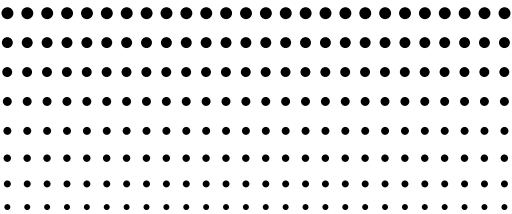


fx-5800P

用户说明书



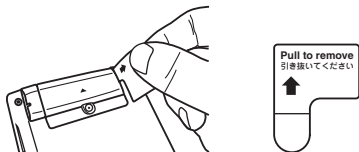
<http://world.casio.com/edu/>

■ 取出绝缘片！

计算器装有一个特殊的绝缘片，将电池与电池舱中的触片隔开使电池在存放和运输期间不会流失电量。首次准备使用计算器之前，请务必取出绝缘片。

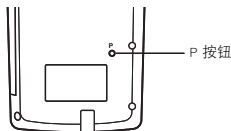
取出绝缘片

1. 按箭头指示方向拉绝缘片的标签以便取出绝缘片。



2. 取出绝缘片后，使用细尖状物品触压计算器背面的 P 按钮以进行对计算器的初始化。

请务必执行此步骤！不可省略！



■ 将计算器复原为初始缺省值设定

执行以下操作以将计算器复原为初始缺省值设定。注意：复原计算器也将删除当前计算器内存中存储的所有数据。

将计算器复原为初始缺省值设定

1. 按 **MODE** **3** (SYSTEM) **3** (Reset All)。
 - 此操作将显示“Reset All?” 确认信息。
2. 按 **EXE** (Yes)。
 - 如果不希望将计算器复原为初始缺省值设定，则按 **EXIT** (No)，而不要按 **EXE** (Yes)。

以下是将计算器复原为初始缺省值设定后所发生的情况。

- 计算模式和设定配置复原为“清除计算模式和设定（复原设定）”（第 13 页）中描述的初始缺省值。
- 删除计算履历数据、存储器数据、统计计算样本数据、程序数据和输入的所有其它数据。

■ 关于本手册

- 大部分键都可以执行多种功能。按住 **SHIFT** 或 **ALPHA**，那么另一个键将执行其它键的备用功能。备用功能标记在键帽上。



本手册中标记的备用功能操作如下所示。

示例: **SHIFT** **sin** (**sin⁻¹**) **1** **EXE**

括号中的符号表示前面的键操作执行的函数。

- 以下内容为手册中在显示器上显示的菜单项所使用的符号。

示例: **FUNCTION** - {**PROG**} - {→}

大括号 ({}) 中的符号表示选定的菜单项。

- 以下内容为手册中在显示器上显示的菜单项所使用的符号 (借助按数字键来执行)。

示例: **FUNCTION** - {**MATH**} **1** (**∫dX**)

括号中的符号表示前面的数字键访问的菜单项。

- 本《用户说明书》中输出的显示和图示 (如键标记) 仅供说明, 可能与所表示的实际项有所不同。
- 本手册的内容如有变更, 恕不另行通知。
- CASIO Computer Co., Ltd. 对于因购买或使用本产品 and 附属品时所产生的特殊、附带、意外和间接损害盖不负责。此外, CASIO Computer Co., Ltd. 对于任意方因使用本产品和附属品所发表的所有声明盖不负责。
- 本手册中使用的公司和产品名称可能是各个公司和产品所有者的注册商标或商标。

■ 示例中使用的符号

本手册中使用了大量符号用来提醒您需要配置哪些设定来正确执行示例操作。

- 如下所示的标记表示您需要变更计算器的显示格式设定。

如果您看到以下内容:	请将显示格式设定变更为:	如果您看到以下内容:	请将显示格式设定变更为:
MATH	普通显示	LINE	线性显示

有关详细信息, 请参照“选择显示格式 (MthIO, LineIO)” (第 11 页)。

- 如下所示的标记表示需要变更计算器的角度单位设定。

如果您看到以下内容:	请将角度单位设定变更为:	如果您看到以下内容:	请将角度单位设定变更为:
Deg	Deg	Rad	Rad

有关详细信息, 请参照“指定角度单位” (第 11 页)。

安全注意事项

使用此计算器之前，请务必阅读以下安全注意事项。确实保存好所有用户文档以供将来参考。



警告

此符号用于表示如果忽略此信息，则可能会导致人身伤害或材料损坏。

电池

- 从计算器取出电池后，请将电池放置在儿童不易接触到的安全地方，以避免儿童误吞。
- 将电池放在儿童不易接触到的地方。如果误吞电池，请立即咨询医生。
- 请勿对电池充电，尝试拆开电池或使电池短路。请勿将电池靠近热源或焚烧电池。
- 使用电池不当可能会导致电池泄漏和损害附近物品，并且可能会引起火灾和造成人身伤害。
 - 向计算器装入电池时，应始终确保正确放置电池正极 ⊕ 和负极 ⊖。
 - 如果长时间不打算使用计算器，请取出电池。
 - 仅使用本手册中为本计算器指定的电池型号。

处置计算器

- 请勿焚烧计算器。焚烧计算器可能会导致某些组件突然爆炸，从而引起火灾和造成人身伤害。

操作注意事项

- **首次使用计算器之前，请务必按计算器背面的 P 按钮。请参照第 1 页有关 P 按钮的信息。**
- **即使计算器正常运行，也要至少一年更换一次电池。**
废旧电池可能会泄漏，从而对计算器造成损害和使计算器产生故障。请勿将废旧电池遗留在计算器中。
- **配备的电池在运输和存放期间可能会产生轻微放电。因此，更换时间可能会比正常电池寿命要早。**
- **请勿对本产品使用镍氢或任何其它使用镍作为材料的电池。电池和产品规格不兼容可能会导致电池寿命缩短和使产品发生故障。**
- **电池容量较低可能会导致内存内容损毁或完全丢失。请始终存放所有重要数据的书面记录。**
- **请避免在超出温度极限的区域使用和存放计算器。**
温度过低可能会导致显示器响应速度降低、显示器完全失灵和缩短电池寿命。还应避免将计算器直接暴露于阳光之下、靠近窗口、靠近热源或可能会使计算器暴露于过高温度的任何地方。过热可能会使计算器外壳变色或变形，并对内部电路造成损害。
- **请避免在湿度过高和灰尘过多的区域使用和存放计算器。**
请勿将计算器放置在可能会被溅湿的地方，或暴露于湿度过高或灰尘过多的地方。此类情况可能会损害内部电路。
- **请勿摔掷计算器或使计算器受到强烈碰撞。**
- **请勿使计算器扭曲或弯曲。**
请避免将计算器放在裤子或其它紧身衣物的口袋中，因为这样可能会使计算器扭曲或弯曲。
- **请勿尝试拆卸计算器。**
- **请勿使用圆珠笔或其它尖锐物品按计算器的键。**
- **请使用柔软的干布清洁计算器的外部。**
如果计算器过脏，可使用沾有稀释、温和的中性家用清洁剂的布料擦除脏物。擦洗计算器之前拧干多余水份。请勿使用稀释剂、苯或其它挥发性物质来清洁计算器。这么做可能会擦除印刷的标记和损害外壳。

目录

取出绝缘片！	1
将计算器复原为初始缺省值设定	1
关于本手册	2
示例中使用的符号	2
安全注意事项	3
操作注意事项	4
开始计算之前	9
打开计算器	9
键标记	9
阅读显示器	9
计算模式和设定	10
选择计算模式	10
计算器设定	11
清除计算模式和设定（复原设定）	13
使用功能菜单	14
输入计算表达式和值	14
输入计算表达式（普通输入）	14
使用普通显示	16
编辑计算	18
查找错误位置	20
选定普通显示作为显示格式时显示 10 进制结果	20
计算示例	21
使用 $\boxed{S\leftrightarrow D}$ 键（S-D 变换）	21
S-D 变换示例	21
基本运算	22
四则运算	22
分数	23
百分比运算	25
度分秒（60 进制）运算	26
计算履历和重放	28
调用计算履历	28
使用重放	28
计算时使用多重语句	29
计算器存储器操作	30
使用答案存储器（Ans）	31
使用独立存储器	32
使用变量	33
清除所有存储器内容	34

保留可变存储器	34
用户存储区域	34
使用额外变量	35
使用 π 和科技常数	36
Pi (π)	36
科技常数	37
科技函数计算	39
三角和反三角函数	39
角度单位变换	40
双曲线和反双曲线函数	40
指数函数和对数函数	40
幂函数和乘方根函数	41
积分计算	42
微分	44
二阶微分	45
Σ 计算	45
坐标变换 (直角坐标 $\leftrightarrow$ 极坐标)	46
随机数函数	48
其它函数	49
使用工程符号	52
使用 10^3 工程符号 (ENG)	52
ENG 变换示例	52
使用工程符号	53
复数计算 (COMP)	54
输入复数	54
复数显示设定	55
复数计算结果显示示例	55
共轭复数 (Conjg)	56
绝对值和幅角 (Abs, Arg)	56
提取复数的实部 (ReP) 和虚部 (ImP)	56
取代缺省复数显示格式	57
矩阵计算 (COMP)	57
矩阵计算概述	57
关于 Mat Ans 屏幕	57
输入和编辑矩阵数据	58
执行矩阵计算	60
序列计算 (RECUR)	64
序列计算概述	64
创建序列表	66
序列计算注意事项	67

方程式计算 (EQN)	68
方程式计算概述	68
选择方程式类型	69
输入系数值	69
查看方程式的解	70
统计计算 (SD/REG)	70
统计样本数据	70
执行单变量统计计算	73
执行双变量统计计算	75
统计计算示例	82
基数计算 (BASE-N)	84
执行基数计算	84
将显示结果变换为其它基数	85
指定特定值的基数	86
使用逻辑运算和负的 2 进制数值执行计算	87
CALC	88
使用 CALC	88
SOLVE 功能	90
SOLVE 功能支持的表达式	90
使用 SOLVE 功能	90
从函数创建数表 (TABLE)	92
TABLE 模式概述	92
创建数表	94
数表创建注意事项	94
内置公式	95
使用内置公式	95
内置公式名称	96
用户公式	100
程序模式 (PROG)	102
程序模式概述	102
创建程序	103
运行程序	106
文件屏幕操作	108
删除程序	109
命令参考	110
程序命令	110
统计计算命令	118
其它 PROG 模式命令	119
数据通信 (LINK)	121
将两个 fx-5800P 计算器互相连接	121
在 fx-5800P 计算器之间传输数据	121

存储器管理器 (MEMORY)	123
可删除数据类型和支持的删除操作	123
使用存储器管理器	124
附录	125
计算的优先顺序	125
堆栈限制	126
计算范围、位数和精确度	126
错误信息	128
假设计算器发生故障之前 ...	130
电池容量低指示器	131
电源要求	131
规格	132

开始计算之前 ...

打开计算器

按 **AC/ON**。此操作将显示上次关闭计算器时所显示的屏幕。

调整显示对比度

如果显示器上的数字难以阅读，则可以尝试调整显示对比度。

1. 按 **MODE** **▼** **3** (SYSTEM) **1** (Contrast)。
 - 此操作将显示对比度调整屏幕。

CONTRAST
LIGHT [◀] DARK [▶]

2. 使用 **◀** 和 **▶** 调整显示对比度。
3. 根据需要进行了设定之后，按 **EXIT**。

注意

显示计算模式菜单（按 **MODE** 键显示）时，您也可以使用 **◀** 和 **▶** 来调整对比度。

关闭计算器

按 **SHIFT** **AC/ON** (OFF)。

键标记



	函数	键标记颜色	要执行函数：
①	ln		按该键。
②	e^x	橙色	按 SHIFT ，然后按该键。
③	[	红色	按 ALPHA ，然后按该键。
④	BIN	绿色	在 BASE-N 模式下，按该键。

阅读显示器

输入表达式和计算结果

此计算器可以在同一屏幕上，同时显示输入的表达式和结果。

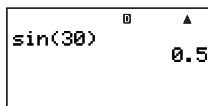
输入表达式	$2 \div 3 + 3 \div 7 i$
计算结果	$\frac{2}{3} + \frac{3}{7} i$

显示符号

下面描述的符号会显示在计算器的显示器上指示当前的计算模式、计算器设定和计算进度等等内容。

旁边的示例屏幕显示 **D** 符号。

将度数 (Deg) 选定为缺省值角度单位 (第 11 页) 时, **D** 符号会打开。



计算模式和设定

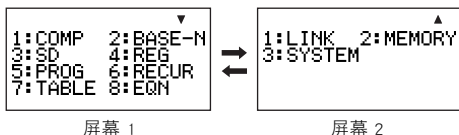
选择计算模式

计算器有 11 种“计算模式”。

选择计算模式

1. 按 **MODE**。

- 此操作将显示计算模式菜单。可使用 **▼** 和 **▲** 在菜单屏幕 1 和屏幕 2 之间进行切换。



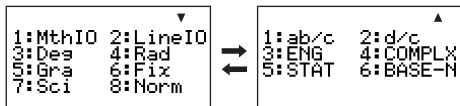
2. 执行以下一种操作, 选择需要的计算模式。

要选择此计算模式:	转至以下屏幕:	然后按以下键:
COMP (计算)	屏幕 1	[1] (COMP)
BASE-N (基数)		[2] (BASE-N)
SD (单变量统计)		[3] (SD)
REG (双变量统计)		[4] (REG)
PROG (编程)		[5] (PROG)
RECUR (递归)		[6] (RECUR)
TABLE (表)		[7] (TABLE)
EQN (方程式)		[8] (EQN)
LINK (通信)	屏幕 2	[1] (LINK)
MEMORY (存储器管理)		[2] (MEMORY)
SYSTEM (对比度调整, 复位)		[3] (SYSTEM)

- 要在不变更计算模式的情况下退出计算模式, 按 **MODE**。

■ 计算器设定

计算器设定可用来配置输入和输出设定、计算参数和其它设定。可使用设定屏幕来配置设定，可通过按 **[SHIFT]** **[MODE]** (SETUP) 来访问设定屏幕。有两个设定屏幕，可以使用 **▲** 和 **▼** 来切换这两个屏幕。



▣ 选择显示格式 (MthIO, LineIO)

可对输入的表达式和计算结果选择普通显示 (MthIO) 或线性显示 (LineIO)。

普通显示 (MthIO)

普通显示可按照分数、平方根、微分、积分、指数、对数和其它数学表达式的书面形式加以显示。此格式可应用于输入表达式和计算结果。选定普通显示时，将在可能的情况下使用分数、平方根或 π 符号显示计算结果。

例如，计算 $1 \div 2$ 会产生结果 $\frac{1}{2}$ ，而 $\pi \div 3$ 会产生结果 $\frac{1}{3}\pi$ 。

线性显示 (LineIO)

在线性显示情况下，将使用计算器定义的特殊格式输入和显示表达式以及函数。

例如， $\frac{1}{2}$ 将输入为 $1 \div 2$ ，而 $\log_2 4$ 将输入为 $\log(2,4)$ 。

选定线性显示时，将使用 10 进制数值显示分数之外的所有计算结果。

要选择此显示格式：	执行以下键操作：
普通显示 (MthIO)	[SHIFT] [MODE] [1] (MthIO)
线性显示 (LineIO)	[SHIFT] [MODE] [2] (LineIO)

注意

有关使用普通显示和线性显示时的输入过程的信息，请参照本手册第 14 页上的“输入计算表达式和值”和本手册此章节有关每个计算类型的说明。

▣ 指定角度单位

选择以下角度单位：	执行以下键操作：
度数	[SHIFT] [MODE] [3] (Deg)
弧度	[SHIFT] [MODE] [4] (Rad)
梯度	[SHIFT] [MODE] [5] (Gra)

$$(90^\circ = \frac{\pi}{2} \text{ 弧度} = 100 \text{ 梯度})$$

指定显示位数

指定以下显示位数设定:	执行以下键操作:
小数位数	SHIFT MODE 6 (Fix) 0 (0) 到 9 (9)
有效位数	SHIFT MODE 7 (Sci) 1 (1) 到 9 (9), 0 (10)
指数显示范围	SHIFT MODE 8 (Norm) 1 (Norm1) 或 2 (Norm2)

以下内容说明计算结果如何根据指定的设定来显示。

- 根据指定的小数位数 (Fix) 来显示 0 到 9 个小数位。计算结果会四舍五入为指定的位数。

示例: $100 \div 7 = 14.286$ (Fix = 3)

14.29 (Fix = 2)

- 在使用 Sci 指定有效位数后, 将通过使用指定的有效位数和 10 的适用乘方来显示计算结果。计算结果会四舍五入为指定的位数。

示例: $1 \div 7 = 1.4286 \times 10^{-1}$ (Sci = 5)

1.429×10^{-1} (Sci = 4)

- 选择 Norm1 或 Norm2 会导致当结果位于以下定义的范围之内时显示会切换为指数符号。

Norm1: $10^{-2} > |x|, |x| \geq 10^{10}$

Norm2: $10^{-9} > |x|, |x| \geq 10^{10}$

示例: $100 \div 7 = 14.28571429$ (Norm1 或 Norm2)

$1 \div 200 = 5. \times 10^{-3}$ (Norm1)

0.005 (Norm2)

指定分数显示格式

为计算结果显示指定以下分数格式:	执行以下键操作:
带分数	SHIFT MODE ▼ 1 (ab/c)
假分数	SHIFT MODE ▼ 2 (d/c)

指定工程符号设定

此设定使您能够接通关闭工程符号。有关更多信息, 请参照第 52 页上的“使用工程符号”。

执行以下操作:	执行以下键操作:
打开工程符号	SHIFT MODE ▼ 3 (ENG) 1 (EngOn)
关闭工程符号	SHIFT MODE ▼ 3 (ENG) 2 (EngOff)

打开工程符号时 (EngOn), 当计算结果超出范围 $1 \leq |x| < 1000$ 时将使用工程符号。

指定复数显示格式

可对复数计算结果指定直角坐标格式或极坐标格式。

为计算结果显示指定以下复数格式:	执行以下键操作:
直角坐标	SHIFT MODE ▼ 4 (COMPLX) 1 ($a+bi$)
极坐标	SHIFT MODE ▼ 4 (COMPLX) 2 ($r\angle\theta$)

选定极坐标格式时，不能使用 ENG 变换（第 52 页）。

指定统计频率设定

可使用以下键操作在 SD 模式和 REG 模式计算（第 70 页）期间打开或关闭统计频率。

选择以下频率设定:	执行以下键操作:
打开频率	SHIFT MODE ▼ 5 (STAT) 1 (FreqOn)
关闭频率	SHIFT MODE ▼ 5 (STAT) 2 (FreqOff)

变更 BASE-N 模式负值设定

可使用以下键操作在 BASE-N 模式中启用或禁用负值。

指定以下设定:	执行以下键操作:
启用负值	SHIFT MODE ▼ 6 (BASE-N) 1 (Signed)
禁用负值	SHIFT MODE ▼ 6 (BASE-N) 2 (Unsigned)

清除计算模式和设定（复原设定）

执行以下键操作复原计算模式和设定。

MODE **▼** **3** (SYSTEM) **2** (Reset Setup) **EXE** (Yes)

如果不希望复原计算器的设定，请在以上操作中代替 **EXE** (Yes) 按 **EXIT** (No)。

计算模式.....COMP

设定

显示格式.....MthIO

角度单位.....Deg

指数显示.....Norm1

分数格式.....d/c

复数格式..... $a+bi$

工程符号.....EngOff

统计频率.....FreqOff

BASE-N 负数.....Signed

使用功能菜单

功能菜单使您能够访问各种数学函数、命令、常数、符号和其它特殊操作。

显示功能菜单

按 **FUNCTION**。例如，以下显示的功能菜单会在 COMP 模式下按 **FUNCTION** 时显示。

D Math▼	
1:MATH	2:COMPLX
3:PROG	4:CONST
5:ANGLE	6:CLR
7:STAT	8:MATRIX

退出功能菜单

按 **EXIT**。

输入计算表达式和值

输入计算表达式（普通输入）

计算器的普通输入系统使您能够通过按 **EXE** 以计算表达式的书面格式输入和执行计算表达式。计算器会自动确定加、减、乘、除、分数和括号的正确优先顺序。

示例： $2(5+4) - 2 \times (-3) =$

LINE

2 (5 + 4) - 2 × (-) 3 EXE

2(5+4)-2×-3
24

输入带括号的科技函数（sin、cos 和 $\sqrt{\quad}$ 等等）。

计算器支持输入如下带括号的科技函数。注意：输入自变量后，需要按 **)** 关闭括号。

sin(, cos(, tan(, $\sin^{-1}$ (, $\cos^{-1}$ (, $\tan^{-1}$ (, sinh(, cosh(, tanh(, $\sinh^{-1}$ (, $\cosh^{-1}$ (, $\tanh^{-1}$ (, log(, ln(, $e^{\wedge}$ (, $10^{\wedge}$ (, $\sqrt{\quad}$ (, $\sqrt[3]{\quad}$ (, Abs(, Pol(, Rec(, $\int$ (, d/dx (, d^2/dx^2 (, Σ (, P(, Q(, R(, Arg(, Conjg(, ReP(, ImP(, Not(, Neg(, Det(, Trn(, Rnd(, Int(, Frac(, Intg(, RanInt#(

示例： $\sin 30 =$

LINE

sin 30) EXE

sin(30)
0.5

注意

使用普通输入时，某些函数需要不同的输入顺序。有关更多信息，请参照第 17 页上的“使用普通显示输入计算表达式”。

❑ 忽略乘号

在以下情况下，可以忽略乘号。

- 后面直接跟开括号：2 $\times$ (5 + 4)
- 后面直接跟带括号的科技函数：2 $\times$ sin(30), 2 $\times$ $\sqrt{\quad}$ (3)
- 后面直接跟前置符号（负号除外）：2 $\times$ h123
- 后跟变量名称、常数或随机数：20 $\times$ A, 2 $\times$ π , 2 $\times$ i

❑ 最后的关括号

在按 **EXE** 键时，可以忽略计算末尾的一个或多个闭合括号。

示例：(2 + 3) (4 - 1) = 15

LINE

(2 + 3)
(4 - 1) **EXE**

(2+3)(4-1)
15

❑ 计算表达式换行（线性显示）

使用线性显示时，长于 16 个字符（数字、字母和运算符）的计算表达式会自动换行到下一行。

示例：123456789 + 123456789 = 246913578

LINE

1 2 3 4 5 6 7 8 9 +
1 2 3 4 5 6 7 8 9 **EXE**

123456789+123456789
246913578

❑ 输入字符数（字节数）

当输入数学表达式时，它会存储在称之为“输入区域”的存储器中，“输入区域”的容量为 127 个字节。这意味着可以对单个数学表达式最多输入 127 个字节。

选择线性显示作为显示格式时，每个函数一般使用 1 个或 2 个字节的存储器。在普通显示格式下，每个函数使用 4 个或更多的存储器字节数。有关更多信息，请参照第 17 页上的“使用普通显示输入计算表达式”。

通常情况下，指示显示器上当前输入位置的光标为闪烁的竖条 (I) 或横条 (—)。当输入区域的剩余容量为 10 字节或更少时，光标会变更为闪烁的方块 (■)。

如果发生这种情况，请停止在合适的位置停止输入当前表达式并计算其结果。

■ 使用普通显示

选定普通显示作为显示格式时（第 11 页），可按照分数和一些科技函数的书面格式加以输入。

◆ 普通显示基础

下表列出了可使用普通显示格式输入的科技函数类型。

- *1 列显示每个科技函数使用的存储器字节数。请参照“输入字符数（字节数）”（第 15 页）以获取更多信息。
- 有关 *2 列的信息，请参照“用值和表达式作自变量”（第 17 页）。

支持普通显示的科技函数

函数	键操作	*1	*2
假分数		9	是
带分数	($\frac{\blacksquare}{\blacksquare}$)	14	否
$\log(a,b)$	- {MATH} (logab)	7	是
10^x	($10^{\blacksquare}$)	4	是
e^x	($e^{\blacksquare}$)	4	是
平方根 ($\sqrt{}$)		4	是
立方根 ($\sqrt[3]{}$)	($\sqrt[3]{\blacksquare}$)	9	是
平方		4	否
倒数	(x^{-1})	5	否
乘方		4	是
乘方根	($\sqrt[\blacksquare]{\blacksquare}$)	9	是
绝对值 (Abs)	- {MATH} (Abs)	4	是
积分	- {MATH} ($\int dx$)	8	是
微分	- {MATH} (d/dx)	7	是
二阶微分	- {MATH} (d^2/dx^2)	7	是
Σ 计算	- {MATH} ($\Sigma()$)	11	是

注意

如果使用普通显示时，在括号（ 和 ）中包括值或表达式，则括号的高度将会根据括号括起的是一行还是两行来自动调整。无论高度是多少，每个前括号和后括号都占用一个字节的存储器。

使用普通显示输入计算表达式

1. 要输入指定函数，请执行“支持普通显示的科技函数”表的“键操作”列中的操作。
2. 在 $\square$ 表示的输入字段处输入所需的值和表达式。
 - 使用光标键来在表达式的输入字段之间进行移动。

示例： 输入 $\frac{1+2}{2 \times 3}$

MATH

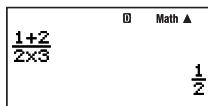
指定分数输入：

输入分子：

将光标移动到分母：

输入分母：

执行计算：



重要事项！

- 某些类型的表达式可能会导致计算公式的高度高于显示行。计算公式的最大允许高度为两个显示器屏幕（31 点 × 2）。如果输入的公式的高度超过允许限制，则无法继续输入。
- 允许嵌套函数和括号。如果嵌套过多的函数和 / 或括号，则无法继续输入。如果发生这种情况，则可将计算划分为多个部分然后分别计算每个部分。

◀ 向左和向右滚动屏幕

使用普通显示进行输入时，屏幕最多显示 14 个字符。当输入的字符数超过 14 个时，屏幕将会自动滚动。如果发生这种情况，则将会显示 ◀ 符号，使您知道显示器的左侧还有表达式。

MATH



- 显示 ◀ 符号时，可使用 ◀ 键来左移光标和滚动屏幕。
- 滚动到左侧将导致部分表达式遗留在显示器右侧，将在右边以 ▶ 符号指示此情况。当屏幕上显示 ▶ 符号时，可使用 ▶ 键来右移光标和滚动屏幕。

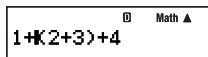
◻ 用值和表达式作自变量

使用普通显示进行输入时，在某些情况下可以将已输入的、被括号括起的值或表达式用作科技函数的自变量（如 $\sqrt{\quad}$ ）和分数的分子等等。为了此处的说明，支持使用先前输入值或带有括号的表达式的普通显示函数称为“可插入普通显示函数”。

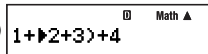
示例： 在以下计算中将普通显示函数 $\sqrt{\quad}$ 插入到带括号的表达式中： $1 + (2 + 3) + 4$

MATH

（将光标立即移动到带括号的表达式的左侧。）



SHIFT DEL (INS)



注意

- 不是所有普通显示函数都是可插入的。只有“支持普通显示的科技函数”（第 16 页）表中显示“是”的列中的科技函数才可插入。
- 光标可以紧邻带括号的表达式、数字值或分数的左侧。插入可插入函数将使带括号的表达式、值或分数成为被插入函数的自变量。
- 如果光标紧邻科技函数的左侧，则整个函数将成为插入函数的自变量。

编辑计算

插入模式和覆盖模式

本计算器有两个输入模式。插入模式会在光标位置插入输入，并会将所有内容移到光标右侧以空出位置。覆盖模式会将光标位置的键操作替换为您的输入。

选定普通显示作为显示格式时，只有插入模式可供使用。不能变更为覆盖模式。选定线性显示作为显示格式时，可选择插入模式或覆盖模式来进行输入。

	原始表达式	按 $\boxed{+}$
插入模式	光标 $\underline{\hspace{1cm}}$ $1+2\boxed{+}4$	$1+2+\boxed{+}4$
覆盖模式	光标 $\underline{\hspace{1cm}}$ $1+2\boxed{3}4$	$1+2+\boxed{4}$

竖条光标 (|) 表示插入模式，横条光标 (■) 表示覆盖模式。

选择输入模式

初始缺省输入模式设定为插入模式。如果选择线性显示作为显示格式并且希望变更为覆盖模式，则按： $\boxed{\text{SHIFT}} \boxed{\text{DEL}} (\text{INS})$ 。

编辑刚刚输入的键操作

当光标位于输入的末尾时，按 $\boxed{\text{DEL}}$ 删除执行的最后键操作。

示例：将 369×13 更正为 369×12

MATH LINE

$\boxed{3} \boxed{6} \boxed{9} \boxed{\times} \boxed{1} \boxed{3}$

$\boxed{0}$ Math $\blacktriangle$

369×13

$\boxed{\text{DEL}}$

$\boxed{0}$ Math $\blacktriangle$

369×11

$\boxed{2}$

$\boxed{0}$ Math $\blacktriangle$

369×12

◆ 删除键操作

在插入模式下，可使用 ◀ 和 ▶ 将光标移动到希望删除的键操作的右侧，然后按 **DEL**。在覆盖模式下，可将光标移动到希望删除的键操作处，然后按 **DEL**。每次按 **DEL** 都会删除一个键操作。

示例：将 $369 \times \times 12$ 更正为 369×12

插入模式

MATH **LINE**

3 6 9 **X** **X** 1 2

Math ▲
369xx12

◀ ▶

Math ▲
369xx12

DEL

Math ▲
369x12

覆盖模式

LINE

3 6 9 **X** **X** 1 2

▲
369xx12_

◀ ◀ ◀

▲
369x_12

DEL

▲
369x_2

◆ 编辑表达式内的键操作

在插入模式下，可使用 ◀ 和 ▶ 将光标移动到希望编辑的键操作的右侧，再按 **DEL** 来删除该键操作，然后执行正确的键操作。在覆盖模式下，可将光标移动到希望更正的键操作处，然后执行正确的键操作。

示例：将 $\cos(60)$ 更正为 $\sin(60)$

插入模式

MATH **LINE**

cos 6 0)

Math ▲
cos(60)

◀ ◀ ◀ **DEL**

Math ▲
E0)

sin

Math ▲
sin(E0)

覆盖模式

LINE

cos 6 0)

▲
cos(60)_

◀ ◀ ◀ ◀

▲
_os(60)

  sin(_0)

将键操作插入到表达式中

在希望将键操作插入到表达式中时，务必选择插入模式。可使用  和  将光标移动到希望插入键操作的位置然后执行键操作。

查找错误位置

如果计算表达式不正确，则当按  执行计算表达式时，将会在显示器上显示错误信息。在显示错误信息后按 、 或  键将会导致光标跳到导致错误的计算中的位置处，以便您能够更正错误。

示例：当输入 $14 \div 0 \times 2 =$ 而不是输入 $14 \div 5 \times 2 =$ 时（以下示例使用插入模式）。

LINE

Math ERROR
Press: [EXIT]

 (或 , )



错误位置





 **5.6**

- 当显示错误信息时除了按 、 或  查找错误的位置，还可以按  清除计算。

选定普通显示作为显示格式时显示 10 进制结果

当选定了普通显示时，按  执行计算将会以普通格式显示结果。按   将会执行计算并以小数格式显示结果。

使用以下格式显示结果：	执行以下键操作：
普通格式	
小数格式	 

注意

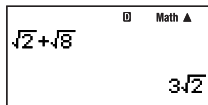
选择线性显示作为显示格式时，不管按 **EXE** 或 **SHIFT** **EXE**，均使用线性（小数）格式显示执行的计算。

■ 计算示例

示例： $\sqrt{2} + \sqrt{8} = 3\sqrt{2}$

MATH

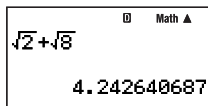
$\sqrt{\square}$ **2** $\rightarrow$ $+$ $\sqrt{\square}$ **8** **EXE**



Calculator display showing the result of the calculation $\sqrt{2} + \sqrt{8}$ in radical format: $3\sqrt{2}$. The display also shows the Math mode indicator and the result of the previous calculation.

