計算作業。
- 若指令省略或未使用 Tol ，則預設容許值的計算方式如下：
 $5E-14 \cdot \max(\text{dim}(\text{矩陣})) \cdot \text{rowNorm}(\text{矩陣})$

LU 因式分解運算法使用部分軸元法搭配列交換法。

$\begin{bmatrix} 6 & 12 & 18 \\ 5 & 14 & 31 \\ 3 & 8 & 18 \end{bmatrix} \rightarrow m1$	$\begin{bmatrix} 6 & 12 & 18 \\ 5 & 14 & 31 \\ 3 & 8 & 18 \end{bmatrix}$
LU $m1, lower, upper, perm$	Done
lower	$\begin{bmatrix} 1 & 0 & 0 \\ \frac{5}{6} & 1 & 0 \\ \frac{1}{2} & \frac{1}{2} & 1 \end{bmatrix}$
upper	$\begin{bmatrix} 6 & 12 & 18 \\ 0 & 4 & 16 \\ 0 & 0 & 1 \end{bmatrix}$
perm	$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$


$\begin{bmatrix} m & n \\ o & p \end{bmatrix} \rightarrow m1$	$\begin{bmatrix} m & n \\ o & p \end{bmatrix}$
LU $m1, lower, upper, perm$	Done
lower	$\begin{bmatrix} 1 & 0 \\ \frac{m}{o} & 1 \\ o & \end{bmatrix}$
upper	$\begin{bmatrix} o & p \\ 0 & n - \frac{m \cdot p}{o} \\ o & \end{bmatrix}$
perm	$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$

mat▶list()目錄 > **mat▶list**(矩陣)⇒列表

傳回填入矩陣元素的列表。系統會從矩陣逐列複製元素。

附註： 如果要從電腦鍵盤插入本函數，可輸入 **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()目錄 > **max**(運算式1, 運算式2)⇒運算式**max**(列表1, 列表2)⇒列表**max**(矩陣1, 矩陣2)⇒矩陣

傳回兩個引數的最大值。若引數是兩個列表或矩陣，則傳回包含每組對應元素最大值的列表或矩陣。

max(列表)⇒運算式

傳回列表的最大元素。

max(矩陣1)⇒矩陣

傳回包含矩陣1中各行最大元素的列向量。


空元素會被忽略。如需空元素的詳細資訊，請參考第 頁碼：219 頁。

附註： 另請參考 **fMax()** 與 **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 $\left(\begin{bmatrix} 1 & -3 & 7 \\ -4 & 0 & 0.3 \end{bmatrix}\right)$	[1 0 7]
--	---------

mean()目錄 > **mean**(列表[, 頻率列表])⇒運算式

傳回列表元素的平均值。

頻率列表的每個元素，代表列表中對應元素的連續出現次數。


mean(矩陣1[, 頻率矩陣])⇒矩陣

傳回矩陣1中所有行之平均數的列向量。

mean ({0.2,0,1,-0.3,0.4})	0.26
mean ({1,2,3},{3,2,1})	$\frac{5}{3}$

直角向量格式：

mean()

目錄 > 

頻率矩陣的每個元素，代表矩陣 I 中對應元素的連續出現次數。


空元素會被忽略。如需空元素的詳細資訊，請參考第 頁碼：219 頁。

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

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

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

median()

目錄 > 

median(列表[, 頻率列表])⇒運算式

傳回列表元素的中位數。

頻率列表的每個元素，代表列表中對應元素的連續出現次數。

median(矩陣 I , 頻率矩陣)⇒矩陣

傳回包含矩陣 I 中各行中位數的列向量。

頻率矩陣的每個元素，代表矩陣 I 中對應元素的連續出現次數。


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

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

附註：

- 列表或矩陣中的所有項目都必須化簡成數字。
- 列表或矩陣中的空元素會遭到忽略。如需空元素的詳細資訊，請參考第 頁碼：219 頁。

MedMed

目錄 > 

MedMed X, Y [, $Freq$] [, 類別, 包含]

計算 X 列表與 Y 列表的 $y = (m \cdot x + b)$ 中位數-中位數線(頻率為 $Freq$)。 *stat.results* 變數會儲存結果摘要(請參閱第 頁碼：162 頁)。

所有列表的維數都必須相同，包含除外。

X 、 Y 是自變數和因變數列表。

Freq 是頻率值列表(非必要)。*Freq* 的每個元素,可用於指定各 X 、 Y 對應資料點的出現頻率。預設值是 1。所有元素都必須是 ≥ 0 的整數。

類別是 X 、 Y 對應資料的類別代號列表。

包含是一個以上類別代號的列表。只有本列表中類別代號所對應的資料項目,才會包含在計算作業中。

如要了解列表中有空元素時的影響,請參考“空元素”,頁碼: 219。

輸出變數	說明
stat.RegEqn	中位數-中位數線方程式: $m \cdot x + b$
stat.m、 stat.b	模型係數
stat.Resid	中位數-中位數線的殘差
stat.XReg	實際用於迴歸的已修改 X 列表中的資料點列表(根據 <i>Freq</i> 、類別列表、包含類別的限制)
stat.YReg	實際用於迴歸的已修改 Y 列表中的資料點列表(根據 <i>Freq</i> 、類別列表、包含類別的限制)
stat.FreqReg	<i>stat.XReg</i> 與 <i>stat.YReg</i> 對應的頻率列表

mid()


mid(來源字串,起點[,個數]) ⇒ 字串

傳回來源字串中的個數字元(從起點位置的字元算起)。

若省略個數或個數超過來源字串長度,則傳回來源字串的所有字元(從起點位置的字元算起)。

個數必須 ≥ 0 。若個數 = 0,則傳回空字串。

mid("Hello there",2)	"ello there"
mid("Hello there",7,3)	"the"
mid("Hello there",1,5)	"Hello"
mid("Hello there",1,0)	""

mid()目錄 > **mid(來源列表, 起點 [, 個數])**⇒列表

傳回來源列表中的個數元素(從起點位置的元素算起)。

若省略個數或個數超過來源列表長度, 則傳回來源列表的所有元素(從起點位置的元素算起)。

個數必須 ≥ 0。若個數 = 0, 則傳回空列表。

mid(來源字符串列表, 起點[, 個數])⇒列表

傳回來源字符串列表中的個數字串(從起點位置的元素算起)。

$\text{mid}\{\{9,8,7,6\},3\}$	$\{7,6\}$
$\text{mid}\{\{9,8,7,6\},2,2\}$	$\{8,7\}$
$\text{mid}\{\{9,8,7,6\},1,2\}$	$\{9,8\}$
$\text{mid}\{\{9,8,7,6\},1,0\}$	$\{\}$

$\text{mid}\{\{"A","B","C","D"\},2,2\}$	$\{"B","C"\}$
---	---------------

min()目錄 > **min(運算式1, 運算式2)**⇒運算式**min(列表1, 列表2)**⇒列表**min(矩陣1, 矩陣2)**⇒矩陣

傳回兩個引數的最小值。若引數是兩個列表或矩陣, 則傳回包含每組對應元素最小值的列表或矩陣。

min(列表)⇒運算式

傳回列表的最小元素。

min(矩陣1)⇒矩陣


傳回包含矩陣1中各行最小元素的列向量。

附註: 另請參考 **fMin()** 與 **max()**。

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

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

$\text{min}\left[\begin{matrix} 1 & -3 & 7 \\ -4 & 0 & 0.3 \end{matrix}\right]$	$[-4 \ -3 \ 0.3]$
---	-------------------

mirr()目錄 > **mirr(融資成本, 現金再投資, CF0, CFList[, CFFreq])**

傳回修正後內部投資報酬率的財務函數。

融資成本是您為現金流量金額支付的利率。

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

現金再投資是現金流量再投資的利率。

CF_0 是時間為 0 時的初始現金流量，必須是實數。

$CFList$ 是初始現金流量 CF_0 之後的現金流量金額列表。

$CFFreq$ 是選擇性的非必要列表，其中各元素可用於指定分組(相鄰)現金流量金額($CFList$ 對應元素)的出現頻率。預設值是 1; 若您輸入值，則必須是 $< 10,000$ 的正整數。

附註：另請參考 $irr()$ ，頁碼：87。

mod()

$mod(\text{運算式1}, \text{運算式2}) \Rightarrow \text{運算式}$

$mod(7,0)$	7
------------	---

$mod(\text{列表1}, \text{列表2}) \Rightarrow \text{列表}$

$mod(7,3)$	1
------------	---

$mod(\text{矩陣1}, \text{矩陣2}) \Rightarrow \text{矩陣}$

$mod(-7,3)$	2
-------------	---

依照下列恆等式的定義，傳回第一個引數對第二個引數的模數：

$mod(7,-3)$	-2
-------------	----

$mod(-7,-3)$	-1
--------------	----

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

$mod(x,0) = x$

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

若第二個引數不是零，則該引數中的結果會呈週期變化。結果如果不是零，就是與第二個引數有相同符號。

若引數是兩個列表或兩個矩陣，則傳回包含每組對應元素模數的列表或矩陣。

附註：另請參考 $remain()$ ，頁碼：137


mRow()

$mRow(Expr, \text{矩陣1}, \text{索引}) \Rightarrow \text{矩陣}$

傳回矩陣 I 的副本，其中矩陣 I 的索引這一行，各元素都要乘以 $Expr$ 。

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

mRowAdd()

目錄 > 


mRowAdd(Expr, 矩陣1, 索引1, 索引2)
⇒ 矩陣

$$\text{mRowAdd}\left(-3, \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}, 1, 2\right) \quad \begin{bmatrix} 1 & 2 \\ 0 & -2 \end{bmatrix}$$
$$\text{mRowAdd}\left(n, \begin{bmatrix} a & b \\ c & d \end{bmatrix}, 1, 2\right) \quad \begin{bmatrix} a & b \\ a \cdot n + c & b \cdot n + d \end{bmatrix}$$

傳回矩陣1的副本，其中矩陣1的索引2這一行，各元素都要取代成：

Expr · 索引1列 + 索引2列

MultReg

目錄 > 

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

計算 Y 列表在 X1, X2, ..., X10 等列表上的多線性迴歸。stat.results 變數會儲存結果摘要(請參閱第 頁碼: 162 頁)。

所有列表的維數都必須相同。

如要了解列表中有空元素時的影響，請參考°空元素°，頁碼: 219。

輸出變數	說明
stat.RegEqn	迴歸方程式: $b_0 + b_1 \cdot x_1 + b_2 \cdot x_2 + \dots$
stat.b0、stat.b1...	迴歸係數
stat.R ²	多元判定係數
stat.yList	$\hat{y} \text{ List} = b_0 + b_1 \cdot x_1 + \dots$
stat.Resid	迴歸殘差

MultRegIntervals

目錄 > 

MultRegIntervals Y, X1[,X2[,X3,...[,X10]]], XValList[,CLevel]

計算預測的 y 值、單次觀測的信賴水準預測區間，以及平均反應的信賴水準信賴區間。


stat.results 變數會儲存結果摘要(請參閱第 頁碼: 162 頁)。

所有列表的維數都必須相同。

如要了解列表中有空元素時的影響，請參考°空元素°，頁碼: 219。

輸出變數	說明
stat.RegEqn	迴歸方程式: $b_0+b_1 \cdot x_1+b_2 \cdot x_2+ \dots$
stat. \hat{y}	點估計值: $\hat{y} = b_0 + b_1 \cdot x_1 + \dots$ (對於 <i>XVallList</i>)
stat.dfError	誤差自由度
stat.CLower、stat.CUpper	平均反應的信賴區間
stat.ME	信賴區間邊際誤差
stat.SE	平均反應的標準誤差
stat.LowerPred、 stat.UpperrPred	單次觀測的預測區間
stat.MEPred	預測區間邊際誤差
stat.SEPred	預測的標準誤差
stat.bList	迴歸係數列表, {b0,b1,b2,...}
stat.Resid	迴歸殘差

MultRegTests

目錄 > 

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

多線性迴歸檢定可計算已知資料的多線性迴歸, 並提供全域 *F* 檢定統計量和係數的 *t* 檢定統計量。

stat.results 變數會儲存結果摘要(請參閱第 頁碼: 162 頁)。

如要了解列表中有空元素時的影響, 請參考“空元素”, 頁碼: 219。

輸出

輸出變數	說明
stat.RegEqn	迴歸方程式: $b_0+b_1 \cdot x_1+b_2 \cdot x_2+ \dots$
stat.F	全域 <i>F</i> 檢定統計量
stat.PVal	與全域 <i>F</i> 統計量相關的 <i>p</i> 值
stat.R ²	多元判定係數
stat.AdjR ²	調整多元判定係數
stat.s	誤差的標準差

輸出變數	說明
stat.DW	Durbin-Watson 統計量; 用於判斷模型中是否有一階自相關現象
stat.dfReg	迴歸自由度
stat.SSReg	迴歸平方和
stat.MSReg	迴歸均方
stat.dfError	誤差自由度
stat.SSError	誤差平方和
stat.MSError	誤差均方
stat.bList	{b0,b1,...}係數表
stat.tList	t 統計量列表; bList 中的每個係數都有一個
stat.PList	每個 t 統計量的 p 值列表
stat.SEList	bList 中的係數標準誤差列表
stat.ŷList	$\hat{y}List = b_0 + b_1 \cdot x_1 + \dots$
stat.Resid	迴歸殘差
stat.sResid	標準化殘差; 將殘差除以其標準差即可得出
stat.CookDist	庫克距離; 根據殘差和槓桿作用測量觀察值的影響力
stat.Leverage	測量自變數的值與其平均值距離

N

nand

ctrl = 鍵

布林運算式1 nand 布林運算式2 傳回布林運算式

$x \geq 3$ and $x \geq 4$ $x \geq 4$

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

布林列表1 nand 布林列表2 傳回布林列表

布林矩陣1 nand 布林矩陣2 傳回布林矩陣

傳回兩個引數的邏輯and運算的負值。
傳回真偽值或方程式的化簡形式。

若是列表和矩陣，則傳回對應元素逐一比較的結果。

整數 1 nand 整數 2 ⇒ 整數

利用 **nand** 功能逐一個別位元比較兩個實際整數。系統內部會把兩個整數轉換成有正負號的 64 位元二進位數字。比較對應的位元時，如果兩個都是 1 則結果為 1；否則結果為 0。傳回的值代表位元結果，是根據基底模式顯示的。

您可以用任何整數做為基底。如果要輸入二進位或十六進位數字，則必須在前面分別加上 **0b** 或 **0h** 字首。如果沒有加上字首，就會把整數視為十進位（以 10 為底）。

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

nCr()**nCr(運算式 1, 運算式 2) ⇒ 運算式**

已知 運算式 1 和 運算式 2 整數且 運算式 1 ≥ 運算式 2 ≥ 0，則 **nCr()** 代表一次取出 運算式 2 件物品時，運算式 1 的組合數（又稱為二項數係數）。兩個引數可以是整數也可以是符號式。

nCr(z,3)	$\frac{z \cdot (z-2) \cdot (z-1)}{6}$
Ans z=5	10
nCr(z,c)	$\frac{z!}{c! \cdot (z-c)!}$
Ans	$\frac{1}{c!}$
nPr(z,c)	c!

nCr(Expr, 0) ⇒ 1**nCr(Expr, 負整數) ⇒ 0****nCr(Expr, 正整數) ⇒ Expr · (Expr-1) ... (Expr-正整數+1) / 正整數!****nCr(Expr, 非整數) ⇒ 運算式! / ((Expr-非整數)! · 非整數!)****nCr(列表 1, 列表 2) ⇒ 列表**

根據兩個列表中的對應成對元素，傳回組合值列表。引數必須是維數大小相同的列表。


nCr({5,4,3},{2,4,2})	{10,1,3}
----------------------	----------

nCr(矩陣 1, 矩陣 2) ⇒ 矩陣

根據兩個矩陣中的對應成對元素，傳回組合值矩陣。引數必須是維度相同的矩陣。

nCr($\begin{bmatrix} 6 & 5 \\ 4 & 3 \end{bmatrix}, \begin{bmatrix} 2 & 2 \\ 2 & 2 \end{bmatrix}$)	$\begin{bmatrix} 15 & 10 \\ 6 & 3 \end{bmatrix}$
---	--

nDerivative()

目錄 > 

nDerivative(運算式1,Var=值[,階數])⇒
值

$nDerivative(x ,x=1)$	1
$nDerivative(x ,x) _{x=0}$	undef
$nDerivative(\sqrt{x-1},x) _{x=1}$	undef

nDerivative(運算式1,Var[,階數]) | Var=
值⇒值

傳回利用自動微分法計算的數值導數。

指定值時，此值會取代之之前賦值的所有變數，或目前以「|」取代的所有該變數。

導數的階數必須是 1 或 2。

newList()

目錄 > 

newList(元素數)⇒列表

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

傳回維數是元素數的列表，每個元素都是零。

newMat()


目錄 > 

newMat(numRows, numColumns)⇒矩陣

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

傳回以零組成的矩陣，維度是 numRows X numColumns。

nfMax()

目錄 > 

nfMax(Expr, Var)⇒值

$nfMax(x^2-2\cdot x-1,x)$	-1.
---------------------------	-----

nfMax(Expr, Var, 下限)⇒值

$nfMax(0.5\cdot x^3-x-2,x,-5,5)$	5.
----------------------------------	----

nfMax(Expr, Var, 下限, 上限)⇒值

nfMax(Expr, Var) | 下限≤Var≤上限⇒
值

傳回 Expr 出現局部極大值時，Var 變數的可能數值。

若您提供下限與上限，則本函數會在閉區間 [下限, 上限] 之間尋找局部極大值。

附註：另請參考 fMax() 與 d()。

nfMin()

目錄 >

nfMin(*Expr*, *Var*)⇒值

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

nfMin(*Expr*, *Var*, 下限)⇒值

$$\text{nfMin}(0.5\cdot x^3-x-2,x,-5.5) \quad -5.$$

nfMin(*Expr*, *Var*, 下限, 上限)⇒值**nfMin**(*Expr*, *Var*) | 下限≤*Var*≤上限⇒值傳回 *Expr* 出現局部極小值時, *Var* 變數的可能數值。

若您提供 下限與 上限, 則本函數會在閉區間 [下限, 上限] 之間尋找最小值。

附註: 另請參考 **fMin()** 與 **d()**。**nInt()**

目錄 >

nInt(運算式 *I*, *Var*, 下限, 上限)⇒運算式

$$\text{nInt}(e^{-x^2},x,-1,1) \quad 1.49365$$

若 運算式 *I* 被積函數只有 *Var* 變數, 且若 下限和 上限是常數、正向 ∞ 或負向 ∞ , 則 **nInt()** 會傳回 \int (運算式 *I*, *Var*, 下限, 上限) 的近似值。這個近似值是 下限<*Var*<上限區間中, 被積函數幾個樣本值的加權平均值。

本函數的目標是求出六個有效位數。適應性運算法在可能達到目標, 或是更多樣本也不太可能改善結果時, 就會停止計算。

如果看似無法達到目標, 則會顯示警告(準確性可疑)。

若要計算多重數值積分, 可嵌套 **nInt()**。積分極限可能受到積分函數外的積分變數影響。

$$\text{nInt}(\cos(x),x,\pi,\pi+1.\text{E}-12) \quad -1.04144\text{E}-12$$

$$\int_{-\pi}^{\pi+10^{-12}} \cos(x)dx \quad -\sin\left(\frac{1}{1000000000000}\right)$$

$$\text{nInt}\left(\text{nInt}\left(\frac{e^{-x\cdot y}}{\sqrt{x^2-y^2}},y,-x,x\right),x,0,1\right) \quad 3.30423$$

附註: 另請參考 $\int()$, 頁碼: 194。**nom()**

目錄 >

nom(實質利率, *CpY*)⇒值

$$\text{nom}(5.90398,12) \quad 5.75$$

可將 *effectiveRate* 實際年利率轉成名目利率的財務函數(已知 *CpY* 是每年複利期數)。

effectiveRate 必須是實數，*CpY* 必須是 >0 的實數。

附註：另請參考 **eff()**，頁碼：55。

nor  鍵

布林運算式 **lnor** 布林運算式2 傳回布林運算式

$x \geq 3$ or $x \geq 4$	$x \geq 3$
--------------------------	------------

布林列表 **lnor** 布林列表2 傳回布林列表

$x \geq 3$ nor $x \geq 4$	$x < 3$
---------------------------	---------

布林矩陣 **lnor** 布林矩陣2 傳回布林矩陣

傳回兩個引數的邏輯 **or** 運算的否定值。
傳回真偽值或方程式的化簡形式。

若是列表和矩陣，則傳回對應元素逐一比較的結果。

整數 **lnor** 整數2 \Rightarrow 整數

利用 **nor** 功能逐一個別位元比較兩個實際整數。系統內部會把兩個整數轉換成有正負號的 64 位元二進位數字。比較對應的位元時，如果兩個都是 1 則結果為 1；否則結果為 0。傳回的值代表位元結果，是根據基底模式顯示的。

3 or 4	7
--------	---


3 nor 4	-8
---------	----

{1,2,3} or {3,2,1}	{3,2,3}
--------------------	---------

{1,2,3} nor {3,2,1}	{-4,-3,-4}
---------------------	------------

您可以用任何整數做為基底。如果要輸入二進位或十六進位數字，則必須在前面分別加上 **0b** 或 **0h** 字首。如果沒有加上字首，就會把整數視為十進位（以 10 為底）。

norm()

目錄 > 

norm(矩陣)⇒運算式

$$\text{norm}\left(\begin{bmatrix} a & b \\ c & d \end{bmatrix}\right) \quad \sqrt{a^2+b^2+c^2+d^2}$$

norm(向量)⇒運算式


$$\text{norm}\left(\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}\right) \quad \sqrt{30}$$

傳回 Frobenius 範數。

$$\text{norm}([1 \ 2]) \quad \sqrt{5}$$

$$\text{norm}\left(\begin{bmatrix} 1 \\ 2 \end{bmatrix}\right) \quad \sqrt{5}$$

normalLine()

目錄 > 

normalLine(運算式 I, Var, 點)⇒運算式

$$\text{normalLine}(x^2, x, 1) \quad \frac{3}{2} \frac{x}{2}$$

normalLine(運算式 I, Var=點)⇒運算式

$$\text{normalLine}((x-3)^2-4, x, 3) \quad x=3$$


針對 運算式 I 代表的曲線，傳回指定 Var=點的法線。

$$\text{normalLine}\left(\frac{1}{x^3}, x=0\right) \quad 0$$

請勿定義自變數。例如若 $f1(x):=5$ 且 $x:=3$ ，則 **normalLine(f1(x), x, 2)** 會傳回「**Error**」。

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

normCdf()


目錄 > 

normCdf(下限, 上限, μ , σ)⇒數字(若下限和上限是數字)或列表(若下限和上限是列表)

針對指定的 μ (預設值=0) 與 σ (預設值=1)，計算下限與上限間的常態分布機率。


對於 $P(X \leq \text{上限})$ ，請設定 下限 = $-\infty$ 。

normPdf()

目錄 > 

normPdf(XVal, μ , σ)⇒數字(若 XVal 是數字)或列表(若 XVal 是列表)

針對指定 μ 與 σ ，計算常態分布在指定 XVal 值的機率密度函數(pdf)。

nPr()目錄 > **nPr(列表1, 列表2) ⇒ 列表**

$nPr(\{5,4,3\},\{2,4,2\})$	$\{20,24,6\}$
----------------------------	---------------

根據兩個列表中的對應成對元素，傳回排列值列表。引數必須是維數大小相同的列表。

nPr(矩陣1, 矩陣2) ⇒ 矩陣

$nPr\left(\begin{bmatrix} 6 & 5 \\ 4 & 3 \end{bmatrix}, \begin{bmatrix} 2 & 2 \\ 2 & 2 \end{bmatrix}\right)$	$\begin{bmatrix} 30 & 20 \\ 12 & 6 \end{bmatrix}$
--	---

根據兩個矩陣中的對應成對元素，傳回排列值矩陣。引數必須是維度相同的矩陣。

npv()目錄 > **npv(利率, CFO, CFList[, CFFreq])**

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

可計算淨現值、現金流入/流出現值總和的財務函數。npv 計算結果如果是正值，代表投資有獲利。

$list2:=\{2,2,2,1\}$	$\{2,2,2,1\}$
----------------------	---------------

利率是現金流量(資金成本)一期的貼現率。

$npv(10,5000,list1,list2)$	4769.91
----------------------------	---------

CFO 是時間為 0 時的初始現金流量，必須是實數。

CFList 是初始現金流量 CFO 之後的現金流量金額列表。

CFFreq 列表各元素可用於指定分組(相鄰)現金流量金額(CFList 對應元素)的出現頻率。預設值是 1; 若您輸入值，則必須是 < 10,000 的正整數。

nSolve()目錄 > **nSolve(方程式, Var[=猜測值]) ⇒ 數字或錯誤字串**

$nSolve(x^2+5x-25=9,x)$	3.84429
-------------------------	---------

nSolve(方程式, Var[=猜測值], 下限) ⇒ 數字或錯誤字串

$nSolve(x^2=4,x=-1)$	-2.
----------------------	-----

nSolve(方程式, Var[=猜測值], 下限, 上限) ⇒ 數字或錯誤字串

$nSolve(x^2=4,x=1)$	2.
---------------------	----

附註: 若有多個解，則可利用猜測值找出特定解。

nSolve(方程式, Var[=猜測值]) | 下限 ≤ Var ≤ 上限 ⇒ 數字或錯誤字串

反覆搜尋方程式其中一個變數的近似實數解。指定變數為：

變數

- 或 -

變數 = 實數

舉例來說, x 和 $x=3$ 都有效。

nSolve() 通常比 **solve()** 或 **zeros()** 快得多, 若您用 `[]` 運算子約束為搜尋較小的區間, 使得其中剛好只有一個簡單解, 則速度更是快。

nSolve() 會試圖求出殘差是零的一個點, 或求出兩個位置較近的點, 且點的殘差正負號相反、殘差數值不會太大。若無法用適當數量的樣本點得到以上結果, 則傳回「找不到解」字串。

附註: 另請參考 **cSolve()**、**cZeros()**、**solve()**、**zeros()**。

$\text{nSolve}(x^2+5\cdot x-25=9,x),x<0$	-8.84429
$\text{nSolve}\left(\frac{(1+r)^{24}-1}{r}=26,r\right)r>0 \text{ and } r<0.25$	0.006886
$\text{nSolve}(x^2=-1,x)$	"No solution found"

O

OneVar

OneVar [1,]X[,*Freq*][,類別,包含]**OneVar** [n,]X1,X2[X3[,...[,X20]]]

計算多達 20 個列表的單元統計值。
stat.results 變數會儲存結果摘要(請參閱第 162 頁)。

所有列表的維數都必須相同, 包含除外。

Freq 是頻率值列表(非必要)。*Freq* 的每個元素, 可用於指定各 X 、 Y 對應資料點的出現頻率。預設值是 1。所有元素都必須是 ≥ 0 的整數。

類別是 X 對應值的數字類別代號列表。

包含是一個以上類別代號的列表。只有本列表中類別代號所對應的資料項目, 才會包含在計算作業中。

X 、 $Freq$ 、類別的任何列表中若有空元素，則所有這些列表的對應元素就會是空元素。 $X1$ 至 $X20$ 的任何列表中若有空元素，則所有這些列表的對應元素就會是空元素。如需空元素的詳細資訊，請參考第 頁碼：219 頁。

輸出變數	說明
stat. \bar{x}	x 值的平均值
stat. Σx	x 值的和
stat. Σx^2	x^2 值的和
stat.sx	x 的樣本標準差
stat. σ_x	x 的母群體標準差
stat.n	資料點數量
stat.MinX	x 值的最小值
stat.Q ₁ X	x 的第一四分位數
stat.MedianX	x 的中位數
stat.Q ₃ X	x 的第三四分位數
stat.MaxX	x 值的最大值
stat.SSX	x 的離均差平方和

or (或)

布林運算式 **lor** 布林運算式2 傳回 布林運算式

$x \geq 3$ or $x \geq 4$ $x \geq 3$

布林列表 **lor** 布林列表2 傳回 布林列表

Define $g(x) = \text{Func}$ *Done*

If $x \leq 0$ or $x \geq 5$

Goto *end*

Return $x \cdot 3$

Lbl *end*

EndFunc

布林矩陣 **lor** 布林矩陣2 傳回 布林矩陣

傳回真偽值或原始輸入的化簡形式。

$g(3)$ 9

$g(0)$ *A function did not return a value*

若其中一個運算式或兩個運算式可化簡為真，則傳回真。只有在兩個運算式求值後結果皆是偽時，才傳回偽。

附註：請參考 **xor**。

輸入範例的注意事項：關於輸入多行程式和函數定義的說明，請參閱產品手冊中的「計算工具」章節。

整數 *l* or 整數 *2* ⇒ 整數

利用 **or** 功能個別位元比較兩個實際整數。系統內部會把兩個整數轉換成有正負號的 64 位元二進位數字。比較對應的位元時，如果其中一個位元是 1，則結果是 1；如果兩個位元都是 0，則結果是 0。傳回的值代表位元結果，並會以基底模式為顯示依據。

您可以用任何整數做為基底。如果要輸入二進位或十六進位數字，則必須在前面分別加上 **0b** 或 **0h** 字首。如果沒有加上字首，就會把整數視為十進位（以 10 為底）。

如果您輸入的十進位整數太大，超出具正負號 64 位元二進位格式的範圍，系統會以對稱模數運算法使數值落入適當範圍。如需詳細資訊，請參閱

►**Base2**，頁碼：17。

附註：請參考 **xor**。

十六進位基底模式：

0h7AC36 or 0h3D5F	0h7BD7F
-------------------	---------

重要訊息：數字零，而非英文字母 O。

二進位基底模式：

0b100101 or 0b100	0b100101
-------------------	----------

附註：您可輸入最長 64 位的二進位數字（不含 **0b** 字首）。您可輸入最長 16 位的十六進位數字。

ord(字串) ⇒ 整數

ord(列表 *l*) ⇒ 列表

傳回字元字串中第一個字元的數字代號，或列表中各元素第一個字元的列表。

ord("hello")	104
char{104}	"h"
ord(char{24})	24
ord({"alpha", "beta"})	{97,98}

P

P►Rx(*rExpr*, *θExpr*) ⇒ 運算式

P►Rx(*r*列表, *θ*列表) ⇒ 列表

P►Rx(*r*矩陣, *θ*矩陣) ⇒ 矩陣

傳回 (*r*, *θ*) 成對值的等值 x 座標。

弧度角模式：

P>Rx()

目錄 >

附註：系統會根據目前的角度模式，將 θ 引數解譯為度數角、梯度角或弧度角。若引數是運算式，則可使用 °、G 或 R 來暫時覆寫角度模式設定。

附註：如果要從電腦鍵盤插入本函數，可輸入 **P@>Rx (...)**。

$P>Rx(r, \theta)$	$\cos(\theta) \cdot r$
$P>Rx(4, 60^\circ)$	2
$P>Rx\left(\{-3, 10, 1.3\}, \left\{\frac{\pi}{3}, \frac{\pi}{4}, 0\right\}\right)$	$\left\{\frac{-3}{2}, 5\sqrt{2}, 1.3\right\}$

P>Ry()

目錄 >

P>Ry(rExpr, θ Expr)⇒運算式

弧度角模式：

P>Ry(r列表, θ 列表)⇒列表

$P>Ry(r, \theta)$	$\sin(\theta) \cdot r$
$P>Ry(4, 60^\circ)$	$2 \cdot \sqrt{3}$

P>Ry(r矩陣, θ 矩陣)⇒矩陣

傳回 (r, θ) 成對值的等值 y 座標。

$P>Ry\left(\{-3, 10, 1.3\}, \left\{\frac{\pi}{3}, \frac{\pi}{4}, 0\right\}\right)$	$\left\{\frac{-3\sqrt{3}}{2}, -5\sqrt{2}, 0\right\}$
--	--

附註：系統會根據目前的角度模式，將 θ 引數解譯為度數角、弧度角或梯度角。若引數是運算式，則可使用 °、G 或 R 來暫時覆寫角度模式設定。

附註：如果要從電腦鍵盤插入本函數，可輸入 **P@>Ry (...)**。

PassErr

目錄 >

PassErr

若需 **PassErr** 的參考範例，請參考第 頁碼：176 頁 **Try** 指令下的例 2。

將錯誤傳遞到下一級。


若 *errCode* 系統變數是零，則 **PassErr** 無作用。

Try...Else...EndTry 段的 **Else** 子句應該使用 **ClrErr** 或 **PassErr**。若要處理或忽略錯誤，請使用 **ClrErr**。若不知如何處理錯誤，請用 **PassErr** 送到下一個錯誤處理程式。若已經沒有待處理的 **Try...Else...EndTry** 錯誤處理程式，則錯誤對話方塊將顯示為正常。

附註：另請參考第 頁碼：25 頁的 **ClrErr** 和第 頁碼：176 頁的 **Try**。

輸入範例的注意事項：關於輸入多行程式和函數定義的說明，請參閱產品手冊中的「計算工具」章節。

piecewise()

目錄 > 

piecewise(運算式1 [, Cond1 [, 運算式2 [, Cond2 [, ...]]])