以小数格式生成结果：

$\sqrt{\square}$ **2** $\rightarrow$ $+$ $\sqrt{\square}$ **8** **SHIFT** **EXE**



Calculator display showing the result of the calculation $\sqrt{2} + \sqrt{8}$ in decimal format: 4.242640687. The display also shows the Math mode indicator and the result of the previous calculation.

使用 **S $\leftrightarrow$ D** 键（S-D 变换）

可以使用 **S $\leftrightarrow$ D** 键在小数 (D) 和标准 (S) 格式（分数、 $\sqrt{\square}$ 和 π ）之间变换值。

重要事项！

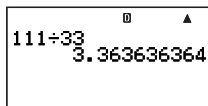
- 根据按 **S $\leftrightarrow$ D** 键时显示器上的计算结果类型，变换过程可能需要花费一些时间来执行。
- 对于某些计算结果，按 **S $\leftrightarrow$ D** 键不会变换显示的值。

■ S-D 变换示例

示例 1：选定线性显示作为显示格式时，执行计算 $111 \div 33$ ，然后将结果变换为分数格式

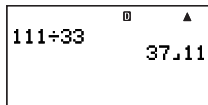
LINE

1 **1** **1** $\div$ **3** **3** **EXE**



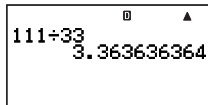
Calculator display showing the result of the calculation $111 \div 33$ in decimal format: 3.363636364. The display also shows the Math mode indicator and the result of the previous calculation.

S $\leftrightarrow$ D



Calculator display showing the result of the calculation $111 \div 33$ in fraction format: 37.11. The display also shows the Math mode indicator and the result of the previous calculation.

S $\leftrightarrow$ D



Calculator display showing the result of the calculation $111 \div 33$ in decimal format: 3.363636364. The display also shows the Math mode indicator and the result of the previous calculation.

注意

- 每次按 **S $\div$ D** 键可在两种格式之间切换显示的结果。
- 分数的格式取决于当前选定的分数显示格式（假分数或带分数）（第 12 页）。

示例 2: 选定普通显示作为显示格式时，执行计算 $111 \div 33$ ，然后将结果变换为小数格式

MATH

1 1 1 $\div$ 3 3 EXE

$$\begin{array}{r} 111 \div 33 \\ \hline 37 \\ 11 \end{array}$$

S $\div$ D

$$\begin{array}{r} 111 \div 33 \\ \hline 3.363636364 \end{array}$$

S $\div$ D

$$\begin{array}{r} 111 \div 33 \\ \hline 37 \\ 11 \end{array}$$

示例 3: 选定普通显示作为显示格式时，如下所示执行 π 计算，然后将结果变换为小数格式

MATH

SHIFT $\times 10^{-1}$ (π) $\times$ $\frac{2}{5}$ 2 ∇ 5 EXE

$$\begin{array}{r} \pi \times \frac{2}{5} \\ \hline \frac{2}{5} \pi \end{array}$$

S $\div$ D

$$\begin{array}{r} \pi \times \frac{2}{5} \\ \hline 1.256637061 \end{array}$$

基本运算

除非另有说明，可以在任意计算器计算模式下，执行本节中的计算，但 BASE-N 模式除外。

四则运算

四则运算可用于执行加 (**+**)、减 (**-**)、乘 (**$\times$**) 和除 (**$\div$**)。

示例 1: $2.5 + 1 - 2 = 1.5$

LINE

$2 \div 5 + 1 - 2 \text{ EXE}$

2.5+1-2
1.5

示例 2: $7 \times 8 - 4 \times 5 = 36$

LINE

$7 \times 8 - 4 \times 5 \text{ EXE}$

7x8-4x5
36

- 计算器会自动确定加、减、乘和除的正确优先顺序。请参照第 125 页的“计算的优先顺序”以获取更多信息。

■ 分数

请记住：在计算器上输入分数时，需要使用的步骤取决于选择普通显示还是线性显示作为显示格式（第 11 页），如下所示。

普通显示：

键操作		显示
假分数	$\boxed{\frac{\Box}{\Box}}$ $7 \downarrow 3$	$\frac{7}{3}$
带分数	$\boxed{\text{SHIFT}} \boxed{\frac{\Box}{\Box}}$ $(\frac{\Box}{\Box})$ $2 \rightarrow 1 \downarrow 3$	$2\frac{1}{3}$

线性显示：

键操作		显示
假分数	$7 \boxed{\frac{\Box}{\Box}} 3$	$\begin{array}{cc} 7 \downarrow 3 \\ \text{分子} & \text{分母} \end{array}$
带分数	$2 \boxed{\frac{\Box}{\Box}} 1 \boxed{\frac{\Box}{\Box}} 3$	$\begin{array}{ccc} 2 \downarrow 1 \downarrow 3 \\ \text{整数} & \text{分子} & \text{分母} \end{array}$

如上所述，普通显示使您按照分数的书面格式输入分数，而线性显示需要输入特殊符号 ($\downarrow$)。

注意

- 在初始缺省值设定下，分数会显示为假分数。
- 在显示之前，始终会自动简化分数计算结果。例如，执行 $2 \downarrow 4 =$ ，将显示结果 $1 \downarrow 2$ 。

分数计算示例

示例 1: $\frac{2}{3} + \frac{1}{2} = \frac{7}{6}$

MATH

2 3

Math ▲

$$\frac{2}{3}$$

+

Math ▲

$$\frac{2}{3} +$$

1 2

Math ▲

$$\frac{2}{3} + \frac{1}{2}$$

EXE

Math ▲

$$\frac{2}{3} + \frac{1}{2}$$

$$\frac{7}{6}$$

LINE

2 3 + 1 2

$$2.\dot{3} + 1.\dot{2}$$

EXE

▲

$$2.\dot{3} + 1.\dot{2}$$

$$7.\dot{6}$$

示例 2: $3\frac{1}{4} + 1\frac{2}{3} = 4\frac{11}{12}$ (分数显示格式: ab/c)

LINE

3 1 4 +
1 2 3 EXE

▲

$$3.\dot{1}\dot{4} + 1.\dot{2}\dot{3}$$

$$4.\dot{1}\dot{1}\dot{1}\dot{2}$$

MATH

SHIFT (= $\frac{a}{b}$) 3 1 4 +

Math ▲

$$3\frac{1}{4} +$$

SHIFT (= $\frac{a}{b}$) 1 2 3 EXE

Math ▲

$$3\frac{1}{4} + 1\frac{2}{3}$$

$$4\frac{11}{12}$$

注意

- 如果组成带分数表达式的元素（整数位 + 分子位 + 分母位 + 分隔符）总数大于 10，则将以小数格式显示计算结果。
- 如果输入计算同时包括分数和小数值，则将以小数格式显示结果。
- 您只能为分数的元素输入整数。

在假分数和带分数格式之间切换

要将假分数变换为带分数（或将带分数变换为假分数），按 **SHIFT** **S<D>** ($a\frac{b}{c} \Leftrightarrow \frac{d}{c}$)。

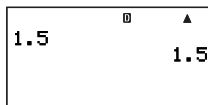
在分数和小数格式之间切换

使用以下过程在分数和小数格式之间，切换显示的计算结果。

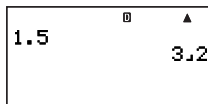
示例： $1.5 = \frac{3}{2}$, $\frac{3}{2} = 1.5$

LINE

1 **.** **5** **EXE**

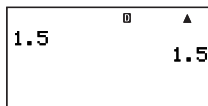


S<D>



当前的分数显示格式设定决定是显示假分数还是带分数。

S<D>



注意

如果构成带分数的元素（整数位 + 分子位 + 分母位 + 分隔符）的总数大于 10，则计算器无法从小数格式切换为分数格式。

百分比运算

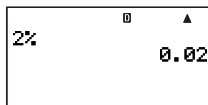
输入一个值和一个百分比号 (%) 会使该值成为百分比值。百分比号 (%) 将它之前紧接的值用作自变量，只需使用该值除以 100 来获得百分比值。

百分比运算示例

所有以下示例都使用线性显示 (**LINE**) 执行。

示例 1: $2\% = 0.02$ ($\frac{2}{100}$)

2 **SHIFT** **.** **(%)** **EXE**



示例 2: $150 \times 20\% = 30$ ($150 \times \frac{20}{100}$)

1 5 0 $\times$ 2 0 SHIFT $\div$ (%) EXE

150×20%
30

示例 3: 660 是 880 的百分之几?

6 6 0 $\div$ 8 8 0
SHIFT $\div$ (%) EXE

660÷880%
75

示例 4: 将 2500 增加 15%。

2 5 0 0 + 2 5 0 0 $\times$
1 5 SHIFT $\div$ (%) EXE

2500+2500×15%
2875

示例 5: 将 3500 减少 25%。

3 5 0 0 - 3 5 0 0 $\times$
2 5 SHIFT $\div$ (%) EXE

3500-3500×25%
2625

■ 度分秒 (60 进制) 计算

可执行使用 60 进制数值的计算，并且可以在 60 进制和 10 进制之间进行变换。

▣ 输入 60 进制数值

以下是输入 60 进制数值的基本句法。

{度} {分} {秒}

示例: 输入 $2^{\circ}30'30''$

LINE

2 $\rightarrow$ 3 0 $\rightarrow$ 3 0 $\rightarrow$ EXE

$2^{\circ}30'30''$
 $2^{\circ}30'30''$

• 注意：即使值为 0，也必须始终为度和分输入内容。

示例：输入 $0^{\circ}00'30''$ ，按 0 $\rightarrow$ 0 $\rightarrow$ 3 0 $\rightarrow$ 。

◆ 60 进制计算示例

- 以下类型的 60 进制计算将产生 60 进制结果。
 - 两个 60 进制数值的加或减
 - 60 进制值和 10 进制值的乘或除

示例 1: $2^{\circ}20'30'' + 39^{\circ}30'' = 3^{\circ}00'00''$

LINE

2 [] 2 0 [] 3 0 [] +
0 [] 3 9 [] 3 0 [] EXE

2° 20' 30" + 0° 39' 30"
3° 0' 0"

示例 2: $2^{\circ}20'00'' \times 3.5 = 8^{\circ}10'00''$

LINE

2 [] 2 0 [] 0 [] X
3 [] . [] 5 [] EXE

2° 20' 0" x 3.5
8° 10' 0"

◆ 执行 10 进制计算以获取 60 进制结果

可以使用“►DMS”命令执行 10 进制计算和获取 60 进制结果。“►DMS”命令只能在 COMP 模式下使用。

示例: 执行计算 $100 \div 3$ ，使其产生 60 进制结果

LINE

1 0 0 [] ÷ [] 3 [] EXE
FUNCTION - {ANGLE} 4 [] (►DMS) [] EXE

100÷3
33.33333333
Ans►DMS
33° 20' 0"

◆ 在 60 进制和 10 进制之间变换

在显示计算结果时，按 [] 将会在 60 进制和 10 进制之间切换。

示例: 将 2.255 变换为 60 进制

LINE

2 [] . [] 2 [] 5 [] 5 [] EXE

2.255
2.255

2.255
2° 15' 18"

计算履历和重放

可在 COMP 和 BASE-N 模式下使用计算履历。

调用计算履历

显示器右上角的 ▲ 符号表示计算履历中存有数据。要查看计算履历中的数据，按 (▲)。每次按 (▲) 将会上滚（返回）一次计算，并同时显示计算表达式及其结果。

示例：

MATH

1 + 1 EXE 2 + 2 EXE
3 + 3 EXE

Math ▲
3+3
6

(▲)

Math ▼▲
2+2
4

(▲)

Math ▼
1+1
2

在计算履历中滚动时，显示器上将显示 ▼ 符号，它表示当前记录的下一个记录（较新的记录）。显示此符号时，按 (▼) 来下滚（向前）计算履历。

重要事项！

- 当变更为其它计算模式或变更显示格式时，计算履历记录会全部清除。
- 计算履历容量有限。如果在计算履历已满的情况下，执行新的计算，则计算履历中的最早记录将会被自动删除，以便为新的记录空出位置。

注意

包含任意以下函数的计算，在执行时将不会存储在计算履历中。

CALC、SOLVE、内置公式和用户公式

使用重放

当显示器上显示计算履历时，按 (◀) 或 (▶) 显示光标，并进入编辑模式。按 (▶) 会在计算表达式的开头处显示光标，而 (◀) 会在计算表达式的末尾显示光标。在执行了所需变更后，按 (EXE) 执行计算。

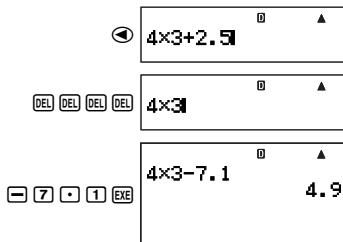
示例：4 × 3 + 2.5 = 14.5

4 × 3 - 7.1 = 4.9

LINE

4 X 3 + 2 . 5 EXE

▲
4x3+2.5
14.5



计算时使用多重语句

多重语句是由特殊分隔符代码 (: 和 $\blacktriangle$) 分隔的多个计算表达式构成的语句。以下示例显示两个分隔符的不同之处。

{ 表达式 1 } : { 表达式 2 } : : { 表达式 n }

按 **EXE** 按顺序 (从 { 表达式 1 } 开始并以序列中的最后一个表达式结束) 执行每个表达式。在此之后, 将会在显示器上显示最后表达式的结果。

示例: 执行计算 $123 + 456$, 然后从结果减去 1000

LINE

1 2 3 + 4 5 6 **SHIFT** **√** (:)
1 0 0 0 - **SHIFT** ($\leftarrow$) (Ans)

123+456:1000-Ans

EXE

123+456:1000-Ans
421

{ 表达式 1 } $\blacktriangle$ { 表达式 2 } $\blacktriangle$ $\blacktriangle$ { 表达式 n }

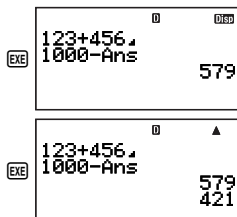
在此情况下, 按 **EXE** 从 { 表达式 1 } 开始执行。当执行遇到 $\blacktriangle$ 分隔符时, 执行会暂停然后在显示器上显示此时的计算结果。再次按 **EXE** 从 $\blacktriangle$ 分隔符后的表达式继续执行计算。

示例: 显示计算 $123 + 456$ 的结果, 并从 1000 减去它

LINE

1 2 3 + 4 5 6 **SHIFT** **x²** ($\blacktriangle$)
1 0 0 0 - **SHIFT** ($\leftarrow$) (Ans)

123+456 $\blacktriangle$
1000-Ans



注意

- 当执行多重语句计算因 $\blacktriangle$ 分隔符而暂停时，将会在显示器的右上角显示 **Disp** 符号。
- 执行多重语句计算时，每次构成多重语句的任意语句产生结果时 Ans(答案存储器)(第 31 页)都会更新。
- 可以在同一计算中混合使用 “ $\blacktriangle$ ” 和 “:” 分隔符。

计算器存储器操作

本计算器包含以下所述的存储器类型，可以用来存储和调用值。

存储器名称	描述
答案存储器	答案存储器包含上次执行的计算的结果。
独立存储器	当加或减多个计算结果时，独立存储器非常方便。
变量	可分别向字母 A 到 Z 指定不同的值并在计算中使用它们。注意：变量 M 也可用于存储独立存储器值。
额外变量	当需要存储从 A 到 Z 的 26 个字母之外的更多值时，可以创建额外变量。最多可以保留 2372 个额外变量，其命名为 Z[1] 和 Z[2] 等等。
公式变量	以下字母变量供计算器的内置公式或用户公式使用。 • 小写字母字符：a 到 z • 希腊字符：α 到 ω，A 到 Ω • 下标字母字符和希腊字符：A_1、a_0、ω_1、和 Δ_x 等等。 有关内置公式和公式变量的详细信息，请参照“内置公式”(第 95 页)。

当按 **AC/ON** 键、变更为其它模式或关闭计算器时，上述存储器的类型不被清除。

■ 使用答案存储器 (Ans)

在计算器上执行的所有新计算的结果，自动存储在答案存储器 (Ans) 中。

▣ 在连续计算中自动插入 Ans

如果在显示器上仍然显示有先前的计算结果时，开始新的计算，则计算器自动在新计算的适当位置插入 Ans。

示例 1: 使用 3×4 的结果除以 30

LINE

3×4 **EXE**

0 ▲
3×4
12

(然后) $\div 30$ **EXE**

0 ▲
3×4
12
Ans÷30
0.4

按 $\div$ 将自动输入 Ans。

示例 2: 确定 $3^2 + 4^2$ 的结果的平方根

LINE

$3 \times^2 + 4 \times^2$ **EXE**

0 ▲
 3^2+4^2
25

$\sqrt{\square}$ **EXE**

0 ▲
 3^2+4^2
25
 $\sqrt{Ans}$
5

注意

- 在上述示例中，当显示器上显示计算结果时，计算器会自动将 Ans 作为所输入的所有的计算运算符或科技函数的自变量插入。
- 在函数带有具有括号的自变量的情况下（第 14 页），仅当单独输入函数然后按 **EXE** 时 Ans 才会自动成为自变量。但是，注意使用普通显示时，如果使用带有括号的函数，则 Ans 可能不会自动成为自变量。
- 基本上，仅当显示器上仍然显示先前的计算结果时，才会自动插入 Ans，并且会紧接在产生 Ans 的计算后。如果希望在使用按 **AC/ON** 清除显示后插入 Ans，则按 **SHIFT** **(←)** (Ans)。

手动将 Ans 插入到计算中

可以通过按 **SHIFT** **(←)** (Ans) 在当前的光标位置将 Ans 插入到计算中。

示例 1: 要在另一个计算中使用 $123 + 456$ 的结果，如下所示

$$123 + 456 = 579 \quad 789 - 579 = 210$$

LINE

1 2 3 + 4 5 6 EXE

123+456	0	▲
		579

7 8 9 - SHIFT (←) (Ans) EXE

123+456	0	▲
789-Ans		210

示例 2: 要确定 $3^2 + 4^2$ 的平方根，然后向结果加 5

LINE

3 x^2 + 4 x^2 EXE

3^2+4^2	0	▲
		25

$\sqrt{\square}$ SHIFT (←) (Ans)) + 5 EXE

3^2+4^2	0	▲
$\sqrt{(Ans)+5}$		10

使用独立存储器

独立存储器 (M) 主要用于计算累积总和。

添加到独立存储器

当显示器上显示输入的值或计算结果时，按 **M+** 将其添加到独立存储器 (M) 中。

示例: 将 $105 \div 3$ 的结果添加到独立存储器 (M)

LINE

1 0 5 $\div$ 3 M+

105 $\div$ 3M+	0	▲
		35

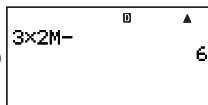
从独立存储器中去除

当显示器上显示输入的值或计算结果时，按 **SHIFT** **M+** (M-) 从独立存储器 (M) 中去除值或结果。

示例：从独立存储器 (M) 中去除 3×2 的结果

LINE

3 **X** **2** **SHIFT** **M+** (M-)



注意

当显示器上显示计算结果时，按 **M+** 或 **SHIFT** **M+** (M-) 会将结果添加到独立存储器中或从独立存储器中去除结果。

重要事项！

当在计算结束时，不按 **EXE** 而是按 **M+** 或 **SHIFT** **M+** (M-) 时，显示器上显示的值是计算的结果（该结果会添加到独立存储器中或从独立存储器中去除）。它不是独立存储器中的当前内容。

查看独立存储器内容

按 **RCL** **9** (M)。

清除独立存储器内容（清除为 0）

0 **SHIFT** **RCL** (STO) **9** (M)

使用独立存储器的计算示例

在执行以下操作之前，按 **0** **SHIFT** **RCL** (STO) **9** (M) 清除独立存储器内容。

示例： $23 + 9 = 32$

2 **3** **+** **9** **M+**

$53 - 6 = 47$

5 **3** **-** **6** **M+**

—) $45 \times 2 = 90$

4 **5** **X** **2** **SHIFT** **M+** (M-)

$99 \div 3 = 33$

9 **9** **÷** **3** **M+**

—————
(共计) 22

RCL **9** (M)

↓
(调用 M 的值)。

使用变量

本计算器支持使用从 A 到 Z 命名的 26 个变量。

向变量指定值或计算结果

使用如下过程，向变量指定值或计算表达式。

示例：向变量 A 指定 $3 + 5$

3 **+** **5** **SHIFT** **RCL** (STO) **i** (A)

❑ 查看指定给变量的值

查看指定给变量的值，按 **RCL**，然后指定变量名称。还可以按 **ALPHA**，再指定变量名称，然后按 **EXE**。

示例：查看指定给变量 A 的值

RCL **i** (A) 或 **ALPHA** **i** (A) **EXE**

❑ 在计算中使用变量

可以如同使用值一样，在计算中使用变量。

示例：计算 $5 + A$

5 **+** **ALPHA** **i** (A) **EXE**

❑ 清除指定给变量的值（清除为 0）

示例：清除变量 A

0 **SHIFT** **RCL** (STO) **i** (A)

❑ 清除所有变量（清除为 0）

使用 MEMORY 模式屏幕清除所有变量的内容。请参照第 123 页的“存储器管理器 (MEMORY)”以获取更多信息。

■ 清除所有存储器内容

当希望将所有变量（包括变量 M）和答案存储器 (Ans) 清除为 0 时，执行以下操作。

FUNCTION – {CLR} – {Memory} **EXE**

保留可变存储器

如果发现计算器的缺省变量 (A 到 Z) 不够用，可以保留可变存储器，并创建“额外变量”来存储值。

指定或调用额外变量的值时，额外变量会类似于名为“Z”阵列的阵列变量。额外变量名称由字母“Z”和字母“Z”后跟的由方括号括起的值组成，例如 Z[5]。

■ 用户存储区域

本计算器拥有 28500 字节的用户存储区域，可以用来保留可变存储器和添加额外变量。

重要事项！

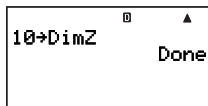
- 可以执行在 COMP 模式或 COMP 模式程序中，保留可变存储器的过程。本节中的所有示例操作都在 COMP 模式 (**MODE** **1**) 下执行。
- 28500 字节的用户存储器可用于存储额外变量和程序。这意味着增加额外变量数，会减少可用于存储程序的存储器容量。因此，在存储器中存储程序，也会减少可用于存储额外变量的存储器容量。

❑ 添加额外变量

示例：增加 10 个变量

LINE

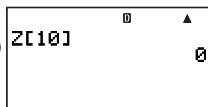
[1] [0] [FUNCTION] - {PROG} - {→} [SHIFT] [•] (Dim Z) [EXE]



- 当显示器上显示“Done”时，表示已添加了指定数量的额外变量。此时，会向所有额外变量赋予 0。

(检查 Z[10] 的值)

[ALPHA] [In] (I) [1] [0] [ALPHA] [x10⁹] (Z) [AC/ON] [ALPHA] [x10⁹] (Z) [ALPHA] [x⁹] (I) [EXE]



注意

保留可变存储器会至少占用 26 个字节，每个添加的额外变量会另外占用 12 个字节。注意：存储复数的额外变量会占用 22 个字节。例如，如上所述地添加 10 个额外变量，将占用 $26 + (12 \times 10) = 146$ 字节的用户存储区域。用户存储器具有 28500 字节的总容量，因此可以添加的额外变量数的限制为 2372 个（假设未向额外变量指定任何复数的话）。

■ 使用额外变量

创建额外变量后，可以象操作缺省变量（从 A 到 Z）一样向它们指定值并将其插入到计算中。请记住：额外变量名称由字母“Z”和字母“Z”后跟的由方括号括起的值组成，例如 Z[5]。

注意

- 可以忽略额外变量名称的后括号（】）。
- 代替额外变量名称的方括号内的值，可以使用计算表达式或缺省阵列名称（A 到 Z）。
- 注意：额外变量名称方括号中的值必须在范围 1 到添加的额外变量数之间。试图使用超过额外变量数的值将错误。

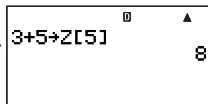
❑ 向额外变量指定值或计算结果

可以使用以下命令句法，向额外变量指定值：{ 值或表达式 } → Z[额外变量值] [EXE]。

示例：向变量 Z[5] 指定 3 + 5

LINE

[3] [+] [5] [FUNCTION] - {PROG} - {→} [ALPHA] [x10⁹] (Z) [ALPHA] [In] (I) [5] [ALPHA] [x⁹] (I) [EXE]



重要事项！

可以在 COMP 模式或 COMP 模式程序中，向额外变量写入数据。

▣ 调用额外变量的内容

输入希望调用内容的额外变量的名称 (Z[n])，然后按 **EXE**。

示例：调用额外变量 Z[5] 的内容

LINE

[ALPHA] **[x10⁻³]** **(Z)** **[ALPHA]** **[In]** **(I)** **[5]** **[ALPHA]** **[x^a]** **(I)** **EXE**

0 ▲
Z[5]
8

▣ 在计算中使用额外变量

可以如同使用值一样，在计算中使用额外变量。

示例：计算 $5 + Z[5]$

LINE

[5] **+** **[ALPHA]** **[x10⁻³]** **(Z)** **[ALPHA]** **[In]** **(I)** **[5]** **[ALPHA]** **[x^a]** **(I)** **EXE**

0 ▲
 $5 + Z[5]$
13

▣ 清除额外变量内容（清除为 0）

示例：清除额外变量 Z[5]

0 **FUNCTION** - {PROG} - {→} **[ALPHA]** **[x10⁻³]** **(Z)** **[ALPHA]** **[In]** **(I)** **[5]** **[ALPHA]** **[x^a]** **(I)** **EXE**

▣ 清除所有额外变量

当希望删除当前位于计算器存储器中的所有额外变量时，执行以下操作。

0 **FUNCTION** - {PROG} - {→} **[SHIFT]** **[•]** (Dim Z) **EXE**

注意

还可以使用 MEMORY 模式屏幕删除所有额外变量。请参照第 123 页的“存储器管理器 (MEMORY)”以获取更多信息。

使用 π 和科技常数

■ Pi (π)

本计算器支持在计算中输入 pi (π)。在所有模式中都支持 Pi (π)，但 BASE-N 模式除外。以下是计算器用于 π 的值。

$\pi = 3.14159265358980$ (**[SHIFT]** **[x10⁹]** (π))