以列表形式傳回分段函數的定義。您也可以利用範本定義分段函數。

附註：另請參考分段函數範本，頁碼：3。

Define $p(x) = \begin{cases} x, & x > 0 \\ \text{undef}, & x \leq 0 \end{cases}$	Done
$p(1)$	1
$p(-1)$	undef

poissCdf()

目錄 > 

poissCdf(λ , 下限, 上限) \Rightarrow 數字 (若下限和上限是數字) 或列表 (若下限和上限是列表)

poissCdf(λ , 上限) for $P(0 \leq X \leq \text{上限}) \Rightarrow$ 數字 (若上限是數字) 或列表 (若上限是列表)

以 λ 指定平均值計算卜松離散分布的累積機率。

對於 $P(X \leq \text{上限})$ ，請設定 下限=0

poissPdf()

目錄 > 

poissPdf(λ , XVal) \Rightarrow 數字 (若 XVal 是數字) 或列表 (若 XVal 是列表)

以 λ 指定平均值計算卜松離散分布的機率。

►Polar

目錄 > 

向量 ►Polar

附註：如果要從電腦鍵盤插入本運算子，可輸入 **@>Polar**。

以極座標形式 [$r \angle \theta$] 顯示向量。向量必須是 2 維，而且可以是列向量也可以是行向量。

附註：►Polar 是顯示格式指令，並非轉換函數。本指令只能用於輸入線末尾，而且無法更新 ans。

附註：另請參考 ►Rect，頁碼：135。

[1 3.]►Polar	[3.16228 \angle 1.24905]
[x y]►Polar	$\left[\sqrt{x^2+y^2} \quad \angle \frac{\pi \cdot \text{sign}(y)}{2} - \tan^{-1}\left(\frac{x}{y}\right) \right]$

複數值 ►Polar

以極座標形式顯示複數向量。

- 若是角度模式則傳回 $(r \angle \theta)$ 。
- 若是弧度角模式則傳回 $re^{i\theta}$ 。

複數值可以是任何複數形式，但輸入 $re^{i\theta}$ 時會導致度數角模式錯誤。

附註： 您必須使用括弧才能輸入 $(r \angle \theta)$ 極座標。

弧度角模式：

$$\begin{array}{l} (3+4 \cdot i) \blacktriangleright \text{Polar} \\ \hline e^{i \cdot \left(\frac{\pi}{2} - \tan^{-1} \left(\frac{3}{4} \right) \right)} \cdot 5 \\ \hline \left(4 \angle \frac{\pi}{3} \right) \blacktriangleright \text{Polar} \\ \hline e^{i \cdot \frac{\pi}{3}} \cdot 4 \end{array}$$

梯度角模式：

$$(4 \cdot i) \blacktriangleright \text{Polar} \qquad (4 \angle 100)$$

度數角模式：

$$(3+4 \cdot i) \blacktriangleright \text{Polar} \qquad \left(5 \angle 90 - \tan^{-1} \left(\frac{3}{4} \right) \right)$$

polyCoeffs()

$\text{polyCoeffs}(\text{Poly} [, \text{Var}]) \Rightarrow$ 列表

傳回 *Poly* 多項式的係數列表 (對 *Var* 變數)。

Poly 必須是 *Var* 中的多項式。建議不要省略 *Var*，除非 *Poly* 是單一變數中的運算式。

$$\text{polyCoeffs}(4 \cdot x^2 - 3 \cdot x + 2, x) \qquad \{4, -3, 2\}$$

$$\text{polyCoeffs}((x-1)^2 \cdot (x+2)^3) \qquad \{1, 4, 1, -10, -4, 8\}$$


展開多項式，並選取 *x* 當作省略的 *Var*。

$$\text{polyCoeffs}((x+y+z)^2, x) \qquad \{1, 2 \cdot (y+z), (y+z)^2\}$$

$$\text{polyCoeffs}((x+y+z)^2, y) \qquad \{1, 2 \cdot (x+z), (x+z)^2\}$$

$$\text{polyCoeffs}((x+y+z)^2, z) \qquad \{1, 2 \cdot (x+y), (x+y)^2\}$$

polyDegree()

目錄 > 

polyDegree(Poly [,Var])⇒值


傳回 *Poly* 多項式的次數(對 *Var* 變數)。若省略 *Var*, 則 **polyDegree()** 函數會從 *Poly* 多項式包含的變數中選取預設值。

Poly 必須是 *Var* 中的多項式。建議不要省略 *Var*, 除非 *Poly* 是單一變數中的運算式。

polyDegree(5)	0
polyDegree(ln(2)+π,x)	0
常數多項式	
<hr/>	
polyDegree(4·x ² -3·x+2,x)	2
polyDegree((x-1) ² ·(x+2) ³)	5
<hr/>	
polyDegree((x+y ² +z ³) ² ,x)	2
polyDegree((x+y ² +z ³) ² ,y)	4
<hr/>	
polyDegree((x-1) ¹⁰⁰⁰⁰ ,x)	10000

雖然次數可以開方, 但係數不行。這是因為您不需要展開多項式, 就能讓次數開方。

polyEval()

目錄 > 

polyEval(列表1, 運算式1)⇒運算式

polyEval(列表1, 列表2)⇒運算式

將第一個引數解譯為降次多項式的係數, 並傳回針對第二個引數值進行求值的多項式。

polyEval({a,b,c},x)	$a \cdot x^2 + b \cdot x + c$
polyEval({1,2,3,4},2)	26
polyEval({1,2,3,4},{2,-7})	{26,-262}

polyGcd()

目錄 > 

polyGcd(運算式1, 運算式2)⇒運算式


傳回兩個引數的最大公因數。

運算式1 與 運算式2 必須是多項式。

不得使用列表、矩陣和布林引數。

polyGcd(100,30)	10
polyGcd(x ² -1,x-1)	x-1
polyGcd(x ³ -6·x ² +11·x-6,x ² -6·x+8)	x-2

polyQuotient()

目錄 > 


polyQuotient(*Poly1*,*Poly2* [,*Var*])⇒運算式

傳回 *Poly1* 多項式除以 *Poly2* 多項式的商 (對 *Var* 指定變數)。

Poly1 與 *Poly2* 必須是 *Var* 中的多項式。建議不要省略 *Var*，除非 *Poly1* 與 *Poly2* 是相同單一變數中的運算式。

$\text{polyQuotient}(x-1, x-3)$	1
$\text{polyQuotient}(x-1, x^2-1)$	0
$\text{polyQuotient}(x^2-1, x-1)$	$x+1$
$\text{polyQuotient}(x^3-6x^2+11x-6, x^2-6x+8)$	x
$\text{polyQuotient}((x-y)\cdot(y-z), x+y+z, x)$	$y-z$
$\text{polyQuotient}((x-y)\cdot(y-z), x+y+z, y)$	$2\cdot x-y+2\cdot z$
$\text{polyQuotient}((x-y)\cdot(y-z), x+y+z, z)$	$-(x-y)$

polyRemainder()

目錄 > 


polyRemainder(*Poly1*,*Poly2* [,*Var*])⇒運算式

傳回 *Poly1* 多項式除以 *Poly2* 多項式的餘 (對 *Var* 指定變數)。

Poly1 與 *Poly2* 必須是 *Var* 中的多項式。建議不要省略 *Var*，除非 *Poly1* 與 *Poly2* 是相同單一變數中的運算式。

$\text{polyRemainder}(x-1, x-3)$	2
$\text{polyRemainder}(x-1, x^2-1)$	$x-1$
$\text{polyRemainder}(x^2-1, x-1)$	0
$\text{polyRemainder}((x-y)\cdot(y-z), x+y+z, x)$	$-(y-z)\cdot(2\cdot y+z)$
$\text{polyRemainder}((x-y)\cdot(y-z), x+y+z, y)$	$-2\cdot x^2-5\cdot x\cdot z-2\cdot z^2$
$\text{polyRemainder}((x-y)\cdot(y-z), x+y+z, z)$	$(x-y)\cdot(x+2\cdot y)$

polyRoots()

目錄 > 

polyRoots(*Poly*,*Var*)⇒列表

polyRoots(係數表)⇒列表

第一個語法 **polyRoots**(*Poly*,*Var*) 可傳回 *Poly* 多項式的實根列表 (對 *Var* 變數)。若無實根，則傳回空白列表。{ }。

Poly 必須是一個變數的多項式。

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

第二個語法 **polyRoots(係數表)** 可傳回係數表中係數的實根列表。

附註：另請參考 **cPolyRoots()**，頁碼：35。

PowerReg

PowerReg *X*, *Y* [, *Freq*] [, 類別, 包含]

計算 *X* 列表與 *Y* 列表的 $y = (a \cdot (x)^b)$ 乘幂迴歸(頻率為 *Freq*)。stat.results 變數會儲存結果摘要(請參閱第 頁碼：162 頁)。

所有列表的維數都必須相同，包含除外。

X、*Y* 是自變數和因變數列表。

Freq 是頻率值列表(非必要)。*Freq* 的每個元素，可用於指定各 *X*、*Y* 對應資料點的出現頻率。預設值是 1。所有元素都必須是 ≥ 0 的整數。

類別是 *X*、*Y* 對應資料的類別代號列表。

包含是一個以上類別代號的列表。只有本列表中類別代號所對應的資料項目，才會包含在計算作業中。

如要了解列表中有空元素時的影響，請參考“空元素”，頁碼：219。

輸出變數	說明
stat.RegEqn	迴歸方程式： $a \cdot (x)^b$
stat.a、stat.b	迴歸係數
stat.r ²	所轉換資料之線性判定的係數
stat.r	所轉換資料 ($\ln(x)$, $\ln(y)$) 的相關係數
stat.Resid	與乘幂模型相關的殘差
stat.ResidTrans	與所轉換資料之線性擬合相關的殘差
stat.XReg	實際用於迴歸的已修改 <i>X</i> 列表中的資料點列表(根據 <i>Freq</i> 、類別列表、包含類別的限制)

輸出變數	說明
stat.YReg	實際用於迴歸的已修改 Y 列表中的資料點列表(根據 Freq、類別列表、包含類別的限制)
stat.FreqReg	stat.XReg 與 stat.YReg 對應的頻率列表

Prgm

目錄 >

Prgm

區段

EndPrgm

製作使用者自行定義程式的範本。必須搭配 **Define**、**Define LibPub** 或 **Define LibPriv** 指令。

區段可以只是一個語句、由 [] 字元分隔的一連串語句，也可以是分成多行的一連串語句。

輸入範例的注意事項：關於輸入多行程式和函數定義的說明，請參閱產品手冊中的「計算工具」章節。

計算最大公因數並顯示中間結果。

```
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()

請參考 Π(), 頁碼: 207。

Product (PI)

請參考 Π(), 頁碼: 207。

product()

目錄 >

product(列表[, 起點[, 終點]]) ⇒ 運算式

傳回列表元素的乘積。起點和終點可選擇性輸入，用於指定元素範圍。

product({1,2,3,4})	24
product({2,x,y})	2·x·y
product({4,5,8,9},2,3)	40

product()

目錄 >

product(矩陣I[,起點[,終點]])⇒矩陣

傳回包含矩陣I中各行元素乘積的列向量。起點和終點可選擇性輸入，用於指定列範圍。

空元素會被忽略。如需空元素的詳細資訊，請參考第 頁碼：219 頁。

$\text{product}\left(\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}\right)$	$[28 \ 80 \ 162]$
$\text{product}\left(\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}, 1, 2\right)$	$[4 \ 10 \ 18]$

propFrac()

目錄 >

propFrac(運算式I[,Var])⇒運算式

propFrac(有理數): 傳回有理數形式的整數與分數總和，兩者正負號必須相同，且分母比分子大。

propFrac(有理式,Var): 傳回適當比例和多項式的總和(對Var變數)。在每個適當比例中，分母的Var次數必須超過分子的Var次數。系統將匯集Var的類似乘幕。各項及其因式以Var為主變數進行排序。

若省略Var，則會根據主變數展開真分數。接著先對多項式部份之係數的主變數，將係數轉為真分數，然後依此類推。

若是有理式，則 **propFrac()** 是 **expand()** 以外速度較快且較不極端的其他選擇。

您可用 **propFrac()** 函數表示帶分數，並說明帶分數的加減運算。

$\text{propFrac}\left(\frac{4}{3}\right)$	$1 + \frac{1}{3}$
$\text{propFrac}\left(\frac{-4}{3}\right)$	$-1 - \frac{1}{3}$
$\text{propFrac}\left(\frac{x^2+x+1}{x+1} + \frac{y^2+y+1}{y+1}, x\right)$	$\frac{1}{x+1} + x + \frac{y^2+y+1}{y+1}$
$\text{propFrac}(\text{Ans})$	$\frac{1}{x+1} + x + \frac{1}{y+1} + y$

$\text{propFrac}\left(\frac{11}{7}\right)$	$1 + \frac{4}{7}$
$\text{propFrac}\left(3 + \frac{1}{11} + 5 + \frac{3}{4}\right)$	$8 + \frac{37}{44}$
$\text{propFrac}\left(3 + \frac{1}{11} - \left(5 + \frac{3}{4}\right)\right)$	$-2 - \frac{29}{44}$

Q**QR**

目錄 >

QR 矩陣, **q**矩陣, **r**矩陣[,**Tol**]

m1 中的浮點數(9.) 會使得系統以浮點數形式計算結果。

計算實數或複數矩陣的 Householder QR 分解值。系統會將求出的 Q、R 矩陣儲存到指定矩陣中。Q 矩陣是單一矩陣。R 矩陣是上三角矩陣。

若任何矩陣元素的絕對值小於 *Tol* (容許值)，則亦可將此元素視為零。只有在矩陣中包含浮點數項目，而且不包含尚未賦值的任何符號變數時，才會使用此容許值。其他時候都會忽略 *Tol* (容許值)。

- 若您使用 **ctrl enter** 或將自動或近似值模式設定成近似值，則系統會利用浮點運算法執行計算作業。
- 若指令省略或未使用 *Tol*，則預設容許值的計算方式如下：
 $5E-14 \cdot \max(\dim(\text{矩陣})) \cdot \text{rowNorm}(\text{矩陣})$

進行 QR 分解的數值計算時，會進行 Householder 轉換。計算符號解時會採用 Gram-Schmidt。*qMatName* 的行向量是單範正交基底向量，跨越矩陣定義的空間。

$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix} \rightarrow mI$	$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$
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>	$\begin{bmatrix} m & -\text{sign}(m \cdot p - n \cdot o) \cdot o \\ \sqrt{m^2 + o^2} & \sqrt{m^2 + o^2} \\ o & m \cdot \text{sign}(m \cdot p - n \cdot o) \\ \sqrt{m^2 + o^2} & \sqrt{m^2 + o^2} \end{bmatrix}$
<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

QuadReg *X,Y[,Freq][,類別,包含]*

計算 *X* 列表與 *Y* 列表的 $y = a \cdot x^2 + b \cdot x + c$ 二次多項式迴歸 (頻率為 *Freq*)。
stat.results 變數會儲存結果摘要 (請參閱第 頁碼: 162 頁)。

所有列表的維數都必須相同，包含除外。

X、*Y* 是自變數和因變數列表。

Freq 是頻率值列表(非必要)。*Freq* 的每個元素,可用於指定各 X 、 Y 對應資料點的出現頻率。預設值是 1。所有元素都必須是 ≥ 0 的整數。

類別是 X 、 Y 對應資料的類別代號列表。

包含是一個以上類別代號的列表。只有本列表中類別代號所對應的資料項目,才會包含在計算作業中。

如要了解列表中有空元素時的影響,請參考“空元素”,頁碼: 219。

輸出變數	說明
stat.RegEqn	迴歸方程式: $a \cdot x^2 + b \cdot x + c$
stat.a、stat.b、stat.c	迴歸係數
stat.R ²	判定係數
stat.Resid	迴歸殘差
stat.XReg	實際用於迴歸的已修改 X 列表中的資料點列表(根據 <i>Freq</i> 、類別列表、包含類別的限制)
stat.YReg	實際用於迴歸的已修改 Y 列表中的資料點列表(根據 <i>Freq</i> 、類別列表、包含類別的限制)
stat.FreqReg	<i>stat.XReg</i> 與 <i>stat.YReg</i> 對應的頻率列表

QuartReg

QuartReg X, Y [, *Freq*] [, 類別, 包含]

計算 X 列表與 Y 列表的 $y = a \cdot x^4 + b \cdot x^3 + c \cdot x^2 + d \cdot x + e$ 四次多項式迴歸(頻率為 *Freq*)。*stat.results* 變數會儲存結果摘要(請參閱第 162 頁)。

所有列表的維數都必須相同,包含除外。

X 、 Y 是自變數和因變數列表。

Freq 是頻率值列表(非必要)。*Freq* 的每個元素,可用於指定各 X 、 Y 對應資料點的出現頻率。預設值是 1。所有元素都必須是 ≥ 0 的整數。

類別是 X 、 Y 對應資料的類別代號列表。

包含是一個以上類別代號的列表。只有本列表中類別代號所對應的資料項目，才會包含在計算作業中。

如要了解列表中有空元素時的影響，請參考“空元素”，頁碼：219。

輸出變數	說明
stat.RegEqn	迴歸方程式： $a \cdot x^4 + b \cdot x^3 + c \cdot x^2 + d \cdot x + e$
stat.a、stat.b、stat.c、stat.d、stat.e	迴歸係數
stat.R ²	判定係數
stat.Resid	迴歸殘差
stat.XReg	實際用於迴歸的已修改 X 列表中的資料點列表(根據 <i>Freq</i> 、類別列表、包含類別的限制)
stat.YReg	實際用於迴歸的已修改 Y 列表中的資料點列表(根據 <i>Freq</i> 、類別列表、包含類別的限制)
stat.FreqReg	<i>stat.XReg</i> 與 <i>stat.YReg</i> 對應的頻率列表

R

R▶P0()

R▶P0 (x 運算式, y 運算式) ⇒ 運算式

度數角模式:

R▶P0 (x 列表, y 列表) ⇒ 列表

R▶P0 (x 矩陣, y 矩陣) ⇒ 矩陣

$$R \blacktriangleright P0(x, y) = 90 \cdot \text{sign}(y) - \tan^{-1} \left(\frac{x}{y} \right)$$

傳回 (x, y) 成對引數的相等 θ 座標。

梯度角模式:

注意:系統會根據目前的角度模式設定，將結果傳回為度數角、梯度角或弧度角。

$$R \blacktriangleright P0(x, y) = 100 \cdot \text{sign}(y) - \tan^{-1} \left(\frac{x}{y} \right)$$

注意:如果要使用電腦鍵盤插入本函數，可輸入 **R@>Ptheta (...)**。

弧度角模式:

R▶Pθ(3,2)	$\tan^{-1}\left(\frac{2}{3}\right)$
R▶Pθ $\left([3 \ -4 \ 2], \left[0 \ \frac{\pi}{4} \ 1.5\right]\right)$	$\left[0 \ \tan^{-1}\left(\frac{16}{\pi}\right) + \frac{\pi}{2} \ 0.643501\right]$

R▶Pr(x運算式,y運算式) ⇒ 運算式

弧度角模式:

R▶Pr(x列表,y列表) ⇒ 列表

$$\frac{R\blacktriangleright Pr(3,2)}{\sqrt{13}}$$

R▶Pr(x矩陣,y矩陣) ⇒ 矩陣

$$\frac{R\blacktriangleright Pr(x,y)}{\sqrt{x^2+y^2}}$$

傳回(x,y)成對引數的等值r座標。

$$\frac{R\blacktriangleright Pr\left([3 \ -4 \ 2], \left[0 \ \frac{\pi}{4} \ 1.5\right]\right)}{\left[3 \ \frac{\sqrt{\pi^2+256}}{4} \ 2.5\right]}$$

注意:如果要使用電腦鍵盤插入本函數,可輸入 **R@>Pr(...)**。

運算式I▶Rad ⇒ 運算式

度數角模式:

把引數轉成弧度角測量單位。

$$\frac{(1.5)\blacktriangleright Rad}{(0.02618)^r}$$

注意:如果要使用電腦鍵盤插入本運算子,可輸入 **@>Rad**。

梯度角模式:

$$\frac{(1.5)\blacktriangleright Rad}{(0.023562)^r}$$

rand() ⇒ 運算式

rand(試驗次數) ⇒ 列表

設定亂數種子。

rand() 會傳回 0 與 1 之間的隨機值。

$$\frac{RandSeed \ 1147}{Done}$$

rand(試驗次數) 會傳回包含 0 與 1 之間試驗次數個隨機值的列表。

$$\frac{rand(2)}{\{0.158206, 0.717917\}}$$

randBin()

目錄 > 

randBin(n, p) ⇒ 運算式


randBin(n, p , 試驗次數) ⇒ 列表

randBin(n, p) 會從指定的二項式分佈傳回隨機實數。

randBin(n, p , 試驗次數) 會從指定的二項分佈，傳回包含試驗次數個隨機實數的列表。

randBin(80,0.5)	42
randBin(80,0.5,3)	{41,32,39}

randInt()

目錄 > 

randInt(下限, 上限)

⇒ 運算式

randInt(下限, 上限, 試驗次數) ⇒ 列表

randInt(下限, 上限)

會傳回下限和上限整數邊界指定範圍內的隨機整數。

randInt(下限, 上限,

試驗次數) 會傳回指定範圍內包含試驗次數個隨機整數的列表。

randInt(3,10)	5
randInt(3,10,4)	{9,7,5,8}

randMat()

目錄 > 

randMat(列數, 欄數) ⇒ 矩陣


傳回指定維度的矩陣，其中元素值是介於 -9 至 9 之間的整數。

兩個引數都必須化簡成整數。

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								

注意:每按一次 **enter**，本矩陣中的數值就會變換。

randNorm()


目錄 > 

randNorm(μ, σ) ⇒ 運算式

randNorm(μ, σ , 試驗次數) ⇒ 列表

RandSeed 1147	Done
randNorm(0,1)	0.492541
randNorm(3,4.5)	-3.54356


randNorm()

目錄 > 

randNorm(μ, σ) 會從指定的常態分佈傳回十進位數字。結果可以是任何實數，但會集中在 $[\mu-3\cdot\sigma, \mu+3\cdot\sigma]$ 區間範圍內。

randNorm($\mu, \sigma, \text{試驗次數}$) 會從指定的常態分佈，傳回包含試驗次數個十進位數字的列表。

randPoly()

目錄 > 


randPoly(變數, 階數) \Rightarrow 運算式

傳回對變數的指定階數多項式，其係數是介於 -9 至 9 範圍內的隨機整數。首項係數不得為零。

階數必須是 0 至 99。

RandSeed 1147	Done
randPoly(x,5)	$-2\cdot x^5+3\cdot x^4-6\cdot x^3+4\cdot x-6$

randSamp()


目錄 > 

randSamp(列表, 試驗次數[, 不放回抽樣]) \Rightarrow 列表

傳回列表，其中元素包含列表中以試驗次數為個數所試驗的隨機樣本，且可選擇替換樣本 (不放回抽樣=0) 或不替換樣本 (不放回抽樣=1)。預設是替換樣本。

Define list3={1,2,3,4,5}	Done
Define list4=randSamp(list3,6)	Done
list4	{2,3,4,3,1,2}

RandSeed


目錄 > 

RandSeed 數字

若數字 = 0，則以亂數產生程式的出廠預設值為種子。若數字 $\neq 0$ ，則用於產生兩個亂數種子，並儲存在系統變數種子1 和種子2 中。

RandSeed 1147	Done
rand()	0.158206

real()

目錄 > 

real(運算式 I) \Rightarrow 運算式

傳回引數的實部。

real(2+3·i)	2
real(z)	z
real(x+i·y)	x

注意:系統會把所有未定義的變數視為實變數。另請參考此處的 **imag()** page 83.

real(列表 l) ⇒ 列表

傳回所有元素的實部。

$$\text{real}\left(\{a+i\cdot b, 3, i\}\right) \quad \{a, 3, 0\}$$

real(矩陣 l) ⇒ 矩陣

傳回所有元素的實部。

$$\text{real}\left(\begin{bmatrix} a+i\cdot b & 3 \\ c & i \end{bmatrix}\right) \quad \begin{bmatrix} a & 3 \\ c & 0 \end{bmatrix}$$

► Rect

向量 ► Rect

注意:如果要使用電腦鍵盤插入本運算子，可輸入 **@>Rect**。

以直角座標形式 [x, y, z] 顯示 **向量**。向量必須是 2 維或 3 維，而且可以是列向量或行向量。

$$\left(3 \angle \frac{\pi}{4} \angle \frac{\pi}{6}\right) \text{►Rect} \quad \begin{bmatrix} 3\cdot\sqrt{2} & 3\cdot\sqrt{2} & 3\cdot\sqrt{3} \\ 4 & 4 & 2 \end{bmatrix}$$

$$\left[a \angle b \angle c\right] \quad \begin{bmatrix} a\cdot\cos(b)\cdot\sin(c) & a\cdot\sin(b)\cdot\sin(c) & a\cdot\cos(c) \end{bmatrix}$$

注意:►Rect 是顯示格式指令，並非轉換函數。本指令只能用於輸入列末尾，而且無法更新 *ans*。

注意:另請參考此處的 ►Polar 頁碼: 122.

複數值 ► Rect

以直角座標形式 a+bi 顯示 **複數值**。複數值可以是任何複數形式，但輸入 $\text{re}^{i\theta}$ 時會導致度數角模式錯誤。

注意:您必須使用括弧才能輸入 (r∠θ) 極座標。

弧度角模式:

$$\left(4\cdot e^{\frac{\pi}{3}}\right) \text{►Rect} \quad 4\cdot e^{\frac{\pi}{3}}$$

$$\left(4 \angle \frac{\pi}{3}\right) \text{►Rect} \quad 2+2\cdot\sqrt{3}\cdot i$$

梯度角模式:

$$\left((1 \angle 100)\right) \text{►Rect} \quad i$$

度數角模式:

$$\left((4 \angle 60)\right) \text{►Rect} \quad 2+2\cdot\sqrt{3}\cdot i$$

注意:若要輸入 ∠，請從「目錄」的符號表中選取。

ref(矩陣I, 容許值) ⇒ 矩陣

傳回矩陣I的列梯形。

若任何矩陣元素的絕對值小於容許值，則亦可將此元素視為零。只有在矩陣中包含浮點數項目，而且不包含尚未賦值的任何符號變數時，才會使用此容許值。其他時候都會忽略容許值。

- 若您使用 **ctrl enter** 或將**自動或近似值**模式設定成近似值，則系統會利用浮點運算法執行計算作業。
- 若指令省略或未使用容許值，則預設容許值的計算方式如下：
 $5E-14 \cdot \max(\dim(\text{矩陣}I)) \cdot \text{rowNorm}(\text{矩陣}I)$

請避免在矩陣I中加入未定義的元素，否則可能出現意外結果。

例如若未定義以下運算式中的 *a*，則會出現警告訊息，並顯示以下結果：

$$\text{ref}\left(\begin{bmatrix} a & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}\right) \quad \begin{bmatrix} 1 & \frac{1}{a} & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

之所以出現警告，是因為若 $a=0$ ，則 $1/a$ 廣義元素會無效。

若要避免這個情形，可先儲存 *a* 的值，或用下例所示的 (||) 取代值。

$$\text{ref}\left(\begin{bmatrix} a & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}\right) | a=0 \quad \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}$$

注意:另請參考此處的 **rref()** page 144.

$$\text{ref}\left(\begin{bmatrix} -2 & -2 & 0 & -6 \\ 1 & -1 & 9 & -9 \\ -5 & 2 & 4 & -4 \end{bmatrix}\right) \quad \begin{bmatrix} 1 & \frac{-2}{5} & \frac{-4}{5} & \frac{4}{5} \\ 0 & 1 & \frac{4}{7} & \frac{11}{7} \\ 0 & 0 & 1 & \frac{-62}{71} \end{bmatrix}$$

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} \rightarrow m1 \quad \begin{bmatrix} a & b \\ c & d \end{bmatrix}$$

$$\text{ref}(m1) \quad \begin{bmatrix} 1 & \frac{d}{c} \\ 0 & 1 \end{bmatrix}$$

RefreshProbeVars

RefreshProbeVars

範例

可讓您透過 TI-Basic 程式存取所有連接感應器探針的感應器數據。

Define temp()=

狀態
變數
值
狀態
變數 正常 (程式繼續運作)
=0

Vernier DataQuest™ 應用程式使用數據收集模式。

狀態
變數
=1 **注意:** Vernier DataQuest™ 應用程式必須使用儀表模式, 才可使此指令運作。 

狀態
變數
=2 Vernier DataQuest™ 應用程式尚未啟動。

狀態
變數
=3 Vernier DataQuest™ 應用程式已啟動, 但尚未連接任何探針。

```
Prgm
© Check if system is ready
RefreshProbeVars status
If status=0 Then
Disp "ready"
For n,1,50
RefreshProbeVars status
temperature:=meter.temperature
Disp "溫度: ",temperature
If temperature>30 Then
Disp "Too hot"
EndIf
© Wait for 1 second between
samples
Wait 1
EndFor
Else
Disp "尚未準備好, 請稍後再
試! "
EndIf
EndPrgm
```

注意:此指令碼可搭配 TI-Innovator™ 分享器使用。

remain()

remain(運算式1, 運算式2) ⇒ 運算式

remain(列表1, 列表2) ⇒ 列表
remain(矩陣1, 矩陣2) ⇒ 矩陣

依照下列恆等式的定義, 傳回第一個引數對第二個引數的餘:

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}

remain(x,0) x
 remain(x,y) $x - y \cdot \text{Part}(x/y)$

請注意到 $\text{remain}(-x,y) - \text{remain}(x,y)$ 這個結果。結果如果不是零，就是與第一個引數有相同符號。

注意：另請參考此處的 `mod()` 頁碼：107。

$$\text{remain} \left(\begin{pmatrix} 9 & -7 \\ 6 & 4 \end{pmatrix}, \begin{pmatrix} 4 & 3 \\ 4 & -3 \end{pmatrix} \right) = \begin{pmatrix} 1 & -1 \\ 2 & 1 \end{pmatrix}$$

Request

Request 提示字串, 變數[, 顯示旗標 [, 狀態變數]]

Request 提示字串, *func*(引數1, ... 引數)[, 顯示旗標 [, 狀態變數]]

程式設計指令：程式暫停執行並顯示包含提示字串訊息的對話方塊，以及讓使用者輸入回覆的方塊。

使用者輸入回覆並按一下 **[確定]** 時，輸入方塊的內容就會指派為變數 *var* 的值。

若使用者按一下 **[取消]**，則程式會繼續進行，且不會接受任何輸入。如果變數已完成定義，則程式會使用先前變數的值。

選用的顯示旗標引數可以是任何運算式。

- 若省略顯示旗標或其求值是 **1**，則在計算工具歷史記錄中顯示提示訊息和使用者的回覆。
- 若顯示旗標的求值是 **0**，則不在歷史記錄中顯示提示訊息和回覆。

選用的狀態變數引數可使程式判斷使用者取消對話方塊的方式。請注意，狀態變數需要顯示旗標引數。

- 如果使用者按一下 **[確定]** 或按 **Enter** 或 **Ctrl+Enter**，則狀態變數的值就會設定為 **1**。
- 否則，狀態變數的值就會設定為 **0**。

寫出以下程式：

```
Define request_demo()=Prgm
  Request "半徑:",r
  Disp "面積 = ",pi*r^2
EndPrgm
```

執行程式並輸入回覆：

request_demo()



選取 **[確定]** 後，計算出結果：

半徑 = 6/2
 面積 = 28.2743

寫出以下程式：

```
Define polynomial()=Prgm
  Request "輸入一個x的多項式:",p
  (x)
  Disp "其實根為:",polyRoots(p
  (x),x)
EndPrgm
```

執行程式並輸入回覆：

polynomial()


`func()` 引數可讓程式把使用者的回覆儲存成函數定義。這個語法的功能，就像使用者執行以下指令：

Define *func*(引數1, ... 引數) = 使用者的回覆

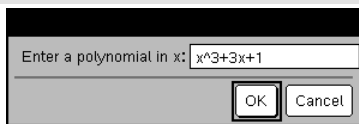
然後，程式就能使用已定義函數 *func* ()。提示字串應該引導使用者輸入適當的使用者的回覆，以便完成整個函數定義。

注意：您可以在使用者自行定義的程式中使用 **Request** 指令，但不能用在函數中。

若要停止包含在無限迴圈內 **Request** 指令的程式，請執行下列動作：

- **計算機：** 按住  鍵並重複按 **Enter** 鍵。
- **Windows®：** 按住 **F12** 鍵並重複按 **Enter** 鍵。
- **Macintosh®：** 按住 **F5** 鍵並重複按 **Enter** 鍵。
- **iPad®：** 應用程式顯示提示。您可以繼續等待或取消。

注意：另請參考 [此處的](#)，page 139。



輸入 x^3+3x+1 並選取 **[確定]** 後，計算出結果：

其實根為： $\{-0.322185\}$

RequestStr

RequestStr 提示字串, 變數[, 顯示旗標]

程式設計指令:與 **Request** 指令的第一個語法原理相同，不過本指令一律將使用者的回覆視為字串。相反地，除非使用者把回覆放在引號(“”)中，否則 **Request** 指令會將回覆解譯成運算式。

注意：您可以在使用者自行定義的程式中使用 **RequestStr** 指令，但不能用在函數中。

若要停止包含在無限迴圈內 **RequestStr** 指令的程式，請執行下列動作：

- **計算機：** 按住  鍵並重複按 **Enter** 鍵。

寫出以下程式：

```
Define requestStr_demo()=Prgm
    RequestStr “你的姓名:”,name,0
    Disp “回應具有 “,dim(name),” 字元。”
EndPrgm
```

執行程式並輸入回覆：

requestStr_demo()



鍵。

- **Windows®**: 按住 **F12** 鍵並重複按 **Enter** 鍵。
- **Macintosh®**: 按住 **F5** 鍵並重複按 **Enter** 鍵。
- **iPad®**: 應用程式顯示提示。您可以繼續等待或取消。

注意:另請參考 [此處的](#), page 138.

選取 **[確定]** 後計算出結果(請注意, **顯示旗標** 指數如果是 **0**, 則歷史記錄中會省略提示訊息和回覆):

```
requestStr_demo()
```

回應具有 5 個字元。

Return

Return [運算式]

傳回運算式作為函數結果。用於 **Func...EndFunc** 區段。

注意:若要結束程式, 請在 **Prgm...EndPrgm** 區段中使用 **Return** 並且不加引數。

輸入範例的注意事項: 關於輸入多行程式和函數定義的說明, 請參閱產品手冊中的 [「計算工具」](#) 章節。

```
Define factorial (nn)=
Func
Local answer,counter
1→answer
For counter,1,nn
answer·counter→answer
EndFor
Return answer|
EndFunc

factorial (3) 6
```

right()

right(列表 *I*, 數字) ⇒ 列表

傳回列表 *I* 中從右邊算來的數字個元素。

若省略數字, 則傳回整個列表 *I*。

right(來源字串 *S*, 數字) ⇒ 字串

傳回來源字串中從右邊算來的數字個字元。

若省略數字, 則傳回整個來源字串。

right(比較) ⇒ 運算式

傳回方程式或不等式右邊的部份。

```
right({1,3,-2,4},3) {3,-2,4}
```

```
right("Hello",2) "lo"
```

```
right(x<3) 3
```

rk23 ()

rk23(運算式, 變數, 因變數, {變數 *0*, 最大變數}, 因變數 *0*, 變數步階 [*diff*tol])

微分方程式:

⇒ 矩陣

rk23(**運算式系統**, **變數**, **因變序列表**, {**變數0**, **最大變數**}, **因變序列表0**, **變數步階**[, **diftol**]) ⇒ 矩陣

rk23(**運算式列表**, **變數**, **因變序列表**, {**變數0**, **最大變數**}, **因變序列表0**, **變數步階**[, **diftol**]) ⇒ 矩陣

使用隆巨—庫塔方法求解方程式組

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

其中, **depVar**(**變數0**)=**因變數0**, 區間為 [**變數0**, **最大變數**]。傳回一個矩陣, 其中第一列定義如**變數步階**所定義的**變數**輸出值。第二列會定義對應**變數值**的第一個求解元素的值, 依此類推。

位於右邊的**運算式**用於定義常微分方程式 (ODE)。

位於右邊的**運算式系統**, 用於定義 ODE 方程式組 (對應於**因變數序列表**中的**因變數順序**)。

位於右邊的**運算式列表**, 用於定義 ODE 方程式組 (對應於**因變數序列表**中的**因變數順序**)。

變數是自變數。

因變數序列表是顯示**因變數**的列表。

{**變數0**, **最大變數**} 是一個雙元素列表, 表示函數從**變數0**積分至**最大變數**。

因變數序列表0是**因變數**初始值的列表。

如果**變數步階**計算出非零數字:**sign**(**變數步階**) = **sign**(**最大變數**-**變數0**), 且傳回 **變數0+i*變數步階**對所有的 **i=0,1,2,...** 的解, 使 **變數0+i*變數步階**位於 [**變數0**, **最大變數**] (在**最大變數**中可能沒有解)。

如果**變數步階**計算出零, 會傳回在「隆巨—庫塔」之**變數值**的解。

$y' = 0.001 \cdot y \cdot (100 - y)$ 與 $y(0) = 10$

$$\text{rk23}\left\{0.001 \cdot y \cdot (100 - y), t, y, \{0, 100\}, 10, 1\right\}$$

0.	1.	2.	3.	4.
10.	10.9367	11.9493	13.042	14.2

若要看到完整結果, 請按 , 然後使用 與 移動游標。

與 **diftol** 設為 $1.E-6$ 時相同的方程式

$$\text{rk23}\left\{0.001 \cdot y \cdot (100 - y), t, y, \{0, 100\}, 10, 1, 1.E-6\right\}$$

0.	1.	2.	3.	4.
10.	10.9367	11.9495	13.0423	14.2189

將以上的結果與使用 **deSolve**() 和 **seqGen**() 得到的 CAS 精確解作比較:

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

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

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

{10., 10.9367, 11.9494, 13.0423, 14.2189, 15.4}

方程式組:


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

其中, $y1(\theta) = 2$ 及 $y2(\theta) = 5$

$$\text{rk23}\left\{\begin{cases} -y1 + 0.1 \cdot y1 \cdot y2 \\ 3 \cdot y2 - y1 \cdot y2 \end{cases}, t, \{y1, y2\}, \{0, 5\}, \{2, 5\}, 1\right\}$$

0.	1.	2.	3.	4.
2.	1.94103	4.78694	3.25253	1.82848
5.	16.8311	12.3133	3.51112	6.27245

rotate()

目錄 > 

傳回列表 I 左右移動互換位數個元素後的副本。請勿更改列表 I 。

若互換位數是正值，則向左輪替。若互換位數是負值，則向右輪替。預設值是 -1 (向右互換一個元素)。

rotate(字串 I , 互換位數) \Rightarrow 字串

傳回字串 I 左右移動互換位數個字元後的副本。請勿更改字串 I 。

若互換位數是正值，則向左輪替。若互換位數是負值，則向右輪替。預設值是 -1 (向右互換一個字元)。

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

$\text{rotate}(\text{"abcd"})$	"dabc"
$\text{rotate}(\text{"abcd"},-2)$	"cdab"
$\text{rotate}(\text{"abcd"},1)$	"bcda"

round()

目錄 > 

round(運算式 I , 位數) \Rightarrow 運算式

傳回四捨五入到小數點後指定位數的引數。

位數必須是在 0 至 12 範圍內的整數。若省略位數，則傳回四捨五入到 12 個有效位數的引數。

注意:顯示位數模式可能影響顯示方式。

round(列表 I , 位數) \Rightarrow 列表

傳回其中元素四捨五入到指定位數的列表。

round(矩陣 I , 位數) \Rightarrow 矩陣


傳回其中元素四捨五入到指定位數的矩陣。

$\text{round}(1.234567,3)$	1.235
----------------------------	-------

$\text{round}(\{\pi,\sqrt{2},\ln(2)\},4)$	$\{3.1416,1.4142,0.6931\}$
---	----------------------------

$\text{round}\left(\begin{bmatrix} \ln(5) & \ln(3) \\ \pi & e^1 \end{bmatrix},1\right)$	$\begin{bmatrix} 1.6 & 1.1 \\ 3.1 & 2.7 \end{bmatrix}$
---	--

rowAdd()

目錄 > 

rowAdd(矩陣 I , $rIndex1$, $rIndex2$) \Rightarrow 矩陣

傳回矩陣 I 的副本，其中 $rIndex1$ 與 $rIndex2$ 兩列的總和取代 $rIndex2$ 列。

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

$\text{rowAdd}\left(\begin{bmatrix} a & b \\ c & d \end{bmatrix},1,2\right)$	$\begin{bmatrix} a & b \\ a+c & b+d \end{bmatrix}$
--	--

rowDim()

目錄 >

rowDim(矩陣) ⇒ 運算式

傳回矩陣列數。

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

注意:另請參考此處的 colDim() 頁碼: 25.

rowNorm()

目錄 >

rowNorm(矩陣) ⇒ 運算式

傳回矩陣中各列元素絕對值的總和最大值。

rowNorm($\begin{bmatrix} -5 & 6 & -7 \\ 3 & 4 & 9 \\ 9 & -9 & -7 \end{bmatrix}$)	25
--	----

注意:矩陣的所有元素都必須化簡成數字。另請參考此處的 colNorm() 頁碼: 26.

rowSwap()

目錄 >

rowSwap(矩陣I, rIndex1, rIndex2) ⇒ 矩陣

傳回 rIndex1 與 rIndex2 兩列互換的矩陣I。

$\begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix} \rightarrow mat$	$\begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix}$
rowSwap(mat,1,3)	$\begin{bmatrix} 5 & 6 \\ 3 & 4 \\ 1 & 2 \end{bmatrix}$

rref()

目錄 >

rref(矩陣I[, 容許值]) ⇒ 矩陣

傳回矩陣I的列簡化梯形矩陣。

rref($\begin{bmatrix} -2 & -2 & 0 & -6 \\ 1 & -1 & 9 & -9 \\ -5 & 2 & 4 & -4 \end{bmatrix}$)	$\begin{bmatrix} 1 & 0 & 0 & \frac{66}{71} \\ 0 & 1 & 0 & \frac{147}{71} \\ 0 & 0 & 1 & \frac{-62}{71} \end{bmatrix}$
--	---

若任何矩陣元素的絕對值小於容許值,則亦可將此元素視為零。只有在矩陣中包含浮點數項目,而且不包含尚未賦值的任何符號變數時,才會使用此容許值。其他時候都會忽略容許值。

rref($\begin{bmatrix} a & b \\ c & d \end{bmatrix}$)	$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$
--	--

- 若您使用 **ctrl** **enter** 或將自動或近似值模式設定成近似值,則系統會利用浮點運算法執行計算作業。
- 若指令省略或未使用容許值,則預設容許值的計算方式如下:
 $5E-14 \cdot \max(\text{dim}(\text{矩陣}I)) \cdot \text{rowNorm}(\text{矩陣}I)$

陣 *I*)注意:另請參考此處的 **ref()** page 136.

S

sec()

 鍵sec(運算式 *I*) ⇒ 運算式

度數角模式:

sec(列表 *I*) ⇒ 列表傳回 運算式 *I* 的正割值, 或傳回包含列表 *I* 中各元素之正割的列表。

$$\sec(45) \quad \sqrt{2}$$

$$\sec(\{1,2,3,4\}) \quad \left\{ \frac{1}{\cos(1)}, 1.00081, \frac{1}{\cos(4)} \right\}$$

附註: 系統會根據目前的角度模式設定, 將引數解譯為度數角、梯度角或弧度角。您可使用 °、 G 或 r 來暫時覆寫角度模式。

sec⁻¹() 鍵sec⁻¹(運算式 *I*) ⇒ 運算式

度數角模式:

sec⁻¹(列表 *I*) ⇒ 列表傳回正割值是 運算式 *I* 的角度, 或傳回包含列表 *I* 中各元素之反正割值的列表。

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

梯度角模式:

附註: 系統會根據目前的角度模式設定, 將結果傳回為度數角、梯度角或弧度角。


$$\sec^{-1}(\sqrt{2}) \quad 50$$

附註: 如果要從鍵盤插入本函數, 可輸入 **arcsec(...)**。

弧度角模式:

$$\sec^{-1}(\{1,2,5\}) \quad \left\{ 0, \frac{\pi}{3}, \cos^{-1}\left(\frac{1}{5}\right) \right\}$$

sech()

目錄 > 

sech(運算式 *I*) ⇒ 運算式


$\text{sech}(3)$	$\frac{1}{\cosh(3)}$
------------------	----------------------

sech(列表 *I*) ⇒ 列表

傳回 運算式 *I* 的雙曲正割值，或傳回包含列表 *I* 各元素之雙曲正割值的列表。

$\text{sech}\{1,2,3,4\}$	$\left\{ \frac{1}{\cosh(1)}, 0.198522, \frac{1}{\cosh(4)} \right\}$
--------------------------	---

sech⁻¹()

目錄 > 

sech⁻¹(運算式 *I*) ⇒ 運算式

弧度角與直角複數模式：

sech⁻¹(列表 *I*) ⇒ 列表

傳回 運算式 *I* 的反雙曲正割值，或傳回包含列表 *I* 中各元素之反雙曲正割值的列表。

$\text{sech}^{-1}(1)$	0
$\text{sech}^{-1}\{1, -2, 2, 1\}$	$\left\{ 0, \frac{2 \cdot \pi}{3}, i, 8.8 \cdot 10^{-15} + 1.07448 \cdot i \right\}$

附註：如果要從鍵盤插入本函數，可輸入 **arcsech(...)**。

Send

分享器功能表

Send *exprOrString1* [, *exprOrString2*] ...

程式設計指令：傳送一個或多個 TI-Innovator™ Hub 指令到已連接的分享器。

exprOrString 必須是有效的 TI-Innovator™ Hub 指令。*exprOrString* 通常包含 "SET ..." 指令以控制裝置，或包含 "READ ..." 指令以要求資料。

引數會連續傳送到分享器。

附註：您可以在使用者自行定義的程式中使用 **Send** 指令，但無法在函數中使用。

附註：另請參考 **Get** (頁碼：73)、**GetStr** (頁碼：79) 和 **eval()** (頁碼：59)。

例如：將內建 RGB LED 的藍色元素開啟 0.5 秒。

Send "SET COLOR.BLUE ON TIME .5"	Done
----------------------------------	------

例如：要求分享器內建亮度等級感應器目前的值。**Get** 指令會擷取數值，並將數值指定給變數 *lightval*。

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

例如：將完成計算的頻率傳送到分享器內建的喇叭。使用特殊變數 *iostr.SendAns* 可顯示分享器指令，並將求得的運算式包含在內。

$n:=50$	50
$m:=4$	4
Send "SET SOUND eval(m·n)"	Done
<i>iostr.SendAns</i>	"SET SOUND 200"

seq()

目錄 >

seq(*Expr*, *Var*, 低, 高[, 步階]) ⇒ 列表

從 Low 到 High 增加 Var 值 (以 Step 為增量級距), 對 Expr 求值, 並以列表將結果傳回。seq() 執行完成後, Var 的原始內容仍然存在。

Step 的預設值 = 1。

$\text{seq}\left(n^2, n, 1, 6\right)$	$\{1, 4, 9, 16, 25, 36\}$
$\text{seq}\left(\frac{1}{n}, n, 1, 10, 2\right)$	$\left\{1, \frac{1}{3}, \frac{1}{5}, \frac{1}{7}, \frac{1}{9}\right\}$
$\text{sum}\left(\text{seq}\left(\frac{1}{n^2}, n, 1, 10, 1\right)\right)$	$\frac{1968329}{1270080}$

附註: 強迫表示結果近似值,

計算機: 按 .

Windows®: 按 **Ctrl+Enter**。

Macintosh®: 按 +Enter。

iPad®: 按住 enter 然後選擇 .

$\text{sum}\left(\text{seq}\left(\frac{1}{n^2}, n, 1, 10, 1\right)\right)$	1.54977
--	---------

seqGen()

目錄 >

seqGen(*Expr*, *Var*, *depVar*, {*Var0*, *VarMax*}[, *ListOfInitTerms* [, *VarStep* [, *CeilingValue*]]) ⇒ 列表

針對序列 $depVar(Var)=Expr$ 產生項列表, 如下所示: 從 *Var0* 到 *VarMax* 增加自變數 *Var* 的值 (以 *VarStep* 為增量級距), 使用 *Expr* 公式及 *ListOfInitTerms* 求 *depVar(Var)* 對應於 *Var* 的值, 並將結果傳回為列表。

seqGen(*ListOrSystemOfExpr*, *Var*, *ListOfDepVars*, {*Var0*, *VarMax*}[, *MatrixOfInitTerms* [, *VarStep* [, *CeilingValue*]]) ⇒ 矩陣

產生序列 $u(n) = u(n-1)^2/2$ 的前 5 項, 其中 $u(1)=2$ 且 $VarStep=1$ 。

$\text{seqGen}\left(\frac{(u(n-1))^2}{n}, n, u, \{1, 5\}, \{2\}\right)$	$\left\{2, 2, \frac{4}{3}, \frac{4}{9}, \frac{16}{405}\right\}$
---	---

Var0=2 的範例:

$\text{seqGen}\left(\frac{u(n-1)+1}{n}, n, u, \{2, 5\}, \{3\}\right)$	$\left\{3, \frac{4}{3}, \frac{7}{12}, \frac{19}{60}\right\}$
---	--

seqGen()

針對序列 $ListOfDepVars(Var)$
 $=ListOrSystemOfExpr$ 產生方程組 (或列表) 的項矩陣, 如下所示: 從 $Var0$ 到 $VarMax$ 增加自變數 Var 的值 (以 $VarStep$ 為增量級距), 使用 $ListOrSystemOfExpr$ 公式及 $MatrixOfInitTerms$ 求 $ListOfDepVars(Var)$ 對應於 Var 的值, 並將結果傳回為矩陣。

seqGen() 執行完成後, Var 的原始內容不會更改。

$VarStep$ 的預設值 = 1。

初始項為符號的範例:

$$\text{seqGen}\left\{u(n-1)+2n,u,\{1,5\},\{a\}\right. \\ \left.\{a,a+2,a+4,a+6,a+8\}\right.$$

兩個序列的方程組:

$$\text{seqGen}\left\{\left[\frac{1}{n}, \frac{u2(n-1)}{2}+u1(n-1)\right], n, \{u1, u2\}, \{1, 5\}, \left[-\right]\right. \\ \left. \begin{bmatrix} 1 & \frac{1}{2} & \frac{1}{3} & \frac{1}{4} & \frac{1}{5} \\ 2 & 2 & \frac{3}{2} & \frac{13}{2} & \frac{19}{2} \end{bmatrix} \right.$$

附註: 以上初始項矩陣中的 $Void()$ 是用來指明 $u1(n)$ 的初始項是使用顯序列公式 $u1(n)=1/n$ 所計算的。

seqn()

seqn($Expr(u, n [, ListOfInitTerms [, nMax [, CeilingValue]])$) ⇒ 列表

針對序列 $u(n)=Expr(u, n)$ 產生項列表, 如下所示: 從 1 到 $nMax$ 增加 n 的值 (以 1 為增量級距), 使用 $Expr(u, n)$ 公式及 $ListOfInitTerms$ 對 n 的對應值求 $u(n)$ 的值, 並將結果傳回為列表。

seqn($Expr(n [, nMax [, CeilingValue]])$) ⇒ 列表

針對非遞迴序列 $u(n)=Expr(n)$ 產生項列表, 如下所示: 從 1 到 $nMax$ 增加 n 的值 (以 1 為增量級距), 使用 $Expr(n)$ 公式求 $u(n)$ 對應於 n 的值, 並將結果傳回為列表。

如果沒有 $nMax$, 設定 $nMax$ 為 2500

如果 $nMax=0$, 則設定 $nMax$ 為 2500

附註: **seqn()** 會呼叫 **seqGen()**, 其中 $n0=1$ 且 $nstep=1$

產生序列 $u(n)=u(n-1)/2$ 的前 6 項, 其中 $u(1)=2$ 。

$$\text{seqn}\left\{\frac{u(n-1)}{n}, \{2\}, 6\right. \\ \left. \left\{2, 1, \frac{1}{3}, \frac{1}{12}, \frac{1}{60}, \frac{1}{360}\right\}\right.$$

$$\text{seqn}\left\{\frac{1}{n^2}, 6\right. \\ \left. \left\{1, \frac{1}{4}, \frac{1}{9}, \frac{1}{16}, \frac{1}{25}, \frac{1}{36}\right\}\right.$$

series(運算式1, Var, 階數 [, 點]) ⇒ 運算式

series(運算式1, Var, 階數 [, 點]) | Var > 點 ⇒ 運算式

series(運算式1, Var, 階數 [, 點]) | Var < 點 ⇒ 運算式

傳回代表運算式1的廣義冪級數截斷式，且該運算式繞著點展開（透過階數）。階數可以是任何有理數。（Var - 點）求出的乘冪可以是負指數，也可以是分數指數。這些乘冪的係數可包括（Var - 點）的對數，以及受制於（Var - 點）所有乘冪（指數符號相同）之 Var 的其他函數。

點的預設值是 0。點可以是 ∞ 也可以是 $-\infty$ 。若是這兩種情形，則展開時是透過 $1/(Var - 點)$ 中的階數次數。

series(...) 在無法求解時會傳回 **series(...)**，例如針對 $z=0$ 的 $\sin(1/z)$ 、 $z=0$ 的 $e^{-1/z}$ ，或 $z = \infty$ 或 $-\infty$ 的 e^z 這類實質奇點時。

若級數或其中一個導數在點處有跳躍不連續性，則結果可能包含 **sign(...)** 形式的子運算式，或實數展開變數的 **abs(...)**，或複數展開變數（以 **[]** 結尾）的 **(-1)^{floor(...angle(...))}**。若只需將級數用於點其中一側的值，請加上 **[| Var > 點|]**、**[| Var < 點|]**、**[| Var ≥ 點|]** 或 **[| Var ≤ 點|]** 的其中一項，以便簡化結果。

series() 可為不定積分和定積分提供符號近似值（若以其他方式無法求得符號解）。

series() 分布在第一個引數列表和矩陣中。

series() 是 **taylor()** 的廣義版本。

如右邊最後一例所示，**series(...)** 求出結果的顯示常式下游可能重排各項順序，讓最左邊的項不是主要項。

$$\text{series}\left(\frac{1-\cos(x-1)}{(x-1)^2}, x, 4, 1\right) \quad \frac{1}{2} \frac{(x-1)^2}{24} + \frac{(x-1)^4}{720}$$

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

$$\text{series}\left(\left(1 + \frac{1}{n}\right)^n, n, 2, \infty\right) \quad e - \frac{e}{2 \cdot n} + \frac{11 \cdot e}{24 \cdot n^2}$$

$$\text{series}\left(\tan^{-1}\left(\frac{1}{x}\right), x, 5\right), x > 0 \quad \frac{\pi}{2} - x + \frac{x^3}{3} - \frac{x^5}{5}$$

$$\text{series}\left(\int \frac{\sin(x)}{x} dx, x, 6\right) \quad x - \frac{x^3}{18} + \frac{x^5}{600}$$

$$\text{series}\left(\int_0^x \sin(x \cdot \sin(t)) dt, x, 7\right) \quad \frac{x^3}{2} - \frac{x^5}{24} - \frac{29 \cdot x^7}{720}$$

$$\text{series}\left(\left(1 + e^x\right)^2, x, 2, 1\right) \quad (e+1)^2 + 2 \cdot e \cdot (e+1) \cdot (x-1) + e \cdot (2 \cdot e+1) \cdot (x-1)^2$$

附註：另請參考 `dominantTerm()`，頁碼：54。

setMode()

setMode(模式名稱整數, 設定整數) ⇒ 整數

利用「顯示位數」的預設值，顯示 π 的近似值，然後以 `Fix2` 的設定顯示 π 。執行程式後，請檢查預設值是否還原。

setMode(列表) ⇒ 整數列表

只在函數或程式中才有效。

setMode(模式名稱整數, 設定整數) 可暫時將模式名稱整數的模式設成設定整數這個新設定，並傳回該模式原始設定的對應整數。這項變動受限於程式/函數的執行時間。

模式名稱整數用於指定您要的模式，必須設定成下表列出的模式整數。

設定整數用於指定該模式的新設定，必須是下列欲設定模式的其中一個設定整數。

setMode(列表) 可讓您更改多項設定。列表內容是成對的模式整數和設定整數。**setMode(列表)**：傳回類似的表，但內容是代表原始模式和設定的成對整數。

若您以 `getMode(0) → var` 儲存所有的模式設定，則可用 `setMode(var)` 還原這些設定，直到函數或程式存在為止。請參考 `getMode()`，頁碼：78。

附註：系統會將目前的模式設定，傳遞給所呼叫的副程式。若任何副程式更動了模式設定，則控制權回到發出呼叫的程式時，所更改的模式就會消失。

輸入範例的注意事項：關於輸入多行程式和函數定義的說明，請參閱產品手冊中的「計算工具」章節。

```
Define prog1()=Prgm Done
  Disp approx( $\pi$ )
  setMode(1,16)
  Disp approx( $\pi$ )
  EndPrgm