科技常数

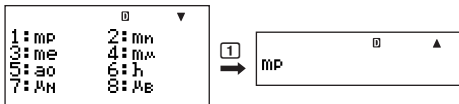
本计算器内置了 40 个常用科技常数，如 π ，每个科技常数都具有唯一的显示符号。在所有模式中都支持科技常数，但 BASE-N 模式除外。

输入科技常数

1. 按 **FUNCTION** 显示功能菜单。
2. 在菜单中，选择“CONST”。
- 此操作将显示科技常数菜单的第 1 页。

1:mp	2:mn
3:me	4:m $\wedge$
5:ao	6:h
7: μ H	8: μ B

- 有 5 个科技命令菜单屏幕，可以使用 $\blacktriangledown$ 和 $\blacktriangle$ 浏览这些屏幕。有关科技常数的更多信息，请参照第 38 页上的“科技常数列表”。
3. 可使用 $\blacktriangledown$ 和 $\blacktriangle$ 滚动页面和显示包含所需科技常数的页面。
 4. 按与希望选择的科技常数相对应的数字键（从 **1** 到 **8**）。
 - 此操作将输入与所按数字键对应的科技常数符号。



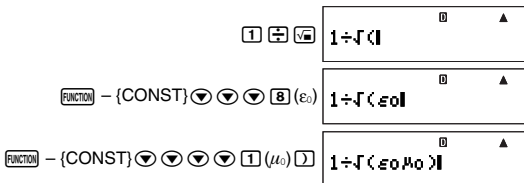
- 此时按 **EXE** 显示当前显示在屏幕上的符号的科技常数的值。

mp
1.67262171 $\times 10^{-27}$

使用科技常数的计算示例

示例：计算真空中的光速的常数（ $c_0 = 1/\sqrt{\epsilon_0 \mu_0}$ ）

LINE



科技常数列表

显示正确的菜单页面时，“页号”列中的数字会在左侧显示常数菜单页号和选择常数所需按的数字键。

页号	科技常数	页号	科技常数
1-1	质子的质量	3-5	μ 介子磁矩
1-2	中子的质量	3-6	法拉第常数
1-3	电子的质量	3-7	元电荷
1-4	μ 介子的质量	3-8	阿伏加德罗常数
1-5	玻尔半径	4-1	玻尔兹曼常数
1-6	普朗克常数	4-2	理想气体的摩尔体积
1-7	核磁子	4-3	摩尔气体常数
1-8	玻尔磁子	4-4	真空中的光速
2-1	合理化普朗克常数	4-5	第一放射常数
2-2	精细结构常数	4-6	第二放射常数
2-3	经典电子半径	4-7	斯蒂芬 -- 玻尔兹曼常数
2-4	康普顿波长	4-8	电性常数
2-5	质子的回磁比	5-1	磁性常数
2-6	质子的康普顿波长	5-2	磁通量
2-7	中子的康普顿波长	5-3	重力加速度
2-8	里德伯常数	5-4	电导量
3-1	原子质量常数	5-5	真空的特性阻抗
3-2	质子磁矩	5-6	摄氏温度
3-3	电子磁矩	5-7	万有引力常数
3-4	中子磁矩	5-8	标准大气压

- 这些值基于 CODATA 推荐值 (2000)。有关详细信息，请参照单独补充资料中的 <#01>。

科技函数计算

除非另有说明，否则可以在计算器任意计算模式下使用本节中的函数，但 BASE-N 模式除外。

科技函数计算注意事项

- 执行包含内置科技函数的计算时，显示计算结果可能会需要一些时间。在计算结果显示之前，不要在计算器上执行任何键操作。
- 中断正在进行的计算操作，按 **AC/ON**。

说明科技函数语法

- 表示函数自变量的文本位于大括号 ({}) 内。自变量通常为 { 值 } 或 { 表达式 }。
- 当大括号 ({}) 位于括号内时，表示括号所有项均需输入。

三角和反三角函数

$\sin(, \cos(, \tan(, \sin^{-1}(, \cos^{-1}(, \tan^{-1}($

句法和输入

$\sin(\{n\})$ (可在自变量中使用其它函数。)

示例: $\sin 30 = 0.5, \sin^{-1} 0.5 = 30$

LINE **Deg**

sin **3** **0** **)** **EXE**

sin(30)
0.5

SHIFT **sin** (**sin**⁻¹) **0** **.** **5** **)** **EXE**

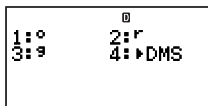
sin⁻¹(0.5)
30

备注

需要在计算中使用的角度单位，是当前选定的缺省角度单位。

■ 角度单位变换

可将一个角度单位输入的值变换为另外一种角度单位。
输入一个值之后，选择 **FUNCTION** - {ANGLE} 可显示如下菜单屏幕。

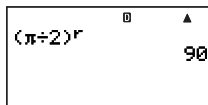


- 1 (°): 度
- 2 (′): 弧度
- 3 (″): 梯度

示例：将 $\frac{\pi}{2}$ 弧度变换为度

LINE **Deg**

() **SHIFT** **[x10⁻¹] (π)** **÷** **2** **)**
FUNCTION - {ANGLE} **2** (′) **EXE**



■ 双曲线和反双曲线函数

$\sinh$ (), $\cosh$ (), $\tanh$ (), $\sinh^{-1}$ (), $\cosh^{-1}$ (), $\tanh^{-1}$ ()

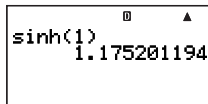
◆ 句法和输入

$\sinh(\{n\})$ (可在自变量中使用其它函数。)

示例： $\sinh 1 = 1.175201194$

LINE

FUNCTION - {MATH} **▼** **▼** **1** ($\sinh$) **1** **)**



◆ 备注

输入双曲线或反双曲线函数，执行以下操作显示功能菜单：**FUNCTION** - {MATH} **▼** **▼**。

■ 指数函数和对数函数

$10^{\wedge}$ (), $e^{\wedge}$ (), $\log$ (), $\ln$ ()

◆ 句法和输入

$10^{\wedge}(\{n\})$	$10^{\{n\}}$	(与 $e^{\wedge}$ 相同)
$\log(\{n\})$	$\log_{10}\{n\}$	(常用对数)
$\log(\{m\},\{n\})$	$\log_{(m)}\{n\}$	(底数为 $\{m\}$ 的对数)
$\ln(\{n\})$	$\log_e\{n\}$	(自然对数)

示例 1: $\log_2 16 = 4$, $\log 16 = 1.204119983$

LINE

$\log$ 2 $\rightarrow$ 16 $\rightarrow$ EXE

$\log(2,16)$
4

$\log$ 16 $\rightarrow$ EXE

$\log(16)$
1.204119983

当不指定底数时，会假定使用底数 10（常用对数）。

MATH

FUNCTION $\rightarrow$ $\{\text{MATH}\}$ $\rightarrow$ 7 (logab)
 2 $\rightarrow$ 16 $\rightarrow$ EXE

$\log_2(16)$
4

示例 2: $\ln 90 (= \log_e 90) = 4.49980967$

LINE

$\ln$ 90 $\rightarrow$ EXE

$\ln(90)$
4.49980967

■ 幂函数和乘方根函数

$x^2, x^{-1}, \wedge, \sqrt{\quad}, \sqrt[n]{\quad}, {}^x\sqrt{\quad}$

◆ 句法和输入

$\{n\} x^2$	$\{n\}^2$	(平方)
$\{n\} x^{-1}$	$\{n\}^{-1}$	(倒数)
$\{m\} \wedge \{n\}$	$\{m\}^{\{n\}}$	(乘方)
$\sqrt{\{n\}}$	$\sqrt{\{n\}}$	(平方根)
$\sqrt[3]{\{n\}}$	$\sqrt[3]{\{n\}}$	(立方根)
$\{m\} \sqrt[n]{\{n\}}$	$\{m\} \sqrt[n]{\{n\}}$	(乘方根)

示例 1: $(\sqrt{2} + 1)(\sqrt{2} - 1) = 1$, $(1 + 1)^{2+2} = 16$

LINE

$($ $\sqrt{\quad}$ 2 $)$ $+$ 1 $)$
 $($ $\sqrt{\quad}$ 2 $)$ $-$ 1 $)$ EXE

$(\sqrt{2}+1)(\sqrt{2}-1)$
1

((1 + 1) x^2 + 2) EXE

(1+1)^(2+2) 16

MATH

((√ 2) + 1)
((√ 2) - 1) EXE

(√2+1)(√2-1) 1

((1 + 1) x^2 + 2) EXE

(1+1)^2+2 16

示例 2: $(-2)^{\frac{2}{3}} = 1.587401052$

LINE

((-) 2) x^ (2) (3) EXE

(-2)^(2/3) 1.587401052

积分计算

本计算器可使用高斯近似积分执行积分运算。计算器使用以下积分函数。

$\int$

句法和输入

$\int (f(x), a, b, tol)$

$f(x)$: x 的函数 (输入变量 x 使用的函数。)

- 除 x 之外的所有变量都被视为常数。

a : 积分区域下限

b : 积分区域上限

tol : 误差允许公差范围 (仅当使用线性显示时才能输入。)

- 此参数可以忽略。在这种情况下, 使用公差 1×10^{-5} 。

示例: $\int (\ln(x), 1, e) = 1$ (tol 值未输入)

MATH

FUNCTION - {MATH} 1 (∫dX)
ln ALPHA 0 (X)) (1) SHIFT ln (e^) 1 EXE

$\int_1^{e^1} \ln(X) dX$ 1

$\boxed{\ln}$ $\boxed{\alpha}$ $\boxed{0}$ $\boxed{(X)}$ $\boxed{)}$ $\boxed{\rightarrow}$ $\boxed{1}$ $\boxed{\rightarrow}$ $\boxed{\text{SHIFT}}$ $\boxed{\ln}$ $\boxed{(e^{\Delta})}$ $\boxed{1}$ $\boxed{)}$ $\boxed{\text{EXE}}$

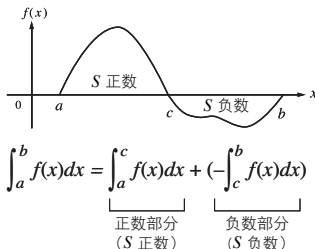
$\boxed{\int}$ $\boxed{(\ln(X), 1, e^{\Delta}(1))}$ $\boxed{\Delta}$
 1

◆ 备注

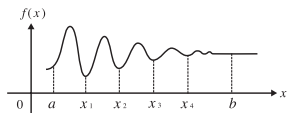
- 仅在 COMP、SD、REG 和 EQN 模式中支持使用 $\int$ 。
- 对 $f(x)$ 、 a 、 b 和 tol 参数不能输入以下函数： $\int$ 、 d/dx 、 d^2/dx^2 、 Σ 。此外，对 $f(x)$ 参数，不能输入 Pol、Rec 函数和随机数函数。
- 当积分区域限值在 $a \leq x \leq b$ 范围内且 $f(x) < 0$ 时，积分结果将为负数。
 示例： $\int(0.5x^2 - 2, -2, 2) = -5.333333333$
- 在三角函数积分的情况下，请选择 Rad 作为角度单位。
- 积分计算完成可能需要很长时间。
- 对 tol 参数指定较小的值可以提高精度，但也会使计算花更多时间。请指定大于 1×10^{-14} 的 tol 值。
- 使用普通显示时将无法输入 tol 值。
- 正在积分的函数类型、积分区域内的正值和负值以及正在使用的积分区域都可以对积分值产生很大的误差和错误。
- 可以通过按 $\boxed{\text{AC/ON}}$ 中断正在进行的积分计算运作。

◆ 成功积分计算技巧

- 对于周期函数以及因使用的积分区域而产生的正负 $f(x)$ 值
 → 将积分按周期分为若干部分（或正数部分和负数部分），得出其中每一个的值，然后将这些值相加。



- 对于因积分区域频繁变动而波动很大的积分值
 → 将积分区间分为多个部分（将波动很大的区域分为若干小部分），对每个部分执行积分，然后合并结果。



$$\int_a^b f(x)dx = \int_a^{x_1} f(x)dx + \int_{x_1}^{x_2} f(x)dx + \dots + \int_{x_4}^b f(x)dx$$

■ 微分

本计算器可通过基于中心差分近似的近似微分法而执行微分计算。使用如下函数执行计算。

$$d/dx($$

▣ 句法和输入

$$d/dx(f(x), a, tol)$$

$f(x)$: x 的函数（输入变量 x 使用的函数。）

- 除 x 之外的所有变量都被视为常数。

a : 所需微分系数的点值（微分点）

tol : 误差允许公差范围（仅当使用线性显示时才能输入。）

- 此参数可以忽略。在这种情况下，使用公差 1×10^{-10} 。

示例： 得出点 $x = \frac{\pi}{2}$ 处函数 $y = \sin(x)$ 的微分系数（ tol 值未输入）

Rad **FUNCTION** $\rightarrow$ {MATH} **2** (d/dX) **SIN** **ALPHA** **0** (X) **□**①

MATH

(从上面的 ① 继续)

▶ **□** **SHIFT** **x10⁻⁹** (π) **▼** **2** **EXE**

LINE

(从上面的 ① 继续)

▶ **SHIFT** **x10⁻⁹** (π) **□** **2** **□** **EXE**

▣ 备注

- 仅在 COMP、SD、REG 和 EQN 模式中支持使用 $d/dx()$ 。
- 对 $f(x)$ 、 a 和 tol 参数不能输入以下函数: $\int$ 、 d/dx 、 d^2/dx^2 、 Σ 。此外，对 $f(x)$ 参数，不能输入 Pol(、Rec(和随机数函数。
- 在三角函数微分的情况下，请选择 Rad 作为角度单位。
- 对 tol 参数指定较小的值可以提高精度，但也会使计算花更多时间。请指定大于 1×10^{-14} 的 tol 值。
- 使用普通显示时将无法输入 tol 值。
- 非连续点、突变波动、极大或极小点、拐点以及包含不能微分的点，以及趋近 0 的微分点或微分计算结果可能会精确度很差或错误。

- 可以通过按 **AC/ON** 中断正在进行的微分计算运作。

■ 二阶微分

本计算器可以计算 $f(x)$ 的二阶微分系数 ($d^2/dx^2(f(x))|_{x=a}$)，其中 $x = a$ 。本计算器使用基于牛顿插值多项式的二阶微分方程的近似。使用如下函数执行计算。

$$d^2/dx^2($$

▣ 句法和输入

$$d^2/dx^2(f(x), a, tol)$$

$f(x)$: x 的函数 (输入变量 x 使用的函数。)

- 除 x 之外的所有变量均被视为常数。

a : 所需二阶微分系数的点值 (二阶微分点)

tol : 误差允许公差范围 (仅当使用线性显示时才能输入。)

- 此参数可以忽略。在这种情况下，使用公差 1×10^{-10} 。

示例 1: 得出 $x = 3$ 时函数 $y = x^3 + 4x^2 + x - 6$ 的二阶微分系数

MATH

$$\text{FUNCTION} \rightarrow \{\text{MATH}\} \rightarrow 3 \rightarrow (d^2/dx^2) \rightarrow \text{ALPHA} \rightarrow 0 \rightarrow (X) \rightarrow x^3 \rightarrow 3 \rightarrow \text{EXE}$$

$$\frac{d^2}{dx^2}(x^3+4x^2+x-6) \Big|_{x=3} = 26$$

示例 2: 执行与示例 1 相同的过程，其中指定 $tol = 1 \times 10^{-12}$

鉴于要为 tol 指定一个值，故需使用线性显示执行此计算。

LINE

$$\text{FUNCTION} \rightarrow \{\text{MATH}\} \rightarrow 3 \rightarrow (d^2/dx^2) \rightarrow \text{ALPHA} \rightarrow 0 \rightarrow (X) \rightarrow x^3 \rightarrow 3 \rightarrow + \rightarrow 4 \rightarrow \text{ALPHA} \rightarrow 0 \rightarrow (X) \rightarrow x^2 \rightarrow + \rightarrow 1 \rightarrow (X) \rightarrow x \rightarrow - \rightarrow 6 \rightarrow \text{EXE}$$

$$\frac{d^2}{dx^2}(x^3+4x^2+x-6, 3, 1 \times 10^{-12}) = 26$$

▣ 备注

请参照第 44 页有关微分的备注。

■ Σ 计算

此函数确定指定范围的输入 $f(x)$ 的和。使用如下函数执行计算。

$$\Sigma($$

下面显示为用于 Σ 计算的计算公式。

$$\Sigma(f(x), x, a, b) = f(a) + f(a+1) + \dots + f(b)$$

❑ 句法和输入

$\Sigma(f(x), x, a, b)$

$f(x)$: x 的函数 (参数变量如下指定)

x : 参数变量 (从 A 到 Z 的任何字母)

- 如果在此处指定的变量名称与 x 的函数中使用的变量名称不匹配, 则把函数中的变量视为常数。

a : 计算范围的始点

b : 计算范围的终点

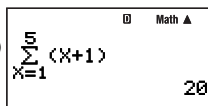
- a 和 b 是范围 $-1 \times 10^{10} < a \leq b < 1 \times 10^{10}$ 中的整数。

- 此计算的阶次固定为 1。

示例: $\Sigma(X+1, X, 1, 5) = 20$

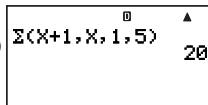
MATH

FUNCTION — {MATH} 4 (Σ) ALPHA 0 (X)
+ 1 ► ALPHA 0 (X) ► 1 ► 5 EXE



LINE

FUNCTION — {MATH} 4 (Σ) ALPHA 0 (X)
+ 1 ► ALPHA 0 (X) ► 1 ► 5) EXE



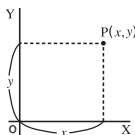
❑ 备注

- 仅在 COMP、SD、REG 和 EQN 模式中支持使用 Σ 。
- 对 $f(x)$ 、 a 和 b 参数不能输入以下函数: $\int$ 、 d/dx 、 d^2/dx^2 、 Σ 。此外, 对 $f(x)$ 参数, 不能输入 Pol、Rec 函数和随机数函数。
- 可以通过按 **AC/ON** 键中断正在进行的 Σ 计算。

■ 坐标变换 (直角坐标 ↔ 极坐标)

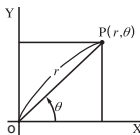
Pol, Rec

本计算器可以在直角坐标和极坐标之间变换。



直角坐标 (Rec)

Pol
Rec



极坐标 (Pol)

◆ 句法和输入

直角坐标变换为极坐标 (Pol)

Pol(x , y)

x : 直角坐标的 x 值

y : 直角坐标的 y 值

极坐标变换为直角坐标 (Rec)

Rec(r , θ)

r : 极坐标的 r 值

θ : 极坐标的 θ 值

示例 1: 将直角坐标 ($\sqrt{2}$, $\sqrt{2}$) 变换为极坐标

LINE **Deg**

SHIFT + (Pol) $\sqrt{\square}$ 2)
2 $\sqrt{\square}$ 2)) EXE

Pol($\sqrt{2}$, $\sqrt{2}$)
r= 2
 θ = 45

MATH **Deg**

SHIFT + (Pol) $\sqrt{\square}$ 2)
2 $\sqrt{\square}$ 2)) EXE

Pol($\sqrt{2}$, $\sqrt{2}$)
r=2, θ =45

示例 2: 将极坐标 (2, 30°) 变换为直角坐标

LINE **Deg**

SHIFT = (Rec) 2)
3 0)) EXE

Rec(2, 30)
x= 1.732050808
y= 1

◆ 备注

- 这些函数可在 COMP 模式中使用。
- 计算生成的 r 值或 x 值赋予变量 I，而 θ 值或 y 值赋予变量 J（第 33 页）。
- 从直角坐标变换为极坐标时得出的 θ 值在范围 $-180^\circ < \theta \leq 180^\circ$ 内。
- 当在计算表达式内执行坐标变换功能时，会使用变换产生的第一个坐标（ r 值或 x 值）执行计算。

示例: $\text{Pol}(\sqrt{2}, \sqrt{2}) + 5 = 2 + 5 = 7$

■ 随机数函数

本计算器备有生成 10 位无序随机数、10 位有序随机数或特定范围内的随机整数的函数。下面是生成随机数的函数。

Ran#, RanInt#(

▣ 无序随机数（小数值）

下面生成范围在 0 和 1 之间的 10 位无序小数。

句法: Ran#

示例: 生成 10 位无序随机数

MATH

FUNCTION - {**MATH**} **6** (Ran#) **EXE**

Ran#
0.6196899081

EXE

Ran#
0.7042860201

EXE

Ran#
0.9671390699

上述值仅作为示例提供。计算器使用此函数生成的实际值将会有所不同。

▣ 有序随机数（小数值）

在这种情况下，会按照 1 到 9 之间的 9 个序数生成范围在 0 到 1 之间的 10 位有序随机数。顺序是由 Ran# 的整数自变量（1 到 9）指定的。按照该自变量生成的随机数以固定顺序生成。

句法: Ran#{*n*} (*n* 是 1 到 9 之间的整数。)

示例: 生成有序随机数

MATH

初始化顺序:

FUNCTION - {**MATH**} **6** (Ran#) **0** **EXE**

Ran# 0
0

FUNCTION - {**MATH**} **6** (Ran#) **1** **EXE**

Ran# 1
0.1450073626

Math ▲
Ran# 1
0.1387437776

EXE

Math ▲
Ran# 1
0.866670424

EXE

▣ 随机整数

此函数生成特定范围内的随机整数。

句法: $\text{RanInt}\#(\{m\}, \{n\})$ (m 和 n 是整数。 $m < n$; $|m|, |n| < 1\text{E}10$; $n - m < 1\text{E}10$)

示例: 生成范围在 0 到 5 之间的随机整数

MATH

FUNCTION {MATH} (RanInt) 0 5 EXE

Math ▲
RanInt#(0,5)
2

Math ▲
RanInt#(0,5)
4

EXE

Math ▲
RanInt#(0,5)
0

EXE

上述值仅作为示例提供。计算器使用此函数生成的实际值将会有所不同。

■ 其它函数

$x!$, Abs, nPr , nCr , Rnd, Int, Frac, Intg

▣ 阶乘 (!)

句法: $\{n\}!$ ($\{n\}$ 必须是自然数或 0。)

示例: $(5 + 3)!$

LINE

(5 + 3)
FUNCTION {MATH} 5 (X!) EXE

Math ▲
(5+3)!
40320

◆ 绝对值 (Abs)

句法: $\text{Abs}\{n\}$

示例: $\text{Abs}(2 - 7) = 5$

MATH

FUNCTION - {MATH} ∇ 1 (Abs) 2 - 7 **EXE**

0 Math ▲
|2-7|
5

LINE

FUNCTION - {MATH} ∇ 1 (Abs) 2 - 7) **EXE**

0 ▲
Abs(2-7)
5

◆ 排列 (nPr)/ 组合 (nCr)

句法: $\{n\}P\{m\}$, $\{n\}C\{m\}$

示例: 10 人组成的小组中可能出现多少种 4 人排列和组合?

LINE

1 0 **FUNCTION** - {MATH} 7 (nPr) 4 **EXE**

0 ▲
10P4
5040

1 0 **FUNCTION** - {MATH} 8 (nCr) 4 **EXE**

0 ▲
10C4
210

◆ 舍入函数 (Rnd)

可以使用舍入函数 (Rnd) 对值表达式或自变量指定的计算结果进行舍入。按照显示位数设定执行舍入, 舍入到有效位数。

显示位数设定: Norm1 或 Norm2

尾数四舍五入到 10 位。

显示位数设定: Fix 或 Sci

值四舍五入为指定的位数。

示例: $200 \div 7 \times 14 = 400$

LINE

2 0 0 $\div$ 7 $\times$ 1 4 **EXE**

0 ▲
200÷7×14
400

(3 位小数)

SHIFT **MODE** (SETUP) **6** (Fix) **3** **EXE**

0 **FIX** **▲**
200÷7×14
400
400.000

(内部计算使用 15 位。)

2 **0** **0** **÷** **7** **EXE**

0 **FIX** **▲**
200÷7
400.000
28.571

× **1** **4** **EXE**

0 **FIX** **▲**
200÷7
Ans×14
28.571
400.000

现在使用舍入 (Rnd) 函数执行同一计算。

AC/ON **2** **0** **0** **÷** **7** **EXE**

0 **FIX** **▲**
200÷7
28.571

(计算使用四舍五入后的值。)

SHIFT **0** (Rnd) **EXE**

0 **FIX** **▲**
200÷7
Rnd(Ans)
28.571
28.571

(四舍五入后的结果)

× **1** **4** **EXE**

0 **FIX** **▲**
Rnd(Ans)
Ans×14
28.571
399.994

▣ 抽取整数部分 (Int)

Int(函数抽取输入实数的整数部分作为其自变量。

句法: Int({n})

示例: 提取 -1.5 的整数部分

LINE

FUNCTION **-** {MATH} **▼** **2** (Int) **(←)** **1** **.** **5** **)** **EXE**

0 **▲**
Int(-1.5)
-1

▣ 提取小数部分 (Frac)

Frac(函数抽取输入实数的小数部分作为其自变量。

句法: $\text{Frac}(\{n\})$

示例: 提取 -1.5 的小数部分

LINE

FUNCTION - {MATH} ∇ 3 (Frac) $\leftarrow$ 1 $\cdot$ 5 $\rightarrow$ EXE

$\text{Frac}(-1.5)$
-0.5

▣ 最大整数 (Intg)

Intg(函数确定不超过输入实数的最大整数作为其自变量。

句法: $\text{Intg}(\{n\})$

示例: 确定不超过 -1.5 的最大整数

LINE

FUNCTION - {MATH} ∇ 4 (Intg) $\leftarrow$ 1 $\cdot$ 5 $\rightarrow$ EXE

$\text{Intg}(-1.5)$
-2

使用工程符号

■ 使用 10^3 工程符号 (ENG)

工程符号 (ENG) 将数量表示为一个 1 和 10 之间的正数和 10 的乘方 (始终为 3 的倍数) 的乘积。有两个函数可以用于将数字变换为工程符号: $\text{ENG} \rightarrow$ 和 $\text{ENG} \leftarrow$ 。

函数	键操作
$\text{ENG} \rightarrow$	SHIFT $\frac{\square}{\square}$ (ENG)
$\text{ENG} \leftarrow$	SHIFT $\frac{\square}{\square}$ ($\overline{\text{ENG}}$)

■ ENG 变换示例

示例 1: 使用 $\text{ENG} \rightarrow$ 将 1234 变换为工程符号

MATH

1 2 3 4 EXE

1234
1234

SHIFT $\frac{\square}{\square}$ (ENG)

1234
1.234×10³

SHIFT $\frac{\square}{\square}$ (ENG)

1234
1234×10⁰

示例 2：使用 ENG $\leftarrow$ 将 123 变换为工程符号

MATH

1 2 3 EXE

123
123

SHIFT $\overline{\square}$ (ENG)

123
0.123×10³

SHIFT $\overline{\square}$ (ENG)

123
0.000123×10⁶

■ 使用工程符号

本计算器支持使用 10 个工程符号 (m、 μ 、n、p、f、k、M、G、T、P)，可以使用它们输入值或显示计算结果。可以在除 BASE-N 模式之外的所有计算模式的计算中，使用工程符号。

▣ 使用工程符号显示计算结果

使用计算器的设定屏幕，选中“EngOn”以支持工程符号（第 12 页）。

▣ 使用工程符号输入值

示例：输入 500 k

5 0 0

500

FUNCTION - {MATH} ∇ ∇ ∇

1:m	2: μ	3:n
4:p	5:f	6:k
7:M	8:G	9:T
0:P		

6 (k) 500k

在工程符号打开的情况下 (EngOn)，使用 ENG (10³) 变换

选中“EngOn”以支持工程符号的情况下，执行 ENG 变换（第 12 页）将把小数点向右移动 3 位，并相应变更工程符号（例如，从 M 变更为 k）。相反，执行 $\leftarrow$ ENG 变换将把小数点向左移动 3 位，并相应变更工程符号（例如，从 k 变更为 M）。

示例： 计算 $999 \text{ k (kilo)} + 25 \text{ k (kilo)} = 1.024 \text{ M (Mega)} = 1024 \text{ (kilo)}$

LINE

9 9 9 FUNCTION - {MATH} 6 (k) +
2 5 FUNCTION - {MATH} 6 (k) EXE

ENG 0 ▲
999k+25k
1.024M

(ENG 变换)

SHIFT $\div$ (ENG)

ENG 0 ▲
999k+25k
1024k

复数计算 (COMP)

为执行本节中的示例操作，首先选择 COMP (MODE 1) 作为计算模式。

输入复数

输入虚数 (i)

使用 i 键输入虚数 i 。

示例： 输入 $2 + 3i$

2 + 3 i

0 Math ▲
2+3i

使用极坐标格式输入复数值

也可以使用极坐标格式 ($r \angle \theta$) 输入复数。

示例： 输入 $5 \angle 30$

5 SHIFT i ($\angle$) 3 0

0 Math ▲
5 $\angle$ 30

重要事项！

当输入自变量 θ 时，输入表示角度的值（按照计算器当前的角度单位设定）。

■ 复数显示设定

请参照“复数显示格式”(第 13 页)。

■ 复数计算结果显示示例

◆ 直角坐标格式 ($a+bi$)