---



prog1()
3.14159
3.14


---


Done
```

模式名稱	模式整數	設定整數
顯示位數	1	1=浮點數、2=浮點數1、3=浮點數2、4=浮點數3、5=浮點數4、6=浮點數5、7=浮點數6、8=浮點數7、9=浮點數8、10=浮點數9、11=浮點數10、12=浮點數11、13=浮點數12、14=固定0、15=固定1、16=固定2、17=固定3、18=固定4、19=固定5、20=固定6、21=固定7、22=固定8、23=固定9、24=固定10、25=固定11、26=固定12
角度	2	1=弧度角、2=度數角、3=梯度角
指數格式	3	1=正常、2=科學、3=工程
實數或複數	4	1=實數、2=直角座標、3=極座標
自動或近似值	5	1=自動、2=近似值、3=精確值
向量格式	6	1=直角座標、2=圓柱座標、3=球面座標
基底	7	1=十進位、2=十六進位、3=二進位
單位制	8	1=國際單位制、2=英制

shift()

目錄 > 

shift(整數 I, 移位位數) ⇒ 整數

移動二進位整數中的位元。您可以用任何數基輸入整數 I；系統會自動轉換成有正負號的 64 位元二進位形式。若整數 I 太大，超出這種格式的範圍，系統會以對稱模數運算法使其落入範圍。如需詳細資訊，請參閱 ▶Base2，頁碼：17。

若移位位數是正值，則向左移位。若移位位數是負值，則向右移位。預設值是 -1(向右移一位)。

向右移位時會捨棄最右邊的位元並補上 0 或 1，以符合最左邊的位元。向左移位時會捨棄最左邊的位元，並補上 0 當作最右邊的位元。

例如向右移位時：

每個位元向右移位。

0b0000000000000111101011000011010

若最左邊的位元是 0，則補上 0；

二進位基底模式：

shift(0b1111010110000110101)	
	0b111101011000011010
shift(256,1)	0b1000000000

十六進位基底模式：

shift(0h78E)	0h3C7
shift(0h78E,-2)	0h1E3
shift(0h78E,2)	0h1E38

重要訊息：若要輸入二進位或十六進位數字，前面一定要有 0b 或 0h 字首 (數字零，而非英文字母 O)。

若最左邊的位元是 1，則補上 1。

求出：

0b00000000000000111101011000011010

所顯示的結果會以基底模式為準。系統不會顯示最前面的零。

shift(列表 I , 移位位數) ⇒ 列表

傳回列表 I 左右移動移位位數個元素後的副本。請勿更改列表 I 。

若移位位數是正值，則向左移位。若移位位數是負值，則向右移位。預設值是 -1(向右移動一個元素)。

因移位而在列表開頭或結尾加入的元素，系統會將其設定為 `undef` 符號。

shift(字串 I , 移位位數) ⇒ 字串

傳回字串 I 左右移動移位位數個字元後的副本。請勿更改字串 I 。

若移位位數是正值，則向左移位。若移位位數是負值，則向右移位。預設值是 -1(向右移動一個字元)。

因移位而在字串開頭或結尾加入的字元，系統會將其設定為空格。

十進位基底模式：

<code>shift({1,2,3,4})</code>	<code>{undef,1,2,3}</code>
<code>shift({1,2,3,4},-2)</code>	<code>{undef,undef,1,2}</code>
<code>shift({1,2,3,4},2)</code>	<code>{3,4,undef,undef}</code>

<code>shift("abcd")</code>	" abc "
<code>shift("abcd",-2)</code>	" ab "
<code>shift("abcd",1)</code>	"bcd "

sign()

sign(運算式 I) ⇒ 運算式

sign(列表 I) ⇒ 列表

sign(矩陣 I) ⇒ 矩陣

若是實數或複數 運算式 I ，則 運算式 $I \neq 0$ 時傳回 `運算式 I / abs(運算式 I)`。

若 運算式 I 是正值，則傳回 1。

若 運算式 I 是負值，則傳回 -1。

sign(0) 傳回 ± 1 的前提是複數格式模式為實數，否則會傳回自己的值。

sign(0) 代表複數值域中的單位圓。

<code>sign(-3.2)</code>	-1.
<code>sign({2,3,4,-5})</code>	{1,1,1,-1}
<code>sign(1+xi)</code>	1

若複數格式模式是實數：

<code>sign([-3 0 3])</code>	[-1 ±1 1]
-----------------------------	-----------

若是列表或矩陣，則傳回各元素的正負號。

simult()

simult(coeffMatrix, constVector[, Tol]) ⇒ 矩陣

傳回行向量，內容包含線性聯立方程式的解。

附註：另請參考 **linSolve()**，頁碼：95。

coeffMatrix 必須是包含方程式係數的方陣。

constVector 的列數必須和 *coeffMatrix* 相同(維數相同)，而且包含常數。

若任何矩陣元素的絕對值小於 *Tol* (容許值)，則亦可將此元素視為零。只有在矩陣中包含浮點數項目，而且不包含尚未賦值的任何符號變數時，才會使用此容許值。其他時候都會忽略 *Tol* (容許值)。

- 若您將 **自動或近似值** 模式設定成近似值，則系統會利用浮點運算法執行計算作業。
- 若指令省略或未使用 *Tol*，則預設容許值的計算方式如下：
 $5E-14 \cdot \max(\dim(\text{coeffMatrix})) \cdot \text{rowNorm}(\text{coeffMatrix})$

simult(coeffMatrix, constMatrix[, Tol]) ⇒ 矩陣

求解多個線性聯立方程式，每個聯立方程式的方程式係數都相同，但常數不同。

constMatrix 各行必須包含聯立方程式的常數。所求得矩陣中的各行，都包含對應聯立方程式的解。

求 x 、 y 的值：

$$x + 2y = 1$$

$$3x + 4y = -1$$

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

解答是 $x=-3$ 、 $y=2$ 。

求解：

$$ax + by = 1$$

$$cx + dy = 2$$

$$\begin{array}{l} \begin{bmatrix} a & b \\ c & d \end{bmatrix} \rightarrow \text{matx1} \quad \begin{bmatrix} a & b \\ c & d \end{bmatrix} \\ \text{simult}\left(\text{matx1}, \begin{bmatrix} 1 \\ 2 \end{bmatrix}\right) \quad \begin{bmatrix} -(2 \cdot b - d) \\ a \cdot d - b \cdot c \\ 2 \cdot a - c \\ a \cdot d - b \cdot c \end{bmatrix} \end{array}$$

求解：

$$x + 2y = 1$$

$$3x + 4y = -1$$

$$x + 2y = 2$$

$$3x + 4y = -3$$

$$\text{simult}\left(\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}, \begin{bmatrix} 1 & 2 \\ -1 & -3 \end{bmatrix}\right) \quad \begin{bmatrix} -3 & -7 \\ 2 & \frac{9}{2} \end{bmatrix}$$

第一個聯立方程式是 $x=-3$ 、 $y=2$ 。第二個聯立方程式是 $x=-7$ 、 $y=9/2$ 。

▶sin

Expr ▶sin

附註：如果要從電腦鍵盤插入本運算子，可輸入 []>sin 。

以正弦表示 *Expr*。這是顯示方式轉換運算子，只能用於輸入線末尾。

▶sin 會約化以下所有乘幂 $\cos(\dots)$ modulo $1 - \sin(\dots)^2$ 讓 $\sin(\dots)$ 剩餘乘幂的指數都在 $(0, 2)$ 範圍內。因此若且唯若 $\cos(\dots)$ 出現在指定運算式且是偶數乘幂時，結果才會沒有 $\cos(\dots)$ 。

附註：度數角或梯度角模式不支援本轉換運算子。使用前請確認是否已將角度模式設成弧度角，而且 *Expr* 中並無明確參照度數角或梯度角。

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

sin()

 鍵

sin(運算式 I) ⇒ 運算式

sin(列表 I) ⇒ 列表

sin(運算式 I) 可將引數的正弦值傳回為運算式。

sin(列表 I)：傳回列表 I 中各元素之正弦值的列表。

附註：系統會根據目前的角度模式，將引數解譯為度數角、梯度角或弧度角。您可使用 $^\circ$ 、 G 或 R 來暫時覆寫角度模式設定。

度數角模式：

$$\frac{\sin\left(\frac{\pi}{4}\right)}{\sin(45)} = \frac{\frac{\sqrt{2}}{2}}{\frac{\sqrt{2}}{2}}$$

$$\frac{\sin(\{0, 60, 90\})}{\left\{0, \frac{\sqrt{3}}{2}, 1\right\}}$$

梯度角模式：

$$\frac{\sin(50)}{2} = \frac{\sqrt{2}}{2}$$

弧度角模式：

$$\frac{\sin\left(\frac{\pi}{4}\right)}{\sin(45^\circ)} = \frac{\frac{\sqrt{2}}{2}}{\frac{\sqrt{2}}{2}}$$

sin(方陣 I) ⇒ 方陣

弧度角模式：

sin()

trig 鍵

傳回方陣 I 的矩陣正弦。這和計算各元素的正弦不同。若要了解計算方式，請參考 **cos()**。

方陣 I 必需可對角化。結果一定會包含浮點數。

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

$$\begin{bmatrix} 0.9424 & -0.04542 & -0.031999 \\ -0.045492 & 0.949254 & -0.020274 \\ -0.048739 & -0.00523 & 0.961051 \end{bmatrix}$$

sin⁻¹()

trig 鍵

sin⁻¹(運算式 I) ⇒ 運算式

度數角模式：

sin⁻¹(列表 I) ⇒ 列表

$$\sin^{-1}(1) \quad 90$$

sin⁻¹(運算式 I) 可將正弦值是運算式 I 的角度傳回為運算式。

梯度角模式：

sin⁻¹(列表 I): 傳回列表 I 各元素之反正弦值的列表。

$$\sin^{-1}(1) \quad 100$$

附註：系統會根據目前的角度模式設定，將結果傳回為度數角、梯度角或弧度角。

弧度角模式：

附註：如果要從鍵盤插入本函數，可輸入 **arcsin(...)**。

$$\sin^{-1}(\{0,0.2,0.5\}) \quad \{0,0.201358,0.523599\}$$

sin⁻¹(方陣 I) ⇒ 方陣

弧度角模式與直角複數格式模式：

傳回方陣 I 的矩陣反正弦。這和計算各元素的反正弦不同。若要了解計算方式，請參考 **cos()**。

$$\sin^{-1} \begin{pmatrix} 1 & 5 \\ 4 & 2 \end{pmatrix} \begin{bmatrix} -0.174533-0.12198 \cdot i & 1.74533-2.35591 \cdot i \\ 1.39626-1.88473 \cdot i & 0.174533-0.593162 \cdot i \end{bmatrix}$$

方陣 I 必需可對角化。結果一定會包含浮點數。

sinh()

目錄 >

sinh(運算式 I) ⇒ 運算式

$$\sinh(1.2) \quad 1.50946$$

sinh(列表 I) ⇒ 列表

$$\sinh(\{0,1.2,3\}) \quad \{0,1.50946,10.0179\}$$


sinh(運算式 I) 可將引數的雙曲正弦值傳回為運算式。

sinh(列表 I): 傳回列表 I 各元素之雙曲正弦值的列表。

sinh(方陣 I) ⇒ 方陣

弧度角模式：

sinh()


目錄 > 

傳回方陣 I 的矩陣雙曲正弦。這和計算各元素的雙曲正弦不同。若要了解計算方式，請參考 **cos()**。

方陣 I 必需可對角化。結果一定會包含浮點數。

$$\sinh \begin{pmatrix} 1 & 5 & 3 \\ 4 & 2 & 1 \\ 6 & -2 & 1 \end{pmatrix} \begin{bmatrix} 360.954 & 305.708 & 239.604 \\ 352.912 & 233.495 & 193.564 \\ 298.632 & 154.599 & 140.251 \end{bmatrix}$$

sinh⁻¹()

目錄 > 

sinh⁻¹(運算式 I) ⇒ 運算式

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

sinh⁻¹(列表 I) ⇒ 列表

$$\sinh^{-1}(\{0, 2, 1, 3\}) \quad \{0, 1.48748, \sinh^{-1}(3)\}$$

sinh⁻¹(運算式 I) 可將引數的反雙曲正弦值傳回為運算式。

sinh⁻¹(列表 I): 傳回列表 I 各元素之反雙曲正弦值的列表。

附註: 如果要從鍵盤插入本函數，可輸入 **arcsinh(...)**。

sinh⁻¹(方陣 I) ⇒ 方陣


弧度角模式:

傳回方陣 I 的矩陣反雙曲正弦。這和計算各元素的反雙曲正弦不同。若要了解計算方式，請參考 **cos()**。

$$\sinh^{-1} \begin{pmatrix} 1 & 5 & 3 \\ 4 & 2 & 1 \\ 6 & -2 & 1 \end{pmatrix} \begin{bmatrix} 0.041751 & 2.15557 & 1.1582 \\ 1.46382 & 0.926568 & 0.112557 \\ 2.75079 & -1.5283 & 0.57268 \end{bmatrix}$$

方陣 I 必需可對角化。結果一定會包含浮點數。

SinReg

目錄 > 

SinReg $X, Y[, [迭代], [週期], [類別, 包含]]$

計算 X 列表與 Y 列表的正弦迴歸。
stat.results 變數會儲存結果摘要(請參閱第 頁碼: 162 頁)。

所有列表的維數都必須相同，包含除外。

X, Y 是自變數和因變數列表。

迭代值用於指定求解次數的上限(1 至 16)。若省略，則會採用 8。通常值越大精度越高，但執行時間越久，反之亦然。

週期用於指定預估週期。若省略，則 X 中各值的差應該相等，而且依照順序。若您指定週期，則 x 中各值的差應該不同。

類別是 X 、 Y 對應資料的類別代號列表。

包含是一個以上類別代號的列表。只有本列表中類別代號所對應的資料項目，才會包含在計算作業中。

無論使用何種角度模式設定，SinReg 輸出都一定是採用弧度角形式。

如要了解列表中有空元素時的影響，請參考 \circledast 空元素 \circledast ，頁碼：219。

輸出變數	說明
stat.RegEqn	迴歸方程式： $a \cdot \sin(bx+c)+d$
stat.a、stat.b、 stat.c、stat.d	迴歸係數
stat.Resid	迴歸殘差
stat.XReg	實際用於迴歸的已修改 X 列表中的資料點列表(根據 <i>Freq</i> 、類別列表、包含類別的限制)
stat.YReg	實際用於迴歸的已修改 Y 列表中的資料點列表(根據 <i>Freq</i> 、類別列表、包含類別的限制)
stat.FreqReg	<i>stat.XReg</i> 與 <i>stat.YReg</i> 對應的頻率列表

solve()

solve(運算式, Var) \Rightarrow 布林運算式

solve(運算式, Var =猜測值) \Rightarrow 布林運算式

solve(不等式, Var) \Rightarrow 布林運算式

傳回 Var 的方程式或不等式之可能實數解。目標是傳回所有可能的解答。但有些方程式或不等式可能有無限多個解。

若是未定義變數的某些賦值組合，可能解答不一定是有限實數解。

$$\text{solve}(a \cdot x^2 + b \cdot x + c = 0, x)$$

$$x = \frac{\sqrt{b^2 - 4 \cdot a \cdot c} - b}{2 \cdot a} \text{ or } x = \frac{-\left(\sqrt{b^2 - 4 \cdot a \cdot c} + b\right)}{2 \cdot a}$$

$$\text{Ans}|a=1 \text{ and } b=1 \text{ and } c=1$$

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

如果在**自動或近似值**模式中設定為自動，則目標是求出簡明的精確解。若精確解不可得，則以計算近似值的迭代搜尋法補足。

由於依預設系統會消去比例之分子與分母的最大公因數，因此解可能只出現在單側或兩側極限處。

對於 \geq 、 \leq 、 $<$ 或 $>$ 等類的不等式，除非不等式屬於線性且只包含 *Var*，否則不太可能有顯解。

如果是精確值模式，則會將無法求解的部份傳回為隱式方程式或不等式。

請用 (`|`) 運算子限制求解區間，或限制方程式或不等式中出現的其他變數。若在一個區間求得解，則可在執行後續搜尋時，用不等式運算子排除該區間。

若無法求得實數解則傳回偽。若 **solve()** 可求出滿足方程式或不等式的 *Var* 有限實數值，則傳回真。

由於 **solve()** 一律傳回布林值結果，因此您可用 `and`、`or` 與 `not` 將 **solve()** 的解互相組合，或與其他布林運算式組合。

這些解可能包含 *nj* 格式的唯一未定義新常數，其中 *j* 是 1-255 區間範圍內的整數。這類變數用於指定任意整數。

在實數模式中，有奇分母的分數幂只代表實數分枝。否則多分枝的運算式（例如分數幂、對數、反三角函數）只代表主枝。因此 **solve()** 只能求得與該實數分枝或主枝對應的解。

附註：另請參考 **cSolve()**、**cZeros()**、**nSolve()**、**zeros()**。

solve(Eqn1 and Eqn2 [and ...], VarOrGuess1, VarOrGuess2 [, ...]) ⇒ 布林運算式

solve(聯立方程式, VarOrGuess1, VarOrGuess2 [, ...]) ⇒ 布林運算式

$$\text{solve}\left\{\left(x-a\right) \cdot e^{x}=x \cdot\left(x-a\right), x\right\} \\ x=a \text{ or } x=-0.567143$$

$$\left(x+1\right) \cdot \frac{x-1}{x-1}+x-3 \quad 2 \cdot x-2$$

$$\text{solve}\left\{5 \cdot x-2 \geq 2 \cdot x, x\right\} \quad x \geq \frac{2}{3}$$

$$\text{exact}\left\{\text{solve}\left\{\left(x-a\right) \cdot e^{x}=x \cdot\left(x-a\right), x\right\}\right\} \\ e^{x}+x=0 \text{ or } x=a$$

弧度角模式：

$$\text{solve}\left\{\tan (x)=\frac{1}{x}, x\right\} x > 0 \text{ and } x < 1 \\ x=0.860334$$

$$\text{solve}\left\{x=x+1, x\right\} \quad \text{false} \\ \text{solve}\left\{x=x, x\right\} \quad \text{true}$$

$$2 \cdot x-1 \leq 1 \text{ and solve}\left\{x^2 \neq 9, x\right\} \quad x \neq -3 \text{ and } x \leq 1$$

弧度角模式：

$$\text{solve}\left\{\sin (x)=0, x\right\} \quad x=n 1 \cdot \pi$$

$$\text{solve}\left\{x^{\frac{1}{3}}=-1, x\right\} \quad x=-1$$

$$\text{solve}\left\{\sqrt{x}=2, x\right\} \quad \text{false}$$

$$\text{solve}\left\{-\sqrt{x}=2, x\right\} \quad x=4$$

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

solve{*Eqn1*, *Eqn2* [...]}{*VarOrGuess1*,
VarOrGuess2 [...]} ⇒ 布林運算式

傳回聯立代數方程式的可能實數解。
每個 *VarOrGuess* 代表應求解的變數。

您可以用 **and** 運算子分隔方程式，或從
目錄]利用範本輸入聯立方程式。
VarOrGuess 引數的數量必須和方程式
相同。您也可以指定或猜測一個變數
的初始值。所有 *VarOrGuess* 的格式必
須如下：

變數

- 或 -

變數 = 實數或非實數

舉例來說， x 和 $x=3$ 都有效。

若所有方程式都是多項式，且若您
未]指定或猜測任何初始值，則 **solve()**
會以 Grobner/Buchberger 詞彙的消去
法，試圖求得所有實數解。

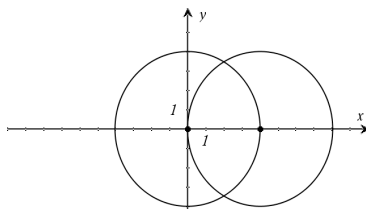
例如假設原點有一個半徑是 r 的圓，第
一個圓和正 x 軸交會之處，則是另一個
半徑是 r 的圓的圓心。利用 **solve()** 可找
出交點。

如右例中的 r 所示，聯立多項式方程式
可包含無具體值的額外變數(但表現出
稍後可代入的指定數值)。

您也可以(或改為)加入方程式中未出
現的求解變數。例如您可加入 z 當作求
解變數，將前例延伸成兩個平行相交
的圓柱(半徑為 r)。

圓柱解能說明求解系可包含 ck 格式的
任意常數，其中 k 是從 1 到 255 的整數
下標。

以多項式系統而言，求解變數的列出
順序對於計算時間或記憶體用量可能
影響甚大。若您最初的選擇用完記憶
體或耗費太多時間，請重排方程式或
VarOrGuess 列表中的變數。



$$\text{solve}\left\{x^2+y^2=r^2 \text{ and } (x-r)^2+y^2=r^2, \{x,y\}\right\}$$

$$x=\frac{r}{2} \text{ and } y=\frac{\sqrt{3}\cdot r}{2} \text{ or } x=\frac{r}{2} \text{ and } y=-\frac{\sqrt{3}\cdot r}{2}$$

$$\text{solve}\left\{x^2+y^2=r^2 \text{ and } (x-r)^2+y^2=r^2, \{x,y,z\}\right\}$$

$$x=\frac{r}{2} \text{ and } y=\frac{\sqrt{3}\cdot r}{2} \text{ and } z=c1 \text{ or } x=\frac{r}{2} \text{ and } y=-\frac{\sqrt{3}\cdot r}{2} \text{ and } z=c1$$

若要看到完整結果，請按 **▲**，然後使用
◀與**▶**移動游標。

若您不加入任何猜測值，且任何方程式都不是任何變數的多項式，但所有方程式都是求解變數的線性方程式，則 **solve()** 會以高斯消去法，試圖求得所有實數解。

若某聯立方程式既非所有變數的多項式，亦非求解變數的線性方程式，則 **solve()** 利用近似迭代法最多只能求得一個解。為此，求解變數的數量必須等於方程式數量，且方程式中的其他變數必須化簡成數字。

每個求解變數都從猜測值開始(如果有)，否則就從 0.0 開始。

使用猜測值可逐一尋找其他解。為了收斂，猜測值必須接近解值。

$$\text{solve}\left(x+e^z \cdot y=1 \text{ and } x-y=\sin(z),\{x,y\}\right)$$

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

$$\text{solve}\left(e^z \cdot y=1 \text{ and } -y=\sin(z),\{y,z\}\right)$$

$$y=2.812\text{E}-10 \text{ and } z=21.9911 \text{ or } y=0.001871$$

若要看到完整結果，請按 **▲**，然後使用 **◀**與**▶**移動游標。

$$\text{solve}\left(e^z \cdot y=1 \text{ and } -y=\sin(z),\{y,z=2 \cdot \pi\}\right)$$

$$y=0.001871 \text{ and } z=6.28131$$

SortA

SortA 列表1[, 列表2][, 列表3] ...

$$\{2,1,4,3\} \rightarrow \text{list1} \quad \{2,1,4,3\}$$

SortA 向量1[, 向量2][, 向量3] ...

$$\text{SortA list1} \quad \text{Done}$$

以升幂順序排列第一個引數的元素。

$$\text{list1} \quad \{1,2,3,4\}$$

若您加入其他引數，則各引數中的元素會重新排列，以便對應第一個引數中各元素的新位置。

$$\{4,3,2,1\} \rightarrow \text{list2} \quad \{4,3,2,1\}$$

所有引數都必須是列表或向量名稱，而且維數必須相同。

$$\text{SortA list2,list1} \quad \text{Done}$$

$$\text{list2} \quad \{1,2,3,4\}$$

$$\text{list1} \quad \{4,3,2,1\}$$

系統會把第一個引數中的空元素移到末尾。如需空元素的詳細資訊，請參考第 頁碼: 219 頁。

SortD 列表1[, 列表2] [, 列表3] ...

 $\{2,1,4,3\} \rightarrow list1$ $\{2,1,4,3\}$

SortD 向量1[, 向量2] [, 向量3] ...

 $\{1,2,3,4\} \rightarrow list2$ $\{1,2,3,4\}$

與 SortA 相同，但 SortD 會以降幂順序排列各元素。

SortD list1,list2 Done

list1 $\{4,3,2,1\}$ list2 $\{3,4,1,2\}$

系統會把第一個引數中的空元素移到末尾。如需空元素的詳細資訊，請參考第 頁碼：219 頁。

►Sphere

向量 ►Sphere

附註： 如果要從電腦鍵盤插入本運算子，可輸入 @>Sphere。以球面座標形式 [ρ $\angle\theta$ $\angle\phi$] 顯示列向量或行向量。

向量必須是 3 維，而且可以是列向量也可以是行向量。

附註： ►Sphere 是顯示格式指令，並非轉換函數。本指令只能用於輸入線的末尾。**附註：** 強迫表示結果近似值，**計算機：** 按 。**Windows®：** 按 **Ctrl+Enter**。**Macintosh®：** 按 **⌘+Enter**。**iPad®：** 按住 **enter** 然後選擇 。
$$\left[\begin{array}{ccc} 1 & 2 & 3 \end{array} \right] \text{►Sphere}$$

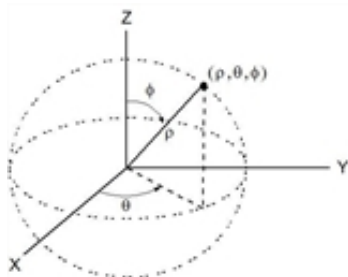

$$\left[3.74166 \quad \angle 1.10715 \quad \angle 0.640522 \right]$$
附註： 強迫表示結果近似值，**計算機：** 按 。**Windows®：** 按 **Ctrl+Enter**。**Macintosh®：** 按 **⌘+Enter**。**iPad®：** 按住 **enter** 然後選擇 。
$$\left(\left[\begin{array}{ccc} 2 & \angle \frac{\pi}{4} & 3 \end{array} \right] \text{►Sphere} \right)$$

$$\left[3.60555 \quad \angle 0.785398 \quad \angle 0.588003 \right]$$

按

$$\left(\left[\begin{array}{ccc} 2 & \angle \frac{\pi}{4} & 3 \end{array} \right] \text{►Sphere} \right)$$

$$\left[\sqrt{13} \quad \angle \frac{\pi}{4} \quad \angle \sin^{-1} \left(\frac{2 \cdot \sqrt{13}}{13} \right) \right]$$

**sqrt()**目錄 > 

sqrt(運算式 I)⇒運算式

$$\sqrt{4} \qquad 2$$

sqrt(列表 I)⇒列表

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

傳回引數的平方根。

若是列表，則傳回列表 I 中各元素的平方根。

附註：另請參考平方根範本，頁碼：1。

stat.results目錄 > 

stat.results

顯示統計計算結果。

系統會將結果顯示為一組成對的 [名稱/值]。所顯示的特定名稱，要看最近求值的統計函數或指令而定。

您可複製名稱或值，然後貼到其他位置。

附註：定義變數時，請避免讓名稱和統計分析變數相同，否則有時可能出現錯誤。下表列出統計分析時所用的變數名稱。

$$xlist=\{1,2,3,4,5\} \qquad \{1,2,3,4,5\}$$

$$ylist=\{4,8,11,14,17\} \qquad \{4,8,11,14,17\}$$

LinRegMx xlist,ylist,1: stat.results


"Title"	"Linear Regression (mx+b)"
"RegEqn"	"m*x+b"
"m"	3.2
"b"	1.2
"r ² "	0.996109
"r"	0.998053
"Resid"	" {... } "

stat.values	"Linear Regression (mx+b)"
	"m*x+b"
	3.2
	1.2
	0.996109
	0.998053
	"{-0.4,0.4,0.2,0,-0.2}"

stat.a	stat.dfDenom	stat.MedianY	stat.Q3X	stat.SSBlock
stat.AdjR ²	stat.dfBlock	stat.MEPred	stat.Q3Y	stat.SSCol
stat.b	stat.dfCol	stat.MinX	stat.r	stat.SSX
stat.b0	stat.dfError	stat.MinY	stat.r ²	stat.SSY
stat.b1	stat.dflInteract	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.σx	stat.SSRow
stat.b5	stat.DW	stat.MSInteract	stat.σy	stat.tList
stat.b6	stat.e	stat.MSReg	stat.σx1	stat.UpperPred
stat.b7	stat.ExpMatrix	stat.MSRow	stat.σx2	stat.UpperVal
stat.b8	stat.F	stat.n	stat.Σx	stat.̄x
stat.b9	stat.FBlock	stat.̂p	stat.Σx ²	stat.̄x1
stat.b10	stat.Fcol	stat.̂p1	stat.Σxy	stat.̄x2
stat.bList	stat.FInteract	stat.̂p2	stat.Σy	stat.̄x Diff
stat.χ ²	stat.FreqReg	stat.̂p Diff	stat.Σy ²	stat.̄xList
stat.c	stat.Frow	stat.PList	stat.s	stat.XReg
stat.CLower	stat.Leverage	stat.PVal	stat.SE	stat.XVal
stat.CLowerList	stat.LowerPred	stat.PValBlock	stat.SEList	stat.XValList
stat.CompList	stat.LowerVal	stat.PValCol	stat.SEPred	stat.ȳ
stat.CompMatrix	stat.m	stat.PValInteract	stat.sResid	stat.ȳ
stat.CookDist	stat.MaxX	stat.PValRow	stat.SESlope	stat.ȳList
stat.CUpper	stat.MaxY	stat.Q1X	stat.sp	stat.YReg
stat.CUpperList	stat.ME	stat.Q1Y	stat.SS	
stat.d	stat.MedianX			

附註： Lists & Spreadsheet 應用程式每次計算統計結果時，就會將 [stat.] 群組變數複製到 [stat#.] 群組。# 代表自動增量值。本功能讓您保留以前的結果，同時執行多次計算。

stat.values

目錄 > 

stat.values

請參考 **stat.results** 範例。

針對最近求值的統計函數或指令，顯示計算出來的數值矩陣。

stat.values 會省略數值的對應名稱，這一點與 **stat.results** 不同。

您可複製數值，然後貼到其他位置。

stDevPop()

stDevPop(列表[, 頻率列表])⇒運算式

傳回列表元素的母群體標準差。

頻率列表的每個元素，代表列表中對應元素的連續出現次數。

附註：列表至少必須有兩個元素。空元素會被忽略。如需空元素的詳細資訊，請參考第 頁碼：219 頁。

stDevPop(矩陣I[, 頻率矩陣])⇒矩陣

傳回矩陣I中各行母群體標準差的列向量。

頻率矩陣的每個元素，代表矩陣I中對應元素的連續出現次數。

附註：矩陣I至少必須有兩列。空元素會被忽略。如需空元素的詳細資訊，請參考第 頁碼：219 頁。

弧度角與自動模式：

$$\text{stDevPop}\{\{a,b,c\}\} = \frac{\sqrt{2 \cdot (a^2 - a \cdot (b+c) + b^2 - b \cdot c + c^2)}}{3}$$

$$\text{stDevPop}\{\{1,2,5,-6,3,-2\}\} = \frac{\sqrt{465}}{6}$$

$$\text{stDevPop}\{\{1.3,2.5,-6.4\},\{3,2,5\}\} = 4.11107$$

$$\text{stDevPop}\left(\begin{bmatrix} 1 & 2 & 5 \\ -3 & 0 & 1 \\ 5 & 7 & 3 \end{bmatrix}, \begin{bmatrix} 4 \cdot \sqrt{6} & \sqrt{78} & 2 \cdot \sqrt{6} \\ 3 & 3 & 3 \end{bmatrix}\right)$$

$$\text{stDevPop}\left(\begin{bmatrix} -1.2 & 5.3 \\ 2.5 & 7.3 \\ 6 & -4 \end{bmatrix}, \begin{bmatrix} 4 & 2 \\ 3 & 3 \\ 1 & 7 \end{bmatrix}\right) = \begin{bmatrix} 2.52608 & 5.21506 \end{bmatrix}$$

stDevSamp()

stDevSamp(列表[, 頻率列表])⇒運算式

傳回列表元素的樣本標準差。

頻率列表的每個元素，代表列表中對應元素的連續出現次數。

附註：列表至少必須有兩個元素。空元素會被忽略。如需空元素的詳細資訊，請參考第 頁碼：219 頁。

$$\text{stDevSamp}\{\{a,b,c\}\} = \frac{\sqrt{3 \cdot (a^2 - a \cdot (b+c) + b^2 - b \cdot c + c^2)}}{3}$$

$$\text{stDevSamp}\{\{1,2,5,-6,3,-2\}\} = \frac{\sqrt{62}}{2}$$

$$\text{stDevSamp}\{\{1.3,2.5,-6.4\},\{3,2,5\}\} = 4.33345$$

stDevSamp()

目錄 > 

stDevSamp(矩陣I, 頻率矩陣)⇒矩陣
傳回矩陣I中各行樣本標準差的列向量。

頻率矩陣的每個元素，代表矩陣I中對應元素的連續出現次數。

附註：矩陣I至少必須有兩列。空元素會被忽略。如需空元素的詳細資訊，請參考第頁碼：219頁。

$\text{stDevSamp}\left(\begin{bmatrix} 1 & 2 & 5 \\ -3 & 0 & 1 \\ 5 & 7 & 3 \end{bmatrix}\right)$	$\left[4 \sqrt{13} \ 2\right]$
$\text{stDevSamp}\left(\begin{bmatrix} -1.2 & 5.3 \\ 2.5 & 7.3 \\ 6 & -4 \end{bmatrix}, \begin{bmatrix} 4 & 2 \\ 3 & 3 \\ 1 & 7 \end{bmatrix}\right)$	$\left[2.7005 \ 5.44695\right]$

Stop

目錄 > 

Stop

程式設計指令：中止程式。

Stop 不得用於函數中。

輸入範例的注意事項：關於輸入多行程式和函數定義的說明，請參閱產品手冊中的「計算工具」章節。

<i>i</i> :=0	0
Define <i>prog1</i> ()=Prgm	Done
For <i>i</i> ,1,10,1	
If <i>i</i> =5	
Stop	
EndFor	
EndPrgm	
<i>prog1</i> ()	Done
<i>i</i>	5

Store

請參考 → (store), 頁碼：216。

string()

目錄 > 

string(Expr)⇒字串

化簡 *Expr* 並將結果傳回為字元字串。

$\text{string}(1.2345)$	"1.2345"
$\text{string}(1+2)$	"3"
$\text{string}(\cos(x)+\sqrt{3})$	"cos(x)+√(3)"

subMat()

目錄 >

subMat(矩陣 I , 起始列 [, 起始行] [, 結束列] [, 結束行]) \Rightarrow 矩陣

傳回矩陣 I 的指定子矩陣。

預設值: 起始列=1、起始行=1、結束列=最後一列、結束行=最後一行。

$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix} \rightarrow m1$	$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$
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)請參考 $\Sigma()$, 頁碼: 208。**sum()**

目錄 >

sum(列表 [, 起點 [, 終點]]) \Rightarrow 運算式

傳回列表中各元素的和。

起點和終點可選擇性輸入, 用於指定元素範圍。

若有任何空引數, 都會求出空結果。列表中的空元素會遭到忽略。如需空元素的詳細資訊, 請參考第 頁碼: 219 頁。

sum(矩陣 I , 起點 [, 終點]]) \Rightarrow 矩陣

傳回包含矩陣 I 中各行所有元素和的列向量。

起點和終點可選擇性輸入, 用於指定列範圍。

若有任何空引數, 都會求出空結果。矩陣 I 中的空元素會遭到忽略。如需空元素的詳細資訊, 請參考第 頁碼: 219 頁。

sum ($\{1, 2, 3, 4, 5\}$)	15
sum ($\{a, 2 \cdot a, 3 \cdot a\}$)	$6 \cdot a$
sum ($\text{seq}(n, n, 1, 10)$)	55
sum ($\{1, 3, 5, 7, 9\}, 3$)	21

sum ($\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}$)	$\begin{bmatrix} 5 & 7 & 9 \end{bmatrix}$
sum ($\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$)	$\begin{bmatrix} 12 & 15 & 18 \end{bmatrix}$
sum ($\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}, 2, 3$)	$\begin{bmatrix} 11 & 13 & 15 \end{bmatrix}$

sumIf()

目錄 >

sumIf(列表, 準則 [, 總和表]) \Rightarrow 值

傳回列表中符合指定準則之所有元素的累積總和。您也可以指定另一個列表(總和表), 以提供要累加的元素。

sumIf ($\{1, 2, e, 3, \pi, 4, 5, 6\}, 2.5 < ? < 4.5$)	$e + \pi + 7$
sumIf ($\{1, 2, 3, 4\}, 2 < ? < 5, \{10, 20, 30, 40\}$)	70

列表可以是運算式、列表或矩陣。總和表(若有指定)與列表的維數必須相同。

準則可以有以下形式：

- 值、運算式、字串。例如 **34** 代表只累加列表中可化簡成 **34** 這個值的元素。
- 包含 **?**符號當作各元素預留位置的布林運算式。例如 **?<10** 代表只累加列表中小於 **10** 的元素。

列表中的某元素若符合準則，就會將該元素計入累積總和。若您加入總和表，則會改而將總和表中的對應元素計入總和。

您可在 Lists & Spreadsheet 應用程式中用儲存格範圍取代列表與總和表。

空元素會被忽略。如需空元素的詳細資訊，請參考第 頁碼：219 頁。

附註：另請參考 **countif()**，頁碼：34。

system(Eqn1 [, Eqn2 [, Eqn3 [, ...]])

$$\text{solve} \left\{ \begin{array}{l} x+y=0 \\ x-y=8 \end{array} \right. \begin{array}{l} x,y \end{array} \quad x=4 \text{ and } y=-4$$

system(運算式1 [, 運算式2 [, 運算式3 [, ...]])

傳回列表格式的聯立方程式。您也可以利用範本建立聯立方程式。

附註：另請參考**聯立方程式**，頁碼：3。

T (transpose)

矩陣 I \Rightarrow 矩陣傳回矩陣 I 的複共軛轉置矩陣。**附註：**如果要從電腦鍵盤插入本運算子，可輸入 **et**。

$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}^T$	$\begin{bmatrix} 1 & 4 & 7 \\ 2 & 5 & 8 \\ 3 & 6 & 9 \end{bmatrix}$
$\begin{bmatrix} a & b \\ c & d \end{bmatrix}^T$	$\begin{bmatrix} a & c \\ b & d \end{bmatrix}$
$\begin{bmatrix} 1+i & 2+i \\ 3+i & 4+i \end{bmatrix}^T$	$\begin{bmatrix} 1-i & 3-i \\ 2-i & 4-i \end{bmatrix}$

tan()

 鍵tan(運算式 I) \Rightarrow 運算式

度數角模式：

tan(列表 I) \Rightarrow 列表tan(運算式 I) 可將引數的正切值傳回為運算式。tan(列表 I): 傳回列表 I 中各元素之正切值的列表。**附註：**系統會根據目前的角度模式，將引數解譯為度數角、梯度角或弧度角。您可使用 °、G 或 π 來暫時覆寫角度模式設定。

梯度角模式：

$\tan\left(\frac{\pi}{4}\right)$	1
tan(45)	1
tan({0,60,90})	{0,√3,undef}

弧度角模式：

$\tan\left(\frac{\pi}{4}\right)$	1
tan(45°)	1
tan($\left\{\pi, \frac{\pi}{3}, \pi, \frac{\pi}{4}\right\}$)	{0,√3,0,1}

tan(方陣 I) \Rightarrow 方陣傳回方陣 I 的矩陣正切。這和計算各元素的正切不同。若要了解計算方式，請參考 **cos()**。方陣 I 必需可對角化。結果一定會包含浮點數。

弧度角模式：

$\tan\left(\begin{bmatrix} 1 & 5 & 3 \\ 4 & 2 & 1 \\ 6 & -2 & 1 \end{bmatrix}\right)$			
	-28.2912	26.0887	11.1142
	12.1171	-7.83536	-5.48138
	36.8181	-32.8063	-10.4594

tan⁻¹()tan⁻¹(運算式 I) ⇒ 運算式

度數角模式:

tan⁻¹(列表 I) ⇒ 列表

tan ⁻¹ (1)	45
-----------------------	----

tan⁻¹(運算式 I) 可將正切值是 運算式 1 的角度傳回為運算式。

梯度角模式:

tan⁻¹(列表 I): 傳回列表 I 各元素之反正切值的列表。

tan ⁻¹ (1)	50
-----------------------	----

附註: 系統會根據目前的角度模式設定, 將結果傳回為度數角、梯度角或弧度角。

弧度角模式:

附註: 如果要從鍵盤插入本函數, 可輸入 arctan(...)

tan ⁻¹ ({0,0.2,0.5})	{0,0.197396,0.463648}
---------------------------------	-----------------------

tan⁻¹(方陣 I) ⇒ 方陣

弧度角模式:

傳回方陣 I 的矩陣反正切。這和計算各元素的反正切不同。若要了解計算方式, 請參考 cos()。

$\tan^{-1}\left(\begin{bmatrix} 1 & 5 & 3 \\ 4 & 2 & 1 \\ 6 & -2 & 1 \end{bmatrix}\right)$			
	-0.083658	1.26629	0.62263
	0.748539	0.630015	-0.070012
	1.68608	-1.18244	0.455126

方陣 I 必需可對角化。結果一定會包含浮點數。

tangentLine()

tangentLine(運算式 I, Var, 點) ⇒ 運算式

tangentLine(x ² , x, 1)	2·x-1
------------------------------------	-------

tangentLine(運算式 I, Var=點) ⇒ 運算式

tangentLine((x-3) ² -4, x=3)	-4
---	----

針對 運算式 I 代表的曲線, 傳回指定 Var=點的切線。

tangentLine($\frac{1}{x^3}$, x=0)	x=0
-------------------------------------	-----

請勿定義自變數。例如若 f1(x):=5 且 x:=3, 則 tangentLine(f1(x), x, 2) 會傳回「偽」。

tangentLine($\sqrt{x^2-4}$, x=2)	undef
------------------------------------	-------

x:=3: tangentLine(x ² , x, 1)	5
--	---

tanh()

目錄 >

tanh(運算式 I) ⇒ 運算式

tanh(1.2)	0.833655
-----------	----------

tanh(列表 I) ⇒ 列表

tanh({0,1})	{0,tanh(1)}
-------------	-------------

tanh(運算式 I) 可將引數的雙曲正切值傳回為運算式。

tanh(列表 I): 傳回列表 I 各元素之雙曲正切值的列表。

tanh(方陣 I) ⇒ 方陣

傳回方陣 I 的矩陣雙曲正切。這和計算各元素的雙曲正切不同。若要了解計算方式，請參考 **cos()**。

方陣 I 必需可對角化。結果一定會包含浮點數。

弧度角模式:

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()

目錄 >

tanh⁻¹(運算式 I) ⇒ 運算式

直角複數格式:

tanh⁻¹(列表 I) ⇒ 列表

tanh ⁻¹ (0)	0
------------------------	---

tanh⁻¹(運算式 I) 可將引數的反雙曲正切值傳回為運算式。

tanh ⁻¹ ({1,2,1,3})	$\left\{ \text{undef}, 0.518046 - 1.5708 \cdot i, \frac{\ln(2)}{2} - \frac{\pi}{2} \cdot i \right\}$
--------------------------------	--

tanh⁻¹(列表 I): 傳回列表 I 各元素之反雙曲正切值的列表。

若要看到完整結果，請按 ▲，然後使用 ◀ 與 ▶ 移動游標。

附註：如果要從鍵盤插入本函數，可輸入 **arctanh(...)**。

tanh⁻¹(方陣 I) ⇒ 方陣

弧度角模式與直角複數格式:


傳回方陣 I 的矩陣反雙曲正切。這和計算各元素的反雙曲正切不同。若要了解計算方式，請參考 **cos()**。

tanh ⁻¹ $\begin{pmatrix} 1 & 5 & 3 \\ 4 & 2 & 1 \\ 6 & -2 & 1 \end{pmatrix}$	$\begin{bmatrix} -0.099353 + 0.164058 \cdot i & 0.267834 - 1.4908 \\ -0.087596 - 0.725533 \cdot i & 0.479679 - 0.94730 \\ 0.511463 - 2.08316 \cdot i & -0.878563 + 1.7901 \end{bmatrix}$
---	--

方陣 I 必需可對角化。結果一定會包含浮點數。

若要看到完整結果，請按 ▲，然後使用 ◀ 與 ▶ 移動游標。

taylor()

目錄 > 


taylor(*運算式* *l*, *Var*, 階數[, 點]) \Rightarrow 運算式

傳回所要求的泰勒展開式。該多項式包括 (*Var* 減點) 中非零的整數次數 (從零到階數) 項。**taylor()** 會在沒有此階級數之截斷式, 或是需要負指數或分數指數時, 傳回自己的值。若要求出更一般形式的冪級數, 請用取代法或臨時乘以 (*Var* 減點) 乘冪。

點的預設值是零, 同時是展開點。

$\text{taylor}(e^{\sqrt{x}}, x, 2)$	$\text{taylor}(e^{\sqrt{x}}, x, 2, 0)$
$\text{taylor}(e^t, t, 4) t = \sqrt{x}$	$\frac{3}{24} + \frac{x}{6} + \frac{x^2}{2} + \sqrt{x} + 1$
$\text{taylor}\left(\frac{1}{x \cdot (x-1)}, x, 3\right)$	$\text{taylor}\left(\frac{1}{x \cdot (x-1)}, x, 3, 0\right)$
$\text{expand}\left(\frac{\text{taylor}\left(\frac{x}{x \cdot (x-1)}, x, 4\right)}{x}, x\right)$	$-x^3 - x^2 - x - \frac{1}{x} - 1$

tCdf()


目錄 > 

tCdf(*下限*, *上限*, *df*) \Rightarrow 數字 (若 *下限* 和 *上限* 是數字) 或列表 (若 *下限* 和 *上限* 是列表)

針對指定自由度 *df* 的 *下限* 和 *上限* 之間, 計算 Student-*t* 分布機率。

對於 $P(X \leq \text{上限})$, 請設定 *下限* = $-\infty$ 。

tCollect()

目錄 > 

tCollect(*運算式* *l*) \Rightarrow 運算式

傳回運算式, 其中正弦和餘弦的乘積和整數冪, 都已經轉成倍角、角和以及角差的正弦和餘弦的線性組合。這種變換功能可將三角多項式, 轉成其調和函數的線性組合。

若預設的三角化簡法無法達成目標, 有時 **tCollect()** 能如您所願。**tCollect()** 可能逆向轉換 **tExpand()** 的結果。有時以兩道步驟將 **tExpand()** 套用到 **tCollect()** 求出的結果, 就能化簡運算式 (反之亦然)。

$\text{tCollect}(\{\cos(\alpha)\}^2)$	$\frac{\cos(2 \cdot \alpha) + 1}{2}$
$\text{tCollect}(\sin(\alpha) \cdot \cos(\beta))$	$\frac{\sin(\alpha - \beta) + \sin(\alpha + \beta)}{2}$

tExpand(運算式 I)⇒運算式

傳回運算式，其中整數倍角、角和以及角差的正弦和餘弦都已經展開。由於恆等式 $(\sin(x))^2 + (\cos(x))^2 = 1$ ，因此有很多可能等值的結果。所以結果可能和其他出版品不同。

若預設的三角化簡法無法達成目標，有時 **tExpand()** 能如您所願。**tExpand()** 可能逆向轉換 **tCollect()** 的結果。有時以兩道步驟將 **tCollect()** 套用到 **tExpand()** 求出的結果，就能化簡運算式(反之亦然)。

附註： $\pi/180$ 的度數角模式換算功能，會干擾 **tExpand()** 的可展開形式辨識能力。為求最佳效果，應將 **tExpand()** 用於弧度角模式。

$$\frac{\text{tExpand}(\sin(3 \cdot \phi))}{\text{tExpand}(\cos(\alpha - \beta))} = \frac{4 \cdot \sin(\phi) \cdot (\cos(\phi))^2 - \sin(\phi)}{\cos(\alpha) \cdot \cos(\beta) + \sin(\alpha) \cdot \sin(\beta)}$$

Text**Text** 提示字串[, 顯示旗標]

程式設計指令：程式暫停執行，並在對話方塊中顯示包含提示字串的字元字串。

使用者選取 **[確定]** 時，程式就會繼續執行。若選取 **[取消]** 可停止執行程式。


旗標選擇性(非必要)引數可以是任何運算式。

- 若省略顯示旗標或其求值為 **1**，則在 Calculator 歷史記錄中加入文字訊息。
- 若顯示旗標求值為 **0**，則不在歷史記錄中加入文字訊息。

若程式需要使用者輸入回答，請參考 **Request(頁碼：138)** 或 **RequestStr(頁碼：139)**。

附註： 您可以在使用者自行定義的程式中使用本指令，但不能用在函數中。

定義讓程式暫停，並在對話方塊中顯示五個隨機數字。

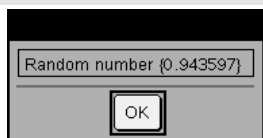
在 Prgm...EndPrgm 範本的各行結尾處按 ，不要按 **[enter]**。按住電腦鍵盤的 Alt 然後按 Enter。