SHIFT **MODE** (SETUP) **▼** **4** (COMPLX) **1** ($a+bi$)

示例 1: $2 \times (\sqrt{3} + i) = 2\sqrt{3} + 2i = 3.464101615 + 2i$

MATH **Deg**

2 **×** **(** **√** **3** **)** **+** **i** **)** **EXE**

Math ▲
 $2 \times (\sqrt{3} + i)$
 $2\sqrt{3} + 2i$

LINE

2 **×** **(** **√** **3** **)** **+** **i** **)** **EXE**

Math ▲
 $2 \times (\sqrt{3} + i)$
3.464101615
+2i

使用线性显示格式时，计算结果会以两行显示，分别显示实部和虚部。

示例 2: $\sqrt{2} \angle 45 = 1 + i$

MATH **Deg**

√ **2** **▶** **SHIFT** **i** (**∠**) **4** **5** **EXE**

Math ▲
 $\sqrt{2} \angle 45$
 $1 + i$

◆ 极坐标格式 ($r \angle \theta$)

SHIFT **MODE** (SETUP) **▼** **4** (COMPLX) **2** ($r \angle \theta$)

示例 1: $2 \times (\sqrt{3} + i) = 2\sqrt{3} + 2i = 4 \angle 30$

MATH **Deg**

2 **×** **(** **√** **3** **)** **+** **i** **)** **EXE**

Math ▲
 $2 \times (\sqrt{3} + i)$
 $4 \angle 30$

LINE

2 **×** **(** **√** **3** **)** **+** **i** **)** **EXE**

Math ▲
 $2 \times (\sqrt{3} + i)$
4
 $\angle 30$

使用线性显示格式时，计算结果会以两行显示，分别显示绝对值和自变量。

示例 2: $1 + i = \sqrt{2} \angle 45$

MATH **Deg**

1 **+** **i** **EXE**

Math ▲
 $1 + i$
 $\sqrt{2} \angle 45$

■ 共轭复数 (Conjg)

可以执行以下操作得出复数 $z = a + bi$ 的共轭复数 $\bar{z} = a - bi$ 。

示例：得出 $2 + 3i$ 的共轭复数

MATH

FUNCTION - {COMPLX} **3** (Conjg) **2** **+** **3** **i** **)** **EXE**

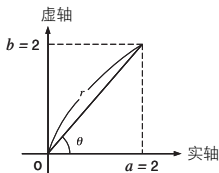
Math ▲
Conjg(2+3i)
2-3i

■ 绝对值和幅角 (Abs, Arg)

可以使用如下过程，得出一个复数的绝对值 ($|z|$) 和在高斯面上的幅角 ($\text{Arg}(z)$)，格式为 $z = a + bi$ 。

示例：得出 $2 + 2i$ 的绝对值和幅角

LINE Deg



绝对值：

FUNCTION - {COMPLX} **1** (Abs) **2** **+** **2** **i** **)** **EXE**

Math ▲
Abs(2+2i)
2.828427125

幅角：

FUNCTION - {COMPLX} **2** (Arg) **2** **+** **2** **i** **)** **EXE**

Math ▲
Arg(2+2i)
45

■ 提取复数的实部 (ReP) 和虚部 (ImP)

可以使用以下操作提取复数 $a + bi$ 的实部 (a) 或虚部 (b)。

示例：得出 $2 + 3i$ 的实部和虚部

MATH

FUNCTION - {COMPLX} **4** (ReP) **2** **+** **3** **i** **)** **EXE**

Math ▲
ReP(2+3i)
2

FUNCTION - {COMPLX} **5** (ImP) **2** **+** **3** **i** **)** **EXE**

Math ▲
ImP(2+3i)
3

■ 取代缺省复数显示格式

可以使用如下所述的过程，取代缺省复数显示格式，并为当前输入的计算指定特定的显示格式。

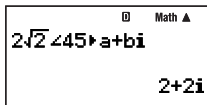
◆ 为计算指定直角坐标格式

在计算的结尾输入 $\text{FUNCTION} - \{\text{COMPLX}\} \text{7} (\blacktriangleright a+bi)$ 。

示例: $2\sqrt{2} \angle 45 = 2 + 2i$ (角度单位: Deg)

MATH Deg

$\text{2} \sqrt{\square} \text{2} \blacktriangleright \text{SHIFT} \text{7} (\angle) \text{4} \text{5}$
 $\text{FUNCTION} - \{\text{COMPLX}\} \text{7} (\blacktriangleright a+bi) \text{EXE}$



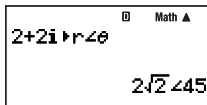
◆ 为计算指定极坐标格式

在计算的结尾输入 $\text{FUNCTION} - \{\text{COMPLX}\} \text{6} (\blacktriangleright r\angle\theta)$ 。

示例: $2 + 2i = 2\sqrt{2} \angle 45$

MATH Deg

$\text{2} \text{+} \text{2} \text{i} \text{FUNCTION} - \{\text{COMPLX}\} \text{6} (\blacktriangleright r\angle\theta) \text{EXE}$



矩阵计算 (COMP)

为执行本节中的示例操作，首先选择 COMP ($\text{MODE} \text{1}$) 作为计算模式。

■ 矩阵计算概述

执行矩阵计算的步骤是: 先将矩阵存储到名为 Mat A 到 Mat F 的六个存储区域中的一个，

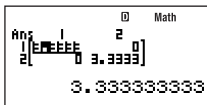
然后使用矩阵区域变量执行实际计算。例如，执行诸如 $\begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} + \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$ 的矩阵计算，则

要将 $\begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}$ 输入 Mat A，将 $\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$ 输入 Mat B，然后执行计算 Mat A + Mat B。

在一个 Mat Ans 屏幕上显示矩阵计算结果。

■ 关于 Mat Ans 屏幕

任何矩阵的计算结果都存储在名为“Mat Ans”的矩阵答案存储器中。可以使用 Mat Ans 屏幕查看当前存储在 Mat Ans 中的矩阵。



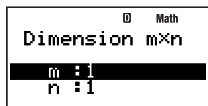
- Mat Ans 屏幕上的每个单元格都显示该单元格中当前存储的值至多 6 位数。
- 检查单元格中存储的值，请使用光标键将高亮显示移到该单元格。这将使包含的完整值在屏幕底部的值显示区域中显示。
- 如果一个单元格包含一个分数或 60 进制数值，则 Mat Ans 屏幕上的适用单元格将显示等值的 10 进制格式数值。当将高亮显示移到该单元格时，该值将以相应的分数或 60 进制格式在值显示区域中显示。
- 尽管 Mat Ans 屏幕看起来与矩阵编辑器屏幕非常相似，但不能在 Mat Ans 屏幕上编辑矩阵。
- Mat Ans 屏幕在显示器上时按 **EXE** 将切换到计算屏幕。
- Mat Ans 在显示器上时，可以按 **+**、**-** 或另一个运算符开始使用当前 Mat Ans 内容的计算（如“Mat Ans+”）。这类似于答案存储器的操作（第 31 页）。有关更多信息，请参照“执行矩阵计算”（第 60 页）。

■ 输入和编辑矩阵数据

最多可以输入 6 个矩阵的数据（名为 Mat A 到 Mat F），然后在计算中使用矩阵名称作为变量。有两种方法可以用于向矩阵输入数据：使用矩阵编辑器屏幕和使用赋值命令（ $\rightarrow$ ）。

◆ 使用矩阵编辑器屏幕输入矩阵数据

1. 按 **FUNCTION** - {MATRIX} **1** (EDIT) 显示矩阵存储区域菜单。
 - 已包含矩阵的存储区域将显示该矩阵的维数（如 2×2 ），而空的区域将指示“None”。
2. 使用 $\blacktriangledown$ 和 $\blacktriangle$ 将高亮显示移到要用于输入数据的矩阵。
3. 按 **EXE**。
 - 此操作将显示一个用于指定矩阵维数的屏幕。 m 是行数， n 是列数。



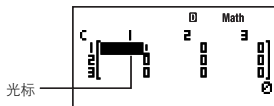
注意

如果在第 2 步中选择已包含矩阵的存储区域，则按 **EXE** 显示的矩阵编辑器屏幕可供编辑该矩阵。如果要删除现有矩阵并制作一个具有不同维数的矩阵，请在第 3 步中按 $\blacktriangleright$ 或 **FUNCTION** **1** (Dim) 而不是 **EXE**。

4. 指定要创建的矩阵的维数（最多 10 行和 10 列）。
 - 为指定行数，请将高亮显示移到 m ，输入一个值，然后按 **EXE**。此操作将使高亮显示移到 n 。
 - 为 n 输入一个值以指定列数，然后按 **EXE**。
 - 可以使用 $\blacktriangledown$ 和 $\blacktriangle$ 在 m 和 n 之间移动高亮显示。

5. 输入所需行值和列值之后，按 **EXE**。

- 此操作将显示矩阵编辑器屏幕。



6. 在矩阵编辑器屏幕上，向矩阵的单元格中输入值。

- 使用光标键将光标移到所需单元格，然后输入值。输入值之后，按 **EXE** 记录它。

7. 输入全部所需的值之后，按 **EXIT**。

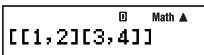
◆ 使用赋值命令 (→) 输入矩阵数据

1. 在 COMP 模式计算屏幕上，使用以下句法输入要输入矩阵存储器的矩阵。

$$\begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{bmatrix} \quad [[a_{11}, a_{12}, \dots, a_{1n}][a_{21}, a_{22}, \dots, a_{2n}] \dots [a_{m1}, a_{m2}, \dots, a_{mn}]]$$

- 例如，可以使用以下键操作完成输入 $\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$ 。

ALPHA **In** **()** **ALPHA** **In** **()** **1** **→** **2** **ALPHA** **x[□]** **()**
ALPHA **In** **()** **3** **→** **4** **ALPHA** **x[□]** **()** **ALPHA** **x[□]** **()**



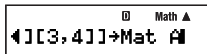
2. 输入赋值命令 (→)。

FUNCTION - {PROG} - {→}

3. 指定要存储矩阵的矩阵存储器 (Mat A 到 Mat F 或 Mat Ans)。

- 例如，将其存储到 Mat A，请执行以下键操作：

FUNCTION - {MATRIX} **2** (Mat) **ALPHA** **i** (A)。

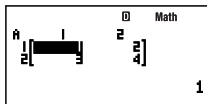


重要事项！

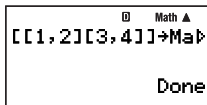
如果指定的是已包含矩阵数据的矩阵存储器，则将使用此处输入的数据取代现有数据。

4. 要存储该矩阵，按 **EXE**。

- 此操作在屏幕上显示所存储的矩阵。显示格式与矩阵编辑器屏幕的显示格式相同，但不能在此屏幕上编辑矩阵。



- 按 **EXIT** 将返回 COMP 模式计算屏幕。



注意

可以通过执行以下操作将矩阵答案存储器的内容指定为一个矩阵变量 (如 Mat A): Mat Ans→Mat A。

查看矩阵的内容

1. 在 COMP 模式计算屏幕上，按 **FUNCTION** - {MATRIX} **1** (EDIT) 显示矩阵存储区域菜单。
2. 使用 **▼** 和 **▲** 将高亮显示移到要查看其内容的矩阵，然后按 **EXE**。
3. 现在就可以编辑矩阵单元格的内容了（如果需要）。
 - 使用光标键将光标移到要变更其值的单元格，然后输入新值。输入值之后，按 **EXE** 记录。
4. 完成所需变更之后，按 **EXIT**。

删除特定矩阵存储区域的内容

1. 在 COMP 模式计算屏幕上，按 **FUNCTION** - {MATRIX} **1** (EDIT) 显示矩阵存储区域菜单。
2. 使用 **▼** 和 **▲** 将高亮显示移到要删除其内容的矩阵存储区。
3. 按 **DEL** 或 **FUNCTION** **2** (Del)。
 - 此操作将使一个确认信息出现。
4. 为删除计算器存储器中当前存储的所有程序，按 **EXE** (Yes)。要取消该操作而不删除任何内容，按 **EXIT** (No)。

执行矩阵计算

本节提供大量矩阵计算的实际示例。

- 执行矩阵计算之前，需要执行“输入和编辑矩阵数据”（第 58 页）中的过程，将数据输入计划在计算中使用的矩阵。
- 下表显示在本节中使用的矩阵名称表示法。当在过程中看到一个矩阵名称时，需要执行如下一种键操作。

当看到以下矩阵名称：	执行以下键操作：
【Mat A】	FUNCTION - {MATRIX} 2 (Mat) ALPHA i (A)
【Mat B】	FUNCTION - {MATRIX} 2 (Mat) ALPHA □ (B)
【Mat C】	FUNCTION - {MATRIX} 2 (Mat) ALPHA □□□ (C)

- 本节中的所有示例都使用普通显示执行。

加减矩阵

仅当矩阵维数相同时，才能进行加减。

示例： $\begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} + \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$

此示例假定 Mat A 包含 $\begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}$ ，Mat B 包含 $\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$ 。

【Mat A】 **+** 【Mat B】

Math **▲**
Mat A+Mat B **EXE**

Ans **1** **2** **3** **4**
EXE $\begin{bmatrix} 3 & 2 \\ 3 & 6 \end{bmatrix}$ **3**

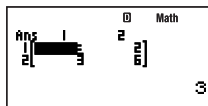
■ 乘以矩阵

仅当两个矩阵行数相同时，才能相乘。

示例: $\left(\begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} + \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}\right) \times \begin{bmatrix} 3 \\ 5 \end{bmatrix}$

此示例假定 Mat A 包含 $\begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}$, Mat B 包含 $\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$, Mat C 包含 $\begin{bmatrix} 3 \\ 5 \end{bmatrix}$ 。

【Mat A】 $\oplus$ 【Mat B】 EXE



$\otimes$ 【Mat C】



EXE



注意

当显示器上呈现计算屏幕时，可以通过按 FUNCTION - {MATRIX} 2 (Mat) SHIFT (-) (Ans) 输入 Mat Ans 变量。

■ 计算矩阵纯量乘法

本计算器支持以下类型的纯量乘法计算。

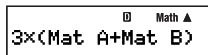
$$n \times \text{Mat A}, n \text{ Mat A}, \text{Mat A} \times n, \text{Mat A} \div n$$

- 可以将“Mat A”替换为计算器的任何一个矩阵，从 Mat A 到 Mat F 或 Mat Ans。
- 对于 n ，可以使用值、独立存储器 (M)、变量、矩阵名称、常数 (π 或科技常数) 或科技函数值 (如 $\sin(30)$)。

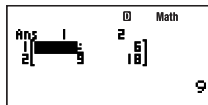
示例: $3 \times \left(\begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} + \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}\right)$

此示例假定 Mat A 包含 $\begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}$, Mat B 包含 $\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$ 。

3 $\otimes$ [] 【Mat A】 $\oplus$ 【Mat B】 []



EXE



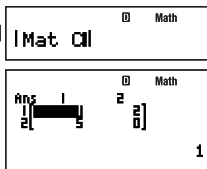
▣ 矩阵元素的绝对值

以下操作确定矩阵元素的绝对值，并将其置于 Mat Ans 屏幕上。

示例： 确定矩阵 $\begin{bmatrix} 1 & -2 \\ 5 & 0 \end{bmatrix}$ 的绝对值。

此示例假定 Mat C 包含 $\begin{bmatrix} 1 & -2 \\ 5 & 0 \end{bmatrix}$ 。

FUNCTION - {MATH} **▼** **1** (Abs) **[Mat C]**



▣ 得出矩阵的行列式

det(函数可以用于得出方阵的行列式。

$$\det \begin{bmatrix} a_{11} \end{bmatrix} = a_{11}$$

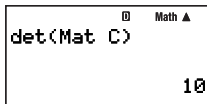
$$\det \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} = a_{11}a_{22} - a_{12}a_{21}$$

$$\det \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} = a_{11}a_{22}a_{33} + a_{12}a_{23}a_{31} + a_{13}a_{21}a_{32} - a_{13}a_{22}a_{31} - a_{12}a_{21}a_{33} - a_{11}a_{23}a_{32}$$

示例： 得出矩阵 $\begin{bmatrix} 1 & -2 \\ 5 & 0 \end{bmatrix}$ 的行列式。

此示例假定 Mat C 包含 $\begin{bmatrix} 1 & -2 \\ 5 & 0 \end{bmatrix}$ 。

FUNCTION - {MATRIX} **3** (det) **[Mat C]** **]**



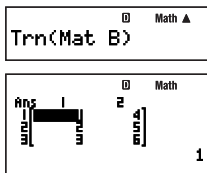
▣ 转置矩阵

转置矩阵本质上表示将矩阵的行列互换。使用以下 Trn(函数执行计算。

示例： 转置矩阵 $\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}$ 。

此示例假定 Mat B 包含 $\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}$ 。

FUNCTION - {MATRIX} **4** (Trn) **[Mat B]** **]**



◆ 求逆矩阵

可以使用如下步骤求逆方阵。

$$[a_{11}]^{-1} = \left[\frac{1}{a_{11}} \right]$$

$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}^{-1} = \frac{\begin{bmatrix} a_{22} & -a_{12} \\ -a_{21} & a_{11} \end{bmatrix}}{a_{11}a_{22} - a_{12}a_{21}}$$

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}^{-1} = \frac{\begin{bmatrix} a_{22}a_{33} - a_{23}a_{32} & -a_{12}a_{33} + a_{13}a_{32} & a_{12}a_{23} - a_{13}a_{22} \\ -a_{21}a_{33} + a_{23}a_{31} & a_{11}a_{33} - a_{13}a_{31} & -a_{11}a_{23} + a_{13}a_{21} \\ a_{21}a_{32} - a_{22}a_{31} & -a_{11}a_{32} + a_{12}a_{31} & a_{11}a_{22} - a_{12}a_{21} \end{bmatrix}}{a_{11}a_{22}a_{33} + a_{12}a_{23}a_{31} + a_{13}a_{21}a_{32} - a_{13}a_{22}a_{31} - a_{12}a_{21}a_{33} - a_{11}a_{23}a_{32}}$$

重要事项！

- 仅对非零方阵支持矩阵求逆。
- 使用 $\text{SHIFT} \square (x^{-1})$ 键输入 “ x^{-1} ”。

示例： 求逆矩阵 $\begin{bmatrix} 1 & -2 \\ 5 & 0 \end{bmatrix}$ 。

此示例假定 Mat C 包含 $\begin{bmatrix} 1 & -2 \\ 5 & 0 \end{bmatrix}$ 。

【Mat C】 $\text{SHIFT} \square (x^{-1}) \text{EXE}$

◆ 计算矩阵的平方

可以使用如下过程计算矩阵的平方。

重要事项！

使用 $\square (x^2)$ 键指定平方操作。

示例： 计算矩阵 $\begin{bmatrix} 1 & -2 \\ 5 & 0 \end{bmatrix}$ 的平方。

此示例假定 Mat C 包含 $\begin{bmatrix} 1 & -2 \\ 5 & 0 \end{bmatrix}$ 。

【Mat C】 $\square (x^2) \text{EXE}$

序列计算 (RECUR)

执行本节中的示例操作，首先选择 RECUR (MODE 6) 作为计算模式。

序列计算概述

可以使用下面两种序列类型中的一种创建序列列表。

① a_n 类型序列

使用这种序列类型，可以输入序列的通项 ($a_n = f(n)$)，以及序列的始值和终值。

② a_{n+1} 类型序列

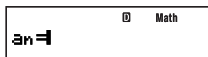
使用这种序列类型，可以输入两项递归的递归公式 ($a_{n+1} = f(a_n)$)，以及序列的始值和终值。

选择序列类型

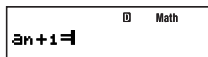
选择以下序列类型:	执行以下键操作:
a_n 类型	FUNCTION - {TYPE} 1 (a_n)
a_{n+1} 类型	FUNCTION - {TYPE} 2 (a_{n+1})

序列编辑器画面

进入 RECUR 模式并选择序列类型之后，下面显示的一个序列编辑器画面会立即出现。使用此屏幕输入定义表达式的表达式（通项或递归公式）。



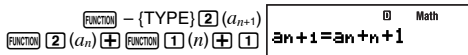
a_n 类型



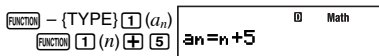
a_{n+1} 类型

在序列编辑器画面上输入公式

示例 1: 输入 $a_{n+1} = a_n + n + 1$



示例 2: 输入 $a_n = n + 5$



注意

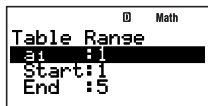
在输入期间清除屏幕，按 AC/ON。

表范围屏幕

按 **EXE** 记录在序列编辑器画面上输入的公式，会显示如下表范围屏幕之一。



a_n 类型序列



a_{n+1} 类型序列

使用此屏幕可指定 n 的始值 (Start) 和终值 (End) 创建序列表。

指定初项、始值和终值

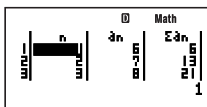
- 在表范围屏幕上，使用 **▼** 和 **▲** 将高亮显示移到要变更的设定位置。
- 输入所需的值或表达式。
 - 在输入期间清除屏幕，按 **AC/ON**。
 - 在输入期间按 **EXIT** 将放弃到此时为止所有的输入，并恢复先前在屏幕上输入的值。
- 输入所需的所有内容后，按 **EXE**。
 - 此操作可记录您的输入。如果输入的是表达式，则记录该表达式的计算结果。
 - 当高亮显示任何设定时，按 **EXE** 可显示序列列表屏幕（第 65 页）。

从表范围屏幕返回序列编辑器画面

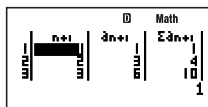
按 **EXIT**。

序列列表屏幕

在“指定初项、始值和终值”下的第 3 步中按 **EXE** 时，计算器将按照输入的公式（通项或递归公式）、初项、始值和终值执行序列计算，并在序列列表屏幕上显示结果。



a_n 类型序列



a_{n+1} 类型序列

- 序列列表屏幕上的每个单元格都显示该单元格中当前存储的值至多为 6 位数。
- 查看单元格中的完整值，将高亮显示移到该单元格。单元格中包含的完整值将在屏幕底部的值显示区域中显示。
- 当一个值在值显示区域中时，可以使用 ENG 变换（第 52 页）、60 进制 -10 进制变换（第 27 页）或小数 - 分数格式变换（第 21 页）变换该值。
- 但是注意不能输入值显示区域或编辑该区域显示的值。

序列表的列

以下列：	包含以下数据：
n	在表范围屏幕上为 n 指定的从始值到终值的值。
a_n	同一行中对应 n 值的 a_n 的值。

以下列:	包含以下数据:
Σa_n	同一行中从 n 始值到 n 值的 a_n 的和
$n+1$	在表范围屏幕上为 $n+1$ 指定的从始值到终值的值。
a_{n+1}	同一行中对应 $n+1$ 值的 a_{n+1} 的值。
Σa_{n+1}	同一行中从 a_1 到 $n+1$ 的 a_{n+1} 的和。

从序列列表屏幕返回表范围屏幕

按 **[EXIT]**。

■ 创建序列表

◆ 创建 a_{n+1} 类型序列表

示例: 重建递归公式为 $a_{n+1} = a_n + n + 1$ 的序列表, 范围为 $1 \leq n \leq 10$ ($n =$ 整数)。但是注意 $a_1 = 2$ 。

进入 RECUR 模式:

[MODE] **[6]** (RECUR)

选择 a_{n+1} 类型:

[FUNCTION] **[-]** {TYPE} **[2]** (a_{n+1})

Math
 $a_{n+1} =$

输入递归公式:

[FUNCTION] **[2]** (a_n) **+** **[FUNCTION]** **[1]** (n) **+** **[1]**

Math
 $a_{n+1} = a_n + n + 1$

记录递归公式:

[EXE]

(此操作可显示表范围屏幕。)

Math
Table Range
 a_1 : 1
Start: 1
End : 5

输入 $a_1 = 2$, 始值 = 1, 终值 = 10:

[2] **[EXE]** **[1]** **[EXE]** **[1]** **[0]** **[EXE]**

Math
Table Range
 a_1 : 2
Start: 1
End : 10

创建序列表:

[EXE]

(此操作可显示序列列表屏幕。)

Math
Table
 a_{n+1} : 2
 a_n : 1
 a_{n+1} : 2
 a_n : 1
 a_{n+1} : 3
 a_n : 2
 a_{n+1} : 4
 a_n : 3
 a_{n+1} : 5
 a_n : 4
 a_{n+1} : 6
 a_n : 5
 a_{n+1} : 7
 a_n : 6
 a_{n+1} : 8
 a_n : 7
 a_{n+1} : 9
 a_n : 8
 a_{n+1} : 10
 a_n : 9
 a_{n+1} : 11
 a_n : 10
 a_{n+1} : 12
 a_n : 11
 a_{n+1} : 13
 a_n : 12
 a_{n+1} : 14
 a_n : 13
 a_{n+1} : 15
 a_n : 14
 a_{n+1} : 16
 a_n : 15
 a_{n+1} : 17
 a_n : 16
 a_{n+1} : 18
 a_n : 17
 a_{n+1} : 19
 a_n : 18
 a_{n+1} : 20
 a_n : 19
 a_{n+1} : 21
 a_n : 20
 a_{n+1} : 22
 a_n : 21
 a_{n+1} : 23
 a_n : 22
 a_{n+1} : 24
 a_n : 23
 a_{n+1} : 25
 a_n : 24
 a_{n+1} : 26
 a_n : 25
 a_{n+1} : 27
 a_n : 26
 a_{n+1} : 28
 a_n : 27
 a_{n+1} : 29
 a_n : 28
 a_{n+1} : 30
 a_n : 29
 a_{n+1} : 31
 a_n : 30
 a_{n+1} : 32
 a_n : 31
 a_{n+1} : 33
 a_n : 32
 a_{n+1} : 34
 a_n : 33
 a_{n+1} : 35
 a_n : 34
 a_{n+1} : 36
 a_n : 35
 a_{n+1} : 37
 a_n : 36
 a_{n+1} : 38
 a_n : 37
 a_{n+1} : 39
 a_n : 38
 a_{n+1} : 40
 a_n : 39
 a_{n+1} : 41
 a_n : 40
 a_{n+1} : 42
 a_n : 41
 a_{n+1} : 43
 a_n : 42
 a_{n+1} : 44
 a_n : 43
 a_{n+1} : 45
 a_n : 44
 a_{n+1} : 46
 a_n : 45
 a_{n+1} : 47
 a_n : 46
 a_{n+1} : 48
 a_n : 47
 a_{n+1} : 49
 a_n : 48
 a_{n+1} : 50
 a_n : 49
 a_{n+1} : 51
 a_n : 50
 a_{n+1} : 52
 a_n : 51
 a_{n+1} : 53
 a_n : 52
 a_{n+1} : 54
 a_n : 53
 a_{n+1} : 55
 a_n : 54
 a_{n+1} : 56
 a_n : 55
 a_{n+1} : 57
 a_n : 56
 a_{n+1} : 58
 a_n : 57
 a_{n+1} : 59
 a_n : 58
 a_{n+1} : 60
 a_n : 59
 a_{n+1} : 61
 a_n : 60
 a_{n+1} : 62
 a_n : 61
 a_{n+1} : 63
 a_n : 62
 a_{n+1} : 64
 a_n : 63
 a_{n+1} : 65
 a_n : 64
 a_{n+1} : 66
 a_n : 65
 a_{n+1} : 67
 a_n : 66
 a_{n+1} : 68
 a_n : 67
 a_{n+1} : 69
 a_n : 68
 a_{n+1} : 70
 a_n : 69
 a_{n+1} : 71
 a_n : 70
 a_{n+1} : 72
 a_n : 71
 a_{n+1} : 73
 a_n : 72
 a_{n+1} : 74
 a_n : 73
 a_{n+1} : 75
 a_n : 74
 a_{n+1} : 76
 a_n : 75
 a_{n+1} : 77
 a_n : 76
 a_{n+1} : 78
 a_n : 77
 a_{n+1} : 79
 a_n : 78
 a_{n+1} : 80
 a_n : 79
 a_{n+1} : 81
 a_n : 80
 a_{n+1} : 82
 a_n : 81
 a_{n+1} : 83
 a_n : 82
 a_{n+1} : 84
 a_n : 83
 a_{n+1} : 85
 a_n : 84
 a_{n+1} : 86
 a_n : 85
 a_{n+1} : 87
 a_n : 86
 a_{n+1} : 88
 a_n : 87
 a_{n+1} : 89
 a_n : 88
 a_{n+1} : 90
 a_n : 89
 a_{n+1} : 91
 a_n : 90
 a_{n+1} : 92
 a_n : 91
 a_{n+1} : 93
 a_n : 92
 a_{n+1} : 94
 a_n : 93
 a_{n+1} : 95
 a_n : 94
 a_{n+1} : 96
 a_n : 95
 a_{n+1} : 97
 a_n : 96
 a_{n+1} : 98
 a_n : 97
 a_{n+1} : 99
 a_n : 98
 a_{n+1} : 100
 a_n : 99
 a_{n+1} : 101
 a_n : 100
 a_{n+1} : 102
 a_n : 101
 a_{n+1} : 103
 a_n : 102
 a_{n+1} : 104
 a_n : 103
 a_{n+1} : 105
 a_n : 104
 a_{n+1} : 106
 a_n : 105
 a_{n+1} : 107
 a_n : 106
 a_{n+1} : 108
 a_n : 107
 a_{n+1} : 109
 a_n : 108
 a_{n+1} : 110
 a_n : 109
 a_{n+1} : 111
 a_n : 110
 a_{n+1} : 112
 a_n : 111
 a_{n+1} : 113
 a_n : 112
 a_{n+1} : 114
 a_n : 113
 a_{n+1} : 115
 a_n : 114
 a_{n+1} : 116
 a_n : 115
 a_{n+1} : 117
 a_n : 116
 a_{n+1} : 118
 a_n : 117
 a_{n+1} : 119
 a_n : 118
 a_{n+1} : 120
 a_n : 119
 a_{n+1} : 121
 a_n : 120
 a_{n+1} : 122
 a_n : 121
 a_{n+1} : 123
 a_n : 122
 a_{n+1} : 124
 a_n : 123
 a_{n+1} : 125
 a_n : 124
 a_{n+1} : 126
 a_n : 125
 a_{n+1} : 127
 a_n : 126
 a_{n+1} : 128
 a_n : 127
 a_{n+1} : 129
 a_n : 128
 a_{n+1} : 130
 a_n : 129
 a_{n+1} : 131
 a_n : 130
 a_{n+1} : 132
 a_n : 131
 a_{n+1} : 133
 a_n : 132
 a_{n+1} : 134
 a_n : 133
 a_{n+1} : 135
 a_n : 134
 a_{n+1} : 136
 a_n : 135
 a_{n+1} : 137
 a_n : 136
 a_{n+1} : 138
 a_n : 137
 a_{n+1} : 139
 a_n : 138
 a_{n+1} : 140
 a_n : 139
 a_{n+1} : 141
 a_n : 140
 a_{n+1} : 142
 a_n : 141
 a_{n+1} : 143
 a_n : 142
 a_{n+1} : 144
 a_n : 143
 a_{n+1} : 145
 a_n : 144
 a_{n+1} : 146
 a_n : 145
 a_{n+1} : 147
 a_n : 146
 a_{n+1} : 148
 a_n : 147
 a_{n+1} : 149
 a_n : 148
 a_{n+1} : 150
 a_n : 149
 a_{n+1} : 151
 a_n : 150
 a_{n+1} : 152
 a_n : 151
 a_{n+1} : 153
 a_n : 152
 a_{n+1} : 154
 a_n : 153
 a_{n+1} : 155
 a_n : 154
 a_{n+1} : 156
 a_n : 155
 a_{n+1} : 157
 a_n : 156
 a_{n+1} : 158
 a_n : 157
 a_{n+1} : 159
 a_n : 158
 a_{n+1} : 160
 a_n : 159
 a_{n+1} : 161
 a_n : 160
 a_{n+1} : 162
 a_n : 161
 a_{n+1} : 163
 a_n : 162
 a_{n+1} : 164
 a_n : 163
 a_{n+1} : 165
 a_n : 164
 a_{n+1} : 166
 a_n : 165
 a_{n+1} : 167
 a_n : 166
 a_{n+1} : 168
 a_n : 167
 a_{n+1} : 169
 a_n : 168
 a_{n+1} : 170
 a_n : 169
 a_{n+1} : 171
 a_n : 170
 a_{n+1} : 172
 a_n : 171
 a_{n+1} : 173
 a_n : 172
 a_{n+1} : 174
 a_n : 173
 a_{n+1} : 175
 a_n : 174
 a_{n+1} : 176
 a_n : 175
 a_{n+1} : 177
 a_n : 176
 a_{n+1} : 178
 a_n : 177
 a_{n+1} : 179
 a_n : 178
 a_{n+1} : 180
 a_n : 179
 a_{n+1} : 181
 a_n : 180
 a_{n+1} : 182
 a_n : 181
 a_{n+1} : 183
 a_n : 182
 a_{n+1} : 184
 a_n : 183
 a_{n+1} : 185
 a_n : 184
 a_{n+1} : 186
 a_n : 185
 a_{n+1} : 187
 a_n : 186
 a_{n+1} : 188
 a_n : 187
 a_{n+1} : 189
 a_n : 188
 a_{n+1} : 190
 a_n : 189
 a_{n+1} : 191
 a_n : 190
 a_{n+1} : 192
 a_n : 191
 a_{n+1} : 193
 a_n : 192
 a_{n+1} : 194
 a_n : 193
 a_{n+1} : 195
 a_n : 194
 a_{n+1} : 196
 a_n : 195
 a_{n+1} : 197
 a_n : 196
 a_{n+1} : 198
 a_n : 197
 a_{n+1} : 199
 a_n : 198
 a_{n+1} : 200
 a_n : 199
 a_{n+1} : 201
 a_n : 200
 a_{n+1} : 202
 a_n : 201
 a_{n+1} : 203
 a_n : 202
 a_{n+1} : 204
 a_n : 203
 a_{n+1} : 205
 a_n : 204
 a_{n+1} : 206
 a_n : 205
 a_{n+1} : 207
 a_n : 206
 a_{n+1} : 208
 a_n : 207
 a_{n+1} : 209
 a_n : 208
 a_{n+1} : 210
 a_n : 209
 a_{n+1} : 211
 a_n : 210
 a_{n+1} : 212
 a_n : 211
 a_{n+1} : 213
 a_n : 212
 a_{n+1} : 214
 a_n : 213
 a_{n+1} : 215
 a_n : 214
 a_{n+1} : 216
 a_n : 215
 a_{n+1} : 217
 a_n : 216
 a_{n+1} : 218
 a_n : 217
 a_{n+1} : 219
 a_n : 218
 a_{n+1} : 220
 a_n : 219
 a_{n+1} : 221
 a_n : 220
 a_{n+1} : 222
 a_n : 221
 a_{n+1} : 223
 a_n : 222
 a_{n+1} : 224
 a_n : 223
 a_{n+1} : 225
 a_n : 224
 a_{n+1} : 226
 a_n : 225
 a_{n+1} : 227
 a_n : 226
 a_{n+1} : 228
 a_n : 227
 a_{n+1} : 229
 a_n : 228
 a_{n+1} : 230
 a_n : 229
 a_{n+1} : 231
 a_n : 230
 a_{n+1} : 232
 a_n : 231
 a_{n+1} : 233
 a_n : 232
 a_{n+1} : 234
 a_n : 233
 a_{n+1} : 235
 a_n : 234
 a_{n+1} : 236
 a_n : 235
 a_{n+1} : 237
 a_n : 236
 a_{n+1} : 238
 a_n : 237
 a_{n+1} : 239
 a_n : 238
 a_{n+1} : 240
 a_n : 239
 a_{n+1} : 241
 a_n : 240
 a_{n+1} : 242
 a_n : 241
 a_{n+1} : 243
 a_n : 242
 a_{n+1} : 244
 a_n : 243
 a_{n+1} : 245
 a_n : 244
 a_{n+1} : 246
 a_n : 245
 a_{n+1} : 247
 a_n : 246
 a_{n+1} : 248
 a_n : 247
 a_{n+1} : 249
 a_n : 248
 a_{n+1} : 250
 a_n : 249
 a_{n+1} : 251
 a_n : 250
 a_{n+1} : 252
 a_n : 251
 a_{n+1} : 253
 a_n : 252
 a_{n+1} : 254
 a_n : 253
 a_{n+1} : 255
 a_n : 254
 a_{n+1} : 256
 a_n : 255
 a_{n+1} : 257
 a_n : 256
 a_{n+1} : 258
 a_n : 257
 a_{n+1} : 259
 a_n : 258
 a_{n+1} : 260
 a_n : 259
 a_{n+1} : 261
 a_n : 260
 a_{n+1} : 262
 a_n : 261
 a_{n+1} : 263
 a_n : 262
 a_{n+1} : 264
 a_n : 263
 a_{n+1} : 265
 a_n : 264
 a_{n+1} : 266
 a_n : 265
 a_{n+1} : 267
 a_n : 266
 a_{n+1} : 268
 a_n : 267
 a_{n+1} : 269
 a_n : 268
 a_{n+1} : 270
 a_n : 269
 a_{n+1} : 271
 a_n : 270
 a_{n+1} : 272
 a_n : 271
 a_{n+1} : 273
 a_n : 272
 a_{n+1} : 274
 a_n : 273
 a_{n+1} : 275
 a_n : 274
 a_{n+1} : 276
 a_n : 275
 a_{n+1} : 277
 a_n : 276
 a_{n+1} : 278
 a_n : 277
 a_{n+1} : 279
 a_n : 278
 a_{n+1} : 280
 a_n : 279
 a_{n+1} : 281
 a_n : 280
 a_{n+1} : 282
 a_n : 281
 a_{n+1} : 283
 a_n : 282
 a_{n+1} : 284
 a_n : 283
 a_{n+1} : 285
 a_n : 284
 a_{n+1} : 286
 a_n : 285
 a_{n+1} : 287
 a_n : 286
 a_{n+1} : 288
 a_n : 287
 a_{n+1} : 289
 a_n : 288
 a_{n+1} : 290
 a_n : 289
 a_{n+1} : 291
 a_n : 290
 a_{n+1} : 292
 a_n : 291
 a_{n+1} : 293
 a_n : 292
 a_{n+1} : 294
 a_n : 293
 a_{n+1} : 295
 a_n : 294
 a_{n+1} : 296
 a_n : 295
 a_{n+1} : 297
 a_n : 296
 a_{n+1} : 298
 a_n : 297
 a_{n+1} : 299
 a_n : 298
 a_{n+1} : 300
 a_n : 299
 a_{n+1} : 301
 a_n : 300
 a_{n+1} : 302
 a_n : 301
 a_{n+1} : 303
 a_n : 302
 a_{n+1} : 304
 a_n : 303
 a_{n+1} : 305
 a_n : 304
 a_{n+1} : 306
 a_n : 305
 a_{n+1} : 307
 a_n : 306
 a_{n+1} : 308
 a_n : 307
 a_{n+1} : 309
 a_n : 308
 a_{n+1} : 310
 a_n : 309
 a_{n+1} : 311
 a_n : 310
 a_{n+1} : 312
 a_n : 311
 a_{n+1} : 313
 a_n : 312
 a_{n+1} : 314
 a_n : 313
 a_{n+1} : 315
 a_n : 314
 a_{n+1} : 316
 a_n : 315
 a_{n+1} : 317
 a_n : 316
 a_{n+1} : 318
 a_n : 317
 a_{n+1} : 319
 a_n : 318
 a_{n+1} : 320
 a_n : 319
 a_{n+1} : 321
 a_n : 320
 a_{n+1} : 322
 a_n : 321
 a_{n+1} : 323
 a_n : 322
 a_{n+1} : 324
 a_n : 323
 a_{n+1} : 325
 a_n : 324
 a_{n+1} : 326
 a_n : 325
 a_{n+1} : 327
 a_n : 326
 a_{n+1} : 328
 a_n : 327
 a_{n+1} : 329
 a_n : 328
 a_{n+1} : 330
 a_n : 329
 a_{n+1} : 331
 a_n : 330
 a_{n+1} : 332
 a_n : 331
 a_{n+1} : 333
 a_n : 332
 a_{n+1} : 334
 a_n : 333
 a_{n+1} : 335
 a_n : 334
 a_{n+1} : 336
 a_n : 335
 a_{n+1} : 337
 a_n : 336
 a_{n+1} : 338
 a_n : 337
 a_{n+1} : 339
 a_n : 338
 a_{n+1} : 340
 a_n : 339
 a_{n+1} : 341
 a_n : 340
 a_{n+1} : 342
 a_n : 341
 a_{n+1} : 343
 a_n : 342
 a_{n+1} : 344
 a_n : 343
 a_{n+1} : 345
 a_n : 344
 a_{n+1} : 346
 a_n : 345
 a_{n+1} : 347
 a_n : 346
 a_{n+1} : 348
 a_n : 347
 a_{n+1} : 349
 a_n : 348
 a_{n+1} : 350
 a_n : 349
 a_{n+1} : 351
 a_n : 350
 a_{n+1} : 352
 a_n : 351
 a_{n+1} : 353
 a_n : 352
 a_{n+1} : 354
 a_n : 353
 a_{n+1} : 355
 a_n : 354
 a_{n+1} : 356
 a_n : 355
 a_{n+1} : 357
 a_n : 356
 a_{n+1} : 358
 a_n : 357
 a_{n+1} : 359
 a_n : 358
 a_{n+1} : 360
 a_n : 359
 a_{n+1} : 361
 a_n : 360
 a_{n+1} : 362
 a_n : 361
 a_{n+1} : 363
 a_n : 362
 a_{n+1} : 364
 a_n : 363
 a_{n+1} : 365
 a_n : 364
 a_{n+1} : 366
 a_n : 365
 a_{n+1} : 367
 a_n : 366
 a_{n+1} : 368
 a_n : 367
 a_{n+1} : 369
 a_n : 368
 a_{n+1} : 370
 a_n : 369
 a_{n+1} : 371
 a_n : 370
 a_{n+1} : 372
 a_n : 371
 a_{n+1} : 373
 a_n : 372
 a_{n+1} : 374
 a_n : 373
 a_{n+1} : 375
 a_n : 374
 a_{n+1} : 376
 a_n : 375
 a_{n+1} : 377
 a_n : 376
 a_{n+1} : 378
 a_n : 377
 a_{n+1} : 379
<

创建 a_n 类型序列表

示例：创建通项为 $a_n = (\frac{1}{2})n^2 + 2n - 3$ 的序列表，范围为 $2 \leq n \leq 6$ ($n =$ 整数)。

MATH

进入 RECUR 模式：

MODE **6** (RECUR)

选择 a_n 类型：

FUNCTION - {TYPE} **1** (a_n)

an = |

输入通项：

1 **▼** **2** **►** **FUNCTION** **1** (n) **x^2** **+**
2 **FUNCTION** **1** (n) **-** **3**

an = $\frac{1}{2}n^2 + 2n - 3$

记录通项：

EXE

(此操作可显示表范围屏幕。)

Table Range
Start: 1
End : 10

输入始值 = 2 和终值 = 6：

2 **EXE** **6** **EXE**

Table Range
Start: 2
End : 6

创建序列表：

EXE

(此操作可显示序列表屏幕。)

n	a_n	Σa_n
2	7.5	10.5
3	11	21.5
4	15.5	37

序列计算注意事项

序列计算期间不能使用以下功能。

- CALC
- SOLVE
- 坐标变换 (Pol, Rec)
- d/dx , d^2/dx^2 , $\int$, Σ
- 独立存储器添加和去除 (**M+**, **SHIFT** **M+** (M-))
- 向变量赋值 (**SHIFT** **RCL** (STO))
- 多重语句输入

生成序列表时的错误

- 一个序列表最多可有 199 行。如果表范围设定使得超过该限值，则会发生范围错误 (Range ERROR)。
- 如果计算器执行序列表计算时用尽计算结果存储器，还会发生“Memory Full”(存储器满) 错误。

序列计算期间的数学错误

如果在序列计算期间发生数学错误 (Math ERROR)，则在显示器上出现序列表屏幕，在错误的单元格中显示“ERROR”。

方程式计算 (EQN)

方程式计算概述

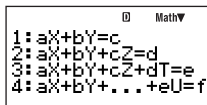
本节显示解二元联立一次方程式的一般操作过程。此过程使用普通显示。

$$X + 0.5Y = 3$$

$$2X + 3Y = 4$$

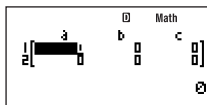
1. 按 **MODE** **8** (EQN)。

- 此操作将显示如下初始 EQN 类型菜单。



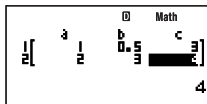
2. 按 **1** (aX+bY=c) 选择二元联立一次方程式。

- 此操作将显示一个如下系数编辑器画面。可以使用此屏幕输入方程式系数的值。



3. 输入系数值。

1 **EXE** **0** **.** **5** **EXE** **3** **EXE** **2** **EXE** **3** **EXE** **4** **EXE**

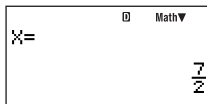


$$\begin{cases} X + 0.5Y = 3 \\ 2X + 3Y = 4 \end{cases}$$

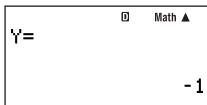
这些值将生成如上联立方程式。

4. 为显示解，按 **EXE**。

- 此操作将显示 X 的解。



- 使用  和  切换 X 和 Y 的解的显示。



- 为退出解屏幕并返回系数编辑器画面，按 **EXIT**。

■ 选择方程式类型

下面显示可以用于选择方程式类型的操作。

选择以下方程式类型：	执行以下键操作：	EQN 类型菜单项：
二元联立一次方程式	MODE 8 (EQN) 1	$aX+bY=c$
三元联立一次方程式	MODE 8 (EQN) 2	$aX+bY+cZ=d$
四元联立一次方程式	MODE 8 (EQN) 3	$aX+bY+cZ+dT=e$
五元联立一次方程式	MODE 8 (EQN) 4	$aX+bY+cZ+dT+eU=f$
二次方程式	MODE 8 (EQN)  1	$aX^2+bX+c=0$
三次方程式	MODE 8 (EQN)  2	$aX^3+bX^2+cX+d=0$

注意

已在使用 EQN 模式执行计算后，变更方程式类型将会清除当前在系数编辑器画面上输入的所有值。

■ 输入系数值

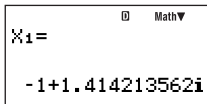
使用系数编辑器可输入方程式系数的值。系数编辑器画面有一些单元格，可以用于输入每个系数的值。在系数编辑器画面上显示的单元格数取决于您选择的方程式类型。

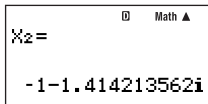
□ 输入和编辑系数值

- 使用光标键将光标移到所需单元格，然后输入值。输入值或表达式后，它会在显示器的左下角显示。
- 为清除光标当前所处的单元格的内容，按 **AC/ON**。
- 必须按 **EXE** 完成向光标所处单元格的输入。此操作将记录输入，并将光标移到下一个单元格。当按 **EXE** 时，单元格中显示输入值最多六位数。
- 可以向每个单元格中输入值或计算表达式。如果输入计算表达式，按 **EXE** 将执行计算并在相应的单元格中仅显示结果。
- 为编辑单元格的内容，使用光标键将光标移到该单元格，然后输入所需内容。

■ 查看方程式的解

在显示系数编辑器时, 按 **EXE** 显示方程式的解。





方程式的解

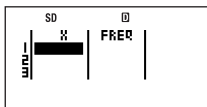
- 可以使用 **▼** 和 **▲** 在多个解之间前后滚动。
- 在显示解时还可以按 **EXE** 轮番显示所有其它可用解。在显示最终解时, 按 **EXE** 将返回系数编辑器画面。
- 为退出解屏幕并返回系数编辑器画面, 按 **EXIT**。
- 解的显示格式取决于设定的显示格式设定和复数显示格式设定。

统计计算 (SD/REG)

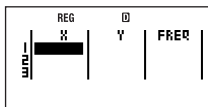
可在 SD 模式或 REG 模式中执行统计计算。SD 模式用于单变量统计计算, REG 模式用于双变量回归计算 (线性、二次、对数、 e 指数、 ab 指数、乘方、逆回归)。

■ 统计样本数据

除了变量的数目, 输入样本数据的过程对 SD 模式和 REG 模式来说是相同的。可使用 STAT 编辑器列表屏幕输入样本数据。



SD 模式 STAT 编辑器



REG 模式 STAT 编辑器

进入 SD 模式 (**MODE** **3**) 或 REG 模式 (**MODE** **4**) 时, 会首先出现相应的 STAT 编辑器画面。

▣ 样本数据输入方法

可以在统计频率打开 (FreqOn) 或关闭 (FreqOff) 的情况下输入样本数据。计算器的初始缺省值是 FreqOff。可以使用设定屏幕的统计频率设定选择所需输入方法 (第 13 页)。当统计频率打开时 (FreqOn), STAT 编辑器画面包含一个 FREQ 列。当统计频率关闭时 (FreqOff), 没有 FREQ 列。

❑ 输入样本数据

以下示例显示如何在 SD 模式中输入以下样本数据。

组值 (X)	频率 (FREQ)
24.5	4
25.5	6
26.5	2

进入 SD 模式：

MODE **3** (SD)

SD	FREQ

在 X 列中输入组值：

2 **4** **.** **5** **EXE** **2** **5** **.** **5** **EXE** **2** **6** **.** **5** **EXE**

SD	FREQ
24.5	1
25.5	
26.5	

所有频率的初始缺省值为 1。

将光标移到 FREQ 列的顶部单元格。



SD	FREQ
24.5	
25.5	
26.5	

在 FREQ 列中输入频率值：

4 **EXE** **6** **EXE** **2** **EXE**

SD	FREQ
24.5	4
25.5	6
26.5	2

在 REG 模式中输入样本数据时的唯一不同是 STAT 编辑器画面有三个列，名为 X、Y 和 FREQ。

注意

- 在 SD 模式中，一组样本数据包含 X 值和 FREQ 值，而一组 REG 模式的样本数据包含 X 值、Y 值和 FREQ 值。通过在 STAT 编辑器画面上对任意列输入值而开始输入样本数据集时，同一样本数据集的其余列会变更为其初始缺省值 (X = 0, Y = 0, FREQ = 1)。
- 向单元格中输入计算表达式并按 **EXE** 将记录计算的结果。
- 在 SD 模式和 REG 模式中，最多可以输入 199 STAT 编辑器行的样本数据。
- 输入的样本数据会一直保留在存储器中，直到删除它或复位计算器。变更为另一计算模式、变更统计频率设定或关闭计算器不会影响样本数据。
- 尽管在 SD 模式中不使用 Y 数据且屏幕上不显示 Y 数据，但计算器会在内部对每个 X 数据提供一个 Y 值 (始终为 0)。因此，如果在 SD 模式中输入单变量数据后，执行涉及 Y 数据的统计计算操作 (如 Σy)，计算器将生成结果而不显示错误。

编辑样本数据

可以使用本节中的步骤，执行以下 STAT 编辑器画面编辑操作：替换单元格内容、插入行、插入单元格、删除单元格和删除所有 STAT 编辑器画面数据。

替换单元格内容

使用光标键将光标移到要替换其内容的单元格，然后输入所需的值或计算表达式。完成后按 **EXE** 记录您的输入。

插入行

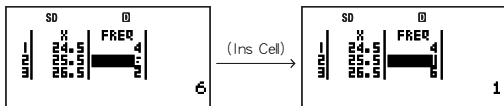
1. 将光标移到要插入行的位置。
2. 按 **FUNCTION** **(5)** (STAT) **(1)** (Edit) **(1)** (Ins Row)。
 - 此操作将将光标所处的行和其下面所有的内容向下移动，并插入一个新行。
 - 新插入行中的所有列都会填充适用的初始缺省值（如果相应则为 $X = 0$, $Y = 0$, $FREQ = 1$ ）。

删除所有 STAT 编辑器数据

1. 按 **FUNCTION** **(5)** (STAT) **(1)** (Edit) **(2)** (Del All)。
 - 此操作将显示“Delete All Data?”确认信息。
2. 要删除所有 STAT 编辑器数据，按 **EXE** (Yes)。要取消该操作而不删除任何内容，按 **EXIT** (No)。

插入单元格

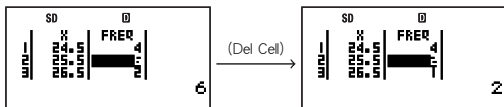
1. 将光标移到要插入单元格的位置。
2. 按 **FUNCTION** **(5)** (STAT) **(1)** (Edit) **(3)** (Ins Cell)。
 - 此操作将将光标所处的单元格和其下面所有的内容向下移动，并插入一个新单元格。插入新单元格的列的底部的单元格会自动删除。



- 插入的单元格会填充适用的初始缺省值 ($X = 0$, $Y = 0$ 或 $FREQ = 1$)。

删除单元格

1. 将光标移到要删除的单元格。
2. 按 **FUNCTION** **(5)** (STAT) **(1)** (Edit) **(4)** (Del Cell)。
 - 此操作将删除光标所处的单元格，并将其下面所有内容向上移动。在删除该单元格的列的底部会插入一个新的单元格。



- 在该列底部插入的新单元格会填充适用的初始缺省值 ($X = 0$, $Y = 0$ 或 $FREQ = 1$)。

■ 执行单变量统计计算

- 执行本节中的示例操作，首先选择 SD (MODE 3) 作为计算模式。
- 以下操作假定已经输入了第 71 页的“输入样本数据”下的样本数据。

▣ 显示统计计算结果屏幕

当 STAT 编辑器画面（包含样本数据）在显示器上出现时，执行以下键操作。

FUNCTION 6 (RESULT)
(使用 $\blacktriangledown$ 和 $\blacktriangle$ 滚动。)

SD 0 $\blacktriangledown$	
1-Variable	
$\bar{x}$	=25.0714285
$\sum x$	=175.5
$\sum x^2$	=4403.75
$x\sigma n$	=0.72843135
$x\sigma n-1$	=0.78679579
n	=7
minX	=24.5
maxX	=26.5

下面是可能的计算结果的一个示例。

注意

- 计算结果最多可显示 10 位。
- 返回 STAT 编辑器画面，按 EXIT。
- 有关在统计计算结果屏幕上显示的值的含义和用于生成它们的公式，请参照“SD 模式统计命令参考”（第 74 页）。

▣ 执行特定统计计算

1. 当 STAT 编辑器画面在显示器上出现时，按 FUNCTION 1 ($\rightarrow$ COMP)。
 - 此操作将显示 COMP 模式计算屏幕。
2. 输入要执行的统计计算的命令，然后按 EXE。
 - 例如，为确定当前输入的样本数据的平均值 ($\bar{x}$)，执行如下操作。

FUNCTION 7 (STAT) 2 (VAR)

0 $\blacktriangledown$	
1:n	2: $\bar{x}$
3: $x\sigma n$	4: $x\sigma n-1$
5: $\bar{y}$	6: $y\sigma n$
7: $y\sigma n-1$	

2 ($\bar{x}$) EXE

0 $\blacktriangle$	
$\bar{x}$	65.68

下面是可能的计算结果的一个示例。

有关统计计算命令的更多信息，请参照下面一节。

SD 模式统计命令参考

n FUNCTION 7 (STAT) 2 (VAR) 1

得出样本数。

$$n = (x_i \text{ 数据项数})$$

$\bar{x}$ FUNCTION 7 (STAT) 2 (VAR) 2

得出平均值。

$$\bar{x} = \frac{\sum x_i}{n}$$

$x\sigma_n$ FUNCTION 7 (STAT) 2 (VAR) 3

得出总体标准偏差。

$$x\sigma_n = \sqrt{\frac{\sum (x_i - \bar{x})^2}{n}}$$

$x\sigma_{n-1}$ FUNCTION 7 (STAT) 2 (VAR) 4

得出样本标准偏差。

$$x\sigma_{n-1} = \sqrt{\frac{\sum (x_i - \bar{x})^2}{n-1}}$$

$\sum x^2$ FUNCTION 7 (STAT) 2 (VAR) ▼ 1

得出样本数据的平方和。

$$\sum x^2 = \sum x_i^2$$

$\sum x$ FUNCTION 7 (STAT) 2 (VAR) ▼ 2

得出样本数据的和。

$$\sum x = \sum x_i$$

$\min X$ FUNCTION 7 (STAT) 2 (VAR) ▼ ▼ 1

确定样本的最小值。

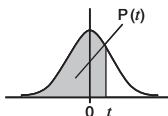
$\max X$ FUNCTION 7 (STAT) 2 (VAR) ▼ ▼ 2

确定样本的最大值。

P(

FUNCTION 7 (STAT) 3 (DISTR) 1

对于自变量 t ，确定标准正态分布 $P(t)$ 的概率。

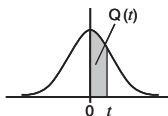


$$P(t) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^t e^{-\frac{x^2}{2}} dx$$

Q(

FUNCTION 7 (STAT) 3 (DISTR) 2

对于自变量 t ，确定标准正态分布 $Q(t)$ 的概率。



$$Q(t) = \frac{1}{\sqrt{2\pi}} \int_0^t e^{-\frac{x^2}{2}} dx$$

R(

FUNCTION 7 (STAT) 3 (DISTR) 3

对于自变量 t ，确定标准正态分布 $R(t)$ 的概率。



$$R(t) = \frac{1}{\sqrt{2\pi}} \int_t^{+\infty} e^{-\frac{x^2}{2}} dx$$

▶t

FUNCTION 7 (STAT) 3 (DISTR) 4

使用紧邻命令之前输入的值 (X) 作为自变量，使用以下公式确定标准化变量 t 。

$$X \blacktriangleright t = \frac{X - \bar{x}}{s/\sqrt{n}}$$

■ 执行双变量统计计算

- 执行本节中的示例操作，首先选择 REG (MODE 4) 作为计算模式。
- 以下操作假定已经输入了第 71 页的“输入样本数据”下的样本数据。

显示统计计算结果屏幕

在 REG 模式中，可以显示列标准偏差计算以及总和计算的结果。也可以显示特定类型回归计算的结果。

注意

- 计算结果最多可显示 10 位。
- 从计算结果屏幕返回 STAT 编辑器画面，按 **EXIT**。
- 有关在结果屏幕上显示的值的含义和用于生成它们的公式，请参照“REG 模式统计命令参考”（第 78 页）。