```
Define text_demo()=Prgm
  For i,1,5
    strinfo:="Random number " &
string(rand(i))
    Text strinfo
  EndFor
EndPrgm
```

執行程式：


text_demo()

對話方塊一例：



Then

請參考 If, 頁碼: 81.

tInterval目錄 > **tInterval** 列表[,Freq[,CLevel]]

(輸入資料列表)


tInterval \bar{x} ,sx,n[,CLevel]

(輸入統計摘要)

計算 t 信賴區間。*stat.results* 變數會儲存結果摘要(請參閱第 頁碼: 162 頁)。

如要了解列表中有空元素時的影響,請參考“空元素”,頁碼: 219。

輸出變數	說明
stat.CLower、stat.CUpper	不明母群體平均值的信賴區間
stat. \bar{x}	常態隨機分布的資料序列平均值樣本
stat.ME	邊際誤差
stat.df	自由度
stat. σ_x	樣本標準差
stat.n	資料序列的長度與平均值樣本

tInterval_2Samp目錄 > **tInterval_2Samp** 列表1,列表2[,Freq1[,Freq2[,CLevel[,合併]]]]

(輸入資料列表)

tInterval_2Samp $\bar{x}1$,sx1,n1, $\bar{x}2$,sx2,n2[,CLevel[,合併]]

(輸入統計摘要)

計算二樣本 t 信賴區間。*stat.results* 變數會儲存結果摘要(請參閱第 頁碼: 162 頁)。

合併=1 合併變異數; 合併=0 不合併變異數。

如要了解列表中有空元素時的影響, 請參考 °空元素°, 頁碼: 219。

輸出變數	說明
stat.CLower、stat.CUpper	包含信心水準分布機率的信賴區間
stat. $\bar{x}1$ - $\bar{x}2$	常態隨機分布的資料序列平均值樣本
stat.ME	邊際誤差
stat.df	自由度
stat. $\bar{x}1$, stat. $\bar{x}2$	常態隨機分布的資料序列平均值樣本
stat. $\sigma x1$ 、stat. $\sigma x2$	列表 1 和列表 2 的標準差樣本
stat.n1、stat.n2	資料序列的樣本數
stat.sp	合併標準差。合併 = YES 時才予計算

tmpCnv()

**tmpCnv(Expr °溫度單位, °溫度單位
2) ⇒ 運算式 °溫度單位2**

將 *Expr* 指定的溫度值轉換成其他單位。有效的溫度單位如下:


°C 攝氏

°F 華氏

°K 絕對

°R 朗肯

若要輸入 °, 請從 [目錄] 符號表中選取。

若要輸入 _, 請按 **ctrl** 。


例如輸入 100_°C 可轉成 212_°F。

若要轉換溫度範圍, 請改用 **ΔtmpCnv()**。

tmpCnv(100_°C,_°F)	212_°F
tmpCnv(32_°F,_°C)	0_°C
tmpCnv(0_°C,_°K)	273.15_°K
tmpCnv(0_°F,_°R)	459.67_°R

附註: 您可使用 [目錄] 選取溫度單位。

Δ tmpCnv()

目錄 > 

Δ tmpCnv(*Expr* °溫度單位, °溫度單位
2) \Rightarrow 運算式 °溫度單位2

附註： 如果要從鍵盤插入本函數，可輸入 **deltaTmpCnv (...)**。

將 *Expr* 指定的溫度範圍(兩個溫度值的差)轉換成其他單位。有效的溫度單位如下：



_°C 攝氏

_°F 華氏

_°K 絕對

_°R 朗肯

若要輸入 °，請從「符號調色盤」選取或輸入 @d。

若要輸入 _，請按  。

1_°C 與 1_°K 的數值範圍相同，1_°F 則與 1_°R 的範圍相同。但 1_°C 數值範圍是 1_°F 的 9/5 倍。


例如 100_°C 的範圍(0_°C 至 100_°C) 等於 180_°F 的範圍。

若要轉換特定溫度值而非溫度範圍，請用 **tmpCnv()**。

Δ tmpCnv(100_°C,_°F)	180._°F
Δ tmpCnv(180_°F,_°C)	100._°C
Δ tmpCnv(100_°C,_°K)	100._°K
Δ tmpCnv(100_°F,_°R)	100._°R
Δ tmpCnv(1_°C,_°F)	1.8_°F

附註： 您可使用「目錄」選取溫度單位。


tPdf()

目錄 > 

tPdf(*XVal*,*df*) \Rightarrow 數字(若 *XVal* 是數字) 或
列表(若 *XVal* 是列表)

針對指定自由度 *df*，計算 Student-*t* 分布在指定 *x* 值的機率密度函數(pdf)。

trace()

目錄 > 

trace(方陣) \Rightarrow 運算式

傳回方陣的描跡(主對角線上的所有元素總和)。

trace $\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix}$	15
trace $\begin{pmatrix} a & 0 \\ 1 & a \end{pmatrix}$	2· <i>a</i>

Try

Try
 區段1

Else
 區段2

EndTry

執行區段1，除非出現錯誤。若區段1出現錯誤，則程式改為執行區段2。*errCode*系統變數含有錯誤代碼，可讓程式執行錯誤復原作業。若需錯誤代碼一覽表，請參考「錯誤代碼與錯誤訊息」，頁碼：226。

區段1與區段2可以只是一個語句，也可以是由「」字元分隔的一連串語句。

輸入範例的注意事項：關於輸入多行程式和函數定義的說明，請參閱產品手冊中的「計算工具」章節。

例 2

若要觀察 Try、ClrErr 與 PassErr 指令的運作方式，請輸入如右所示的 *eigenvals()* 程式。請執行以下各個運算式，以便執行程式。

$$\text{eigenvals}\left(\begin{bmatrix} -3 \\ -41 \\ 5 \end{bmatrix}, [-1 \ 2 \ -3.1]\right)$$

$$\text{eigenvals}\left([1 \ 2 \ 3], \begin{bmatrix} 1 \\ 2 \end{bmatrix}\right)$$

附註：另請參考第 頁碼：25 頁的 ClrErr 和第 頁碼：121 頁的 PassErr。

```
Define prog1()=Prgm
  Try
    z:=z+1
    Disp "z incremented."
  Else
    Disp "Sorry, z undefined."
  EndTry
EndPrgm
```

Done

z:=1:prog1()

z incremented.

Done

```
DelVar z:prog1()
```

Sorry, z undefined.

Done

```
Define eigenvals(a,b)=Prgm
```

```
c 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
```

```
EndPrgm
```

tTest μ_0 , 列表[,Freq[,Hypoth]]

(輸入資料列表)

tTest μ_0, \bar{x}, sx, n , [Hypoth]

(輸入統計摘要)

執行單個不明母群體平均值 μ 的假設檢定，且母群體標準差 σ 不明。

stat.results 變數會儲存結果摘要(請參閱第頁碼: 162 頁)。

以 $H_0: \mu = \mu_0$ 檢定以下項目:

$H_a: \mu < \mu_0$ 時，設定 *Hypoth*<0

$H_a: \mu \neq \mu_0$ (預設值) 時，設定 *Hypoth*=0

$H_a: \mu > \mu_0$ 時，設定 *Hypoth*>0

如要了解列表中有空元素時的影響，請參考“空元素”，頁碼: 219。

輸出變數	說明
stat.t	$(\bar{x} - \mu_0) / (\text{stdev} / \text{sqrt}(n))$
stat.PVal	無效假說被否定之最低意義標準
stat.df	自由度
stat. \bar{x}	列表中資料序列的平均值樣本
stat.sx	資料序列的樣本標準差
stat.n	樣本數

tTest_2Samp

tTest_2Samp 列表1, 列表2[,Freq1[,Freq2[,Hypoth[,合併]]]]

(輸入資料列表)

tTest_2Samp $\bar{x}1, sx1, n1, \bar{x}2, sx2, n2$, [Hypoth[,合併]]

(輸入統計摘要)

計算二樣本 *t* 檢定。*stat.results* 變數會儲存結果摘要(請參閱第頁碼: 162 頁)。

以 $H_0: \mu_1 = \mu_2$ 檢定以下項目：

$H_a: \mu_1 < \mu_2$ 時，設定 $Hypoth < 0$

$H_a: \mu_1 \neq \mu_2$ (預設值) 時，設定 $Hypoth = 0$

$H_a: \mu_1 > \mu_2$ 時，設定 $Hypoth > 0$

合併=1 合併變異數

合併=0 不合併變異數

如要了解列表中有空元素時的影響，請參考 [空元素](#)，頁碼：219。

輸出變數	說明
stat.t	平均值差的標準常態計算值
stat.PVal	無效假說被否定之最低意義標準
stat.df	t-統計量自由度
stat.x1、stat.x2	列表 1 和列表 2 中資料序列的平均值樣本
stat.sx1、stat.sx2	列表 1 和列表 2 中資料序列的標準差樣本
stat.n1、stat.n2	樣本大小
stat.sp	合併標準差。合併=1 時才予計算。

tvmFV()

tvmFV(*N,I,PV,Pmt,[PpY],[CpY],[PmtAt]*)
⇒ 值

tvmFV(120,5,0,-500,12,12) 77641.1

可計算貨幣未來價值的財務函數。

附註：貨幣時間價值引數表說明了貨幣時間價值函數使用的引數(第頁碼：179 頁)。另請參考 **amortTbl()**，頁碼：8。

tvmI()

tvmI(*N,PV,Pmt,FV,[PpY],[CpY],[PmtAt]*)
⇒ 值

tvmI(240,100000,-1000,0,12,12) 10.5241

可計算每年利率的財務函數。

附註：貨幣時間價值引數表說明了貨幣時間價值函數使用的引數(第頁碼: 179 頁)。另請參考 **amortTbl()**, 頁碼: 8。

tvmN()

tvmN(*I,PV,Pmt,FV,[PpY],[CpY],[PmtAt]*)
⇒ 值

tvmN(5,0,-500,77641,12,12) 120.

可計算付款期數的財務函數。

附註：貨幣時間價值引數表說明了貨幣時間價值函數使用的引數(第頁碼: 179 頁)。另請參考 **amortTbl()**, 頁碼: 8。

tvmPmt()

tvmPmt(*N,I,PV,FV,[PpY],[CpY],[PmtAt]*)
⇒ 值

tvmPmt(60,4,30000,0,12,12) -552.496

可計算每期付款金額的財務函數。

附註：貨幣時間價值引數表說明了貨幣時間價值函數使用的引數(第頁碼: 179 頁)。另請參考 **amortTbl()**, 頁碼: 8。

tvmPV()

tvmPV(*N,I,Pmt,FV,[PpY],[CpY],[PmtAt]*)
⇒ 值

tvmPV(48,4,-500,30000,12,12) -342.67

可計算現值的財務函數。


附註：貨幣時間價值引數表說明了貨幣時間價值函數使用的引數(第頁碼: 179 頁)。另請參考 **amortTbl()**, 頁碼: 8。

貨幣時間價值引數*	說明	資料類型
<i>N</i>	付款期數	實數
<i>I</i>	年利率	實數

貨幣時間 價值引數*	說明	資料類型
<i>PV</i>	現值	實數
<i>Pmt</i>	付款金額	實數
<i>FV</i>	未來值	實數
<i>PpY</i>	每年付款期數, 預設值=1	>0 的整數
<i>CpY</i>	每年複利期數, 預設值=1	>0 的整數
<i>PmtAt</i>	每期付款時間是期末還是期初, 預設值=期末	整數(0=期末, 1=期初)

* 這些貨幣時間價值引數名稱, 與 Calculator 應用程式財務解題系統所用的貨幣時間價值變數名稱(例如 **tvm.pv** 與 **tvm.pmt**) 類似。但是財務函數不會把引數值或計算結果儲存到 TVM 變數。

TwoVar

目錄 > 

TwoVar *X*, *Y*, [*Freq*] [, 類別, 包含]

計算 **TwoVar** 統計。*stat.results* 變數會儲存結果摘要(請參閱第 頁碼: 162 頁)。

所有列表的維數都必須相同, 包含除外。

X、*Y* 是自變數和因變數列表。

Freq 是頻率值列表(非必要)。*Freq* 的每個元素, 可用於指定各 *X*、*Y* 對應資料點的出現頻率。預設值是 1。所有元素都必須是 ≥ 0 的整數。

類別是 *X*、*Y* 對應資料的數字類別代號列表。

包含是一個以上類別代號的列表。只有本列表中類別代號所對應的資料項目, 才會包含在計算作業中。

X、*Freq*、類別的任何列表中若有空元素, 則所有這些列表的對應元素就會是空元素。*X1* 至 *X20* 的任何列表中若有空元素, 則所有這些列表的對應元素就會是空元素。如需空元素的詳細資訊, 請參考第 頁碼: 219 頁。

輸出變數	說明
stat. \bar{x}	x 值的平均值
stat. x	x 值的和
stat. x2	x ² 值的和
stat.sx	x 的樣本標準差
stat. x	x 的母群體標準差
stat.n	資料點數量
stat. \bar{y}	y 值的平均值
stat. y	y 值的和
stat. y ²	y ² 值的和
stat.sy	y 的樣本標準差
stat. y	y 的母群體標準差
stat. xy	x · y 值的和
stat.r	相關係數
stat.MinX	x 值的最小值
stat.Q ₁ X	x 的第一四分位數
stat.MedianX	x 的中位數
stat.Q ₃ X	x 的第三四分位數
stat.MaxX	x 值的最大值
stat.MinY	y 值的最小值
stat.Q ₁ Y	y 的第一四分位數
stat.MedY	y 的中位數
stat.Q ₃ Y	y 的第三四分位數
stat.MaxY	y 值的最大值
stat. (x-) ²	x 的離均差平方和
stat. (y-) ²	y 的離均差平方和

unitV()

目錄 >

unitV(向量 I) ⇒ 向量

依照 向量 I 的形式傳回列單位向量或行單位向量。

向量 I 必須是單列矩陣或單行矩陣。

$$\text{unitV}\left(\begin{bmatrix} a & b & c \end{bmatrix}\right) = \left[\frac{a}{\sqrt{a^2+b^2+c^2}} \quad \frac{b}{\sqrt{a^2+b^2+c^2}} \quad \frac{c}{\sqrt{a^2+b^2+c^2}} \right]$$

$$\text{unitV}\left(\begin{bmatrix} 1 & 2 & 1 \end{bmatrix}\right) = \left[\frac{\sqrt{6}}{6} \quad \frac{\sqrt{6}}{3} \quad \frac{\sqrt{6}}{6} \right]$$

$$\text{unitV}\left(\begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}\right) = \begin{bmatrix} \frac{\sqrt{14}}{14} \\ \frac{14}{\sqrt{14}} \\ 7 \\ 3\sqrt{14} \\ 14 \end{bmatrix}$$

若要看到完整結果，請按 ▲，然後使用 ◀與▶ 移動游標。

unLock

目錄 >

unLock Var1[, Var2] [, Var3] ...

unLock Var.

解開鎖定指定變數或變數群組。您無法修改或刪除已鎖定的變數。

請參考第 頁碼: 98 頁的 Lock 和第 頁碼: 78 頁的 getLockInfo()。

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()

目錄 >

varPop(列表[, 頻率列表]) ⇒ 運算式

傳回列表的母群體變異數。

頻率列表的每個元素，代表列表中對應元素的連續出現次數。

附註：列表至少必須包含兩個元素。

varPop({5,10,15,20,25,30})	875
	12
Ans*1.	72.9167

若任何列表中有空元素，則系統會忽略該元素和另一列表中的對應元素。如需空元素的詳細資訊，請參考第 頁碼：219 頁。

varSamp()

varSamp(列表[, 頻率列表])⇒運算式

傳回列表的樣本變異數。

頻率列表的每個元素，代表列表中對應元素的連續出現次數。

附註：列表至少必須包含兩個元素。

若任何列表中有空元素，則系統會忽略該元素和另一列表中的對應元素。如需空元素的詳細資訊，請參考第 頁碼：219 頁。

varSamp(矩陣I[, 頻率矩陣])⇒矩陣

傳回包含矩陣I中各行樣本變異數的列向量。

頻率矩陣的每個元素，代表矩陣I中對應元素的連續出現次數。

若任何矩陣中有空元素，則系統會忽略該元素和另一矩陣中的對應元素。如需空元素的詳細資訊，請參考第 頁碼：219 頁。

附註：矩陣I至少必須包含兩列。

$$\text{varSamp}\{\{a,b,c\}\}$$

$$\frac{a^2 - a \cdot (b+c) + b^2 - b \cdot c + c^2}{3}$$

$$\text{varSamp}\{\{1,2,5,6,3,-2\}\}$$

$$\frac{31}{2}$$

$$\text{varSamp}\{\{1,3,5\},\{4,6,2\}\}$$

$$\frac{68}{33}$$

$$\text{varSamp}\left(\begin{pmatrix} 1 & 2 & 5 \\ -3 & 0 & 1 \\ .5 & .7 & 3 \end{pmatrix}\right) \quad [4.75 \quad 1.03 \quad 4]$$

$$\text{varSamp}\left(\begin{pmatrix} -1.1 & 2.2 \\ 3.4 & 5.1 \\ -2.3 & 4.3 \end{pmatrix}, \begin{pmatrix} 6 & 3 \\ 2 & 4 \\ 5 & 1 \end{pmatrix}\right) \quad [3.91731 \quad 2.08411]$$

W

Wait

Wait timeInSeconds

停止執行 *timeInSeconds* 秒。

Wait 特別適合需要短時間延遲以等待請求的資料的程式。

引數 *timeInSeconds* 必須包含一個簡化為 0 到 100 範圍內十進制值的表達式。此命令將此值四捨五入至到最接近的 0.1 秒。

要等待 4 秒：

Wait 4

要等待 1/2 秒：

Wait 0.5

要使用變量 *seccount* 等待 1.3 秒：

seccount:=1.3

如要取消進行中的 **Wait** ,

- **計算機**: 按住 鍵並重複按 **Enter** 鍵。
- **Windows®**: 按住 **F12** 鍵並重複按 **Enter** 鍵。
- **Macintosh®**: 按住 **F5** 鍵並重複按 **Enter** 鍵。
- **iPad®**: 應用程式顯示提示。您可以繼續等待或取消。

附註:可以在使用者自行定義的程式中使用 **Wait** 指令,而非在函數中。

Wait seccount

此範例將開啟一個綠色 LED 0.5 秒,然後關閉。