显示标准偏差和总和计算结果屏幕

当 STAT 编辑器画面（包含样本数据）在显示器上出现时，执行以下键操作。

FUNCTION **6** (RESULT) **1** (S-Var)
(使用 $\blacktriangledown$ 和 $\blacktriangle$ 滚动。)

REG	
2-Variable	
Σx^2	=25.0714285
Σx	=175.5
Σx^2	=4403.75

$x_{\sigma n}$	=0.72843135
$x_{\sigma n-1}$	=0.78679579
n	=7
$\bar{y}$	=1.57142857
Σy	=11
Σy^2	=21
$y_{\sigma n}$	=0.72843135
$y_{\sigma n-1}$	=0.78679579
Σxy	=279.5
Σx^3	=110596.875
$\Sigma x^2 y$	=7108.75
Σx^4	=2780005.43
minX	=24.5
maxX	=26.5
minY	=1
maxY	=3

下面是可能的计算结果的一个示例。

显示回归计算结果

1. 当 STAT 编辑器画面（包含样本数据）在显示器上出现时，请执行以下键操作。

FUNCTION **6** (RESULT) **2** (Reg)

REG	
1:Line	2:Quad
3:Log	4:eExp
5:abExp	6:Power
7:Inv	

- 此操作将显示回归类型菜单。

2. 按对应需要显示其结果的回归类型的键。

为显示以下类型的回归计算的结果：	按以下键：
线性 ($y = ax + b$)	[1] (Line)
二次 ($y = ax^2 + bx + c$)	[2] (Quad)
对数 ($y = a + b \ln x$)	[3] (Log)
e 指数 ($y = ae^{bx}$)	[4] (Exp)
ab 指数 ($y = ab^x$)	[5] (abExp)
乘方 ($y = ax^b$)	[6] (Power)
逆回归 ($y = a + b/x$)	[7] (Inv)

(按 **[1]** 时的示例屏幕)

REG	D
$y=ax+b$	
a	=0.1347162
b	=-4.6081604
r	=0.90910777

下面是可能的计算结果的一个示例。

▣ 执行特定统计计算

- 当 STAT 编辑器画面在显示器上出现时，按 **FUNCTION** **[1]** ($\rightarrow$ COMP)。
 - 此操作将显示 COMP 模式计算屏幕。
- 输入需要执行的统计计算的命令，然后按 **EXE**。
 - 例如，为确定当前输入样本数据的平均值 ($\bar{x}$ 和 $\bar{y}$)，执行如下操作。

FUNCTION **[7]** (STAT) **[2]** (VAR)

	D	
1:n	2: $\bar{x}$	▼
3: $\sum x_n$	4: $\sum x_{n-1}$	
5: $\bar{y}$	6: $\sum y_n$	
7: $\sum y_{n-1}$		

[2] ($\bar{x}$) **EXE**

	D	
$\bar{x}$	65.68	▲

FUNCTION **[7]** (STAT) **[2]** (VAR) **[5]** ($\bar{y}$) **EXE**

	D	
$\bar{y}$	4.24	▲

下面是可能的计算结果的一个示例。

▣ 执行特定回归计算

示例：使用如下统计数据执行对数回归，以得出相关系数以及当 $x = 100$ 时 y 的估计值

x	y
29	1.6
50	23.5
74	38.0
103	46.4
118	48.9

1. 显示对数回归计算结果屏幕。

FUNCTION **6** (RESULT) **2** (Reg) **3** (Log)

REG	0
$y = a + b \cdot \ln x$	
a	$= -111.12839$
b	$= 34.0201475$
r	$= 0.99401394$

2. 按 **EXIT** 返回 STAT 编辑器画面。
3. 按 **FUNCTION** **1** ($\rightarrow$ COMP) 显示 COMP 模式计算屏幕。
4. 确定相关系数 r 以及 $x = 100$ 时 y 的估计值。

- 相关系数

FUNCTION **7** (STAT) **2** (VAR)
▼ ▼ ▼ 4 (r) **EXE**

0	▲
r	0.9940139466

- $x = 100$ 时 y 的估计值

1 **0** **0** **FUNCTION** **7** (STAT) **2** (VAR)
▼ ▼ ▼ 7 ($\hat{y}$) **EXE**

0	▲
100 $\hat{y}$	45.54017135

注意

- 此示例中生成的 r 和 $\hat{y}$ 值是用于对数回归的，因为在使用计算所需值的命令之前，在第 1 步中显示的是对数回归计算屏幕。如果不先显示一个特定的回归计算结果，则缺省情况下这些命令生成的值用于线性回归。
- 有关统计计算命令的详细信息，请参照“REG 模式统计命令参考”（第 78 页）。

▣ REG 模式统计命令参考

平均值和标准偏差命令

n

FUNCTION **7** (STAT) **2** (VAR) **1**

得出样本数。

$$n = (x_i \text{ 数据项数})$$

$\bar{x}$

FUNCTION 7 (STAT) 2 (VAR) 2

得出样本 x 数据的平均值。

$$\bar{x} = \frac{\sum x_i}{n}$$

 $x\sigma_n$

FUNCTION 7 (STAT) 2 (VAR) 3

得出样本 x 数据的总体标准偏差。

$$x\sigma_n = \sqrt{\frac{\sum (x_i - \bar{x})^2}{n}}$$

 $x\sigma_{n-1}$

FUNCTION 7 (STAT) 2 (VAR) 4

得出样本 x 数据的样本标准偏差。

$$x\sigma_{n-1} = \sqrt{\frac{\sum (x_i - \bar{x})^2}{n-1}}$$

 $\bar{y}$

FUNCTION 7 (STAT) 2 (VAR) 5

得出样本 y 数据的平均值。

$$\bar{y} = \frac{\sum y_i}{n}$$

 $y\sigma_n$

FUNCTION 7 (STAT) 2 (VAR) 6

得出样本 y 数据的总体标准偏差。

$$y\sigma_n = \sqrt{\frac{\sum (y_i - \bar{y})^2}{n}}$$

 $y\sigma_{n-1}$

FUNCTION 7 (STAT) 2 (VAR) 7

得出样本 y 数据的样本标准偏差。

$$y\sigma_{n-1} = \sqrt{\frac{\sum (y_i - \bar{y})^2}{n-1}}$$

和命令

 $\sum x^2$

FUNCTION 7 (STAT) 2 (VAR) ▼ 1

得出样本 x 数据的平方和。

$$\sum x^2 = \sum x_i^2$$

 $\sum x$

FUNCTION 7 (STAT) 2 (VAR) ▼ 2

得出样本 x 数据的和。

$$\sum x = \sum x_i$$

Σy^2

FUNCTION 7 (STAT) 2 (VAR) ▼ 3

得出样本 y 数据的平方和。

$$\Sigma y^2 = \Sigma y_i^2$$

 Σy

FUNCTION 7 (STAT) 2 (VAR) ▼ 4

得出样本 y 数据的和。

$$\Sigma y = \Sigma y_i$$

 Σxy

FUNCTION 7 (STAT) 2 (VAR) ▼ 5

得出样本 x 数据和 y 数据的乘积的和。

$$\Sigma xy = \Sigma x_i y_i$$

 Σx^3

FUNCTION 7 (STAT) 2 (VAR) ▼ 6

得出样本 x 数据的立方和。

$$\Sigma x^3 = \Sigma x_i^3$$

 $\Sigma x^2 y$

FUNCTION 7 (STAT) 2 (VAR) ▼ 7

得出样本 x 数据乘以样本 y 数据的平方和。

$$\Sigma x^2 y = \Sigma x_i^2 y_i$$

 Σx^4

FUNCTION 7 (STAT) 2 (VAR) ▼ 8

得出样本 x 数据的四次方的和。

$$\Sigma x^4 = \Sigma x_i^4$$

最小值和最大值命令

minX

FUNCTION 7 (STAT) 2 (VAR) ▼ ▼ 1

得出样本 x 数据的最小值。

maxX

FUNCTION 7 (STAT) 2 (VAR) ▼ ▼ 2

得出样本 x 数据的最大值。

minY

FUNCTION 7 (STAT) 2 (VAR) ▼ ▼ 3

得出样本 y 数据的最小值。

maxY

FUNCTION 7 (STAT) 2 (VAR) ▼ ▼ 4

得出样本 y 数据的最大值。

回归系数和估计值命令

以下命令得出的值取决于用于计算的回归公式。使用的回归公式是相应于上次在 REG 模式中显示的回归计算结果屏幕（第 76 页）的回归公式。

a FUNCTION 7 (STAT) 2 (VAR) ▼ ▼ ▼ 1

得出回归公式的系数 a。

b FUNCTION 7 (STAT) 2 (VAR) ▼ ▼ ▼ 2

得出回归公式的系数 b。

c FUNCTION 7 (STAT) 2 (VAR) ▼ ▼ ▼ 3

仅对二次回归支持此命令。

它得出回归公式的系数 c。

r FUNCTION 7 (STAT) 2 (VAR) ▼ ▼ ▼ 4

得出相关系数 r。

对二次回归不支持此命令。

$\hat{x}_1$ FUNCTION 7 (STAT) 2 (VAR) ▼ ▼ ▼ 5

得出 y 值（假定是紧跟此命令输入的值）的 x 估计值。

$\hat{x}_2$ FUNCTION 7 (STAT) 2 (VAR) ▼ ▼ ▼ 6

仅对二次回归支持此命令。

它得出 y 值（假定是紧跟此命令输入的值）的第二个 x 估计值。当在不是二次回归计算的计算中使用此命令时，此命令生成的结果与 $\hat{x}_1$ 命令相同。

$\hat{y}$ FUNCTION 7 (STAT) 2 (VAR) ▼ ▼ ▼ 7

得出 x 值（假定是紧跟此命令输入的值）的 y 估计值。

回归系数和估计值计算

回归系数和估计值命令使用的公式取决于回归计算类型。有关回归计算公式的详细信息，请参照单独补充资料中的 <#02> 和 <#08>。

有关以下类型的回归计算的更多信息：	请参照：
线性回归	<#02>
二次回归	<#03>
对数回归	<#04>
e 指数回归	<#05>
ab 指数回归	<#06>
乘方回归	<#07>
逆回归	<#08>

统计计算示例

本节提供在计算器上执行的一些实际统计计算示例。

示例 1: 旁边的表显示总注册学生数有 1000 人的男子中学中 50 个学生的脉搏次数。

- ① 确定样本数据的平均值和标准偏差。
- ② 假定总体分布是正态分布，确定脉搏次数大于 70 的学生的分布概率。

脉搏次数	学生数
54 - 56	1
56 - 58	2
58 - 60	2
60 - 62	5
62 - 64	8
64 - 66	9
66 - 68	8
68 - 70	6
70 - 72	4
72 - 74	3
74 - 76	2

操作过程

选择 SD 模式: **MODE** **3** (SD)

统计频率设定选择 FreqOn:

SHIFT **MODE** (SETUP) **▼** **5** (STAT) **1** (FreqOn)

将脉搏次数数据输入 X 列:

5 **5** **EXE** **5** **7** **EXE** **5** **9** **EXE** **6** **1** **EXE** **6** **3** **EXE** **6** **5** **EXE**
6 **7** **EXE** **6** **9** **EXE** **7** **1** **EXE** **7** **3** **EXE** **7** **5** **EXE**

将学生数数据输入 FREQ 列:

▼ **►** **1** **EXE** **2** **EXE** **2** **EXE** **5** **EXE** **8** **EXE**
9 **EXE** **8** **EXE** **6** **EXE** **4** **EXE** **3** **EXE** **2** **EXE**

(1) 平均值和标准偏差

得出平均值:

FUNCTION **1** ($\rightarrow$ COMP)
FUNCTION **7** (STAT) **2** (VAR) **2** ($\bar{x}$) **EXE**

$\bar{x}$ 65.68

得出样本标准偏差:

FUNCTION **7** (STAT) **2** (VAR) **4** ($s\sigma_{n-1}$) **EXE**

$s\sigma_{n-1}$ 4.635444632

(2) 分布概率

FUNCTION **7** (STAT) **3** (DISTR) **3** (R) **7** **0**
FUNCTION **7** (STAT) **3** (DISTR) **4** ($P\rightarrow t$) **7** **EXE**

$P(70 \rightarrow t)$ 0.17325

示例 2: 旁边的数据显示新生儿在出生后不同天数的体重情况。

- ① 得出回归公式以及数据线性回归生成的相关系数。
- ② 得出回归公式以及数据对数回归生成的相关系数。
- ③ 根据按照回归结果最适合数据趋势的回归公式，预测出生后 350 天的体重。

天数	体重 (克)
20	3150
50	4800
80	6420
110	7310
140	7940
170	8690
200	8800
230	9130
260	9270
290	9310
320	9390

操作过程

选择 REG 模式: **MODE** **4** (REG)

统计频率设定选择 FreqOff:

SHIFT **MODE** (SETUP) **▼** **5** (STAT) **2** (FreqOff)

将天数数据输入 X 列:

2 **0** **EXE** **5** **0** **EXE** **8** **0** **EXE** **1** **1** **0** **EXE** **1** **4** **0** **EXE** **1** **7** **0** **EXE**
2 **0** **0** **EXE** **2** **3** **0** **EXE** **2** **6** **0** **EXE** **2** **9** **0** **EXE** **3** **2** **0** **EXE**

将体重数据输入 Y 列:

▼ **►** **3** **1** **5** **0** **EXE** **4** **8** **0** **0** **EXE** **6** **4** **2** **0** **EXE** **7** **3** **1** **0** **EXE**
7 **9** **4** **0** **EXE** **8** **6** **9** **0** **EXE** **8** **8** **0** **0** **EXE** **9** **1** **3** **0** **EXE**
9 **2** **7** **0** **EXE** **9** **3** **1** **0** **EXE** **9** **3** **9** **0** **EXE**

(1) 线性回归

显示线性回归计算结果屏幕:

FUNCTION **6** (RESULT) **2** (Reg) **1** (Line)

REG	D
$y=ax+b$	
a	=18.8757575
b	=4446.57575
r	=0.90479356

(2) 分布对数回归

显示对数回归计算结果屏幕:

EXIT **FUNCTION** **6** (RESULT) **2** (Reg) **3** (Log)

REG	D
$y=a+b \cdot \ln x$	
a	=-4209.3565
b	=2425.75622
r	=0.99149312

(3) 体重预测

相关系数 r 的绝对值接近 1，所以使用对数回归进行计算。

得出 $x = 350$ 时的 $\hat{y}$:

EXIT **FUNCTION** **1** ($\rightarrow$ COMP) **3** **5** **0** **FUNCTION** **7** (STAT)
2 (VAR) **▼** **▼** **▼** **7** ($\hat{y}$) **EXE**

D	▲
350	
	10000.56129

执行本节中的示例操作，首先选择 BASE-N ($\boxed{\text{MODE}} \boxed{2}$) 作为计算模式。

当按 **MODE** **2** 进入 BASE-N 模式时，在显示器上显示的当前设定如下。

基数指示符 负值设定指示符

```

**Dec Signed**

```

基数指示符：指示当前基数（见下表）。

负值设定指示符：指示是否允许使用负值（请参照第 13 页的“变更 BASE-N 模式负值设定”）。

使用如下键指定基数。

 DEC
  HEX
  BIN [
  OCT]

选择以下基数:	按以下键:	显示以下屏幕指示符:
10 进制	 (DEC)	Dec
16 进制	 (HEX)	Hex
2 进制	 (BIN)	Bin
8 进制	 (OCT)	Oct

示例：选择 2 进制作为基础数，并计算 $1_2 + 1_2$

AC/ON In (BIN) 1 + 1 EXE

```

**Bin Signed**
1+1
000000000000000000
000000000000000010

```

- 输入无效值会产生句法错误 (Syntax ERROR)。
- 在 BASE-N 模式中, 不支持输入分数 (小数) 值和指数值。计算结果小数点右边的所有内容都会被裁切。

16 进制数值输入和计算示例

使用以下键输入 16 进制数值所需的字母 (A, B, C, D, E, F)。



示例：选择 16 进制作为基数，并计算 $1F_{16} + 1_{16}$

AC/ON log (HEX) 1 tan (F) + 1 EXE

▲
Hex Signed
1F+1
00000020

有效计算范围

当设定的负值设定为“Signed”时

基数	有效范围
2 进制	正数: 00000000000000000000000000000000 $\leq x \leq 01111111111111111111111111111111$ 负数: 10000000000000000000000000000000 $\leq x \leq 11111111111111111111111111111111$
8 进制	正数: 00000000000 $\leq x \leq 17777777777$ 负数: 20000000000 $\leq x \leq 37777777777$
10 进制	$-2147483648 \leq x \leq 2147483647$
16 进制	正数: 00000000 $\leq x \leq 7FFFFFFF$ 负数: 80000000 $\leq x \leq FFFFFFFF$

当设定的负值设定为“Unsigned”时

基数	有效范围
2 进制	00000000000000000000000000000000 $\leq x \leq 11111111111111111111111111111111$
8 进制	00000000000 $\leq x \leq 37777777777$
10 进制	$0 \leq x \leq 4294967295$
16 进制	00000000 $\leq x \leq FFFFFFFF$

- 当计算结果超出当前缺省基数的相应范围时，会发生数学错误 (Math ERROR)。

将显示结果变换为其它基数

当显示计算结果时，按 x^D (DEC)、log (HEX)、ln (BIN) 或 x^O (OCT) 将把结果变换为相应的基数。

示例：将 10 进制数值 30_{10} 变换为 2 进制、8 进制和 16 进制格式

$\boxed{\text{AC/ON}}$ $\boxed{x^2}$ (DEC) $\boxed{3}$ $\boxed{0}$ $\boxed{\text{EXE}}$

```

      ▲
**Dec Signed**
30                                     30

```

$\boxed{\text{In}}$ (BIN)

```

      ▲
**Bin Signed**
30
0000000000000000
00000000000011110

```

$\boxed{x^2}$ (OCT)

```

      ▲
**Oct Signed**
30                                00000000036

```

$\boxed{\log}$ (HEX)

```

      ▲
**Hex Signed**
30                                0000001E

```

■ 指定特定值的基数

当输入一个值时，可以指定不同于当前缺省基数的基数。

▣ 输入期间指定基数

例如，输入 10 进制数值 3，可以使用以下键操作执行。

$\boxed{\text{FUNCTION}}$ $\boxed{1}$ (BASE-N)

```

      ▼
1:d      2:h
3:b      4:o

```

$\boxed{1}$ (d) $\boxed{3}$

```

      ▲
**Bin Signed**
d3

```

❑ 使用指定基数计算示例

示例：执行计算 $5_{10} + 5_{16}$ ，并以 2 进制显示结果

AC/ON In (BIN)
FUNCTION 1 (BASE-N) 1 (d) 5 +
FUNCTION 1 (BASE-N) 2 (h) 5 EXE

▲
Bin Signed
d5+h5
0000000000000000
0000000000001010

■ 使用逻辑运算和负的 2 进制数值执行计算

本计算器可以执行 32 位 2 进制逻辑运算和负值计算。如下所有示例都是在将 2 进制 (In (BIN)) 设定为缺省基数的情况下执行的。

注意

负的 2 进制、8 进制和 16 进制数值可通过取 2 进制 2 的补码，然后将结果返回原基数而生成。对于 10 进制基数，使用负号显示负值。

❑ 逻辑积 (and)

返回位与的结果。

示例： 1010_2 and $1100_2 = 1000_2$

1 0 1 0 FUNCTION 1 (BASE-N)
▼ 3 (and) 1 1 0 0 EXE

▲
Bin Signed
1010and1100
0000000000000000
0000000000001000

❑ 逻辑和 (or)

返回位或的结果。

示例： 1011_2 or $11010_2 = 11011_2$

1 0 1 1 FUNCTION 1 (BASE-N)
▼ 4 (or) 1 1 0 1 0 EXE

▲
Bin Signed
1011or11010
0000000000000000
00000000000011011

❑ 排他逻辑和 (xor)

返回排他逻辑和的结果。

示例： 1010_2 xor $1100_2 = 110_2$

1 0 1 0 FUNCTION 1 (BASE-N)
▼ 5 (xor) 1 1 0 0 EXE

▲
Bin Signed
1010xor1100
0000000000000000
000000000000110

❑ 排他非逻辑和 (xnor)

返回排他非逻辑和的结果。

示例：为表达式 $3 \times A + B$ 赋值 $A = 5$ 、 $B = 3$ 和 $A = 5$ 、 $B = 10$

LINE

$\boxed{3} \boxed{\times} \boxed{\text{ALPHA}} \boxed{i} \boxed{(A)} \boxed{+} \boxed{\text{ALPHA}} \boxed{\text{B}} \boxed{=}$

$3 \times A + B$

CALC

(此操作可显示赋值屏幕。)

$3 \times A + B$
 $A = 5$
 $B = 3$

赋值 $A = 5$ 和 $B = 3$ ：

$\boxed{5} \boxed{\text{EXE}} \boxed{3} \boxed{\text{EXE}}$

$3 \times A + B$
 $A = 5$
 $B = 3$

执行计算：

EXE

$3 \times A + B$ 18

再次显示赋值屏幕：

CALC

$3 \times A + B$
 $A = 5$
 $B = 3$

A 保持不变，将 10 赋予 B：

$\boxed{\nabla} \boxed{1} \boxed{0} \boxed{\text{EXE}}$

$3 \times A + B$
 $A = 5$
 $B = 10$

执行计算：

EXE

$3 \times A + B$ 25

注意

- 如果选择普通显示作为计算器的显示格式，则赋值屏幕一次仅显示一个变量。
- 使用 ∇ 和 $\blacktriangle$ 将高亮显示在变量之间移动，选择要输入的变量。

重要事项！

在赋值屏幕上不能使用以下功能。

- 输入和显示复数
- SOLVE 功能

在赋值屏幕上显示注释文本

可以使用以下句法向使用 CALC 输入的表达式添加注释文本：“注释文本”：{ 计算表达式 }。该注释文本将在赋值屏幕顶行替换表达式而显示。

示例：使用 CALC 和输入 "AREA": $S = A \times B \div 2$ ，赋值 $A = 7$ 、 $B = 8$ ，然后执行计算

MATH

(SHIFT) (ALPHA) (A-LOCK) (√) (") (i) (A) (÷) (R) (COS) (E)
(i) (A) (√) (") (SHIFT) (√) (:) (ALPHA) (1) (S) (ALPHA) (RCL) (=)
(ALPHA) (i) (A) (X) (ALPHA) (B) (÷) (2)

Math
"AREA": S=A×B÷2

Math
AREA
A=7

CALC

Math
S=A×B÷2
28

7 [EXE] 8 [EXE] [EXE]

SOLVE 功能

SOLVE 功能使用基于牛顿定律的近似解方程式。SOLVE 功能仅可用于 COMP 模式 (MODE 1)。

SOLVE 功能支持的表达式

SOLVE 功能支持输入以下格式的方程式

示例： $Y = X + 5$ 、 $X = \sin(M)$ 、 $X + 3 = B + C$ 、 $XY + C$ （视为 $XY + C = 0$ ）

重要事项！

- SOLVE 功能的方程式不支持以下功能。
 - 输入积分、微分、 Σ 、Pol 和 Rec 函数
 - 多重语句输入
- SOLVE 功能不支持输入复数系数或显示复数解。

使用 SOLVE 功能

输入方程式并按 **SOLVE** 键可显示“赋值屏幕”，可用于向每个变量赋值。向变量赋值之后，将高亮显示移到要求解的变量，然后按 **EXE** 求解。

使用 SOLVE 功能解方程式

示例：解方程式 $y = ax^2 + b$ 中的 x ，其中 $y = 0$ ， $a = 1$ ， $b = -2$

MATH

(ALPHA) (Y) (ALPHA) (RCL) (=) (ALPHA) (i) (A)
(ALPHA) (0) (X) (x²) (+) (ALPHA) (B) (B)

Math
Y=AX²+B

(此操作可显示赋值屏幕。)

Calculator screen showing the equation $Y=AX^2+B$ and the assignment $Y=0$.

将 0 赋予 Y:

将 1 赋予 A:

为 X 输入一个初始值 (此处输入 1):

将 -2 赋予 B:

Calculator screen showing the equation $Y=AX^2+B$ and the assignment $B=-2$.

指定要求解的变量:

(此处我们要对 X 求解, 所以将高亮显示移到 X。)

Calculator screen showing the equation $Y=AX^2+B$ and the assignment $X=$.

解方程式:

Calculator screen showing the equation $Y=AX^2+B$ and the solution $X=1.414213562$.

- 按 **EXE** 返回赋值屏幕。然后可以向变量赋予不同的值, 变更初始值, 再次求解。
- 如果 SOLVE 功能无法解该方程式, 将显示一条错误信息 (Can't Solve)。

注意

- 如果选择普通显示作为计算器的显示格式, 则赋值屏幕一次仅显示一个变量。
- 使用 **▼** 和 **▲** 将高亮显示在变量之间移动, 并选择要输入的变量。

在赋值屏幕上滚动方程式

如果一个方程式太长, 变量赋值屏幕不能全部显示, 可以左右滚动以将它的其余部分显示在视图中。为启用滚动, 按 **FUNCTION** **6** (LOOK)。

使用 **▶** 和 **◀** 键左右滚动方程式。

Calculator screen showing the equation $Y=\sin(X)+\cos(AX)$ and the assignment $Y=0$.