```
Send "SET GREEN 1 ON"
Wait 0.5
Send "SET GREEN 1 OFF"
```

warnCodes()

warnCodes(Expr1, StatusVar)⇒運算式

計算運算式 *Expr1*, 傳回結果, 並將任何所產生警告的代碼儲存在 *StatusVar* 列表變數中。如果未產生警告, 則此函數會指定空的列表給 *StatusVar*。

Expr1 可以是任何有效的 TI-Nspire™ 或 TI-Nspire™ CAS 數學運算式。無法對 *Expr1* 使用指令或指定工作。

StatusVar 必須是有效的變數名稱。

如需警告代碼與相關訊息的列表, 請參考第 頁碼: 234 頁。

warnCodes $\left(\text{solve} \left(\sin(10 \cdot x) = \frac{x^2}{x}, x \right), \text{warn} \right)$

$x = -0.84232$ or $x = -0.706817$ or $x = -0.2852$

warn {10007, 10009}

若要看到完整結果, 請按 , 然後使用 與 移動游標。

when()

when(條件, 真結果 [, 偽結果], [, 不明結果]) ⇒運算式

根據條件是真、偽還是不明, 傳回真結果、偽結果或不明結果。若引數太少, 不足以求出適當結果, 則傳回輸入值。

若省略偽結果和不明結果, 則可得出定義僅限於條件是真的範圍內運算式。

用 **undef** 偽結果可定義出只在某區間範圍內繪圖的運算式。

$\text{when}(x < 0, x + 3) | x = 5$ undef

when()

目錄 >

when() 可用於定義遞迴函數。

$\text{when}(n>0, n \cdot \text{factorial}(n-1), 1) \rightarrow \text{factorial}(n)$	Done
$\text{factorial}(3)$	6
$3!$	6

While

目錄 >

While 條件
區段

EndWhile

只要條件是真，就執行區段中的語句。

區段可以只是一個語句，也可以是由「`]`」字元分隔的一系列語句。

輸入範例的注意事項：關於輸入多行程式和函數定義的說明，請參閱產品手冊中的「計算工具」章節。

Define $\text{sum_of_recip}(n) = \text{Func}$	
Local $i, \text{tempsum}$	
$1 \rightarrow i$	
$0 \rightarrow \text{tempsum}$	
While $i \leq n$	
$\text{tempsum} + \frac{1}{i} \rightarrow \text{tempsum}$	
$i+1 \rightarrow i$	
EndWhile	
Return tempsum	
EndFunc	Done
$\text{sum_of_recip}(3)$	$\frac{11}{6}$

X

xor (互斥)

目錄 >

布林運算式1 **xor** 布林運算式2 傳回布林運算式

true xor true	false
$5>3 \text{ xor } 3>5$	true

布林列表1 **xor** 布林列表2 傳回布林列表

布林矩陣1 **xor** 布林矩陣2 傳回布林矩陣

若布林運算式1是真而布林運算式2是偽則傳回真，反之亦然。

若兩個引數同時是真或同時是偽則傳回偽。若其中一個引數無法解出真偽值，則傳回化簡的布林運算式。

附註：請參考 **or**，頁碼：119。

整數1 **xor** 整數2 \Rightarrow 整數

十六進位基底模式：

請注意：數字零，而非英文字母O。

利用 **xor** 功能個別位元比較兩個實際整數。系統內部會把兩個整數轉換成有正負號的 64 位元二進位數字。比較對應的位元時，如果只有一個位元是 1，則結果是 1；如果兩個位元都是 0 或都是 1，則結果是 0。傳回的值代表位元結果，並會以基底模式為顯示依據。

您可以用任何整數做為基底。如果要輸入二進位或十六進位數字，則必須在前面分別加上 **0b** 或 **0h** 字首。如果沒有加上字首，就會把整數視為十進位(以 10 為底)。

如果您輸入的十進位整數太大，超出具正負號 64 位元二進位格式的範圍，系統會以對稱模數運算法使數值落入適當範圍。如需詳細資訊，請參閱

►**Base2**，頁碼：17。

附註：請參考 **or**，頁碼：119。

0h7AC36 xor 0h3D5F

0h79169

二進位基底模式：

0b100101 xor 0b100

0b100001

附註：您可輸入最長 64 位的二進位數字(不含 **0b** 字首)。您可輸入最長 16 位的十六進位數字。

Z

zeros()

zeros(Expr, Var) ⇒ 列表

zeros(Expr, Var=猜測值) ⇒ 列表

傳回可以使 $Expr=0$ 的 Var 之可能實數值列表。**zeros()** 為了完成運算，必須計算 **explist(solve(Expr=0, Var), Var)**。

對某些目的來說，**zeros()** 的結果形式比 **solve()** 更方便。但 **zeros()** 的結果形式無法表達隱解、需要不等式的解，或沒有 Var 的解。

附註：另請參考 **cSolve()**、**cZeros()**、**solve()**。

zeros({運算式1, 運算式2}, {VarOrGuess1, VarOrGuess2 [, ...]}) ⇒ 矩陣

傳回聯立代數運算式的可能實數零點解。每個 $VarOrGuess$ 用於指定您尋找的未知數值。

$$\text{zeros}\left(a \cdot x^2 + b \cdot x + c, x\right)$$

$$\left[\frac{\sqrt{b^2 - 4 \cdot a \cdot c} - b}{2 \cdot a}, \frac{-\left(\sqrt{b^2 - 4 \cdot a \cdot c} + b\right)}{2 \cdot a} \right]$$

$$a \cdot x^2 + b \cdot x + c | x = \text{Ans}[2] \quad 0$$

$$\text{exact}\left(\text{zeros}\left(a \cdot \left(e^x + x\right) \cdot \left(\text{sign}(x) - 1\right), x\right)\right) \quad \left\{ \left[\right] \right\}$$

$$\text{exact}\left(\text{solve}\left(a \cdot \left(e^x + x\right) \cdot \left(\text{sign}(x) - 1\right) = 0, x\right)\right)$$

$$e^x + x = 0 \text{ or } x > 0 \text{ or } a = 0$$

您也可以指定或猜測一個變數的初始值。所有 *VarOrGuess* 的格式必須如下：

變數

- 或 -

變數 = 實數或非實數

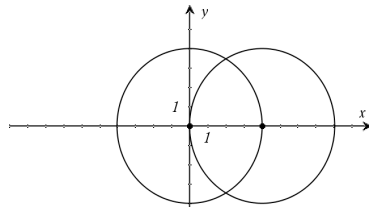
舉例來說， x 和 $x=3$ 都有效。

若所有運算式都是多項式，且若您並未指定或猜測任何初始值，則 **zeros()** 會以 Grobner/Buchberger 詞彙的消去法，試圖求得所有實數零點。

例如假設原點有一個半徑是 r 的圓，第一個圓和正 x 軸交會之處，則是另一個半徑是 r 的圓的圓心。利用 **zeros()** 可找出交點。

如右例中的 r 所示，聯立多項式可包含無具體值的額外變數(但表現出稍後可代入的指定數值)。

所產生矩陣的每一列都代表一個零點，其元素順序和 *varOrGuess* 列表相同。為擷取某列字串，請依 [row] 製作矩陣的索引。



$$\text{zeros}\left(\left\{x^2+y^2-r^2, (x-r)^2+y^2-r^2\right\}, \{x,y\}\right)$$

$\frac{r}{2}$	$\frac{-\sqrt{3}\cdot r}{2}$
$\frac{r}{2}$	$\frac{\sqrt{3}\cdot r}{2}$

擷取第 2 列：

$$\text{Ans}[2]$$

$\frac{r}{2}$	$\frac{\sqrt{3}\cdot r}{2}$
---------------	-----------------------------

您也可以(或改為)加入運算式中未出現的未知數。例如您可加入 z 當作未知數，將前例延伸成兩個平行相交的圓柱(半徑為 r)。圓柱零點能說明零點系可包含 *ck* 格式的任意常數，其中 k 是從 1 到 255 的整數下標。

$$\text{zeros}\left(\left\{x^2+y^2-r^2, (x-r)^2+y^2-r^2\right\}, \{x,y,z\}\right)$$

$\frac{r}{2}$	$\frac{-\sqrt{3}\cdot r}{2}$	$c1$
$\frac{r}{2}$	$\frac{\sqrt{3}\cdot r}{2}$	$c1$

以多項式系統而言，未知數的列出順序對於計算時間或記憶體用量可能影響甚大。若您最初的選擇用完記憶體或耗費太多時間，請重排運算式或 *varOrGuess* 列表中的變數。

若您不加入任何猜測值，且任何運算式都不是任何變數的多項式，但所有運算式都是未知數的一次式，則 **zeros()** 會以高斯消去法，試圖求得所有實數零點。

$$\text{zeros}\left(\left\{x+e^z \cdot y-1, x-y-\sin(z)\right\},\{x,y\}\right)$$

$$\begin{bmatrix} e^z \cdot \sin(z)+1 & -(\sin(z)-1) \\ e^z+1 & e^z+1 \end{bmatrix}$$

若某聯立方程式既非所有變數的多項式，亦非未知數的線性方程式，則 **zeros()** 利用近似迭代法最多只能求得一個零點。為此，未知數的數量必須等於運算式數量，且運算式中的其他變數必須化簡成數字。

$$\text{zeros}\left(\left\{e^z \cdot y-1, y-\sin(z)\right\},\{y,z\}\right)$$

$$\begin{bmatrix} 0.041458 & 3.18306 \\ 0.001871 & 6.28131 \\ 4.76\text{E-}11 & 1796.99 \\ 2.\text{E-}13 & 254.469 \end{bmatrix}$$

每個未知數都從猜測值開始(如果有)，否則就從 0.0 開始。

使用猜測值可逐一尋找其他零點。為收斂，猜測值可能必須接近零點。

$$\text{zeros}\left(\left\{e^z \cdot y-1, y-\sin(z)\right\},\{y,z-2 \cdot \pi\}\right)$$

$$\begin{bmatrix} 0.001871 & 6.28131 \end{bmatrix}$$

zInterval

zInterval σ , 列表[,Freq[,CLevel]]

(輸入資料列表)

zInterval σ, \bar{x}, n [,CLevel]

(輸入統計摘要)

計算 z 信賴區間。*stat.results* 變數會儲存結果摘要(請參閱第 頁碼: 162 頁)。

如要了解列表中有空元素時的影響，請參考 \emptyset 空元素 \emptyset ，頁碼: 219。

輸出變數	說明
stat.CLower、stat.CUpper	不明母群體平均值的信賴區間
stat. \bar{x}	常態隨機分布的資料序列平均值樣本
stat.ME	邊際誤差
stat.sx	樣本標準差
stat.n	資料序列的長度與平均值樣本
stat. σ	列表資料序列的已知母群體標準差

zInterval_1Prop**zInterval_1Prop** $x, n, [CLevel]$

計算單一母群體比例 z 信賴區間。
stat.results 變數會儲存結果摘要(請參閱第 頁碼: 162 頁)。

x 是不為負的整數。

如要了解列表中有空元素時的影響, 請參考“[空元素](#)”, 頁碼: 219。

輸出變數	說明
stat.CLower、stat.CUpper	包含信心水準分布機率的信賴區間
stat. \hat{p}	計算出的成功比例
stat.ME	邊際誤差
stat.n	資料序列的樣本數

zInterval_2Prop**zInterval_2Prop** $x1, n1, x2, n2, [CLevel]$

計算二母群體比例 z 信賴區間。
stat.results 變數會儲存結果摘要(請參閱第 頁碼: 162 頁)。

$x1$ 和 $x2$ 是不為負的整數。

如要了解列表中有空元素時的影響, 請參考“[空元素](#)”, 頁碼: 219。

輸出變數	說明
stat.CLower、stat.CUpper	包含信心水準分布機率的信賴區間
stat. \hat{p} Diff	計算出的兩個比例差值
stat.ME	邊際誤差
stat. \hat{p} 1	第一個樣本比例估計值
stat. \hat{p} 2	第二個樣本比例估計值
stat.n1	資料序列一的樣本數
stat.n2	資料序列二的樣本數

zInterval_2Samp

zInterval_2Samp σ_1, σ_2 , 列表1, 列表2
[,Freq1[,Freq2],[CLevel]]

(輸入資料列表)

zInterval_2Samp $\sigma_1, \sigma_2, \bar{x}1, n1, \bar{x}2, n2$
[,CLevel]

(輸入統計摘要)

計算二樣本 z 信賴區間。*stat.results* 變數會儲存結果摘要(請參閱第 頁碼: 162 頁)。

如要了解列表中有空元素時的影響, 請參考 \emptyset 空元素 \emptyset , 頁碼: 219。

輸出變數	說明
stat.CLower、stat.CUpper	包含信心水準分布機率的信賴區間
stat. $\bar{x}1$ - $\bar{x}2$	常態隨機分布的資料序列平均值樣本
stat.ME	邊際誤差
stat. $\bar{x}1$ 、stat. $\bar{x}2$	常態隨機分布的資料序列平均值樣本
stat. σ_1 、stat. σ_2	列表 1 和列表 2 的標準差樣本
stat.n1、stat.n2	資料序列的樣本數
stat.r1、stat.r2	列表 1 和列表 2 資料序列的已知母群體標準差

zTest

zTest μ_0, σ , 列表, [Freq[,Hypoth]]

(輸入資料列表)

zTest $\mu_0, \sigma, \bar{x}, n$, [Hypoth]

(輸入統計摘要)

執行 z 檢定(頻率列表為頻率)。*stat.results* 變數會儲存結果摘要(請參閱第 頁碼: 162 頁)。

以 $H_0: \mu = \mu_0$ 檢定以下項目:

$H_a: \mu < \mu_0$ 時, 設定 *Hypoth*<0

$H_a: \mu \neq \mu_0$ (預設值)時, 設定 *Hypoth*=0

$H_a: \mu > \mu_0$ 時, 設定 *Hypoth*>0

如要了解列表中有空元素時的影響，請參考“空元素”[®]，頁碼：219。

輸出變數	說明
stat.z	$(\bar{x} - \mu_0) / (\sigma / \sqrt{n})$
stat.P 值	無效假說被否定之最低機率
stat. \bar{x}	列表中資料序列的平均值樣本
stat.sx	資料序列的樣本標準差。只在輸入資料時才傳回結果。
stat.n	樣本數

zTest_1Prop

zTest_1Prop $p_0, x, n[, Hypoth]$

計算單一母群體比例 z 檢定。*stat.results* 變數會儲存結果摘要(請參閱第 頁碼：162 頁)。

x 是不為負的整數。

以 $H_0: p = p_0$ 檢定以下項目：

$H_a: p > p_0$ 時，設定 $Hypoth > 0$

$H_a: p \neq p_0$ (預設值) 時，設定 $Hypoth = 0$

$H_a: p < p_0$ 時，設定 $Hypoth < 0$

如要了解列表中有空元素時的影響，請參考“空元素”[®]，頁碼：219。

輸出變數	說明
stat.p0	假設的母群體比例
stat.z	比例的標準常態計算值
stat.PVal	無效假說被否定之最低意義標準
stat. \hat{p}	樣本比例估計值
stat.n	樣本數

zTest_2Prop

zTest_2Prop $x_1, n_1, x_2, n_2[, Hypoth]$

計算二母群體比例 z 檢定。 *stat.results* 變數會儲存結果摘要(請參閱第 頁碼: 162 頁)。

$x1$ 和 $x2$ 是不為負的整數。

以 $H_0: p1 = p2$ 檢定以下項目:

$H_a: p1 > p2$ 時, 設定 *Hypoth*>0

$H_a: p1 \neq p2$ (預設值) 時, 設定 *Hypoth*=0

$H_a: p < p0$ 時, 設定 *Hypoth*<0

如要了解列表中有空元素時的影響, 請參考 *空元素*, 頁碼: 219。

輸出變數	說明
stat.z	比例差的標準常態計算值
stat.PVal	無效假說被否定之最低意義標準
stat.p1	第一個樣本比例估計值
stat.p2	第二個樣本比例估計值
stat.p	合併樣本比例估計值
stat.n1、stat.n2	試驗 1 和試驗 2 採集的樣本數

zTest_2Samp

zTest_2Samp σ_1, σ_2 , 列表1, 列表2[, Freq1
[, Freq2[, Hypoth]]]

(輸入資料列表)

zTest_2Samp $\sigma_1, \sigma_2, \bar{x}1, n1, \bar{x}2, n2$ [, Hypoth]

(輸入統計摘要)

計算二樣本 z 檢定。 *stat.results* 變數會儲存結果摘要(請參閱第 頁碼: 162 頁)。

以 $H_0: \mu1 = \mu2$ 檢定以下項目:

$H_a: \mu1 < \mu2$ 時, 設定 *Hypoth*<0

$H_a: \mu1 \neq \mu2$ (預設值) 時, 設定 *Hypoth*=0

$H_a: \mu1 > \mu2$ 時, *Hypoth*>0

如要了解列表中有空元素時的影響，請參考^o空元素^o，頁碼：219。

輸出變數	說明
stat.z	平均值差的標準常態計算值
stat.PVal	無效假說被否定之最低意義標準
stat.x1, stat.x2	列表1和列表2中資料序列的平均值樣本
stat.sx1、stat.sx2	列表1和列表2中資料序列的標準差樣本
stat.n1、stat.n2	樣本大小

符號

+(加)



運算式1 + 運算式2⇒運算式

傳回兩個引數的和。

56	56
56+4	60
60+4	64
64+4	68
68+4	72

列表1 + 列表2⇒列表

矩陣1 + 矩陣2⇒矩陣

傳回包含列表1和列表2(或矩陣1和矩陣2)對應元素相加之和的列表(或矩陣)。

引數的維數必須相同。

$\left\{22,\pi,\frac{\pi}{2}\right\} \rightarrow I1$	$\left\{22,\pi,\frac{\pi}{2}\right\}$
$\left\{10,5,\frac{\pi}{2}\right\} \rightarrow I2$	$\left\{10,5,\frac{\pi}{2}\right\}$
$I1+I2$	$\{32,\pi+5,\pi\}$
$Ans+\{\pi,5,\pi\}$	$\{\pi+32,\pi,0\}$
$\begin{bmatrix} a & b \\ c & d \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$	$\begin{bmatrix} a+1 & b \\ c & d+1 \end{bmatrix}$

Expr + 列表1⇒列表

列表1 + Expr⇒列表

傳回列表，其元素包含列表1中各元素加上 Expr 的和。

傳回列表，其元素包含列表1中各元素加上值的和。

Expr + 矩陣1⇒矩陣

矩陣1 + Expr⇒矩陣

傳回矩陣，其元素是矩陣1中對角線上各元素加上 Expr 的和。矩陣1必需是方陣。

傳回矩陣，其元素是矩陣1中對角線上各元素加上值的和。矩陣1必需是方陣。

附註：請用 .+(點加)讓各元素加上運算式。

$15+\{10,15,20\}$	$\{25,30,35\}$
$\{10,15,20\}+15$	$\{25,30,35\}$

$20+\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$	$\begin{bmatrix} 21 & 2 \\ 3 & 24 \end{bmatrix}$
---	--

-(減)



運算式1 - 運算式2 ⇒ 運算式

$$6 - 2 = 4$$

傳回 運算式1 減 運算式2 的差。

$$\pi - \frac{\pi}{6} = \frac{5\pi}{6}$$

列表1 - 列表2 ⇒ 列表

$$\left\{ 22, \pi, \frac{\pi}{2} \right\} - \left\{ 10, 5, \frac{\pi}{2} \right\} = \left\{ 12, \pi - 5, 0 \right\}$$

矩陣1 - 矩陣2 ⇒ 矩陣

$$\begin{bmatrix} 3 & 4 \end{bmatrix} - \begin{bmatrix} 1 & 2 \end{bmatrix} = \begin{bmatrix} 2 & 2 \end{bmatrix}$$

使列表1(或矩陣1)的各元素減掉列表2(或矩陣2)的對應元素,然後傳回結果。

引數的維數必須相同。

Expr - 列表1 ⇒ 列表

$$15 - \{10, 15, 20\} = \{5, 0, -5\}$$

列表1 - Expr ⇒ 列表

$$\{10, 15, 20\} - 15 = \{-5, 0, 5\}$$

使 Expr 減掉列表1 的各元素,或使列表1 的各元素減掉 Expr,然後傳回結果列表。

使值減掉列表1 的各元素,或使列表1 的各元素減掉值,然後傳回結果列表。

Expr - 矩陣1 ⇒ 矩陣

$$20 - \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} = \begin{bmatrix} 19 & -2 \\ -3 & 16 \end{bmatrix}$$

矩陣1 - Expr ⇒ 矩陣

Expr - 矩陣1 : 傳回 Expr 乘以單位矩陣後減掉矩陣1 的矩陣。矩陣1 必須是方陣。

矩陣1 - Expr : 傳回矩陣1 減掉 Expr 和單位矩陣相乘結果的矩陣。矩陣1 必須是方陣。

附註: 請用 .-(點減) 使各元素減掉運算式。

·(乘)



運算式1 · 運算式2 ⇒ 運算式

$$2 \cdot 3.45 = 6.9$$

傳回兩個引數的乘積。

$$x \cdot y \cdot x = x^2 \cdot y$$

列表1 · 列表2 ⇒ 列表

$$\{1, 2, 3\} \cdot \{4, 5, 6\} = \{4, 10, 18\}$$

傳回列表,其元素包含列表1 和列表2 中對應元素的乘積。

$$\left\{ \frac{2}{a}, \frac{3}{2} \right\} \cdot \left\{ a^2, \frac{b}{3} \right\} = \left\{ 2 \cdot a, \frac{b}{2} \right\}$$

列表的維數必須相同。

· (乘)**☒ 鍵**

矩陣1 · 矩陣2 ⇒ 矩陣

傳回矩陣1 和矩陣2 的矩陣乘積。

矩陣1 的行數必須等於矩陣2 的列數。

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix} \cdot \begin{bmatrix} a & d \\ b & e \\ c & f \end{bmatrix} = \begin{bmatrix} a+2\cdot b+3\cdot c & d+2\cdot e+3\cdot f \\ 4\cdot a+5\cdot b+6\cdot c & 4\cdot d+5\cdot e+6\cdot f \end{bmatrix}$$

Expr · 列表1 ⇒ 列表

列表1 · Expr ⇒ 列表

傳回列表，其元素包含 Expr 乘以列表1 各元素的乘積。

傳回列表，其元素包含值乘以列表1 各元素的乘積。

Expr · 矩陣1 ⇒ 矩陣

矩陣1 · Expr ⇒ 矩陣

傳回矩陣，其元素包含 Expr 乘以矩陣1 各元素的乘積。

傳回矩陣，其元素包含值乘以矩陣1 各元素的乘積。

附註：請用 · (點乘) 讓運算式乘以各元素。

$$\pi \cdot \{4,5,6\} \qquad \{4\cdot\pi, 5\cdot\pi, 6\cdot\pi\}$$

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \cdot 0.01 \qquad \begin{bmatrix} 0.01 & 0.02 \\ 0.03 & 0.04 \end{bmatrix}$$

$$1 \cdot \text{identity}(3) \qquad \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

/ (除)**☒ 鍵**

運算式1 / 運算式2 ⇒ 運算式

傳回 運算式1 除以 運算式2 的商。

附註：另請參考**分數範本**，頁碼：1。

列表1 / 列表2 ⇒ 列表

傳回列表，其元素包含列表1 除以列表2 的商。

列表的維數必須相同。

Expr / 列表1 ⇒ 列表

列表1 / Expr ⇒ 列表

傳回列表，其元素包含 Expr 除以列表1 或列表1 除以 Expr 的商。

$$\frac{2}{3.45} \qquad .57971$$

$$\frac{x^3}{x} \qquad x^2$$

$$\frac{\{1,2,3\}}{\{4,5,6\}} \qquad \left\{0.25, \frac{2}{5}, \frac{1}{2}\right\}$$

$$\frac{a}{\{3,a,\sqrt{a}\}} \qquad \left\{\frac{a}{3}, 1, \sqrt{a}\right\}$$

$$\frac{\{a,b,c\}}{a\cdot b\cdot c} \qquad \left\{\frac{1}{b\cdot c}, \frac{1}{a\cdot c}, \frac{1}{a\cdot b}\right\}$$

/ (除)



傳回列表，其元素包含值除以列表1或列表1除以值的商。

矩陣1 / Expr ⇒ 矩陣

$$\frac{\begin{bmatrix} a & b & c \end{bmatrix}}{a \cdot b \cdot c} \quad \begin{bmatrix} \frac{1}{b \cdot c} & \frac{1}{a \cdot c} & \frac{1}{a \cdot b} \end{bmatrix}$$

傳回矩陣，其元素包含矩陣1/Expr的商。

附註：請用 ./ (點除) 使運算式除以各元素。

^ (乘幂)



運算式1 ^ 運算式2 ⇒ 運算式

$$4^2 \quad 16$$

$$\{a, 2, c\}^{\{1, b, 3\}} \quad \{a, 2^b, c^3\}$$

列表1 ^ 列表2 ⇒ 列表

傳回以第一個引數為底，第二個引數為幕的指數值。

附註：另請參考指數範本，頁碼：1。

若是列表，則傳回以列表1中各元素為底，列表2對應元素為幕的指數值。

在實數領域中，有奇分母的簡化指數之分數幕會使用實數分枝，複數模式則使用主枝。

Expr ^ 列表1 ⇒ 列表

$$p^{\{a, 2, -3\}} \quad \left\{ p^a, p^2, \frac{1}{p^3} \right\}$$

傳回以 Expr 為底，列表1各元素為幕的指數值。

列表1 ^ Expr ⇒ 列表

$$\{1, 2, 3, 4\}^{-2} \quad \left\{ 1, \frac{1}{4}, \frac{1}{9}, \frac{1}{16} \right\}$$

傳回以列表1中各元素為底，Expr 為幕的指數值。

方陣1 ^ 整數 ⇒ 矩陣

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}^2 \quad \begin{bmatrix} 7 & 10 \\ 15 & 22 \end{bmatrix}$$

傳回以方陣1為底，整數為幕的指數值。

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}^{-1} \quad \begin{bmatrix} -2 & 1 \\ 3 & -1 \\ 2 & 2 \end{bmatrix}$$

方陣1 必須是方陣。

若整數 = -1，則計算反矩陣。

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}^{-2} \quad \begin{bmatrix} \frac{11}{4} & -\frac{5}{4} \\ 2 & 2 \\ -\frac{15}{4} & \frac{7}{4} \end{bmatrix}$$

若整數 < -1，則以適當的正幕次方計算反矩陣。

x^2 (平方)

x^2 鍵

運算式 $I^2 \Rightarrow$ 運算式

傳回引數的平方值。

列表 $I^2 \Rightarrow$ 列表

傳回列表 I 中各元素平方值的列表。

方陣 $I^2 \Rightarrow$ 矩陣

傳回方陣 I 的矩陣平方值。這和計算各元素的平方值不同。請用 $.^2$ 計算各元素的平方值。

4^2	16
$\{2,4,6\}^2$	$\{4,16,36\}$
$\begin{bmatrix} 2 & 4 & 6 \\ 3 & 5 & 7 \\ 4 & 6 & 8 \end{bmatrix}^2$	$\begin{bmatrix} 40 & 64 & 88 \\ 49 & 79 & 109 \\ 58 & 94 & 130 \end{bmatrix}$
$\begin{bmatrix} 2 & 4 & 6 \\ 3 & 5 & 7 \\ 4 & 6 & 8 \end{bmatrix}.^2$	$\begin{bmatrix} 4 & 16 & 36 \\ 9 & 25 & 49 \\ 16 & 36 & 64 \end{bmatrix}$

$+$ (點和)

$+$ 鍵

矩陣 $I +$ 矩陣 $2 \Rightarrow$ 矩陣

$Expr +$ 矩陣 $1 \Rightarrow$ 矩陣

矩陣 $I +$ 矩陣 2 : 傳回矩陣, 其元素是矩陣 I 和矩陣 2 中每組成對對應元素的和。

$Expr +$ 矩陣 1 : 傳回矩陣, 其元素是 $Expr$ 和矩陣 1 各元素的和。

$\begin{bmatrix} a & 2 \\ b & 3 \end{bmatrix} + \begin{bmatrix} c & 4 \\ 5 & d \end{bmatrix}$	$\begin{bmatrix} a+c & 6 \\ b+5 & d+3 \end{bmatrix}$
$x + \begin{bmatrix} c & 4 \\ 5 & d \end{bmatrix}$	$\begin{bmatrix} x+c & x+4 \\ x+5 & x+d \end{bmatrix}$

$-$ (點差)

$-$ 鍵

矩陣 $I -$ 矩陣 $2 \Rightarrow$ 矩陣

$Expr -$ 矩陣 $1 \Rightarrow$ 矩陣

矩陣 $I -$ 矩陣 2 : 傳回矩陣, 其元素是矩陣 I 和矩陣 2 中每組成對對應元素的差。

$Expr -$ 矩陣 1 : 傳回矩陣, 其元素是 $Expr$ 和矩陣 1 各元素的差。

$\begin{bmatrix} a & 2 \\ b & 3 \end{bmatrix} - \begin{bmatrix} c & 4 \\ 5 & d \end{bmatrix}$	$\begin{bmatrix} a-c & -2 \\ b-d & -2 \end{bmatrix}$
$x - \begin{bmatrix} c & 4 \\ 5 & d \end{bmatrix}$	$\begin{bmatrix} x-c & x-4 \\ x-d & x-5 \end{bmatrix}$

\cdot (點乘)

\cdot 鍵

矩陣 $I \cdot$ 矩陣 $2 \Rightarrow$ 矩陣

$Expr \cdot$ 矩陣 $1 \Rightarrow$ 矩陣

$\begin{bmatrix} a & 2 \\ b & 3 \end{bmatrix} \cdot \begin{bmatrix} c & 4 \\ 5 & d \end{bmatrix}$	$\begin{bmatrix} a \cdot c & 8 \\ 5 \cdot b & 3 \cdot d \end{bmatrix}$
$x \cdot \begin{bmatrix} a & b \\ c & d \end{bmatrix}$	$\begin{bmatrix} a \cdot x & b \cdot x \\ c \cdot x & d \cdot x \end{bmatrix}$

.(點乘)

□ □ × 鍵

矩陣1 . 矩陣2: 傳回矩陣, 其元素是矩陣1 和矩陣2 中每組成對對應元素的乘積。

Expr . 矩陣1: 傳回矩陣, 其元素包含 Expr 乘以矩陣1 各元素的乘積。

./(點除)

□ □ ÷ 鍵

矩陣1 ./ 矩陣2 ⇒ 矩陣

Expr ./ 矩陣1 ⇒ 矩陣

矩陣1 ./ 矩陣2: 傳回矩陣, 其元素是矩陣1 和矩陣2 中每組成對對應元素的商。

Expr ./ 矩陣1: 傳回矩陣, 其元素是 Expr 乘以矩陣1 各元素的商。

$\begin{bmatrix} a & 2 \\ b & 3 \end{bmatrix} ./ \begin{bmatrix} c & 4 \\ 5 & d \end{bmatrix}$	$\begin{bmatrix} \frac{a}{5} & \frac{1}{d} \\ \frac{c}{5} & \frac{2}{d} \end{bmatrix}$
$x ./ \begin{bmatrix} c & 4 \\ 5 & d \end{bmatrix}$	$\begin{bmatrix} \frac{x}{5} & \frac{x}{d} \\ \frac{c}{5} & \frac{4}{d} \end{bmatrix}$

.(點冪)

□ □ ^ 鍵

矩陣1 .^ 矩陣2 ⇒ 矩陣

Expr .^ 矩陣1 ⇒ 矩陣

矩陣1 .^ 矩陣2: 傳回矩陣, 使矩陣2 的各元素是矩陣1 對應元素的指數。

Expr .^ 矩陣1: 傳回矩陣, 使矩陣1 的各元素是 Expr 的指數。

$\begin{bmatrix} a & 2 \\ b & 3 \end{bmatrix} .^ \begin{bmatrix} c & 4 \\ 5 & d \end{bmatrix}$	$\begin{bmatrix} a^c & 16 \\ b^5 & 3^d \end{bmatrix}$
$x .^ \begin{bmatrix} c & 4 \\ 5 & d \end{bmatrix}$	$\begin{bmatrix} x^c & x^4 \\ x^5 & x^d \end{bmatrix}$

-(負值)

□ □ (-) 鍵

-運算式1 ⇒ 運算式

-列表1 ⇒ 列表

-矩陣1 ⇒ 矩陣

傳回引數的負值。

若是列表或矩陣, 則傳回所有元素的負值。

若引數是二進位或十六進位整數, 則負值會提供二的補數。

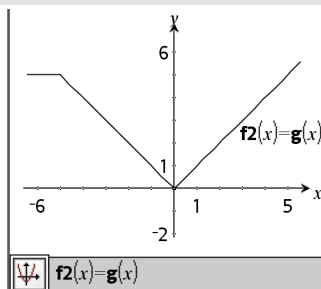
-2.43	-2.43
$-\{-1, 0.4, 1.2\mathbb{E}19\}$	$\{-1, -0.4, -1.2\mathbb{E}19\}$
$-a \cdot b$	$a \cdot b$

二進位基底模式:

重要訊息: 數字零, 而非英文字母 O

= (等於)

 鍵



≠ (不等於)

  鍵

運算式1 \neq 運算式2 \Rightarrow 布林運算式

請參考 $=$ (等於) 的範例。

列表1 \neq 列表2 \Rightarrow 布林列表

矩陣1 \neq 矩陣2 \Rightarrow 布林矩陣

若確定 運算式1 不等於 運算式2, 則傳回真。

若確定 運算式1 等於 運算式2, 則傳回偽。

若是其他情形, 則傳回方程式的化簡形式。

若是列表和矩陣, 則傳回對應元素逐一比較的結果。

附註: 如果要從鍵盤插入本運算子, 可輸入 \neq

< (小於)

  鍵

運算式1 $<$ 運算式2 \Rightarrow 布林運算式

請參考 $=$ (等於) 的範例。

列表1 $<$ 列表2 \Rightarrow 布林列表

矩陣1 $<$ 矩陣2 \Rightarrow 布林矩陣

若確定 運算式1 小於 運算式2, 則傳回真。

若確定 運算式1 大於或等於 運算式2, 則傳回偽。

< (小於)

ctrl = 鍵

若是其他情形，則傳回方程式的化簡形式。

若是列表和矩陣，則傳回對應元素逐一比較的結果。

≤ (小於或等於)

ctrl = 鍵

運算式1 ≤ 運算式2 ⇒ 布林運算式

請參考 = (等於) 的範例。

列表1 ≤ 列表2 ⇒ 布林列表

矩陣1 ≤ 矩陣2 ⇒ 布林矩陣

若確定 運算式1 小於或等於 運算式2，則傳回真。

若確定 運算式1 大於 運算式2，則傳回偽。

若是其他情形，則傳回方程式的化簡形式。

若是列表和矩陣，則傳回對應元素逐一比較的結果。

附註：如果要從鍵盤插入本運算子，可輸入 <=

> (大於)

ctrl = 鍵

運算式1 > 運算式2 ⇒ 布林運算式

請參考 = (等於) 的範例。

列表1 > 列表2 ⇒ 布林列表

矩陣1 > 矩陣2 ⇒ 布林矩陣

若確定 運算式1 大於 運算式2，則傳回真。

若確定 運算式1 小於或等於 運算式2，則傳回偽。

若是其他情形，則傳回方程式的化簡形式。

若是列表和矩陣，則傳回對應元素逐一比較的結果。

≥(大於或等於)

ctrl = 鍵

運算式1 ≥ 運算式2 ⇒ 布林運算式

請參考 E|(等於) 的範例。

列表1 ≥ 列表2 ⇒ 布林列表

矩陣1 ≥ 矩陣2 ⇒ 布林矩陣

若確定 運算式1 大於或等於 運算式2，則傳回真。

若確定 運算式1 小於 運算式2，則傳回偽。

若是其他情形，則傳回方程式的化簡形式。

若是列表和矩陣，則傳回對應元素逐一比較的結果。

附註：如果要從鍵盤插入本運算子，可輸入 >=

⇒(邏輯隱含)

ctrl = 鍵

布林運算式1 ⇒ 布林運算式2 傳回 布林運算式

5 > 3 or 3 > 5	true
----------------	------

5 > 3 ⇒ 3 > 5	false
---------------	-------

布林列表1 ⇒ 布林列表2 傳回 布林列表

3 or 4	7
--------	---

3 ⇒ 4	-4
-------	----

布林矩陣1 ⇒ 布林矩陣2 傳回 布林矩陣

{1,2,3} or {3,2,1}	{3,2,3}
--------------------	---------

{1,2,3} ⇒ {3,2,1}	{-1,-1,-3}
-------------------	------------

整數1 ⇒ 整數2 傳回 整數

計算運算式 not <引數1> or <引數2> 並傳回真偽值或方程式的化簡形式。

若是列表和矩陣，則傳回對應元素逐一比較的結果。

附註：如果要從鍵盤插入本運算子，可輸入 =>

⇔(邏輯雙隱含, XNOR)

ctrl = 鍵

布林運算式1 ⇔ 布林運算式2 傳回
布林運算式

5>3 xor 3>5	true
-------------	------

布林列表1 ⇔ 布林列表2 傳回 布林
列表

5>3 ⇔ 3>5	false
-----------	-------

布林矩陣1 ⇔ 布林矩陣2 傳回 布林
矩陣

3 xor 4	7
---------	---

3 ⇔ 4	-8
-------	----

{1,2,3} xor {3,2,1}	{2,0,2}
---------------------	---------

{1,2,3} ⇔ {3,2,1}	{-3,-1,-3}
-------------------	------------

整數1 ⇔ 整數2 傳回 整數

傳回兩個引數 **XOR** 布林運算的負值。
傳回真偽值或方程式的化簡形式。

若是列表和矩陣，則傳回對應元素逐
一比較的結果。

附註: 如果要從鍵盤插入本運算子，可
輸入<=>

!(階乘)

?! 鍵

運算式1!⇒ 運算式

5!	120
----	-----

列表1!⇒ 列表

{{5,4,3}}!	{120,24,6}
------------	------------

矩陣1!⇒ 矩陣

$\begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}!$	$\begin{bmatrix} 1 & 2 \\ 6 & 24 \end{bmatrix}$
---	---

傳回引數的階乘。

若是列表或矩陣，則傳回由各元素階
乘所組成的列表或矩陣。

&(接上)


ctrl 鍵

字串1 & 字串2 ⇒ 字串

"Hello "&"Nick"	"Hello Nick"
-----------------	--------------

傳回將字串2 接上字串1 的文字字串。

d() (導數)

目錄 > 

d(運算式 I, Var[, 階數]) ⇒ 運算式

d(列表 I, Var[, 階數]) ⇒ 列表

d(矩陣 I, Var[, 階數]) ⇒ 矩陣

傳回第一個引數對 Var 變數的一階導數。

階數(若加入)必須是整數。若階數小於零,則會得出反導數。

附註: 如果要從鍵盤插入本函數,可輸入 **derivative(...)**。

d() 並不會遵守一般求值方法(亦即將引數完全化簡後,再將函數定義套用到這些完全化簡過的引數)。d() 所執行的是以下步驟:

1. 只把第二個引數化簡到無法求出非變數為止。
2. 只把第一個引數化簡到可回收第 1 步所求出變數的任何儲存值為止。
3. 求出第 2 步結果對於第 1 步所求變數的符號導數。

若第 1 步求出的變數有儲存值或以約束(『』)運算子指定的值,請將該值代入第 3 步的結果。


附註: 另請參考第 頁碼: 5 頁的一階導數、第 頁碼: 6 頁的二階導數或第 頁碼: 6 頁的 N 階導數。

$$\frac{d}{dx}(f(x) \cdot g(x)) = \frac{d}{dx}(f(x)) \cdot g(x) + \frac{d}{dx}(g(x)) \cdot f(x)$$

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

$$\frac{d}{dx} \left(\left\{ x^2, x^3, x^4 \right\} \right) = \left\{ 2 \cdot x, 3 \cdot x^2, 4 \cdot x^3 \right\}$$

∫() (積分)

目錄 > 

∫(運算式 I, Var[, 下限, 下限]) ⇒ 運算式

∫(運算式 I, Var[, 常數]) ⇒ 運算式

傳回 運算式 I 對於 Var 變數從下限到上限的積分。

附註: 另請參考定積分範本或不定積分範本, 頁碼: 6。

附註: 如果要從鍵盤插入本函數,可輸入 **integral(...)**。

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

若省略下**限**和**上**限，則傳回反導數。
除非提供**常數**引數，否則會省略積分的符號**常數**。

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

$$\int (a \cdot x^2, x, c)$$

$$\frac{a \cdot x^3}{3} + c$$

數值**常數**可能區分同樣有效的反導數。這類**常數**可能隱而不見。反導數包含**對數**或**反三角函數**時更是如此。此外有時會加入**分段常數**運算式，以便在時間區間超過普通公式的情況下，讓反導數能維持有效。

∫() 在無法判定**分段 運算式 I** 是不是內建**函數**與**運算子**的顯式有限組合時，會傳回自身的值。