为在滚动后将方程式返回其初始视图, 按 **EXIT**。

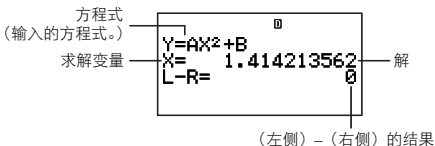
SOLVE 功能警告

- 利用为方程式输入的初始值, SOLVE 功能可能无法求出解。如果发生这种情况, 则输入估计的接近正在求解的变量的实际值的不同初始值, 并重试。
- SOLVE 功能可能无法求出方程式的解, 即使实际存在解。
- 由于牛顿定律的特点, 以下科技函数很难得出解。
 - 周期函数 (如 $y = \sin(x)$)
 - 当以图形表示时产生陡坡的函数 (如 $y = e^x$ 、 $y = \frac{1}{x}$)
 - 不连续函数 (如 $y = \sqrt{x}$)

解屏幕内容

SOLVE 功能的解屏幕显示如下信息。

LINE

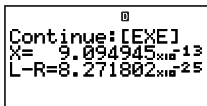


- 解始终以小数格式显示。
- 当选择普通显示作为计算器的显示格式时，“X=”和“L-R”这两行会在一行显示。如果在屏幕上不能全部显示一行中的全部数据，可使用 $\blacktriangleright$ 和 $\blacktriangleleft$ 滚动。
- “(左侧) - (右侧) 的结果”显示当将得出的值赋予正在求解的变量后方程式左侧减右侧的结果。此结果越接近 0，解的精度越高。

正在趋近屏幕

当 SOLVE 功能在进行一定的计算之后无法得出解时，显示器上会显示如下正在趋近屏幕。当显示此屏幕时，按 EXE 将继续计算。

LINE



中断正在进行的计算操作，按 AC/ON 。

从函数创建数表 (TABLE)

执行本节中的示例操作，首先选择 TABLE (MODE 7) 作为计算模式。

TABLE 模式概述

TABLE 模式可以创建 x 和 $f(x)$ 数表，方法是为函数 $f(x)$ 和 x 指定一个范围的可替换的值。

表达式编辑器画面

当进入 TABLE 模式时，会首先显示表达式编辑器画面。可以使用此屏幕输入变量 x 的函数，该函数将用于生成数表。

在表达式编辑器画面上输入

示例: 输入 $f(x) = x^2 + \frac{1}{2}$

MATH

ALPHA 0 (X) x^2 + $\frac{1}{2}$

Math
 $f(X)=X^2+\frac{1}{2}$

注意

- 在输入期间清除屏幕, 按 **AC/ON**。
- 在表达式编辑器画面上, 输入的除 x 之外的任何其它变量将视为值 (按照当前赋予它的值)。

表范围屏幕

按 **EXE** 记录在表达式编辑器画面上输入的表达式, 可显示如下表范围屏幕。

Math
Table Range
Start: 1
End : 5
Step : 1

使用此屏幕可指定 x 值的始值 (Start)、终值 (End) 和步值 (Step), 均用于生成数表。

指定始值、终值和步值

- 在表范围屏幕上, 使用 **▼** 和 **▲** 将高亮显示移到要变更的设定。
- 输入所需的值或表达式。
 - 在输入期间清除屏幕, 按 **AC/ON**。
 - 在输入期间按 **EXIT** 将放弃到此时为止的所有输入, 并恢复先前在屏幕上输入的值。
- 输入所需的所有内容后, 按 **EXE**。
 - 此操作可记录所作的输入。如果输入的是表达式, 则记录该表达式的计算结果。
 - 当高亮显示任何设定时, 按 **EXE** 可显示数表屏幕 (第 93 页)。

从表范围屏幕返回表达式编辑器画面

按 **EXIT**。

数表屏幕

当在“指定始值、终值和步值”的第 3 步中按 **EXE** 时, 计算器将按照输入的函数表达式、始值、终值和步值执行数表计算, 并在数表屏幕上显示结果。

Math

X	F(X)
1	1.5
2	4.5
3	9.5

1

- 数表屏幕上的每个单元格, 都显示该单元格中当前存储的值至多 6 位数。
- 查看单元格中的完整值, 将高亮显示移到该单元格。单元格中包含的完整值将在屏幕底部的值显示区域中显示。

- 当一个值在值显示区域中时，可以使用 ENG 变换（第 52 页）、60 进制 -10 进制变换（第 27 页）或小数 - 分数格式变换（第 21 页）变换该值。
- 但是注意不能输入值显示区域或编辑该区域显示的值。

数表的列

列:	包含以下数据:
X	依据数表范围屏幕指定的初值、终值和步长计算的 x 值
F(X)	同一行中对应 x 值的 $f(x)$ 值

从数表屏幕返回表范围屏幕

按 **EXIT**。

■ 创建数表

示例：原本金金额为 \$100,000，年利率为 3%，每年以复利计息，计算一年、三年和五年的年底本金和利息总额
 x 年后的本金和利息总额可以使用公式 $100000 \times (1 + 0.03)^x$ 计算。输入上述函数，对 x 值指定始值为 1，终值为 5，步长为 2，然后创建数表。

进入 TABLE 模式：

MODE **7** (TABLE)

输入函数表达式 $f(x) = 100000 \times (1 + 0.03)^x$ ：

1 **0** **0** **0** **0** **0** **0** **(** **1** **+** **0** **.** **0** **3** **)**
x^α **ALPHA** **0** **(X)**

f(X)=100000(1+0.03)^X

记录函数表达式：

EXE

(此操作可显示表范围屏幕。)

Table Range
Start:1
End :5
Step :1

输入 Start = 1、End = 5 和 Step = 2：

1 **EXE** **5** **EXE** **2** **EXE**

Table Range
Start:1
End :5
Step :2

创建数表：

EXE

(此操作可显示数表屏幕。)

X	F(X)
1	103000
3	109272
5	115927

1

■ 数表创建注意事项

数表创建的注意事项与 RECUR 模式的注意事项相同。有关更多信息，请参照第 67 页的“序列计算注意事项”。

内置公式

本计算器带有 128 个不同的数学公式和科技公式。内置公式仅可用于 COMP 模式 (MODE 1)。

使用内置公式

输入字符搜索内置公式

- 按 **FMLA**。
 - 此操作显示内置公式的字母顺序菜单。
- 输入所需内置公式名称的第一个字母。
 - 例如，如果要调用 Sector Area（扇形面积）公式，按 **1** (S)。此操作将转移到公式菜单中名称以指定的字母（此示例中为“S”）开头的部分。名称以该字母开头的第一个公式将高亮显示，表示它是选中的。使用 **▼** 和 **▲** 将高亮显示滚动到所需的公式（此示例中为“Sector Area”）。



- 由于空间限制，显示器上显示的公式名称是缩写的。有关公式的完整列表、它们的缩写名称和完整名称，请参照“内置公式名称”（第 96 页）。

滚动菜单搜索内置公式

- 按 **FMLA**。
- 使用 **▼** 和 **▲** 滚动内置公式名称，直到选中要调用的公式。

使用内置公式执行计算

以下示例显示如何在已知三条边长度 (8, 5, 5) 的情况下，使用希罗公式确定三角形的面积。

操作过程

LINE

搜索希罗公式：

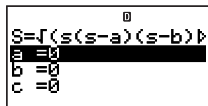
FMLA **1** (H) **▼** (HeronFormula)



开始计算：

(此操作将显示提示输入第一个变量值。)

EXE



输入 $a = 8$ 、 $b = 5$ 和 $c = 5$ ：

[8] **[EXE]** **[5]** **[EXE]** **[5]** **[EXE]**

```

      0
S=√(s(s-a)(s-b)
a =8
b =5
c =5

```

[EXE]
(此操作将显示计算结果。)

```

      0
(s-c))":s=((a+b+
c).2):S=√(s(s-a)
(s-b)(s-c))
      12

```

- 如上所示，向所有必需变量赋值之后会显示计算结果。
- 如果有生成多个结果的公式（如 $\Delta \rightarrow Y$ 变换），则当显示器上显示第一个计算结果时，**[Disp]** 符号将打开。按 **[EXE]** 显示该公式生成的其它结果。当显示器上显示最后一个结果时，**[Disp]** 符号将关闭。
- 当显示器上显示最后一个结果时（**[Disp]** 符号关闭），按 **[EXE]** 将从头重新执行该公式。

◆ 显示内置公式

当显示器上显示提示输入公式变量值时，可以按 **[FUNCTION]** **[6]** (LOOK) 显示整个公式。

使用 **[▶]** 和 **[◀]** 键左右滚动公式。

```

      0
S=√(s(s-a)(s-b)
a =8
b =5
c =5

```

按 **[EXIT]** 或 **[EXE]** 返回按 **[FUNCTION]** **[6]** (LOOK) 之前显示的屏幕。

■ 内置公式名称

有关每个内置公式使用的计算公式的详细信息，请参照单独补充资料中的 <#09>。

序号	显示名称	公式名称
1	2-Line Int	两条直线的夹角
2	Acceleration	加速度
3	Advance Dist	前进距离
4	Area&IntAngl	三角形的面积和内角（3 条边）
5	ArithProgSum	等差数列的和
6	AvgGasMolSpd	平均气体分子速度
7	AxisMov&Rota	坐标轴的移动和旋转
8	Bernoulli 1	柏努利定理（1）
9	Bernoulli 2	柏努利定理（2）
10	Bernoulli 3	柏努利定理（3）
11	Binomial PD	二项分布的概率函数
12	C-PointCoord	中点坐标
13	CarnotCycEf1	卡诺循环的效率（1）

序号	显示名称	公式名称
14	CarnotCycEf2	卡诺循环的效率 (2)
15	CentriForce1	离心力 (1)
16	CentriForce2	离心力 (2)
17	ChordWarVel	弦传输的波速
18	CircCone Lat	圆锥的侧面积
19	CircCone Vol	圆锥的体积
20	CircCylinLat	圆柱的侧面积
21	CircCylinVol	圆柱的体积
22	CircMotCyc 1	圆周运动的周期 (1)
23	CircMotCyc 2	圆周运动的周期 (2)
24	CircMotCyc 3	圆周运动的周期 (3)
25	Circle Area	圆面积
26	ConductResis	导体的电阻
27	Cont Equa 1	连续性方程式 (1)
28	Cont Equa 2	连续性方程式 (2)
29	Conv Y→ Δ	Y→ Δ 变换
30	Conv Δ →Y	Δ →Y 变换
31	Coord Calc	坐标计算 (方位角和距离)
32	CosinTheorem	余弦定理
33	Coulomb's Law	库仑定律
34	Cubes Sum	立方和
35	Current Gain	电流增益
36	Deviation	偏差
37	Dist&DirecAn	距离和方位角 (坐标)
38	Doppler	多普勒效应
39	DropDistance	下落距离
40	Elastic Enr	弹性能
41	ElecCap ES 1	静电电容中存储的能量 (1)
42	ElecCap ES 2	静电电容中存储的能量 (2)
43	ElecCap ES 3	静电电容中存储的能量 (3)
44	ElecFldEDS 1	静电场中存储的能量密度 (1)
45	ElecFldEDS 2	静电场中存储的能量密度 (2)
46	ElecFldStren	电场强度
47	ElecOsciFreq	电振荡频率
48	Ellipse Area	椭圆的面积
49	Enthalpy	热函
50	Exponent PD	指数分布的概率函数
51	Geom PD	几何分布的概率函数
52	GeomProg Sum	等比数列的和

序号	显示名称	公式名称
53	HeatQuantity	热量
54	HeronFormula	希罗公式
55	HyperGeom PD	超几何分布的概率函数
56	I E Force	感应电动势
57	IdlGasStaEq1	理想气体的状态方程式 (1)
58	IdlGasStaEq2	理想气体的状态方程式 (2)
59	IdlGasStaEq3	理想气体的状态方程式 (3)
60	IdlGasStaEq4	理想气体的状态方程式 (4)
61	IncCritAngle	临界入射角
62	InducMagnEnr	感应磁能
63	IntsecCoord1	交点坐标 (4 个点)
64	IntsecCoord2	交点坐标 (3 个点和 1 个距离)
65	Joule Law 1	焦耳定律 (1)
66	Joule Law 2	焦耳定律 (2)
67	Kinetic Enr	动能
68	LC S-Cir CR	LC 串行电路中的组合电抗
69	LR S-CirImp	LR 串行电路中的阻抗
70	LRC P-CirImp	LRC 并行电路中的阻抗
71	LRC S-CirImp	LRC 串行电路中的阻抗
72	MagPolFrcExr	磁极上施加的压力
73	Magn Fld EKE	磁场中的电子动能
74	Magn Force	磁力
75	MinLossMatch	最小损耗匹配
76	Module 1	模数 (1)
77	Module 2	模数 (2)
78	Module 3	模数 (3)
79	Module 4	模数 (4)
80	Normal Dist	正态分布 (概率密度函数)
81	P-Reson Cir	并行共振电路
82	ParalleArea	平行四边形的面积
83	PlateElecCap	平行面之间的静电电容
84	Point-Line	点和直线之间的距离
85	Point-Point	两点之间的距离
86	Poisson PD	泊桑分布的概率函数
87	PotentialEnr	势能
88	Power Factor	功率系数
89	Power Gain	功率增益
90	Pyramid Vol	楞锥的体积
91	Pythagorean	毕达哥拉斯定理

序号	显示名称	公式名称
92	Quad Area	四边形的面积（坐标）
93	R T-VoltChng	RC 串行电路中 R 终端电压的变化
94	RC S-CirImp	RC 串行电路中的阻抗
95	RadiTraverse	辐射导线
96	RefracRelInd	折射的相对指数
97	Repeat Comb	重复组合
98	RepeatPermut	重复排列
99	Reynolds Num	雷诺兹数
100	RotatBodyEnr	旋转体能量
101	S-Reson Cir	串行共振电路
102	Scir&ArcArea	部分圆和圆弧的面积（弦和半径）
103	Sector Area	扇形的面积
104	ShearStress1	抗剪应力（1）
105	ShearStress2	抗剪应力（2）
106	Simp Pend 1	单摆（1）
107	Simp Pend 2	单摆（2）
108	SimpHarmMot1	简谐运动（1）
109	SimpHarmMot2	简谐运动（2）
110	SimpPendCyc	单摆的周期
111	SineTheorem1	正弦定理（1）
112	SineTheorem2	正弦定理（2）
113	SineTheorem3	正弦定理（3）
114	Single Curve	单曲线
115	Sound Intens	声强
116	SphereS-Area	球体的表面积
117	SphereVolume	球体的体积
118	SpringPendCyc	弹簧摆周期
119	Squares Sum	平方和
120	Stadia Calc	使用视距的计算
121	T-Zoid Area	梯形的面积
122	Tension&Comp	张力和压力
123	Triangle 1	三角形的面积
124	Triangle 2	三角形的面积（坐标）
125	Uniform PD	均匀分布的概率函数
126	UnivGravitat	万有引力定律
127	V-Line&Dist	垂线和距离（3 个点）
128	Voltage Gain	电压增益

■ 用户公式

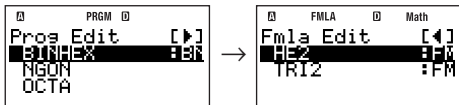
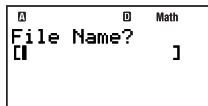
除了计算器的内置公式，也可以自行输入公式供以后需要时调用。还可以使用不同的名称存储内置公式并按需要编辑其内容。创建的以及使用所需其它名称保存的公式称为“用户公式”。

重要事项！

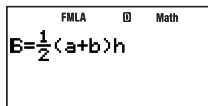
用户公式实际上是运行模式为“Formula”的程序。当执行以下过程时，也称为“程序模式 (PROG)”（第 102 页）。

□ 使用不同名称保存内置公式

1. 按 **FMLA** 显示内置公式菜单。
 - 有关此过程的详细信息，请参照“使用内置公式”（第 95 页）。
2. 按 **FUNCTION** **2** (Save formula)。
 - 此操作将显示文件名输入屏幕，并自动将计算器的键盘锁定为字母格式 (**SHIFT** **ALPHA**)。
3. 对文件名最多输入 12 个字符，然后按 **EXE**。
 - 此操作把指定的内置公式另存为一个程序（用户公式），显示 Fmla List 屏幕，并高亮显示刚刚保存的用户公式。
 - 此处可以通过按 **EXE** 执行刚刚保存的用户公式。
4. 按 **EXIT** 关闭 Fmla List 屏幕。
5. 按 **MODE** **5** (PROG) 进入 PROG 模式。
6. 按 **3** (EDIT)。
 - 此操作将显示 Prog Edit 或 Fmla Edit 文件菜单。
 - 如果显示 Prog Edit 文件菜单，则按 **▶** 可变更为 Fmla Edit 文件菜单。



7. 使用 **▼** 和 **▲** 将高亮显示移到在第 3 步中输入的程序名，然后按 **EXE**。
 - 此操作显示公式编辑屏幕。
8. 使用 **▶** 和 **◀** 将光标移到公式上，并执行必需的操作进行编辑。
9. 在根据需要设定了程序内容后，按 **EXIT**。
 - 此操作返回 Fmla Edit 文件菜单。



创建和保存新公式

1. 按 **MODE** **5** (PROG) 进入 PROG 模式。
2. 按 **1** (NEW)。
 - 此操作显示文件名输入屏幕，并自动将计算器的键盘锁定为字母格式 (**SHIFT** **ALPHA**)。
3. 对文件名最多输入 12 个字符，然后按 **EXE**。
 - 此操作记录文件名并显示运行模式选择屏幕。
4. 按 **3** (Formula) 以选择 Formula 模式。
 - 此操作将显示公式编辑屏幕。
5. 输入公式。
 - 有关可以输入的表达式、变量和字符的信息，请参照“公式编辑屏幕操作”（第 101 页）。
6. 完成输入之后，按 **EXIT**。
 - 此操作将变更为 PROG 模式 Fmla Edit 文件菜单，于是会显示已保存公式的文件名。

```
PRGM 0
File Mode
1:COMP 2:BASE-N
3:Fomula
```

编辑现有用户公式

请参照“编辑现有程序”（第 105 页）。

公式编辑屏幕操作

公式编辑屏幕用于使用新名称保存内置公式、编辑公式和创建新公式。在公式编辑屏幕上可执行以下操作。

- 公式编辑屏幕支持输入可以使用 CALC 输入的计算公式（第 88 页）。和使用 CALC 一样，还可以输入显示计算公式时出现的注释文本。有关更多信息，请参照第 90 页的“在赋值屏幕上显示注释文本”。
- 公式中可以输入以下类型的字符。
 - 单字符字母或希腊字符变量（如 z 或 α ）
 - 带有索引的 2 字符变量

为输入此类变量，按 **FUNCTION** **4** (ALPHA) 显示如下菜单。

```
0 Math
1:abc 2:ABT
3:αβγ 4:123
5:ABC 6:abc
```

输入以下字符：	按以下键：
小写字母	1 (abc)
大写希腊字符	2 (ABT)
小写希腊字符	3 ($\alpha\beta\gamma$)
数字索引	4 (123)
大写字母索引	5 (ABC)
小写字母索引	6 (abc)

输入大写字母字符，使用普通大写输入操作，如 **ALPHA** **i** (A)。

❖ 使用用户公式执行计算

由于用户公式保存为程序，所以执行用户公式的过程与执行程序的过程相同。有关此过程的详细信息，请参照“从 Prog List 或 Fmla List 屏幕运行程序”（第 106 页）。执行用户公式后的操作（向变量赋值）与执行内置公式后的操作相同。

程序模式 (PROG)

可以使用 PROG 模式 (MODE 5) 创建和存储需要定期执行的计算程序。可以在存储程序时为其指定一个名称，这可简化调用、编辑、删除和其它文件管理操作。

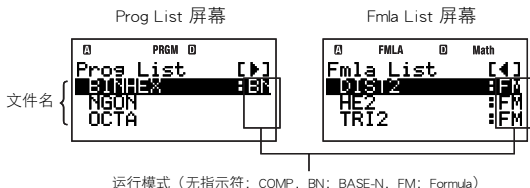
■ 程序模式概述

❖ 指定程序运行模式

创建新程序时，需要指定其“运行模式”，即运行程序的计算器模式。有三种运行模式 COMP、BASE-N 和 Formula。

当程序包含以下内容时：	选择以下运行模式：
在 COMP 模式中执行的计算（包括矩阵、复数和统计计算）	COMP
在 BASE-N 模式中执行的计算	BASE-N
内置公式类型的计算	Formula

每个程序的运行模式在文件列表屏幕上显示（第 108 页）。有两种类型的文件列表屏幕，即显示运行模式为 COMP 或 BASE-N 程序的“Prog List”屏幕，以及显示运行模式为 Formula 程序的“Fmla List”屏幕。



这些示例使用 COMP 或 BASE-N 作为运行模式。有关使用 Formula 作为运行模式的程序（用户公式）的信息，请参照“用户公式”（第 100 页）。

❖ 程序存储器

本计算器有 28500 字节的程序存储器。

重要事项！

28500 字节的用户存储器可用于存储额外变量和程序。这意味着增加额外变量数会减少可用于存储程序的存储器容量。因此，在存储器中存储程序也会减少可用于存储额外变量的存储器容量。

■ 创建程序

▣ 创建新程序

示例：创建一个程序，确定三个规则八面体的表面积和体积，边长分别为 7 cm、10 cm 和 15 cm

下面是用于确定已知边长 (A) 时计算规则八面体表面积 (S) 和体积 (V) 的公式。

$$S = 2\sqrt{3} A^2, V = \frac{\sqrt{2}}{3} A^3$$

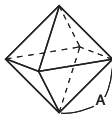
以下程序提示输入 A，然后按照上述程序输出 S 和 V。

"A" ?→A

$2 \times \sqrt{(3)} \times A^2$ ▴

$\sqrt{(2)} \div 3 \times A^3$

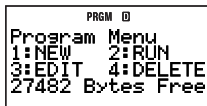
- 指定 COMP 作为该程序的运行模式，并指定文件名 "OCTAHEDRON"。



操作过程

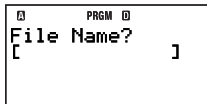
- 按 **MODE** **5** (PROG) 进入 PROG 模式。

- 此程序显示“程序菜单”屏幕。



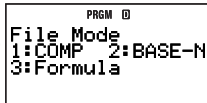
- 按 **1** (NEW)。

- 此操作显示文件名输入屏幕，并自动将计算器的键盘锁定为字母格式 (**SHIFT** **ALPHA**)。



- 对文件名最多输入 12 个字符，然后按 **EXE**。

- 此操作记录文件名，并显示运行模式选择屏幕。



5 (O) **2** (T) **1** (A) **7** (H) **cos** (E) **sin** (D)
2 (R) **5** (O) **4** (N) **EXE**

- 按对应需要指定为程序运行模式的数字键。

- 此处按 **1** (COMP)。此操作选择 COMP 作为运行模式，并显示程序编辑屏幕。

重要事项！

仅当创建新程序时，才可指定运行模式。指定程序的运行模式之后，将不能变更。

5. 输入程序。

- 此处输入如下程序。

ALPHA $\sqrt{\square}$ (") ALPHA $\frac{\square}{\square}$ (A) ALPHA $\sqrt{\square}$ (")
 FUNCTION 3 (PROG) 1 (?) FUNCTION 3 (PROG) 2 ($\rightarrow$) ALPHA $\frac{\square}{\square}$ (A) EXE
 2 $\times$ $\sqrt{\square}$ 3 $\square$ $\times$ ALPHA $\frac{\square}{\square}$ (A) $\times^2$ SHIFT $\times^2$ ($\blacktriangle$)
 $\sqrt{\square}$ 2 $\square$ $\div$ 3 $\times$ ALPHA $\frac{\square}{\square}$ (A) $\times^3$ 3 $\square$

```

PRGM 0
"A"? $\rightarrow$ A $\leftarrow$ 
2 $\times$  $\sqrt{(3)\times A^2}$ 
 $\sqrt{(2)\div 3\times A^3}$ 
    
```

- 按 **EXE** 输入换行符号 ($\blacktriangle$)。

6. 完成输入整个程序后，按 **EXIT**。

- 此操作显示 Prog Edit 文件菜单（第 105 页）。刚输入的程序名将会在屏幕上高亮显示（选中）。

```

PRGM 0
Prog Edit  [▶]
OCTAGON
OCTAHEDRON
OCTOPUS
    
```

7. 尝试运行刚创建的程序 (OCTAHEDRON)。

按 **EXIT** 显示“程序菜单”屏幕，然后按 **2** (RUN)。

- 此操作可显示 Prog List 屏幕。按 **EXE** 运行 Prog List 屏幕上当前名称高亮显示（选中）的程序。

```

PRGM 0
Prog List  [▶]
OCTAGON
OCTAHEDRON
OCTOPUS
    
```

8. 由于“OCTAHEDRON”已高亮显示，只需按 **EXE** 即可运行。
运行该程序：

(此操作显示提示输入变量 A 的值。)
 EXE

```

PRGM 0
A?
    
```

对 A 输入 7:

(此操作显示计算出的表面积结果 S。)
 7 **EXE**

```

PRGM 0
A?
7
169.7409791
    
```

程序中的 $\blacktriangle$ 命令使其暂停并显示到此时为止的计算结果。

使程序继续执行下一个计算：

(此操作显示体积 V 的计算结果，然后结束程序。)
 EXE

```

PRGM 0
A?
7
169.7409791
161.6917506
    
```

9. 为计算第二个八面体的表面积和体积，按 **EXIT** 或 **EXE** 以返回 Prog List 屏幕，并重复第 8 步，对 A 输入 10。再对 A 输入 15，对第三个八面体执行一次此过程。

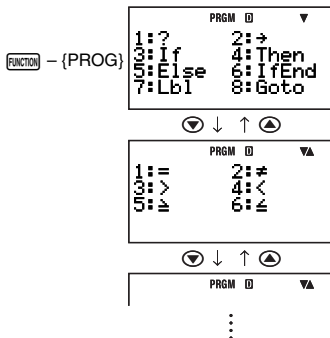
- 可以在第 8 步结束后任何时间，通过按 **MODE** **1** (COMP) 进入 COMP 模式，返回普通计算屏幕。

注意

- 如果在上述过程的第 3 步中输入的文件名已由另一个程序使用，则按 **EXE** 显示现有程序的编辑屏幕。
- 除了此过程中所述的方法，还有许多不同的方式可以运行程序。有关更多信息，请参照“运行程序”（第 106 页）。

程序命令

在程序编辑屏幕上，选择 **FUNCTION** - {PROG} 显示一个可用于输入命令的程序命令菜单。

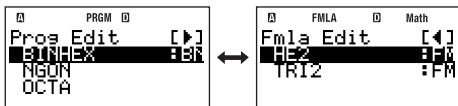


注意

- 在菜单上显示的可输入的程序命令取决于程序的运行模式。有关更多信息，请参照“命令参考”（第 110 页）。
- 当显示器上显示 COMP 模式或 BASE-N 模式计算屏幕时，也可以使用上述过程输入命令。但是，注意当对显示格式设定选择普通显示时，仅可在 COMP 模式中才可输入变量赋值命令（→）。
- 在 BASE-N 模式计算屏幕上不能输入某些命令。有关更多信息，请参照“命令参考”（第 110 页）。

编辑现有程序

- 按 **MODE** **5** (PROG) **3** (EDIT)。
 - 此操作可显示 Prog Edit 或 Fmla Edit 文件菜单。
- 按 **▶** 或 **◀** 可在 Prog Edit 和 Fmla Edit 文件菜单之间切换。



为编辑使用以下运行模式的程序：	显示以下屏幕：
COMP 或 BASE-N	Prog Edit 文件菜单
Formula	Fmla Edit 文件菜单

- 使用 **▼** 和 **▲** 将高亮显示移到需要编辑的程序名，然后按 **EXE**。
 - 此操作显示程序编辑屏幕。

- 使用 **▶** 和 **◀** 将光标移到程序上，并执行必需的操作，编辑该程序的内容或添加新的内容。
 - 可以按 **SHIFT** **▲** 跳至程序的开头，或按 **SHIFT** **▼** 跳至结尾。
- 在根据需要设定了程序内容后，按 **EXIT**。

■ 运行程序

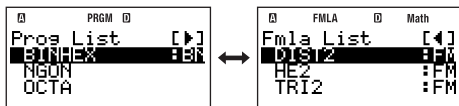
可以 COMP 模式、BASE-N 模式或 PROG 模式执行现有程序。

注意

- 如果从 PROG 模式运行程序，则程序完成后按 **EXE** 或 **EXIT** 将返回 Prog List 或 Fmla List 屏幕。由于刚刚运行程序的名称在 Prog List 或 Fmla List 屏幕上高亮显示，故可通过按 **EXE** 再次运行它。
- 如果从 COMP 模式或 BASE-N 模式运行程序，则程序完成后按 **EXE** 则再次运行同一程序。按 **EXIT** 不会执行任何操作。
- 中断程序的执行，按 **AC/ON**。

▣ 从 Prog List 或 Fmla List 屏幕运行程序

- 执行以下操作之一。
 - 在 PROG 模式中显示器上显示“程序菜单”屏幕时，按 **2** (RUN)。
 - 在 COMP 或 BASE-N 模式中，按 **FILE**。
- 此操作显示 Prog List 屏幕或 Fmla List 屏幕。
- 按 **▶** 或 **◀** 可在 Prog List 和 Fmla List 屏幕之间切换。

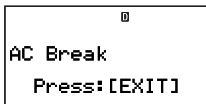


运行时使用以下运行模式的程序：	显示以下屏幕：
COMP 或 BASE-N	Prog List 屏幕
Formula	Fmla List 屏幕

- 使用 **▼** 和 **▲** 将高亮显示移到需要运行的程序名，然后按 **EXE**。
 - 此操作将运行该程序。

注意

运行程序之后，可按 **AC/ON** 键手动中断程序的运行。此操作会显示如下屏幕。



按 **EXIT** 显示程序编辑屏幕，光标位于执行中断的位置。

❑ 使用 Prog 命令运行程序

Prog 可用于指定程序文件名并直接从 COMP 模式或 BASE-N 模式运行该程序。请在 COMP 模式或 BASE-N 模式中执行以下过程。

重要事项！

如果没有使用 Prog 命令指定的文件名的程序，则将发生转移错误（Go ERROR）。

示例：使用 Prog 命令执行名为“OCTAHEDRON”的程序

LINE

AC/ON SHIFT FILE (Prog)

Prog I

SHIFT ALPHA (A-LOCK) (")

5 (O) (C) 2 (T) i (A) (H) cos (E)

sin (D) (R) 5 (O) 4 (N) (")

Prog "OCTAHEDRON"

EXE

(此操作运行该程序。)

Prog "OCTAHEDRON"

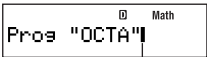
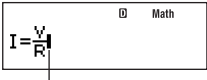
A?