$$\int b \cdot e^{-x^2} + \frac{a}{x^2+a^2} dx \quad b \cdot \int e^{-x^2} dx + \tan^{-1}\left(\frac{x}{a}\right)$$

若提供**下**限和**上**限，則系統會在**下**限 < *Var* < **上**限的區間內尋找任何不連續點或不連續導數，並在這些位置再細分該區間。

如果在**自動**或**近似值**模式中設定為自動，則會在無法確定反導數或極限時，依情況採用數值積分。

如果是**近似值**設定，則會在可行時先嘗試數值積分。只有在不適用數值積分或計算失敗時，才會採用反導數。

附註： 強迫表示結果近似值，

計算機： 按  。

Windows®： 按 **Ctrl+Enter**。

Macintosh®： 按 **⌘+Enter**。

iPad®： 按住 **enter** 然後選擇 。

$$\int_{-1}^1 e^{-x^2} dx \quad 1.49365$$

您可嵌套 **∫()**，以便計算**多重積分**。積分極限可能受到積分函數外的積分變數影響。

附註： 另請參考 **nInt()**，頁碼：113。

$$\int_0^a \int_0^x \ln(x+y) dy dx$$

$$\frac{a^2 \cdot \ln(a)}{2} + \frac{a^2 \cdot (4 \cdot \ln(2) - 3)}{4}$$

$\sqrt{()}$ (平方根)

ctrl x² 鍵 $\sqrt{(\text{運算式 } I)} \Rightarrow \text{運算式}$

$$\sqrt{4} \qquad 2$$

 $\sqrt{(\text{列表 } I)} \Rightarrow \text{列表}$

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

傳回引數的平方根。

若是列表，則傳回列表1中各元素的平方根。

附註：如果要從鍵盤插入本函數，可輸入 **sqrt(...)****附註：**另請參考平方根範本，頁碼：1。

$\prod()$ (prodSeq)

目錄 >

 $\prod(\text{運算式 } I, \text{Var}, \text{低}, \text{高}) \Rightarrow \text{運算式}$

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

附註：如果要從鍵盤插入本函數，可輸入 **prodSeq(...)**。計算 運算式 I 在 Var 從低到高取值時的結果，並傳回結果的乘積。

$$\prod_{k=1}^n (k^2) \qquad (n!)^2$$

附註：另請參考乘積範本 (\prod)，頁碼：5。

$$\prod_{n=1}^5 \left\{ \left\{ \frac{1}{n}, n, 2 \right\} \right\} \qquad \left\{ \frac{1}{120}, 120, 32 \right\}$$

 $\prod(\text{運算式 } I, \text{Var}, \text{低}, \text{低}-1) \Rightarrow 1$

$$\prod_{k=4}^3 (k) \qquad 1$$

 $\prod(\text{運算式 } I, \text{Var}, \text{低}, \text{高}) \Rightarrow 1/\prod(\text{運算式 } I, \text{Var}, \text{高}+1, \text{低}-1)$ if 高 < 低-1

所用的乘積公式乃由以下參考資料導出：

Ronald L. Graham, Donald E. Knuth, and Oren Patashnik. Concrete Mathematics: A Foundation for Computer Science. Reading, Massachusetts: Addison-Wesley, 1994.

$$\prod_{k=4}^1 \left(\frac{1}{k}\right) \qquad 6$$

$$\prod_{k=4}^1 \left(\frac{1}{k}\right) \cdot \prod_{k=2}^4 \left(\frac{1}{k}\right) \qquad \frac{1}{4}$$

$\Sigma()$ (sumSeq)

目錄 >

$\Sigma(\text{運算式 } I, \text{Var}, \text{低}, \text{高}) \Rightarrow \text{運算式}$

附註：如果要從鍵盤插入本函數，可輸入 **sumSeq(...)**。

計算 **運算式 I** 在 **Var** 從低到高取值時的結果，並傳回結果的總和。

附註：另請參考**總和範本**，頁碼：5。

$\Sigma(\text{運算式 } I, \text{Var}, \text{低}, \text{低}-0) \Rightarrow 1$

$\Sigma(\text{運算式 } I, \text{Var}, \text{低}, \text{高}) \Rightarrow -\Sigma(\text{運算式 } I, \text{Var}, \text{高}+1, \text{低}-1)$ if $\text{高} < \text{低}-1$

所用的加總法公式乃由以下參考資料導出：

Ronald L. Graham, Donald E. Knuth, and Oren Patashnik. Concrete Mathematics: A Foundation for Computer Science. Reading, Massachusetts: Addison-Wesley, 1994.

$$\sum_{n=1}^5 \left(\frac{1}{n}\right) \quad \frac{137}{60}$$

$$\sum_{k=1}^n (k^2) \quad \frac{n \cdot (n+1) \cdot (2 \cdot n+1)}{6}$$

$$\sum_{n=1}^{\infty} \left(\frac{1}{n^2}\right) \quad \frac{\pi^2}{6}$$

$$\sum_{k=4}^3 (k) \quad 0$$

$$\sum_{k=4}^1 (k) \quad -5$$

$$\sum_{k=4}^1 (k) + \sum_{k=2}^4 (k) \quad 4$$

$\Sigma\text{Int}()$

目錄 >

$\Sigma\text{Int}(\text{NPmt}1, \text{NPmt}2, N, I, PV, [\text{Pmt}], [\text{FV}], [\text{PpY}], [\text{CpY}], [\text{PmtAt}], [\text{四捨五入值}]) \Rightarrow \text{值}$

$$\Sigma\text{Int}(1, 3, 12, 4.75, 20000., 12, 12) \quad -213.48$$

$\Sigma\text{Int}(\text{NPmt}1, \text{NPmt}2, \text{amortTable}) \Rightarrow \text{值}$

針對指定付款期數範圍，計算利息總和的攤銷函數。

NPmt1 與 **NPmt2** 用於界定付款期數範圍的起點和終點。

N, I, PV, Pmt, FV, PpY, CpY, PmtAt 的說明位於貨幣時間價值指數表(第頁碼：179頁)。

- 如省略 **Pmt**，則會以 **Pmt=tvmPmt(N, I, PV, FV, PpY, CpY, PmtAt)** 為預設

值。

- 如省略 *FV*，則會以 *FV*=0 為預設值。
- PpY*、*CpY*、*PmtAt* 的預設值和貨幣時間價值函數相同。

四捨五入值用於指定四捨五入的小數位數。預設值 = 2。

ΣInt(*NPmt1*,*NPmt2*,*amortTable*) 可根據 *amortTable* 攤銷表計算利息總和。*amortTable* 引數必須是 **amortTbl()** 一節所描述的矩陣形式(第頁碼: 8頁)。

附註: 另請參考下一節 ΣPrn() 和 Bal() (頁碼: 17)。

tbl:=amortTbl(12,12,4.75,20000.,12,12)

0	0.	0.	20000.
1	-77.49	-1632.43	18367.6
2	-71.17	-1638.75	16728.8
3	-64.82	-1645.1	15083.7
4	-58.44	-1651.48	13432.2
5	-52.05	-1657.87	11774.4
6	-45.62	-1664.3	10110.1
7	-39.17	-1670.75	8439.32
8	-32.7	-1677.22	6762.1
9	-26.2	-1683.72	5078.38
10	-19.68	-1690.24	3388.14
11	-13.13	-1696.79	1691.35
12	-6.55	-1703.37	-12.02

ΣInt(1,3,*tbl*) -213.48

ΣPrn()

ΣPrn(*NPmt1*,*NPmt2*,*N*,*I*,*PV*, [*Pmt*], [*FV*], [*PpY*], [*CpY*], [*PmtAt*], [四捨五入])
⇒ 值

ΣPrn(1,3,12,4.75,20000.,12,12) -4916.28

ΣPrn(*NPmt1*,*NPmt2*,*amortTable*)⇒ 值

針對指定付款期數範圍，計算本金總和的攤銷函數。

NPmt1 與 *NPmt2* 用於界定付款期數範圍的起點和終點。

N,*I*,*PV*,*Pmt*,*FV*,*PpY*,*CpY*,*PmtAt* 的說明位於貨幣時間價值引數表(第頁碼: 179頁)。

- 如省略 *Pmt*，則會以 *Pmt*=*tvmpmt*(*N*,*I*,*PV*,*FV*,*PpY*,*CpY*,*PmtAt*) 為預設值。
- 如省略 *FV*，則會以 *FV*=0 為預設值。
- PpY*、*CpY*、*PmtAt* 的預設值和貨幣時間價值函數相同。

四捨五入值用於指定四捨五入的小數位數。預設值 = 2。

tbl:=amortTbl(12,12,4.75,20000.,12,12)

0	0.	0.	20000.
1	-77.49	-1632.43	18367.57
2	-71.17	-1638.75	16728.82
3	-64.82	-1645.1	15083.72
4	-58.44	-1651.48	13432.24
5	-52.05	-1657.87	11774.37
6	-45.62	-1664.3	10110.07
7	-39.17	-1670.75	8439.32
8	-32.7	-1677.22	6762.1
9	-26.2	-1683.72	5078.38
10	-19.68	-1690.24	3388.14
11	-13.13	-1696.79	1691.35
12	-6.55	-1703.37	-12.02

ΣPrn(1,3,*tbl*) -4916.28

ΣPrn(*NPmt1*,*NPmt2*,*amortTable*) 可根據 *amortTable* 攤銷表計算已付本金總和。*amortTable* 引數必須是 **amortTbl()** 一節所描述的矩陣形式(第頁碼: 8 頁)。

附註: 另請參考上一節 ΣInt() 和 Bal() (頁碼: 17)。

#(間接)

  **鍵**

var 名稱字串

#("x"&"y"&"z")	xyz
----------------	-----

參照名稱是 *var* 名稱字串的變數。本功能讓您在函數中用字串建立變數名稱。

建立或參照 xyz 變數。

10 → r	10
"r" → s1	"r"
#s1	10

傳回名稱儲存在 s1 變數中的變數(r) 值。

E(科學記號)

 **鍵**

尾數E指數

23000.	23000.
以科學記號輸入數字。系統會將數字解譯為尾數 × 10 ^{指數} 。	2300000000.+4.1E15
3·10 ⁴	30000

以科學記號輸入數字。系統會將數字解譯為尾數 × 10^{指數}。

提示: 若要輸入 10 的次方卻不要結果是十進位值, 請用 10[^]整數。

附註: 如果要從電腦鍵盤插入本運算子, 可輸入 **@E**。例如輸入 **2.3@E4** 即可得到 2.3E4。

g(梯度)

 **鍵**

運算式 *lg* ⇒ 運算式

度數角、梯度角或弧度角模式:

列表 *lg* ⇒ 列表

cos(50 ^g)	$\frac{\sqrt{2}}{2}$
-----------------------	----------------------

矩陣 *lg* ⇒ 矩陣

cos{{0,100 ^g ,200 ^g }}	{1,0,-1}
--	----------

本函數讓您在度數角或弧度角模式下指定梯度角。

g(梯度)

π 鍵

若是弧度角模式，則將 運算式 I 乘以 $\pi/200$ 。

若是度數角模式，則將 運算式 I 乘以 $g/100$ 。

若是梯度角模式，則傳回 運算式 I 不變。

附註： 如果要從電腦鍵盤插入本符號，可輸入 @g。

r(弧度)

π 鍵

運算式 $I \Rightarrow$ 運算式

列表 $I \Rightarrow$ 列表

矩陣 $I \Rightarrow$ 矩陣

本函數讓您在度數角或梯度角模式下指定弧度角。

若是度數角模式，則將引數乘以 $180/\pi$ 。

若是弧度角模式，則傳回引數不變。

若是梯度角模式，則將引數乘以 $200/\pi$ 。

提示：若無論使用函數時所處模式為何，都要在函數定義中強迫採用弧度角，請用 r。

附註： 如果要從電腦鍵盤插入本符號，可輸入 @r。

度數角、梯度角或弧度角模式：

$$\cos\left(\frac{\pi}{4^r}\right) \quad \frac{\sqrt{2}}{2}$$
$$\cos\left(\left\{0^r, \frac{\pi}{12}, r, -(\pi)r\right\}\right) \quad \left\{1, \frac{(\sqrt{3+1})\sqrt{2}}{4}, -1\right\}$$

° (度數)

π 鍵

運算式 $I^\circ \Rightarrow$ 運算式

列表 $I^\circ \Rightarrow$ 列表

矩陣 $I^\circ \Rightarrow$ 矩陣

本函數讓您在梯度角或弧度角模式下指定度數角。

度數角、梯度角或弧度角模式：

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

弧度角模式：

附註： 強迫表示結果近似值，

° (度數)

 鍵

若是弧度角模式，則將引數乘以 $\pi/180$ 。

若是度數角模式，則傳回引數不變。

若是梯度角模式，則將引數乘以 10/9。

附註：如果要從電腦鍵盤插入本符號，可輸入 `@d`。

計算機：按 `ctrl` `enter`。

Windows®：按 `Ctrl+Enter`。

Macintosh®：按 `⌘+Enter`。

iPad®：按住 `enter` 然後選擇 。

$$\cos\left\{\left\{0, \frac{\pi}{4}, 90^\circ, 30.12^\circ\right\}\right\}$$

$$\{1, 0.707107, 0, 0.864976\}$$

°、'、" (度/分/秒)

`ctrl`  鍵

`dd°mm'ss.ss"` ⇒ 運算式

`dd` 正數或負數

`mm` 不可是負數

`ss.ss` 不可是負數

傳回 $dd+(mm/60)+(ss.ss/3600)$ 。

這個 -60 進位輸入格式有以下功能：

- 可輸入「度/分/秒」格式的角度，無須理會目前的角度模式。
- 可輸入「時/分/秒」格式的時間。

附註：`ss.ss` 後面接的是兩個單引號 (")，而非雙引號 (")。

度數角模式：

<code>25°13'17.5"</code>	25.2215
<code>25°30'</code>	$\frac{51}{2}$

∠ (角度)

`ctrl`  鍵

`[弧度, ∠θ_角度]` ⇒ 向量

(極座標輸入)

`[弧度, ∠θ_角度, Z_座標]` ⇒ 向量

(圓柱座標輸入)

`[弧度, ∠θ_角度, ∠θ_角度]` ⇒ 向量

(球面座標輸入)

根據直角、圓柱、球面的向量格式模式，傳回向量形式的座標。

附註：如果要從電腦鍵盤插入本符號，可輸入 `@<`。

在弧度角模式將向量格式設定成：

直角

$$\left[5 \angle 60^\circ \angle 45^\circ \right] \left[\frac{5\sqrt{2}}{4} \quad \frac{5\sqrt{6}}{4} \quad \frac{5\sqrt{2}}{2} \right]$$

圓柱

$$\left[5 \angle 60^\circ \angle 45^\circ \right] \left[\frac{5\sqrt{2}}{2} \quad \angle \frac{\pi}{3} \quad \frac{5\sqrt{2}}{2} \right]$$

∠(角度)

  **鍵**

球面

$$\left[5 \angle 60^\circ \angle 45^\circ \right] \quad \left[5 \angle \frac{\pi}{3} \angle \frac{\pi}{4} \right]$$

弧度角模式與直角複數格式:


$$5+3 \cdot i \left(10 \angle \frac{\pi}{4} \right) \quad 5-5 \cdot \sqrt{2} + (3-5 \cdot \sqrt{2}) \cdot i$$

(量 ∠ 角度) ⇒ 複數值

(極座標輸入)

以 $(r \angle \theta)$ 極座標形式輸入複數值。系統會以目前的角度模式設定解譯角度。

附註: 強迫表示結果近似值,

計算機: 按  .

Windows®: 按 **Ctrl+Enter**。

Macintosh®: 按 **⌘+Enter**。

iPad®: 按住 **enter** 然後選擇 .

$$5+3 \cdot i \left(10 \angle \frac{\pi}{4} \right) \quad -2.07107-4.07107 \cdot i$$

'(上標)

 **鍵**

變數'

變數''

在微分方程式中輸入上標符號。一個上標符號代表一階微分方程式，二個上標符號代表二階，以此類推。

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

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

_(代表空元素的底線)

請參考 °β空元素°, 頁碼: 219。



_(代表單位指示符號的底線)

  **鍵**

*Expr*_單位

$$3 \cdot \text{m} \blacktriangleright \text{ft} \quad 9.84252 \cdot \text{ft}$$

指定 *Expr* 的單位。所有單位名稱的開頭都必須是底線。

附註: 您可以在「目錄」中找到轉換符號 。按一下 ，然後按一下 [數學運算子]。

_(代表單位指示符號的底線)

ctrl  鍵

您可使用預設單位，亦可自行定義單位。如需預設單位一覽表，請打開 [目錄] 並顯示 [單位換算] 索引標籤。您可以在 [目錄] 中選取單位名稱，亦可直接輸入單位名稱。

變數_

變數無賦值時，系統會將其視為代表複數。依預設，若無 _，則系統會將變數視為實數。

若變數擁有值，則忽略 _，且變數會保留原來資料類型。

附註：若要將複數儲存到變數，其實不用 _。但若要有 **cSolve()** 和 **cZeros()** 的最佳計算結果，則建議輸入 _。

假設尚未指定 z:

$\text{real}(z)$	z
$\text{real}(z_)$	$\text{real}(z_)$
$\text{imag}(z)$	0
$\text{imag}(z_)$	$\text{imag}(z_)$

►(換算)

ctrl  鍵

$\text{Expr}_\text{單位1} \blacktriangleright \text{單位2} \Rightarrow \text{Expr}_\text{單位2}$



$3 \cdot \text{m} \blacktriangleright \text{ft}$ 9.84252 · ft

將運算式轉換成其他單位。

_底線字元可用於指定單位。單位必須屬於同一類，例如長度或面積。


如需預設單位一覽表，請打開 [目錄] 並顯示 [單位換算] 索引標籤：

- 您可以在列表中選取單位名稱。
- 您可以在列表最上方選取轉換運算子 \blacktriangleright 。

您也可以手動輸入單位名稱。在計算機上輸入單位名稱時若要輸入 []，請按  。

附註：若要換算溫度單位，請用 **tmpCnv()** 和 **ΔtmpCnv()**。 \blacktriangleright 轉換運算子無法處理溫度單位。

10^()

目錄 > 

$10^\wedge(\text{運算式1}) \Rightarrow \text{運算式}$

$10^{1.5}$ 31.6228

$10^\wedge(\text{列表1}) \Rightarrow \text{列表}$

$10^{\{0, -2.2, a\}}$ $\left\{ 1, \frac{1}{100}, 100, 10^a \right\}$

傳回以 10 為底，以引數為冪的指數值。

若是列表，則傳回以 10 為底，以列表 I 中各元素為冪的指數值。

10^ (方陣 I) ⇒ 方陣

傳回以 10 為底，以方陣 I 為冪的指數值。這和計算以 10 為底、以各元素為冪的指數值不同。若要了解計算方式，請參考 **cos()**。

方陣 I 必需可對角化。結果一定會包含浮點數。

10^{10^6}	$\begin{bmatrix} 1 & 5 & 3 \\ 4 & 2 & 1 \\ 6 & -2 & 1 \end{bmatrix}$	
		$\begin{bmatrix} 1.14336\text{E}7 & 8.17155\text{E}6 & 6.67589\text{E}6 \\ 9.95651\text{E}6 & 7.11587\text{E}6 & 5.81342\text{E}6 \\ 7.65298\text{E}6 & 5.46952\text{E}6 & 4.46845\text{E}6 \end{bmatrix}$

^-1 (倒數)

運算式 I ^-1 ⇒ 運算式

列表 I ^-1 ⇒ 列表

傳回引數的倒數。

若是列表，則傳回列表 I 中各元素的倒數。

方陣 I ^-1 ⇒ 方陣

傳回方陣 I 的反矩陣。

方陣 I 不得是奇異方陣。

$(3.1)^{-1}$	0.322581
$\{a, 4, 0.1, x, -2\}^{-1}$	$\left\{ \frac{1}{a}, \frac{1}{4}, -10, \frac{1}{x}, \frac{-1}{2} \right\}$

$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}^{-1}$	$\begin{bmatrix} -2 & 1 \\ 3 & -1 \\ 2 & 2 \end{bmatrix}$
$\begin{bmatrix} 1 & 2 \\ a & 4 \end{bmatrix}^{-1}$	$\begin{bmatrix} -2 & 1 \\ a-2 & a-2 \\ a & -1 \\ 2 \cdot (a-2) & 2 \cdot (a-2) \end{bmatrix}$

[(約束運算子)

運算式 | 布林運算式 I [and 布林運算式 2]...

$x+1 x=3$	4
-----------	---

運算式 | 布林運算式 I [or 布林運算式 2]...

$x+y x=\sin(y)$	$\sin(y)+y$
-----------------	-------------

$x+y \sin(y)=x$	$x+y$
-----------------	-------

([]) 符號可充當二元運算子。| 左邊的運算元是運算式。| 右邊的運算元可用於指定一個以上的關係，目的是決定運算式的化簡方法。| 後的多種關係必須由 **and** 或 **or** 邏輯連接。

|(約束運算子)

ctrl  鍵

約束運算子提供三種基本功能：

- 代換
- 區間限制
- 排除

代換的形式是等式，例如 $x=3$ 或 $y=\sin(x)$ 。為提高效果，左側應該是簡單變數。**Expr | 變數 = 值**：將值代入 Expr 中的每個變數。

區間約束的形式為 **[and]** 或 **[or]** 邏輯運算子連接的一個以上不等式。用其他方式可能化簡無效或無法計算時，區間約束法也能加以化簡。

$x^3-2\cdot x+7 \rightarrow f(x)$	Done
-----------------------------------	------

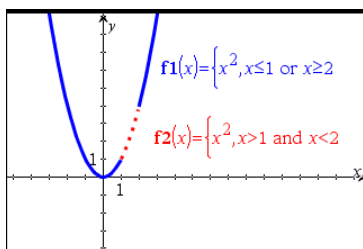
$f(x) _{x=\sqrt{3}}$	$\sqrt{3}+7$
----------------------	--------------

$(\sin(x))^2+2\cdot\sin(x)-6 \sin(x)=d$	$d^2+2\cdot d-6$
---	------------------

$\text{solve}(x^2-1=0,x) x>0 \text{ and } x<2$	$x=1$
--	-------

$\sqrt{x}\cdot\sqrt{\frac{1}{x}} x>0$	1
---------------------------------------	---

$\sqrt{x}\cdot\sqrt{\frac{1}{x}}$	$\sqrt{\frac{1}{x}}\cdot\sqrt{x}$
-----------------------------------	-----------------------------------



$\text{solve}(x^2-1=0,x) x\neq 1$	$x=-1$
-----------------------------------	--------

排除功能會以「不等於」(\neq 或 \neq) 關係運算子，在計算期間排除特定值。排除功能的主要用途是在使用 **cSolve()**、**cZeros()**、**fMax()**、**fMin()**、**solve()**、**zeros()** 等等函數時，能夠排除精確解。

→(儲存)

ctrl **var** 鍵

Expr → Var

列表 → Var

矩陣 → Var

Expr → 函數(Param1,...)

列表 → 函數(Param1,...)

矩陣 → 函數(Param1,...)

$\frac{\pi}{4} \rightarrow myvar$	$\frac{\pi}{4}$
-----------------------------------	-----------------

$2\cdot\cos(x) \rightarrow y1(x)$	Done
-----------------------------------	------

$\{1,2,3,4\} \rightarrow lst5$	$\{1,2,3,4\}$
--------------------------------	---------------

$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix} \rightarrow matg$	$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}$
---	--

"Hello" → str1	"Hello"
----------------	---------

→(儲存)

ctrl var 鍵

若 *Var* 變數不存在，則自動建立並賦予初值，以成為 *Expr*、列表或矩陣。

若 *Var* 變數已經存在，而且並未遭到鎖定或保護，則以 *Expr*、列表或矩陣取代其值。

提示：若有意用未定義的變數進行符號計算作業，請避免將任何資料儲存到常用的單字母變數，例如 a、b、c、x、y、z.....等等。

附註：如果要從鍵盤插入本運算子，可輸入 `=:` 當作快速鍵。例如輸入 `pi/4`
`=: myvar`。

:=(賦值)

ctrl := 鍵

Var := *Expr*

Var := 列表

Var := 矩陣

函數(*Param1*,...):= *Expr*

函數(*Param1*,...):= 列表

函數(*Param1*,...):= 矩陣

若 *Var* 變數不存在，則自動建立 *Var* 並賦予初值，以成為 *Expr*、列表或矩陣。

若 *Var* 已經存在，而且並未遭到鎖定或保護，則以 *Expr*、列表或矩陣取代其值。

提示：若有意用未定義的變數進行符號計算作業，請避免將任何資料儲存到常用的單字母變數，例如 a、b、c、x、y、z.....等等。

$myvar := \frac{\pi}{4}$	$\frac{\pi}{4}$
$y1(x) := 2 \cdot \cos(x)$	Done
$lst5 := \{1, 2, 3, 4\}$	$\{1, 2, 3, 4\}$
$matg := \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}$	$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}$
$str1 := "Hello"$	"Hello"

©(註解)

ctrl  鍵

© [文字]

© 符號可將文字視為註釋行處理，讓您為自己建立的函數和程式加上註釋。

© 可放在指令行開頭或任何位置。c 符號右邊到行尾的所有內容都是註解。

輸入範例的注意事項：關於輸入多行程式和函數定義的說明，請參閱產品手冊中的「計算工具」章節。

Define $g(n)=Func$

© Declare variables

Local $i,result$

result:=0

For $i,1,n,1$ ©Loop n times

result:=result+i²

EndFor

Return result

EndFunc

Done

$g(3)$

14

0b、0h

 鍵、 鍵

0b 二進位數字

十進位基底模式：

0b10+0hF+10

27

0h 十六進位數字

二進位基底模式：

0b10+0hF+10

0b11011

分別代表二進位和十六進位數字。若要輸入二進位或十六進位數字，無論使用哪種基底模式，前面一定要輸入 0b 或 0h 字首。如果沒有加上字首，就會把數字視為十進位(以 10 為底)。

所顯示的結果會以基底模式為準。

十六進位基底模式：

0b10+0hF+10

0h1B

空元素

分析現實世界的資料時，不一定每次都能取得完整的資料。TI-Nspire™ CAS 軟體能容許空資料元素，讓您根據幾乎完整的資料進行運算，而不用重頭來過或捨棄不完整的案例。

Lists & Spreadsheet 一章的〈試算表資料繪圖〉一節列出空元素的資料範例。

delVoid() 函數可讓您刪除列表中的空元素。**isVoid()** 函數可讓您檢定空元素。詳情請參考第 頁碼：47 頁的 **delVoid()** 和第 頁碼：88 頁的 **isVoid()**。

附註：若要在數學式中手動輸入空元素，請輸入 `[]` 或 **void** 關鍵字。系統在對運算式求值時，會自動把 **void** 關鍵字轉成 `[]` 符號。若要在計算機上輸入 `[]`，請按 `ctrl` `[]`。

涵蓋空元素的計算作業

計算內容涵蓋空元素時，大部分情況下都會求出空結果。請見以下特殊案例。

<code>[]</code>	-
<code>gcd(100,_)</code>	-
<code>3+_</code>	-
<code>{5,_,10}</code>	<code>{3,6,9}</code>
<code>{5,_,10}</code>	<code>{2,_,1}</code>

列出包含空元素的引數

以下函數和指令會忽略(跳過)列表引數中的空元素。

count、**countIf**、**cumulativeSum**、**freqTable**、**list**、**frequency**、**max**、**mean**、**median**、**product**、**stDevPop**、**stDevSamp**、**sum**、**sumIf**、**varPop**、**varSamp**、迴歸計算式、**OneVar**、**TwoVar**、**FiveNumSummary** 統計、信賴區間、統計檢定

<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> $\begin{pmatrix} 1 & 2 \\ 3 & - \\ 5 & 6 \end{pmatrix}$	$\begin{pmatrix} 1 & 2 \\ 4 & - \\ 9 & 8 \end{pmatrix}$

SortA 與 **SortD** 會把第一個引數中的所有空元素移到末尾。

<code>{5,4,3,_,1}</code>	<code>→ list1</code>	<code>{5,4,3,_,1}</code>
<code>{5,4,3,2,1}</code>	<code>→ list2</code>	<code>{5,4,3,2,1}</code>
<code>SortA list1,list2</code>		<i>Done</i>
<code>list1</code>		<code>{1,3,4,5,_}</code>
<code>list2</code>		<code>{1,3,4,5,2}</code>

列出包含空元素的引數

$\{1,2,3,_,5\} \rightarrow list1$	$\{1,2,3,_,5\}$
$\{1,2,3,4,5\} \rightarrow list2$	$\{1,2,3,4,5\}$
SortD list1,list2	Done
list1	$\{5,3,2,1,_\}$
list2	$\{5,3,2,1,4\}$

以迴歸來說，X、Y 列表中若有空元素，則殘差的對應元素就會是空元素。

ll:= $\{1,2,3,4,5\}$; l2:= $\{2,_,3,5,6,6\}$	$\{2,_,3,5,6,6\}$
LinRegMx ll,l2	Done
stat.Resid	$\{0.434286,_,-0.862857,-0.011429,0.44\}$
stat.XReg	$\{1,_,3,4,5\}$
stat.YReg	$\{2,_,3,5,6,6\}$
stat.FreqReg	$\{1,_,1,1,1,1\}$

若在迴歸中省略類別，則殘差的對應元素就會是空元素。

ll:= $\{1,3,4,5\}$; l2:= $\{2,3,5,6,6\}$	$\{2,3,5,6,6\}$
cat:= $\{ "M", "M", "F", "F" \}$; incl:= $\{ "F" \}$	$\{ "F" \}$
LinRegMx ll,l2,1,cat,incl	Done
stat.Resid	$\{_,_,0,0,0\}$
stat.XReg	$\{_,_,4,5\}$
stat.YReg	$\{_,_,5,6,6\}$
stat.FreqReg	$\{_,_,1,1,1\}$

若迴歸的頻率是 0，則殘差的對應元素就會是空元素。

ll:= $\{1,3,4,5\}$; l2:= $\{2,3,5,6,6\}$	$\{2,3,5,6,6\}$
LinRegMx ll,l2, $\{1,0,1,1\}$	Done
stat.Resid	$\{0.069231,_,-0.276923,0.207692\}$
stat.XReg	$\{1,_,4,5\}$
stat.YReg	$\{2,_,5,6,6\}$
stat.FreqReg	$\{1,_,1,1,1\}$

輸入數學式的快速鍵

快速鍵讓您自行輸入數學式的元素，而不需使用「目錄」或「符號調色盤」。例如若要輸入 $\sqrt{6}$ 運算式，可在輸入線上輸入 **sqrt(6)**。當您按下 **enter** 時，**sqrt(6)** 運算式就會變成 $\sqrt{6}$ 。有的快速鍵可同時用在計算機和電腦鍵盤上。有的快速鍵則主要用在電腦鍵盤上。

計算機和電腦鍵盤快速鍵

若要輸入這個符號：	請輸入這個快速鍵：
π	pi
θ	theta
∞	infinity
\leq	<=
\geq	>=
\neq	/=
\Rightarrow (邏輯隱含)	=>
\Leftrightarrow (邏輯雙隱含, XNOR)	<=>
\rightarrow (儲存運算子)	=:
$ $ (絕對值)	abs (...)
$\sqrt{()}$	sqrt (...)
d()	derivative (...)
$\int()$	integral (...)
$\Sigma()$ (總和範本)	sumSeq (...)
$\Pi()$ (乘積範本)	prodSeq (...)
$\sin^{-1}()$ 、 $\cos^{-1}()$	arcsin (...) 、 arccos (...)
ΔList()	deltaList (...)
ΔtmpCnv()	deltaTmpCnv (...)

電腦鍵盤快速鍵

若要輸入這個符號：	請輸入這個快速鍵：
c1 、 c2 (常數)	@c1 、 @c2
n1 、 n2 (整數常數)	@n1 、 @n2
i (虛常數)	@i
e (以 e 為底的自然對)	@e

若要輸入這個符號：	請輸入這個快速鍵：
數)	
E(科學記號)	@E
↑(轉置)	@t
⋈(弧度)	@r
°(度數)	@d
g(梯度)	@g
∠(角度)	@<
↔(轉換)	@>
↪Decimal、↪approxFraction ().....	@>Decimal、@>approxFraction().....

EOS™(方程式作業系統)階層

i的章節描述TI-Nspire™ CAS 數學與科學學習技術所使用的Operating System (EOS™)。數學、變數和函數以簡單、直接的序列輸入。EOS™軟體使用包含在括號內的群組並根據以下描述的優先順序來計算運算式和方程式。

計算順序

等級	運算子
1	小括弧 ()、中括弧 []、大括弧 { }
2	間接 (#)
3	函數呼叫
4	後置運算子: 度-分-秒 (°、'、")、分數 (!)、百分比 (%)、弧度角 (ʳ)、下標 ([])、轉置 (ᵀ)
5	指數化、乘幕 (^)
6	負值 (-)
7	字串連接 (&)
8	乘法 (•)、除法 (/)
9	加法 (+)、減法 (-)
10	等式關係: 等於 (=)、不等於 (≠ 或 ≠)、小於 (<)、小於或等於 (≤ 或 <=)、大於 (>)、大於或等於 (≥ 或 >=)
11	邏輯 not
12	邏輯 and
13	邏輯 or
14	xor、nor、nand
15	邏輯隱含 (⇒)
16	邏輯雙隱含, XNOR (⇔)
17	約束運算子 ([])
18	儲存 (→)

小括弧、中括弧、大括弧

它將先計算包含在圓括號、括弧或大括號內的所有計算。例如，在 $4(1+2)$ 運算式中，EOS™軟體將先計算圓括號內的運算式 $1+2$ ，然後將結果乘以4。

運算式或方程式中的左右小括弧、中括弧和大括弧數量必須相同，否則會顯示錯誤訊息，指出缺少元素。例如 $(1+2)/(3+4$ 的式子會導致錯誤訊息「缺少」產生。

附註：因為TI-Nspire™ CAS軟體可以讓您定義自己的函數，所以變數名稱和後面的包含於圓括號內的運算式，將作為「函數調用」而不是隱式的乘法。例如， $a(b+c)$ 是 $b+c$ 運算的 a 函數。若要將 $b+c$ 運算式乘以 a ，請使用明顯的乘法： $a*(b+c)$ 。

Indirection

間接運算子 (#) 可將字串轉成變數或函數名稱。例如 #("x"&"y"&"z") 這個式子可建立 xyz 這個變數名稱。間接運算子也可讓您在程式內建立、修改變數。舉例來說，若 $10 \rightarrow r$ 且 " r " $\rightarrow s1$ ，則 $\#s1=10$ 。

後置運算子

後置運算子是引數後面緊接的運算子，例如 $5!$ 、 25% 或 $60^\circ 15' 45''$ 。後置運算子前面的引數，擁有第四級的求值優先順序。例如 $4^3!$ 運算式的 $3!$ 會先進行取值。接著計算結果 6 就會成為 4 的指數，得出 4096 的值。

指數化

指數化 (^) 和各元素指數化 (.^) 的求值順序是從右到左。例如 2^3^2 運算式的求值結果，就和 $2^3(3^2)$ 一樣是 512 。 $(2^3)^2$ 就不同了，結果是 64 。


負值

若要輸入負數，請按 $\boxed{-}$ 然後輸入數字。後置計算和指數化的執行順序都比負值早。例如 $-x^2$ 的結果是負數， $-9^2 = -81$ 。如果用小括弧將負數括起，例如 $(-9)^2$ ，結果就是 81 。

約束 (||)

(||) 運算子後面的引數是一組約束條件，影響運算子前面的引數求值結果。

常數和數值

以下表格列出執行單位換算時可用的常數及其數值。您可手動輸入數值，或從 [公用程式] > [單位換算] 中的常數清單中選擇 (計算機:按  3)。



常數	名稱	值
_c	光速	299792458 _m/_s
_Cc	庫倫常數	8987551787.3682 _m/_F
_Fc	法拉第常數	96485.33289 _coul/_mol
_g	重力加速度	9.80665 _m/_s ²
_Gc	重力常數	6.67408E-11 _m ³ /_kg/_s ²
_h	普朗克常數	6.626070040E-34 _J_s
_k	波茲曼常數	1.38064852E-23 _J/_°K
_μ0	真空磁導率	1.2566370614359E-6 _N/_A ²
_μb	波耳磁元	9.274009994E-24 _J_m ² /_Wb
_Me	電子靜質量	9.10938356E-31 _kg
_Mμ	介子質量	1.883531594E-28 _kg
_Mn	中子靜質量	1.674927471E-27 _kg
_Mp	質子靜質量	1.672621898E-27 _kg
_Na	亞佛加厥常數	6.022140857E23 /_mol
_q	電子電荷	1.6021766208E-19 _coul
_Rb	波耳半徑	5.2917721067E-11 _m
_Rc	莫耳氣體常數	8.3144598 _J/_mol/_°K
_Rdb	芮得柏常數	10973731.568508/_m
_Re	電子半徑	2.8179403227E-15 _m
_u	原子質量	1.660539040E-27 _kg
_Vm	莫耳體積	2.2413962E-2 _m ³ /_mol
_ε0	真空介電係數	8.8541878176204E-12 _F/_m
_σ	史特凡波茲曼常數	5.670367E-8 _W/_m ² /_°K ⁴
_φ0	磁通量量子	2.067833831E-15 _Wb

錯誤代碼和訊息

出現錯誤時，errCode 變數會有指定代碼。使用者自行定義的程式和函數可檢查 errCode，以判斷錯誤原因。若需 errCode 的使用參考範例，請參考第 頁碼：176 頁 Try 指令下的例 2。

附註：有的錯誤狀況只適用於 TI-Nspire™ CAS 產品，有的只適用於 TI-Nspire™ 產品。

錯誤代碼	說明
10	函數並未傳回值
20	檢定並未解出 TRUE 或 FALSE。 系統通常無法比較未定義的變數。例如若執行 If 語句時，a 或 b 其中之一屬於未定義，則 If a<b 檢定就會導致這項錯誤。
30	引數不得是資料夾名稱。
40	引數錯誤
50	引數不符 兩個以上的引數必須屬於相同類型。
60	引數必須是布林運算式或整數
70	引數必須是十進位數
90	引數必須是列表
100	引數必須是矩陣
130	引數必須是字串
140	引數必須是變數名稱。 名稱的條件： <ul style="list-style-type: none">• 開頭不得是數字• 不可包含空格或特殊字元• 不得以無效方式使用底線或句號• 不可超過長度限制 詳情請參考說明文件中的 <Calculator> 一節。
160	引數必須是運算式
165	電池電力不足，無法收發資料 請裝入新電池再收發資料。
170	界限 下限必須小於上限，才能界定搜尋區間。

錯誤代碼	說明
180	中斷 在冗長計算或執行程式期間按  或  鍵。
190	循環定義 在化簡期間無盡期取代變數值時，為避免用完記憶體而顯示此訊息。例如在 $a+1 \rightarrow a$ 中， a 是未定義的變數，因此會導致本錯誤。
200	約束運算式無效 例如 $\text{solve}(3x^2-4=0,x) x < 0 \text{ or } x > 5$ 就會產生這個錯誤訊息，因為約束條件是以 <code>br</code> 分隔，而非 <code>and</code> 。
210	資料類型無效 引數的資料類型錯誤。
220	相依限制
230	維度 列表或矩陣指數無效。例如若將 <code>list {1,2,3,4}</code> 儲存在 <code>L1</code> ，則 <code>L1[5]</code> 會導致維度錯誤，因為 <code>L1</code> 只包含四個元素。
235	維度錯誤。列表中的元素不足。
240	維度不符 兩個以上引數的維數必須相同。例如 <code>[1,2]+[1,2,3]</code> 的維數就不符，因為兩個矩陣的元素數目不同。
250	分母是零
260	值域錯誤 引數必須在指定值域。例如 <code>rand(0)</code> 就無效。
270	變數名稱重複
280	<code>Else</code> 和 <code>Elseif</code> 在 <code>If...EndIf</code> 段以外時無效
290	<code>EndTry</code> 缺少相對的 <code>Else</code> 語句
295	迭代過多
300	預期的列表或矩陣是 2 元或 3 元
310	<code>nSolve</code> 的第一個引數必須是只有一個變數的方程式。其中不能包含需注意變數以外的未取值變數。
320	<code>solve</code> 或 <code>cSolve</code> 的第一個引數必須是方程式或不等式 例如 <code>solve(3x^2-4,x)</code> 無效，因為第一個引數不是方程式。
345	單位不一致

錯誤代碼	說明
350	指數在範圍外
360	間接字串不是有效的變數名稱
380	未定義 Ans 如果不是前一項計算作業沒有產生 Ans，就是根本沒有輸入前一項計算作業。
390	指定無效
400	指定值無效
410	指令無效
430	無法用於目前的模式設定
435	猜測值無效
440	隱式乘法無效 例如 $x(x+1)$ 無效，但 $x*(x+1)$ 是正確語法。這是為了避免隱式乘法和函數呼叫有所混淆。
450	無法用於函數或目前的運算式 只有特定指令可用於使用者自行定義的函數。
490	無法用於 Try..EndTry 段
510	列表或矩陣無效
550	不可在函數或程式以外 幾個指令不可用在函數或程式以外。例如不可使用 Local，除非在函數或程式中。
560	不可在 Loop..EndLoop、For..EndFor、While..EndWhile 段以外 例如 Exit 指令只有在這些迴圈段落中才有效。
565	不可用在程式以外
570	路徑名稱無效 例如 \var 就無效。
575	複極無效
580	程式參照無效 不得在 $1+p(x)$ 這類函數或運算式中參照程式。p 代表程式。
600	表格無效
605	單位用法無效

錯誤代碼	說明
610	Local 語句中的變數名稱無效
620	變數或函數名稱無效
630	變數參照無效
640	向量語法無效
650	連結傳輸 兩台計算機無法傳輸資料。請檢查兩端的連接線是否已經接牢。
665	矩陣無法對角化
670	記憶體不足 1.刪除本文件的某些資料 2.儲存本文件然後關閉 如果 1、2 失敗，請取出電池然後重新裝入
672	資源耗盡
673	資源耗盡
680	缺少 (
690	缺少)
700	缺少“
710	缺少]
720	缺少 }
730	缺少段落語法的開頭或結尾
740	缺少 if..Endif 段中的 Then
750	名稱不是函數或程式
765	尚未選取函數
780	找不到解
800	非實數結果 例如若軟體使用「實數」設定， $\sqrt{-1}$ 就無效。 如果要進行複數計算，請把「實數或複數」模式設定改成 RECTANGULAR 或 POLAR。
830	溢位
850	找不到程式

錯誤代碼	說明
	執行期間無法在所提供的路徑中，找到其他程式內部的程式參照項目。
855	繪圖時不可使用 Rand 類型函數
860	遞迴太深
870	保留名稱或系統變數
900	引數錯誤 無法將中位數-中位數模型應用到資料集。
910	語法錯誤
920	找不到文字
930	引數太少 函數或指令缺少一個以上的引數。
940	引數太多 運算式或方程式包含過多引數，無法求值。
950	下標太多
955	未定義的變數太多
960	尚未定義變數 尚未指定變數值。請使用以下其中一個指令： <ul style="list-style-type: none"> • <code>sto →</code> • <code>:=</code> • Define 以便指定變數值。
965	作業系統無使用授權
970	正在使用該變數，因此無法參照或更改
980	變數已經受到保護
990	變數名稱無效 名稱絕不可超過長度限制
1000	Window 變數值域
1010	縮放
1020	內部錯誤
1030	受保護的記憶體違規

錯誤代碼	說明
1040	不支援此函數。本函數需要「電腦代數系統」。請嘗試 TI-Nspire™ CAS。
1045	不支援此運算子。本運算子需要「電腦代數系統」。請嘗試 TI-Nspire™ CAS。
1050	不支援此功能。本運算子需要「電腦代數系統」。請嘗試 TI-Nspire™ CAS。
1060	輸入引數必須是數字。只允許包含數值的輸入。
1070	三角函數引數太大，無法精確約化
1080	不支援使用 Ans。此應用程式不支援 Ans。
1090	尚未定義函數。請使用以下其中一個指令： <ul style="list-style-type: none"> • Define • := • sto →
1100	非實數計算 例如若軟體使用「實數」設定， $\sqrt{-1}$ 就無效。 如果要進行複數計算，請把「實數或複數」模式設定改成 RECTANGULAR 或 POLAR。
1110	邊界無效
1120	不得更改正負號
1130	引數不得是列表或矩陣
1140	引數錯誤 第一個引數必須是第二個引數中的多項式。如果省略第二個引數，軟體會試圖選取預設值。
1150	引數錯誤 前二個引數必須是第三個引數中的多項式。如果省略第三個引數，軟體會試圖選取預設值。
1160	資料庫路徑名稱無效 路徑名稱的格式必須是 xxx\yyy: <ul style="list-style-type: none"> • xxx 部份可包含 1 至 16 個字元。 • yyy 部份可包含 1 至 15 個字元。 詳情請參考說明文件中的〈資料庫〉一節。
1170	資料庫路徑名稱用法無效 <ul style="list-style-type: none"> • 不可使用 Define、:= 或 sto → 指定路徑名稱的值。 • 不可將路徑名稱宣告為區域性變數，或將該名稱用在函數或程式定義中當作參數。
1180	資料庫變數名稱無效。

錯誤代碼	說明
	名稱的條件： <ul style="list-style-type: none"> • 不包含句點 • 開頭不得是底線 • 長度不可超過 15 個字元 詳情請參考說明文件中的〈資料庫〉一節。
1190	找不到資料庫文件： <ul style="list-style-type: none"> • 確認資料庫是否在 MyLib 資料夾中。 • 重新整理資料庫。 詳情請參考說明文件中的〈資料庫〉一節。
1200	找不到資料庫變數： <ul style="list-style-type: none"> • 確認資料庫變數是否在資料庫的第一個問題中。 • 請務必將資料庫變數界定為 LibPub 或 LibPriv。 • 重新整理資料庫。 詳情請參考說明文件中的〈資料庫〉一節。
1210	資料庫捷徑名稱無效。 名稱的條件： <ul style="list-style-type: none"> • 不包含句點 • 開頭不得是底線 • 長度不可超過 16 個字元 • 不得是保留名稱 詳情請參考說明文件中的〈資料庫〉一節。
1220	值域錯誤： tangentLine 與 normalLine 函數只支援實值函數。
1230	值域錯誤。 度數角或梯度角模式不支援三角轉換運算子。
1250	引數錯誤 請使用線性方程式。 變數是 x 和 y 的兩個線性聯立方程式舉例如下： $3x+7y=5$ $2y-5x=-1$
1260	引數錯誤： nfMin 或 nfMax 的第一個引數必須是只有一個變數的運算式。其中不能包含需注意變數以外的未取值變數。

錯誤代碼	說明
1270	引數錯誤 導數階數必須等於 1 或 2。
1280	引數錯誤 請使用只有一個變數的展開多項式。
1290	引數錯誤 請使用只有一個變數的多項式。
1300	引數錯誤 多項式的係數在求值後必須是數值。
1310	引數錯誤： 無法對函數的一個以上引數求值。
1380	引數錯誤： 不允許巢狀調用 <code>domain()</code> 函數。

警告代碼和訊息

可使用 `warnCodes()` 函數以儲存由計算運算式所產生的警告代碼。下表列出每一個數值警告代碼及其相關訊息。

如需儲存警告代碼的範例，請參考第 頁碼：184 頁的 `warnCodes()`。

警告代碼	訊息
10000	作業可能產生錯誤的解。
10001	對方程式作微分可能產生錯誤的方程式。
10002	有問題的解
10003	有問題的準確性
10004	作業可能遺失解。
10005	<code>cSolve</code> 可能指定更多的零。
10006	<code>Solve</code> 可能指定更多的零。
10007	可能存在更多的解。
10007	可能存在其他解決方案。請指定適當的下限或上限及/或猜測值。 <code>solve()</code> 的使用範例： <ul style="list-style-type: none">• <code>solve(方程式, 變數=估計值) 下限 < 變數 < 上限</code>• <code>solve(方程式, 變數) 下限 < } 變數 < 上限</code>• <code>solve(方程式, 變數=估計值)</code>
10008	結果的值域可能比輸入的值域小。
10009	結果的值域可能比輸入的值域大。
10012	非實數計算
10013	1 取代 ∞^0 或 <code>undef^0</code>
10014	1 取代 <code>undef^0</code>
10015	1 取代 1^∞ 或 <code>1^undef</code>
10016	1 取代 <code>1^undef</code>
10017	∞ 或 ∞ 取代 溢位
10018	作業需要並傳回 64 位元的值
10019	記憶體用盡，化簡可能不完整。
10020	三角函數引數太大，無法精確約化。
10021	輸入包含未定義的參數。 結果不一定對所有可能的參數值有效。

警告代碼	訊息
10022	指定適當的下限和上限會產生解。
10023	單位矩陣已乘以純量。
10024	使用適當的近似值運算法取得結果。
10025	無法在 [精確] 模式中驗證等值。
10026	可能忽略約束。由形式 "\'變數 MathTestSymbol 約束' 或結合這些形式以指定約束，例如 'x<3 and x>-12'

Texas Instruments 支持与服务

一般信息

有关 TI 产品和服务的更多信息，请通过电子邮件与 TI 联系或访问 TI 网址。

电子邮件咨询:	ti-cares@ti.com
主页:	education.ti.com

维修和保修信息

关于保修期限和条款，及产品维修的信息，请参阅本产品附带的保修声明，或者联系当地的 Texas Instruments 零售商/分销商。

索引

-		^	
-次方根		\wedge^{-1} , 倒數	215
範本	1	\wedge , 乘幂	197
-, 減[*]	195	-	
!		┌, 單位指示符號	213
!, 階乘	204		
"		, 約束運算子	215
", 秒符號	212	+	
#		+, 加	194
#, 間接	210	/	
#, 間接運算子	224	/, 除[*]	196
%		≠	
%, 百分比	200	≠, 不等於[*]	201
&		=	
&, 接上	204	=, 等於	200
*		>	
*, 乘	195	>, 大於	202
,		Π	
, 上標	213	Π, 乘積[*]	207
, 分符號	212	Σ	
.		Σ(), 總和[*]	208
., 點差	198	ΣInt()	208
*, 點乘	198	ΣPrn()	209
./, 點除	199	√	
^, 點幂	199	√, 平方根[*]	207
+, 點和	198	∫	
:		∫, 積分[*]	205
:=, 賦值	217		

\leq , 大於或等於	203	$^{\circ}$, 度/分/秒[*]	212
\leq , 小於或等於	202	$^{\circ}$, 度數符號[*]	211
▶		0	
▶, 換算單位[*]	214	0b, 二進位指示	218
▶, 轉成梯度角 [Grad]	81	0h, 十六進位指示	218
▶approxFraction()	13	1	
▶Base10, 以十進位整數顯示 [Base10]	19	10的次方, 10 ⁿ ()	214
▶Base16, 以十六進位顯示 [Base16]	19	10 ⁿ (), 10的次方	214
▶Base2, 以二進位顯示 [Base2] ...	17	A	
▶cos, 以餘弦 [cos] 表示	29	abs(), 絕對值	8
▶Cylind, 以圓柱座標形式顯示向 量 [Cylind]	41	amortTb(), 攤銷表	8, 17
▶DD, 以十進位角顯示 [DD]	44	and, 布林運算子	9
▶Decimal, 以十進位顯示結果 [Decimal]	44	angle(), 角度	9
▶DMS, 以度/分/秒顯示 [DMS] ...	52	ANOVA, 單因子變異數分析	10
▶exp, 以 e[exp] 顯示	60	ANOVA2way, 雙因子變異數分析	11
▶Polar, 以極座標形式顯示向量 [Polar]	122	Ans, 最近結果	13
▶Rad, 轉換成弧度角	132	answer (last), Ans	13
▶Rect, 顯示為直角向量	135	approx(), 近似值	13-14
▶sin, 以正弦 [sin] 表示	154	approxRational()	14
▶Sphere, 以球面座標形式顯示向 量 [Sphere]	161	arccos()	14
→		arccosh()	14
→, 儲存	216	arccot()	14
⇒		arccoth()	14
⇒, 邏輯隱含[*]	203	arccsc()	14
⇒, 邏輯隱含[*]	221	arccsch()	14
⇔		arcLen(), 弧長	15
⇔, 邏輯雙隱含[*]	204	arcsec()	15
©		arcsech()	15
©, 註解	218	arcsin()	15
		arcsinh()	15
		arctan()	15
		arctanh()	15
		augment(), 擴展/串連	15
		avgRC(), 平均變化率	16
		B	
		binomCdf()	20, 86

binomPdf()	20	Cycle, 迴圈	40
C		cZeros(), 複數零點	41
Cdf()	65	D	
ceiling(), 無條件進入法	20	d(), 一階導數	205
centralDiff()	21	dbd(), 日期之間的天數	43
cFactor(), 複因式	22	Define	45
char(), 字元字串	22	Define LibPriv	45
charPoly()	23	Define LibPub	46
χ^2 2way	23	Define, 定義	45
χ^2 Cdf()	23	deltaList()	46
χ^2 GOF	24	deltaTmpCnv()	47
χ^2 Pdf()	24	DelVar, 刪除變數	47
ClearAZ	25	delVoid(), 移除空元素	47
ClrErr, 清除錯誤	25	derivative()	47
colAugment	25	deSolve(), 解	48
colDim(), 矩陣行維數	25	det(), 矩陣行列式	49
colNorm(), 矩陣行範數	26	diag(), 矩陣對角線	50
comDenom(), 公分母	26	dim(), 維數	50
completeSquare(), 完全平方	27	Disp, 顯示資料	50
conj(), 共軛複數	27	DispAt	51
constructMat(), 建立矩陣	28	domain(), 域函數	53
corrMat(), 相關矩陣	28	dominantTerm(), 主要項	54
\cos^{-1} , 反餘弦	30	dotP(), 點積	55
cos(), 餘弦	29	E	
cosh ⁻¹ (), 反雙曲餘弦	31	e 指數	2
cosh(), 雙曲餘弦	31	範本	2
cot ⁻¹ (), 反餘切	32	E, 科學記號	210
cot(), 餘切	32	e, 運算式表現方式	60
coth ⁻¹ (), 反雙曲餘切	33	e [^] (), 以 e 為底乘冪	55
coth(), 雙曲餘切	33	eff), 名目利率轉成實際年利率	55
count(), 計算列表中的項目數	33	eigVc(), 特徵向量	56
countif(), 有條件計算列表中的 項目數	34	eigVl(), 特徵值	56
cPolyRoots()	35	else if, ElseIf	57
crossP(), 外積	35	Elseif, else if	57
csc ⁻¹ (), 反餘割	36	end	
csc(), 餘割	35	for, EndFor	68
csch ⁻¹ (), 反雙曲餘割	36	while, EndWhile	185
csch(), 雙曲餘割	36	end while, EndWhile	185
cSolve(), 複數解	37	EndTry, 結束嘗試	176
CubicReg, 三次迴歸	39	EndWhile, end while	185
cumulativeSum(), 累積總和	40	EOS (方程式作業系統)	223

euler(), 尤拉函數	58	GetStr	79
exact(), 精確	60	getType(), 取得變數類型	79
Exit, 結束	60	Goto, 轉到	81
exp(), 以 e 為底乘幕	61		
exp▶list(), 運算式至列表	61	I	
expand(), 展開	62	identity(), 單位矩陣	81
expr(), 字串至運算式	63, 99	If, 如果	81
ExpReg, 指數迴歸	63	ifFn()	82
		imag(), 虛部	83
F		ImpDif(), 隱導函數	84
factor(), 因式	64	Input, 輸入	84
Fill, 矩陣填充	66	inString(), 在字串內	84
FiveNumSummary	66	int(), 整數	84
floor(), 無條件捨去法	67	intDiv(), 整數除法	85
fMax(), 函數上限	67	interpolate(), 插入	85
fMin(), 函數下限	67	invF()	86
For	68	invNorm(), 逆累積常態分佈) ...	87
for, For	68	invt()	87
For, for	68	lnx ² ()	85
format(), 格式字串	68	iPart(), 整數部份	87
fpart(), 函數部分	69	irr(), 內部投資報酬率	
freqTable()	70	內部投資報酬率, irr()	87
frequency()	70	isPrime(), 質數檢定	88
Frobenius 範數, norm()	115	isVoid(), 空值檢定	88
Func, 函數	71		
Func, 程式函數	71	L	
		Lbl, 標籤	88
G		lcm, 最小公倍數	89
g, 梯度	210	left(), 左邊	89
gcd(), 最大公因數	72	LibPriv	45
geomCdf()	72	LibPub	46
geomPdf()	73	libShortcut(), 建立資料庫群組快	
Get	73	速鍵	89
getDenom(), 取得/傳回分母	74	limit() or lim(), 極限	90
getKey()	74	LinRegBx, 線性迴歸	90
getLangInfo(), 取得/傳回語言資		LinRegMx, 線性迴歸	91
訊	77	LinRegtIntervals, 線性迴歸	92
getLangInfo(), 取得/傳回變數資		LinRegtTest	94
訊	80	linSolve()	95
getLockInfo(), 檢驗變數或變數群		Δlist(), 列表差異	95
組的鎖定狀態	78	list▶mat(), 列表至矩陣	96
getMode(), 取得模式設定	78	ln(), 自然對數	96
getNum(), 取得/傳回數字	79	LnReg, 對數迴歸	97

Local, 區域性變數	98	nSolve(), 數值解	117
Lock, 鎖定變數或變數群組	98	O	
Logistic, 羅吉斯迴歸	99	OneVar, 單變數統計	118
LogisticD, 羅吉斯迴歸	100	or(布林), 或	119
Loop, 迴圈	102	or, 布林運算子	119
LU, 矩陣上下分解值	102	ord(), 數字代號	120
M			
mat►list(), 矩陣至列表	103	P	
max(), 最大值	103	P►Rx(), 成對值的等值 x 座標 ...	120
mean(), 平均值	103	P►Ry(), 成對值的等值 y 座標 ...	121
median(), 中位數	104	PassErr, 傳遞錯誤	121
MedMed, 中位數-中位數線性迴歸	104	Pdf()	69
mid(), 字串中	105	piecewise()	122
min(), 最小值	106	poissCdf()	122
mirr(), 修正後內部投資報酬率 ..	106	poissPdf()	122
mod(), 模數	107	polyCoeff()	123
mRow(), 矩陣行運算	107	polyDegree()	124
mRowAdd(), 矩陣行乘法與加法 ..	108	polyEval(), 多項式求值	124
MultReg	108	polyGcd()	124-125
MultRegIntervals()	108	PolyRoots()	125
MultRegTests()	109	PowerReg, 乘幕迴歸	126
N			
nand, 布林運算子	110	Prgm, 定義程式	127
nCr(), 組合	111	prodSeq()	127
nDerivative(), 數值導數	112	product(), 乘積	127
newList(), 新列表	112	propFrac, 真分數	128
newMat(), 新矩陣	112	Q	
nfMax(), 數值函數上限	112	QR 分解值, QR	128
nfMin(), 數值函數下限	113	QR, QR 分解值	128
nInt(), 數值積分	113	QuadReg, 二次迴歸	129
nom), 實際年利率轉成名目利率 ..	113	QuartReg, 四次迴歸	130
nor, 布林運算子	114	R	
norm(), Frobenius 範數	115	R, 弧度	211
normalLine()	115	R►Pr(), 極座標	132
normCdf()	115	R►Pθ(), 極座標	131
normPdf()	115	rand(), 亂數	132
not, 布林運算子	116	randBin, 亂數	133
nPr(), 排列	116	randInt(), 隨機整數	133
npv(), 淨現值	117	randMat(), 隨機矩陣	133
		randNorm(), 隨機範數	133

randPoly(), 隨機多項式	134	stat.results	162
randSamp()	134	stat.values	163
RandSeed, 亂數種子	134	stdDevPop(), 母群體標準差	164
real(), 實部	134	stdDevSamp(), 樣本標準差	164
ref(), 列梯形	136	string(), 運算式至字串	165
RefreshProbeVars	136	student-t 分布機率, tCdf()	171
remain(), 餘	137	student-t 機率密度, tPdf()	175
RequestStr	139	subMat(), 子矩陣	166-167
Return, 傳回	140	sum(), 求和	166
right(), 從右取字串	140	sumIf()	166
rk23(), Runge Kutta 函數	140	sumSeq()	167
rotate(), 互換	142		
round(), 四捨五入法	143	T	
rowAdd(), 矩陣列加法	143	t 檢定, tTest	177
rowDim(), 矩陣的列數	144	T, 轉置	168
rowNorm(), 矩陣的列範數	144	tan ⁻¹ (), 反正切	169
rowSwap(), 矩陣列交換	144	tan(), 正切	168
rref(), 列簡化梯形	144	tangentLine()	169
		tanh ⁻¹ (), 反雙曲正切	170
S		tanh(), 雙曲正切	170
sec ⁻¹ (), 反正割	145	taylor(), 泰勒多項式	171
sec(), 正割	145	tCdf(), studentt 分布機率	171
sech ⁻¹ (), 反雙曲正割	146	tCollect(), 三角集合	171
sech(), 雙曲正割	146	Test_2S, 二樣本 F 檢定	71
seq(), 序列	147	tExpand(), 三角擴張	172
seqGen()	147	tInterval, t 信賴區間	173
seqn()	148	tInterval_2Samp, 二 t 信賴區間	173
series(), 系列	149	ΔtmpCnv() [tmpCnv]	175
setMode(), 設定模式	150	tmpCnv()	174-175
shift(), 移位	151	tPdf(), studentt 機率密度	175
sign(), 符號	152	trace()	175
simult(), 聯立方程式	153	Try, 錯誤處理指令	176
sin ⁻¹ (), 反正弦	155	tTest, t 檢定	177
sin(), 正弦	154	tTest_2Samp, 二樣本 t 檢定	177
sine		tvmFV()	178
運算式表現方式	154	tvmI()	178
sinh ⁻¹ (), 反雙曲正弦	156	tvmN()	179
sinh(), 雙曲正弦	155	tvmPmt()	179
SinReg, 正弦迴歸	156	tvmPV()	179
solve(), 求解	157	TwoVar, 二變數結果	180
SortA, 升冪排列	160		
SortD, 降冪排列	161	U	
sqrt(), 平方根	162	unitV(), 單位向量	182

unlock, 解開鎖定變數或變數群組	182	二進位 指示, 0b	218
V		顯示, ▶Base2	17
varPop()	182	二階導數 範本	6
varSamp(), 樣本變異數	183	二樣本 F 檢定	71
W		二變數結果, TwoVar	180
Wait 指令	183	十	
warnCodes(), 警告代碼	184	十六進位 指示, 0h	218
when(), when	184	顯示, ▶Base16	19
when, when()	184	十進位 角度顯示, ▶DD	44
while, While	185	整數顯示, ▶Base10	19
While, while	185	三	
X		三次迴歸, CubicReg	39
x ² , 平方	198	三角集合, tCollect()	171
XNOR	204	三角擴張, tExpand()	172
xor, 布林互斥的「或」	185	上	
Z		上標,	213
zeroes(), 零點	186	大	
zInterval, z 信賴區間	188	大於, >	202
zInterval_1Prop, 單一母群體比例 z 信賴區間	189	大於或等於, 	203
zInterval_2Prop, 二母群體比例 z 信賴區間	189	子	
zInterval_2Samp, 二樣本 z 信賴區 間	190	子矩陣, subMat()	166-167
zTest	190	小	
zTest_1Prop, 單一母群體比例 z 檢定	191	小於,	201
zTest_2Prop, 二母群體比例 z 檢 定	191	小於或等於, {	202
zTest_2Samp, 二樣本 z 檢定	192	不	
一		不定積分 範本	6
一階導數 範本	5	不等於, ≠	201
二			
二次迴歸, QuadReg	129		

中

中位數-中位數線性迴歸, MedMed	104
中位數, median()	104

互

互換, rotate()	142
--------------------	-----

公

公分母, comDenom()	26
-----------------------	----

分

分布函數	
normCdf()	115
normPdf()	115
poissCdf()	122
poissPdf()	122
tCdf()	171
tPdf()	175
χ^2 2way()	23
χ^2 Cdf()	23
χ^2 GOF()	24
χ^2 Pdf()	24
分母	26
分佈函數	
binomCdf()	20, 86
binomPdf()	20
invNorm()	87
invt()	87
Inv χ^2 ()	85
分段函數(2段)	
範本	2
分段函數(N段)	
範本	3
分符號,	212
分數	
propFrac	128
範本	1

切

切線, tangentLine()	169
-------------------------	-----

反

反正切, $\tan^{-1}()$	169
反正弦, $\sin^{-1}()$	155
反向, \wedge^{-1}	215
反餘弦, $\cos^{-1}()$	30
反雙曲	
正切, $\tanh^{-1}()$	170
正弦, $\sinh^{-1}()$	156
餘弦, $\cosh^{-1}()$	31

文

文字命令	172
------------	-----

方

方陣	
QR 分解值, QR	128
上下分解值, LU	102
子矩陣, subMat()	166-167
列表至矩陣, list►mat()	96
行列式, det()	49
行乘法與加法, mRowAdd()	108
行運算, mRow()	107
行維數, colDim()	25
行範數, colNorm()	26
求和, sum()	166
乘積, product()	127
特徵向量, eigVc()	56
特徵值, eigVl()	56
矩陣至列表, mat►list()	103
累積總和, cumulativeSum()	40
最大值, max()	103
最小值, min()	106
填充, Fill	66
新, newMat()	112
對角線, diag()	50
維數, dim()	50
點和, .+	198
點乘, .*	198
點差 N	198
點除, ./	199
點冪, .^	199
擴展/串連, augment()	15

轉置, T	168
方程式作業系統 (EOS)	223
日	
日期之間的天數, dbd()	43
主	
主要項, dominantTerm()	54
以	
以 e 為底乘冪, e^()	55, 61
加	
加, +	194
右	
右邊, right()	27, 58, 184
四	
四次迴歸, QuartReg	130
四捨五入法, round()	143
外	
外積, crossP()	35
左	
左邊, left()	89
布	
布林運算子	
⇒	203
⇐	204
and	9
nand	110
nor	114
⊖	221
互斥	185
或	119
非	116

平	
平方根	
範本	1
平方根, √()	162, 207
平均值, mean()	103
平均變化率, avgRC()	16
本	
本金總和	209
正	
正切, tan()	168
正弦, sin()	154
正弦迴歸, SinReg	156
用	
用 [] 運算子代換	215
用 [] 運算子排除	215
列	
列表	
升冪排列, SortA	160
外積, crossP()	35
列表中的差異, Δlist()	95
列表至矩陣, list▶mat()	96
字串中, mid()	105
求和, sum()	166
空元素, 在	219
降冪排列, SortD	161
乘積, product()	127
差異, Δlist()	95
矩陣至列表, mat▶list()	103
累積總和, cumulativeSum() ..	40
最大值, max()	103
最小值, min()	106
新, newList()	112
運算式至列表, exp▶list()	61
點積, dotP()	55
擴展/串連, augment()	15
列表, 有條件項目計數	34
列表, 項目計數	33

序		自訂的函數或程式	45
序列, seq()	147-148	定義, Define	45
求		定積分	
求和, sum()	166	範本	6
求值, 順序	223		
求解, solve()	157	底	
系		底線, _	213
系列, series()	149	弧	
角		弧長, arcLen()	15
角度, angle()	9	弧度, R	211
使		法	
使用者定義的函數	45	法線, normalLine()	115
使用者定義的函數和程式	45-46	物	
函		物件	
函數		建立資料庫快速鍵	89
上限, fMax()	67	直	
下限, fMin()	67	直角向量顯示, ▶Rect	135
使用者定義的	45	空	
部分, fpart()	69	空元素	219
程式函數, Func	71	空元素, 移除	47
函數和變數		空值, 檢定	88
複製	28	空值檢定, isVoid()	88
取		近	
取得/傳回		近似值, approx()	13-14
分母, getDenom()	74	度	
數字, getNum()	79	度/分/秒符號	212
變數資訊, getVarInfo()	77, 80	度/分/秒顯示, ▶DMS	52
和		度數符號, -	211
和, 	215	建	
定		建立矩陣, constructMat()	28
定義			
公用的函數或程式	46		

指	倒
指數	倒數, \wedge^{-1} 215
範本 1	展
指數迴歸, ExpReg 63	展開, expand() 62
相	格
相關矩陣, corrMat() 28	格式字串, format() 68
科	泰
科學記號, E 210	泰勒多項式, taylor() 171
秒	特
秒符號, " 212	特徵向量, eigVc() 56
約	特徵值, eigVl() 56
約束運算子 [] 215	真
約束運算子, 計算順序 223	真分數, propFrac 128
要	矩
要求 138	矩陣
計	列加法, rowAdd() 143
計算日期之間的天數, dbd() 43	列交換, rowSwap() 144
計算列表中的項目數, count() .. 33	列梯形, ref() 136
負	列數, rowDim() 144
負值, 輸入負數 224	列範數, rowNorm() 144
乘	列簡化梯形, rref() 144
乘, * 195	單位, identity() 81
乘幕, \wedge 197	隨機, randMat() 133
乘幕迴歸, PowerReg 125-126, 172	矩陣 (1×2)
乘幕迴歸, PowerReg 138-139	範本 4
乘積 (Π)	矩陣 (2×1)
範本 5	範本 4
乘積, product() 127	矩陣 (2×2)
乘積, Π() 207	範本 4
修	矩陣 (m×n)
修正後內部投資報酬率, mirr() . 106	範本 4
	矩陣至列表, mat►list() 103
修	財
	財務函數, tvmfV() 178

財務函數, tvml()	178
財務函數, tvnN()	179
財務函數, tvnPmt()	179
財務函數, tvmPV()	179

迴

迴圈, Cycle	40
迴圈, Loop	102
迴歸	
MultReg	108
二次, QuadReg	129
三次, CubicReg	39
中位數-中位數線性, MedMed	104
四次, QuartReg	130
正弦, SinReg	156
指數, ExpReg	63
乘幂迴歸, PowerReg	125-126, 172
乘幂迴歸, PowerReg	138-139
對數, LnReg	97
線性迴歸, LinRegAx	91
線性迴歸, LinRegBx	90, 92
羅吉斯	99
羅吉斯, Logistic	100

逆

逆累積常態分佈 (invNorm())	87
---------------------	----

除

除, /	196
------	-----

停

停止指令	165
------	-----

區

區域性, Local	98
區域性變數, Local	98

域

域函數, domain()	53
---------------	----

帶

帶分數, 使用 propFrac(> with	128
-------------------------	-----

常

常態分布機率, normCdf()	115
常數	
以 cSolve()	38
以 cZeros()	43
以 deSolve()	48
以 solve()	158-159
以 zeros()	187
快速鍵	221

從

從右取字串, right()	85, 140
----------------	---------

排

排列	
升幂, SortA	160
降幂, SortD	161
排列, nPr()	116

接

接上, &	204
-------	-----

梯

梯度, g	210
-------	-----

淨

淨現值, npv()	117
------------	-----

清

清除	
錯誤, ClrErr	25

球

球面座標形式向量顯示 ▶Sphere	161
--------------------	-----

移

移位, shift()	151
-------------	-----

函數, EndFunc	71
迴圈, EndLoop	102
嘗試, EndTry	176
結束, Exit	60
結束函數, EndFunc	71
結束迴圈, EndLoop	102
結果	
以e表現	60
以正弦表示	154
以餘弦表示	29
結果, 統計	162
結果值, 統計	163

絕

絕對值	
範本	3-4

虛

虛部, imag()	83
------------------	----

註

註解, ©	218
-------------	-----

間

間接, #	210
間接運算子 (#)	224

階

階乘, !	204
-------------	-----

亂

亂	
數種子, RandSeed	134

傳

傳回, Return	140
傳遞錯誤, PassErr	121

圓

圓柱座標形式向量顯示, ►Cylind	41
---------------------	----

新

新

列表, newList()	112
矩陣, newMat()	112

極

極

座標, R►Pr()	132
座標, R►Pθ()	131

極限

lim()	90
limit()	90
範本	6

極座標形式

顯示向量, ►Polar	122
--------------------	-----

群

群組, 檢驗鎖定狀態	78
群組, 鎖定和解除鎖定	98, 182

解

解, deSolve()	48
解開鎖定變數和變數群組	182

資

資料庫	
建立物件快速鍵	89

運

運算子	
求值順序	223
運算式	
字串至運算式, expr()	63, 99
運算式至列表 exp►list()	61

零

零點, zeroes()	186
--------------------	-----

實

實部, real()	134
實際年利率, eff()	55

對		分數	1
對數	96	平方根	1
範本	2	定積分	6
對數迴歸, LnReg	97	指數	1
精		乘積 (P)	5
精確, exact()	60	矩陣 (1×2)	4
維		矩陣 (2×1)	4
維數, dim()	50	矩陣 (2×2)	4
語		矩陣 (m×n)	4
語言		絕對值	3-4
取得語言資訊	77	極限	6
數		對數	2
數值		導數或 N 階導數	6
解, nSolve()	117	總和 (Σ)	5
導數, nDeriv()	112-113	聯立方程式 (2 個方程式) ..	3
導數, nDerivative()	112	聯立方程式 (N 個方程式) ..	3
積分, nInt()	113	線	
標		線性迴歸, LinRegAx	91
標準差, stdDev()	164, 182	線性迴歸, LinRegBx	90, 92
標籤, Lbl	88	編	
模		編程	
模式		定義程式, Prgm	127
設定, setMode()	150	傳遞錯誤, PassErr	121
模式設定, getMode()	78	顯示資料, Disp	50
模數, mod()	107	複	
範		複	
範本		因式, cFactor()	22
e 指數	2	複製變數或函數, CopyVar	28
N 次方根	1	複數	
一階導數	5	共軛, conj()	27
二階導數	6	解, cSolve()	37
不定積分	6	零點, cZeros()	41
分段函數 (2 段)	2	質	
分段函數 (N 段)	3	質數檢定, isPrime()	88
餘		餘	
質		餘, remain()	137
餘		餘切, cot()	32

餘弦	
表示	29
餘弦, cos()	29

導

導數	
一階導數, d()	205
數值導數, nDeriv()	112-113
數值導數, nDerivative()	112
導數或 N 階導數	
範本	6

整

整數, int()	84
整數除法, intDiv()	85
整數部份, iPart()	87

機

機率密度, normPdf()	115
-----------------------	-----

積

積分, J	205
-------------	-----

輸

輸入, Input	84
-----------------	----

錯

錯誤代碼和訊息	226
錯誤與疑難排除	
清除錯誤, ClrErr	25
傳遞錯誤, PassErr	121

隨

隨機	
多項式, randPoly()	134
矩陣, randMat()	133
範數, randNorm()	133
隨機樣本	134

儲

儲存	
符號, &	216-217

總

總和 (Σ)	
範本	5
總和, $\Sigma()$	208

聯

聯立方程式(2 個方程式)	
範本	3
聯立方程式(N 個方程式)	
範本	3
聯立方程式, simult()	153

隱

隱導函數, Impdif()	84
----------------------	----

點

點	
和, .+	198
乘, .*	198
差, -	198
除, /	199
冪, .^	199
積, dotP()	55

擴

擴展/串連, augment()	15
------------------------	----

轉

轉成	
4Grad	81
轉到, Goto	81
轉置, T	168

鎖

鎖定變數和變數群組	98
-----------------	----

	雙		度/分/秒, ▶DMS	52
			球面座標形式向量, ▶Sphere	161
			圓柱座標形式向量, ▶Cylind .	41
雙曲			顯示資料, Disp	50
	正切, tanh()	170	顯示資料, 顯示	146
	正弦, sinh()	155		
	餘弦, cosh()	31		
	羅			
	羅吉斯迴歸, Logistic	99		
	羅吉斯迴歸, LogisticD	100		
	警			
	警告代碼和訊息	234		
	攤			
	攤銷表, amortTbl()	8, 17		
	變			
	變異數, variance()	183		
	變數			
	刪除, DelVar	47		
	區域性, Local	98		
	從字元字符串建立名稱	224		
	清除所有單字元	25		
	變數, 鎖定和解開鎖定	78, 98, 182		
	變數和函數			
	複製	28		
	邏			
	邏輯隱含, ⇒	203, 221		
	邏輯雙隱含, ⇔	204		
	顯			
	顯示			
	極座標形式向量, ▶Polar	122		
	顯示, 顯示資料	146		
	顯示為			
	二進位, ▶Base2	17		
	十六進位, ▶Base16	19		
	十進位角, ▶DD	44		
	十進位整數, ▶Base10	19		
	直角向量, ▶Rect	135		