❑ 出现错误信息时的操作

如果出现错误信息，则可以按以下任何键以清除错误信息：**EXIT**、◀ 或 ▶。随后发生的事情取决于试图运行该程序时使用的计算器模式。

COMP 模式或 BASE-N 模式

会显示以下屏幕之一（取决于执行该程序的方式）。

如果是以下面的方式执行该程序：	出现以下屏幕：
使用 Prog 命令	 光标位于刚才在运行的程序文件名的结尾。
从 Prog List 屏幕	
从 Fmla List 屏幕	 光标位于用户公式的结尾。

于是可以进入 PROG 模式，显示发生错误的程序的程序编辑屏幕，并更正错误的原因。

注意

当清除错误信息后，计算器自动进入的模式取决于该程序的运行模式，如下所示。

程序运行模式	计算模式
COMP 模式或 Formula 模式	COMP 模式
BASE-N 模式	BASE-N 模式

PROG 模式

清除错误信息可显示程序的编辑屏幕，光标位于错误的位置，以便更正。

■ 文件屏幕操作

从以下屏幕中选择 PROG 模式的程序菜单。

- **Prog List** 屏幕 / **Fmla List** 屏幕：[2] (RUN)
使用这些屏幕选择程序或用户公式并加以运行。
- **Prog Edit** 文件菜单 / **Fmla Edit** 文件菜单：[3] (EDIT)
使用这些菜单选择程序或用户公式并进行编辑。
- **Prog Delete** 文件菜单 / **Fmla Delete** 文件菜单：[4] (DELETE) [1] (One File)
使用这些菜单选择程序或用户公式并进行删除。

显示器上显示任何上述屏幕时，都可以执行本节中的所有操作。

重要事项！

本节中的操作假定这六个屏幕中的一个已在显示器上显示。

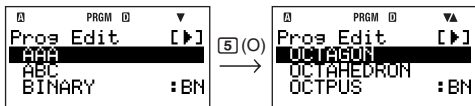
▣ 搜索程序

输入字符搜索文件名

在文件屏幕上，输入所需程序名的第一个字母。

示例：搜索名为“OCTAHEDRON”的文件

按 [5] (O)。



此操作跳至文件屏幕中文件名称以指定字母（此示例中为“O”）开头的部分。名称以该字母开头的第一个程序将高亮显示，表示选中。可以使用 ▼ 和 ▲ 将高亮显示滚动到所需的程序（此示例中为“OCTAHEDRON”）。

滚动列表搜索程序

在文件屏幕上，使用 ▼ 和 ▲ 在程序名之间滚动，直到高亮显示所需的程序。

❑ 将文件名添加到“Favorites”

可以将常用程序的文件名添加到“Favorites”，此操作使该名称在文件屏幕的顶部显示。

操作过程

1. 在文件屏幕上，高亮显示需要添加到“Favorites”的文件名。
2. 按 **FUNCTION** **1** (Favorite-Add)。
 - 此操作在文件屏幕的顶部显示该文件名。



注意

- 添加到“Favorites”的文件名既在文件屏幕的顶部显示，也在文件屏幕中其普通字母顺序位置显示。
- 当通过输入程序文件名的第一个字母搜索文件时，不会检查“Favorites”中文件的名称。
- 文件屏幕顶部的“Favorites”文件名与其它名称之间使用一条线隔开。
- 添加到“Favorites”中的文件名会以添加的顺序显示（不以字母顺序）。

❑ 从“Favorites”中删除文件名

1. 在文件屏幕的“Favorites”文件名列表（分隔线上方）中，高亮显示需要删除的文件名。
2. 按 **FUNCTION** **1** (Favorite-Off)。

❑ 重新命名程序文件

1. 在文件屏幕上，选择需要重新命名的文件的名称。
2. 按 **FUNCTION** **2** (Rename)。
 - 此操作显示一个文件名输入屏幕。
3. 输入新名称，然后按 **EXE**。

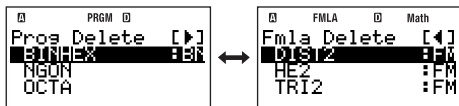
重要事项！

如果输入的文件名已由另一个程序使用，将会出现一条错误信息。若发生这种情况，按 **AC/ON**、**EXIT**、**◀** 或 **▶** 可返回文件名输入屏幕，可输入另外一个名称。

■ 删除程序

❑ 删除特定程序

1. 按 **MODE** **5** (PROG) **4** (DELETE) **1** (One File)。
 - 此操作会显示 Prog Delete 或 Fmla Delete 文件菜单。
2. 按 **▶** 或 **◀** 可在 Prog Delete 和 Fmla Delete 文件菜单之间切换。



欲删除使用以下运行模式的程序：	显示以下屏幕：
COMP 或 BASE-N	Prog Delete 文件菜单
Formula	Fmla Delete 文件菜单

- 使用 和 将高亮显示移到要删除的程序名，然后按 **EXE**。
• 此操作显示“Delete File?”确认信息。
- 删除指定的程序，按 **EXE** (Yes)。取消该操作而不删除任何内容，按 **EXIT** (No)。

◆ 删除所有程序

- 按 **MODE** **5** (PROG) **4** (DELETE) **2** (All Files)。
• 此操作显示“Delete All Files?”确认信息。
- 要删除计算器存储器中当前存储的所有程序，按 **EXE** (Yes)。要取消该操作而不删除任何内容，按 **EXIT** (No)。

命令参考

本节提供有关可在程序中使用的每个命令的详细信息。

注意

- 当在命令句法中看到 < 变量 > 时，系从 A 到 Z 的任何变量或阵列（如 Z[5]）。
- 当在程序编辑屏幕上输入命令时，可使用换行符号 () 或输出命令 () 代替句法或示例中的分隔符 (:)。向程序编辑屏幕中输入 < 换行 > 字符，按 **EXE**。
- 除了分隔符 (:) 和输出命令 ()，本参考中说明的任何命令都不能输入使用 Formula 作为运行模式的程序（用户公式）。有关创建用户公式的信息，请参照第 100 页的“用户公式”。

■ 程序命令

有两种类型的程序命令：从选择 **FUNCTION** - {PROG} 时出现的菜单输入的命令，和使用以下键操作输入的命令：分隔符 (:) (**SHIFT**)、输出命令 () (**SHIFT** **X²**) 和 Prog (**SHIFT** **FILE**)。本节提供每个程序命令的详细信息。

注意

命令名称右侧的“(COMP)”表示仅能输入使用 COMP 作为运行模式的程序的命令，或仅能输入到 COMP 模式计算屏幕上（选择线性显示作为显示格式时）的命令。

◆ 基本操作命令

：（分隔符）	(SHIFT)
--------	------------------

句法 < 语句 > : < 语句 > : ... : < 语句 >
功能 分隔语句。不停止程序的执行。
示例 ? → A : A² : Ans²

▲ (输出命令)

(SHIFT) (X²)

句法	< 语句 > ▲ < 语句 >
功能	暂停程序的执行并显示当前执行的结果。当程序的执行由此命令暂停时， Disp 符号打开。
示例	? → A : A ² ▲ Ans ²
注意	由 ▲ 命令暂停的程序执行可通过按 EXE 键继续，此操作从 ▲ 命令后继续执行程序。

? (输入提示)

句法 1	? → < 变量 > "< 字符串 >"? → < 变量 >
功能 1	向变量赋值。当程序的执行到达此命令时，在显示器上将出现输入提示“?”或“< 字符串 >?”。
示例 1	? → A
句法 2	?< 变量 > "< 字符串 >"?< 变量 >
功能 2	向变量赋值。当程序的执行到达此命令时，在显示器上将出现输入提示“< 变量 >?”或“< 字符串 >?”。 与 ? → < 变量 > 句法不一样，此句法伴随提示符显示该变量的当前值。如果响应该提示按 EXE ，而不输入任何内容，执行程序则使用该变量的当前值继续。
示例 2	? A

→ (变量赋值)

句法	< 表达式 > → < 变量 >
功能	将左侧元素得出的值赋予右侧的变量。
示例	A + 5 → A

关系运算符

=, ≠, >, ≥, <, ≤

句法	< 表达式 > < 关系运算符 > < 表达式 >
功能	这些命令运算两侧的表达式并返回真 (1) 或假 (0)。这些命令与分支命令 ⇒ 使用，以及在构造 If、While 和 Do 语句的 < 条件表达式 > 时使用。
示例	请参照 ⇒ (第 112 页)、If 语句 (第 112 页)、While 语句 (第 114 页) 和 Do 语句 (第 114 页) 的条目。
注意	这些命令运算两侧的表达式，如果真则返回 1，如果假则返回 0。

转移命令

Goto ~ Lbl

句法	$\text{Goto } n : \dots : \text{Lbl } n$ 或 $\text{Lbl } n : \dots : \text{Goto } n$ (n 是从 0 到 9 之间的整数, 或从 A 到 Z 之间的变量名称。)
功能	执行 $\text{Goto } n$ 会转移到相应的 $\text{Lbl } n$ 。
示例	$? \rightarrow A : \text{Lbl } 1 : ? \rightarrow B : A \times B \div 2 \blacktriangle \text{Goto } 1$
重要事项!	如果在 $\text{Goto } n$ 所处的同一程序中没有相应的 $\text{Lbl } n$, 则会发生转移错误 (Go ERROR)。

Dsz (递减, 如果等于 0 则跳过)

句法	$\text{Dsz } \langle \text{变量} \rangle : \langle \text{语句 } 1 \rangle : \langle \text{语句 } 2 \rangle : \dots$
功能	将 $\langle \text{变量} \rangle$ 的值递减 1。 $\langle \text{变量} \rangle$ 值非零, 则执行 $\langle \text{语句 } 1 \rangle$, 然后执行 $\langle \text{语句 } 2 \rangle$ 以及后面的所有内容。 $\langle \text{变量} \rangle$ 值为零, 则会跳过 $\langle \text{语句 } 1 \rangle$ 和 $\langle \text{语句 } 2 \rangle$ 以及执行该命令后的所有内容。
示例	$10 \rightarrow A : 0 \rightarrow C : \text{Lbl } 1 : ? \rightarrow B : B + C \rightarrow C : \text{Dsz } A : \text{Goto } 1 : C \div 10$

Isz (递增, 如果等于 0 则跳过)

句法	$\text{Isz } \langle \text{变量} \rangle : \langle \text{语句 } 1 \rangle : \langle \text{语句 } 2 \rangle : \dots$
功能	将 $\langle \text{变量} \rangle$ 的值递增 1。 $\langle \text{变量} \rangle$ 值非零, 则执行 $\langle \text{语句 } 1 \rangle$, 然后执行 $\langle \text{语句 } 2 \rangle$ 以及后面的所有内容。 $\langle \text{变量} \rangle$ 值为零, 则会跳过 $\langle \text{语句 } 1 \rangle$ 而执行 $\langle \text{语句 } 2 \rangle$ 及其后面的所有内容。

$\Rightarrow$

句法	① $\langle \text{表达式} \rangle \langle \text{关系运算符} \rangle \langle \text{表达式} \rangle \Rightarrow \langle \text{语句 } 1 \rangle : \langle \text{语句 } 2 \rangle : \dots$ ② $\langle \text{表达式} \rangle \Rightarrow \langle \text{语句 } 1 \rangle : \langle \text{语句 } 2 \rangle : \dots$
功能	这是一个条件分支命令。条件分支命令与关系运算符 ($=, \neq, >, \geq, <, \leq$) 组合使用。 句法 ①: 如果 $\Rightarrow$ 命令左侧的条件为真, 则执行 $\langle \text{语句 } 1 \rangle$, 然后执行 $\langle \text{语句 } 2 \rangle$ 并依次执行该命令执行后的所有内容。如果 $\Rightarrow$ 命令左侧的条件为假, 则跳过 $\langle \text{语句 } 1 \rangle$, 然后执行 $\langle \text{语句 } 2 \rangle$ 及其后面的所有内容。 句法 ②: $\Rightarrow$ 命令左侧的条件运算结果非零会解释为“真”, 所以会执行 $\langle \text{语句 } 1 \rangle$, 然后执行 $\langle \text{语句 } 2 \rangle$ 以及后面的所有内容。 $\Rightarrow$ 命令左侧的条件运算结果为零会解释为“假”, 因此会跳过 $\langle \text{语句 } 1 \rangle$, 而执行 $\langle \text{语句 } 2 \rangle$ 及其后面的所有内容。
示例	$\text{Lbl } 1 : ? \rightarrow A : A \geq 0 \Rightarrow \sqrt{}(A) \blacktriangle \text{Goto } 1$

控制结构命令: If 语句

If 语句用于按照 If 后面的表达式 (即分支条件) 的真假来控制程序的执行分支。

If 语句注意事项

- If 语句必须始终伴随有 Then 语句。使用 If 而没有相应的 Then 将产生句法错误(Syntax ERROR)。
- 表达式、Return 或 Stop 命令可用作 Then 和 Else 后面的 < 表达式 *>。

If ~ Then (~ Else) ~ IfEnd

句法	If < 条件表达式 > :Then < 表达式 *> :Else < 表达式 *> :IfEnd :< 语句 > : ...
功能	• Then 后面的语句会执行到 Else, 然后当 If 后面的条件语句为真时执行 IfEnd 后面的语句。当 If 后面的条件语句为假时, 会先执行 Else 后面的语句, 然后执行 IfEnd 后面的语句。 • “Else< 表达式 >”可以省略。 • 必须有 “IfEnd”。省略它不会错误, 但某些程序内容可能会由于 If 语句之后的所有内容导致意外的执行结果。
示例 1	? → A : If A < 10 : Then 10A ▲ Else 9A ▲ IfEnd : Ans×1.05
示例 2	? → A : If A > 0 : Then A×10 → A : IfEnd : Ans×1.05

控制结构命令：For 语句

只要赋予控制变量的值位于指定的范围内, For 语句就会重复执行 For 和 Next 之间的语句。

For 语句注意事项

For 语句必须始终伴随有 Next 语句。使用 For 而没有相应的 Next 将产生句法错误(Syntax ERROR)。

For ~ To ~ Next

(COMP)

句法	For < 表达式 (始值)> → < 变量 (控制变量)> To < 表达式 (终值)> : < 语句 > : ... < 语句 > : Next : ...
功能	For 到 Next 之间语句的执行重复进行, 每次执行控制变量都加 1 (从始值开始)。当控制变量的值到达终值时, 执行会跳至 Next 后面的语句。如果 Next 后面没有语句, 则停止程序的执行。
示例	For 1 → A To 10 : A ² → B : B ▲ Next

For ~ To ~ Step ~ Next

(COMP)

句法	For < 表达式 (始值)> → < 变量 (控制变量)> To < 表达式 (终值)> Step < 表达式 (步长)> : < 语句 > : ... < 语句 > : Next : ...
功能	For 到 Next 之间语句的执行重复进行, 每次执行控制变量都加步长 (从始值开始)。除此之外, 此命令与 For-To-Next 相同。
示例	For 1 → A To 10 Step 0.5 : A ² → B : B ▲ Next

控制结构命令：While 语句

While - WhileEnd

(COMP)

- 句法 While < 条件语句 > : < 语句 > : ... < 语句 > : WhileEnd : ...
- 功能 只要 While 后面的条件语句为真（非零），则从 While 到 WhileEnd 之间的语句就会重复。当 While 后面的语句变为假 (0) 时，则执行 WhileEnd 后面的语句。
- 示例 ? $\rightarrow$ A : While A < 10 : A² $\blacktriangle$ A + 1 $\rightarrow$ A : WhileEnd : A $\div$ 2
- 注意
- 如果第一次执行此命令 While 语句的条件即为假，则执行会直接跳至 WhileEnd 后面的语句，而一次都不执行从 While 到 WhileEnd 之间的语句。
 - 为输入 WhileEnd 命令，请在功能菜单上选择 “W · End”。

Do - LpWhile

(COMP)

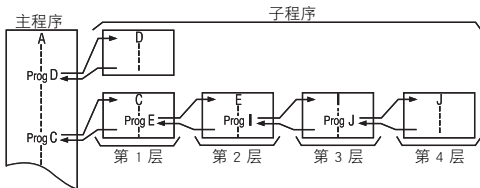
- 句法 Do : < 语句 > : ... < 语句 > : LpWhile < 条件语句 >
- 功能 只要 LpWhile 后面的条件语句为真（非零），则从 Do 到 LpWhile 之间的语句就会重复。由于在执行 LpWhile 之后才评估该条件，所以从 Do 到 LpWhile 之间的语句至少执行一次。
- 示例 Do : ? $\rightarrow$ A : A $\times$ 2 $\rightarrow$ B : B $\blacktriangle$ LpWhile B > 10
- 注意 为输入 LpWhile 命令，在功能菜单上选择 “Lp · W”。

子程序调用命令

Prog

(SHIFT FILE)

- 句法 ... : Prog " 文件名 " : ...
- 功能 从当前程序（主程序）执行另一个其它的程序（子程序）。



- 一个子程序可以被主程序调用任意次。一个子程序可以被任意数目的主程序调用。
- 执行 Prog " 文件名 " 命令会跳至该子程序并从头运行。当到达该子程序的结尾时，处理则跳回主程序并从 Prog " 文件名 " 命令后面的语句继续执行。
- 可以在子程序的内部使用 Prog 命令使进程跳至另一个子程序。这称为“嵌套”。最多可以嵌套 10 层子程序。如果试图嵌套超过 10 层，则会嵌套错误 (Ne ERROR)。

- 注意子程序内的 Goto ~ Lbl 转移仅可在同一子程序内执行。
- 如果由于某种原因找不到 Prog "文件名" 命令指定的程序，则会发生转移错误 (Go ERROR)。

重要事项！

- 使用 Formula 作为运行模式的程序不能用作子程序。
- 主程序和子程序必须是同一运行模式。这意味着，举个例子说，运行模式为 BASE-N 的程序（子程序）不能被运行模式为 COMP 的程序（主程序）调用。

示例
注意

Ans → A : Prog "SUB1" : Prog "SUB2"

有关在 PROG 模式之外使用 Prog 命令的信息，请参照“使用 Prog 命令运行程序”（第 107 页）。

▣ 程序控制命令

Break (COMP)	
--------------	--

句法	... : Break : ...
功能	此命令在 For、While 或 Do 循环中强制中断程序，并跳至下一个命令。通常，此命令在 Then 语句中使用，以便应用 Break 条件。
示例	While A > 0 : If A > 2 : Then Break : IfEnd : WhileEnd : A ▲

Return (COMP)	
---------------	--

句法	... : Return : ...
功能	从子程序返回调用此子程序的程序。在主程序中，此命令会终止程序。
示例	主程序 子程序（文件名：SB） 1 → A : Prog "SB" : C ▲ For A → B To 10 : B + 1 → C : Next : Return

Stop (COMP)	
-------------	--

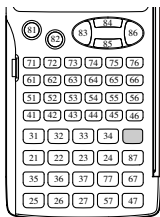
句法	... : Stop : ...
功能	强制终止程序的执行。在子程序中执行此命令会终止所有执行，包括所有子程序和主程序。
示例	For 2 → A To 10 : If A=5 : Then "STOP" : Stop : IfEnd : Next

▣ 输入 / 输出命令

Getkey	
--------	--

句法	... : Getkey : ...
----	--------------------

功能 返回如下代码之一（对应按的最后一个键）。如果未按键，则返回 0。



注意 Getkey 命令可以插入与数学函数相同的表达式。

示例 Do : Cls : Locate 1,1, Ran# : Locate 1,2, "PRESS 0" : LpWhile Getkey ≠ 25

◆ 显示屏幕命令

" "

句法 ... : "< 字符串 >" : ...

功能 将括在引号 (" ") 中的字母数字字符、命令或其它文本显示为注释文本。如果有超过 16 个字符，则文本会换行。如果当前行是屏幕的底行，则当执行文本换行时，屏幕内容将向上滚动。

Cls

句法 ... : Cls : ...

功能 清除显示器中的所有表达式和计算结果。

Locate

句法 Locate < 列号 >, < 行号 >, < 值 >

Locate < 列号 >, < 行号 >, < 表达式 >

Locate < 列号 >, < 行号 >, "< 字符串 >"

($1 \leq \text{行号} \leq 4$, $1 \leq \text{列号} \leq 16$)

功能 在指定的屏幕位置显示指定值或字符串。

- 屏幕位置指定为 (< 列号 >, < 行号 >)，左上角的位置为 (1, 1)，右下角为 (16, 4)。如果指定的位置超出上面定义的范围，则会自变量错误 (Argument ERROR)。
- 指定 < 表达式 > 将使表达式的结果在指定位置显示。如果该表达式是一个变量，则将显示赋予该变量的值。注意如果计算结果是复数、列表或矩阵，则会错误 (Math ERROR 或 Syntax ERROR)。

示例 Locate 5, 2, "CASIO fx"

❑ 逻辑运算子命令

And	(COMP)
------------	--------

句法 < 表达式 > And < 表达式 >
功能 运算两端的表达式（相等或不相等），并基于它们的逻辑积返回真或假。
示例 $? \rightarrow A : ? \rightarrow B : \text{If } A = 2 \text{ And } B > 0 : \text{Then } A \div B : \text{Else } B : \text{IfEnd}$

Or	(COMP)
-----------	--------

句法 < 表达式 > Or < 表达式 >
功能 运算两端的表达式（相等或不相等），并基于它们的逻辑和返回真或假。
示例 $\text{While } A < 10 \text{ Or } B < 5 : A + B \blacktriangle A + 1 \rightarrow A : B + 1 \rightarrow B : \text{WhileEnd}$

Not	(COMP)
------------	--------

句法 Not < 表达式 >
功能 运算后面紧邻的表达式（相等或不相等），并返回其非值。
示例 $\text{Do} : ? \rightarrow A : A \times 2 \rightarrow B : B \blacktriangle \text{LpWhile Not } B < 10$

❑ 清除命令

可以从选择 **FUNCTION** – {CLR} 时出现的菜单上输入数据清除命令。

ClrStat

句法 ClrStat
功能 清除所有列表数据 (List X, List Y, List Freq)。

ClrMemory

句法 ClrMemory
功能 将所有变量 (A 到 Z) 和 Ans 存储器清零。
注意 清除特定变量，使用 $0 \rightarrow \text{< 变量 >}$ 。

ClrMat

句法 ClrMat
功能 清除所有矩阵存储器的内容 (Mat A 到 Mat F 和 Mat Ans)。

ClrVar

句法 ClrVar
功能 清除所有公式变量。

统计计算命令

可以从选择 **FUNCTION** - {STAT} 时出现的菜单上，输入统计计算命令。

注意

有关用于计算平均值、标准偏差和其它统计值的单个命令的信息，请参照“统计计算 (SD/REG)” (第 70 页)。

x 列表命令: **FUNCTION** - {STAT} **1** (LIST)

{ } (输入列表)		(COMP)
句法	... : { < 表达式 > , < 表达式 > , ..., < 表达式 > } → List < X , Y , Freq > : ...	
功能	向列表指定列表数据。	
示例	见下面的 List 命令。	

List (列表操作)		(COMP)
句法	① ... : < 列表数据 > → List < X , Y , Freq > : ... (向列表指定列表数据)。 ② ... : < 表达式 > → List < X , Y , Freq > [< 值 (单元格位置) >] : (向列表中的特定单元格赋值)。 ③ ... : List < X , Y , Freq > [< 值 (单元格位置) >] : (从列表中的特定单元格调用值)。 	

重要事项！

List X、List Y 和 List Freq 分别对应 STAT 模式中 STAT 编辑器画面的 X 列、Y 列和 FREQ 列。

功能	List 与 X (ALPHA 0)、Y (ALPHA •)、Freq (FUNCTION - {STAT} 1 (LIST) 2) 使用可执行上述括号中所述的操作。	
示例	① ? → A : { A , A + 2 , A + 3 } → List X ② ? → A : ? → B : A → List Y [B] ③ ? → A : List X [A] ▲ List X [A + 1]	

回归命令: **FUNCTION** - {STAT} **4** (Reg)

LinearReg 等		(COMP)
句法	... : LinearReg : ... (线性回归) ... : QuadReg : ... (二次回归) ... : LogReg : ... (对数回归) ... : ExpReg : ... (<i>e</i> 指数回归) ... : abExpReg : ... (<i>ab</i> 指数回归) ... : PowerReg : ... (乘方回归) ... : InvReg : ... (逆回归)	
功能	这些命令基于 STAT 编辑器中输入的数据 (List X, List Y, List Freq) 执行指定的回归计算。执行这些命令时出现的屏幕与“显示回归计算结果”(第 76 页)中描述的屏幕相同。	

■ 其它 PROG 模式命令

本节描述仅能在 PROG 模式中使用的命令。注意每个命令仅能在使用命令名称右侧的“(COMP)”或“(BASE-N)”指示的运行模式的程序中使用。

▣ 设定命令

这些命令的运行方式与计算器的各种设定相同。有关更多信息，请参照第 11 页的“计算器设定”。

重要事项！

对于一些设定命令，所配置的设定在完成运行程序之后仍然有效。

角度单位设定命令

Deg, Rad, Gra		(COMP)
---------------	--	--------

句法 ... : Deg : ...

... : Rad : ...

... : Gra : ...

功能 这些命令指定角度单位设定。

显示格式设定命令

Fix		(COMP)
-----	--	--------

句法 ... : Fix <*n*> : ... (*n* 是从 0 到 9 之间的整数。)

功能 固定计算结果输出的小数位数 (从 0 到 9)。

Sci		(COMP)
-----	--	--------

句法 ... : Sci <*n*> : ... (*n* 是从 0 到 9 之间的整数。)

功能 固定计算结果输出的有效位数 (从 1 到 10)。*n* 值为 0 (Sci 0) 将指定有效位数为 10。

Norm		(COMP)
------	--	--------

句法 ... : Norm <1 ; 2> : ...

功能 为计算结果输出指定 Norm 1 还是 Norm 2。

ab/c, d/c		(COMP)
-----------	--	--------

句法 ... : ab/c : ...

... : d/c : ...

功能 指定应使用带分数格式 (ab/c) 还是假分数格式 (d/c) 作为计算结果的显示格式。

EngOn, EngOff	(COMP)
----------------------	--------

句法 ... : EngOn : ...
 ... : EngOff : ...

功能 这两个命令打开工程符号 (EngOn) 或关闭工程符号 (EngOff)。

a+bi, r∠θ	(COMP)
------------------	--------

句法 ... : a+bi : ...
 ... : r∠θ : ...

功能 指定应使用直角坐标格式 ($a+bi$) 还是极坐标格式 ($r\angle\theta$) 作为复数计算结果的显示格式。

统计频率设定命令

FreqOn, FreqOff	(COMP)
------------------------	--------

句法 ... : FreqOn : ...
 ... : FreqOff : ...

功能 这两个命令打开统计频率 (FreqOn) 或关闭统计频率 (FreqOff)。

❑ 基数命令

Dec, Hex, Bin, Oct	(BASE-N)
---------------------------	----------

句法 ... : Dec : ... / ... : Hex : ... / ... : Bin : ... / ... : Oct : ...

功能 这些命令指定用于基数计算的基数。

Signed, Unsigned	(BASE-N)
-------------------------	----------

句法 ... : Signed : ...
 ... : Unsigned : ...

功能 这两个命令指定基数计算值有正负号 (允许负数) 还是无正负号 (不允许负数)。

❑ 舍入 (Rnd) 命令

Rnd(	(COMP)
-------------	--------

句法 ... : < 表达式 > : Rnd(Ans : ...

功能 将当前在 Ans 中的计算结果四舍五入为显示格式设定中指定的位数。

数据通信 (LINK)

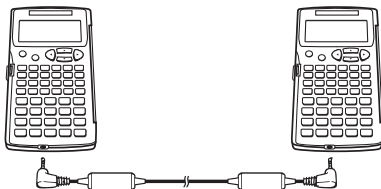
数据通信可以用于在两个 fx-5800P 计算器之间传输程序。

■ 将两个 fx-5800P 计算器互相连接

连接两个计算器需要有一个可选购的数据通信线 (SB-62)。

◆ 连接计算器

如下图所示连接数据通信线。



■ 在 fx-5800P 计算器之间传输数据

连接两个 fx-5800P 计算器之后，执行以下过程以传输数据。

◆ 传输所有程序

1. 在接收计算器（接收方）上执行以下操作。

MODE $\blacktriangledown$ 1 (LINK) 2 (Receive)

Receiving...
Cancel:[AC]

- 这将使接收方设定为待收状态，由显示器上的“Receiving...”指示。

2. 在发送计算器（发送方）上执行以下操作。

MODE $\blacktriangledown$ 1 (LINK) 1 (Transmit) 1 (All)

Transmit OK?
Yes:[EXE]
No:[EXIT]

3. 开始数据传输，按发送方的 [EXE] 键。

- 在数据传输期间，发送方的屏幕将显示右侧显示的信息。接收方的屏幕继续显示在第 1 步中出现的屏幕。

Transmitting...
Cancel:[AC]

- 数据传输完成后，发送方和接收方的屏幕均会变更为右侧显示的信息。

Complete!
Press:[EXIT]

❑ 传输特定程序

1. 在接收计算器（接收方）上执行以下操作，将其设定为待收状态。

MODE  1 (LINK) 2 (Receive)

Receiving...
Cancel:[AC]

2. 在发送计算器（发送方）上，执行以下操作。

MODE  1 (LINK) 1 (Transmit) 2 (Select)

Select Data
AAA
BBB
[1]:SEL [0]:TRAN

3. 在发送方上，使用  和  将高亮显示移到需要传输的程序，然后按 1 (SEL)。
 - 这将在该文件名的左侧显示一个“▶”标记，表示已选中它进行传输。每次按 1 (SEL) 都将会切换文件名旁边“▶”标记显示的开（选中）关（不选中）。
 - 重复第 3 步，直到需要传输的所有程序的名称旁边都显示“▶”标记。
4. 选择所需所有文件之后，按发送方上的 0 (TRAN)。
 - 这将在发送方的显示器上显示“Transmit OK?”。
5. 开始传输程序，按发送方上的 。
 - 在数据传输期间，发送方的屏幕将显示右侧显示的信息。接收方的屏幕继续显示在第 1 步中出现的屏幕。

Transmitting...
Cancel:[AC]

- 数据传输完成后，发送方和接收方的屏幕均会变更为右侧显示的信息。

Complete!
Press:[EXIT]

❑ 中断正在进行的数据传输操作

在发送或接收 fx-5800P 计算器上，按 。

❑ 当接收计算器上已有文件名相同的程序时

如果数据传输期间，发送计算器（发送方）发现接收计算器（接收方）在存储器中已有名称相同的程序，则发送方将显示如右侧所示的信息。

Already Exists
[AAA]
Overwrite?
[1]:Yes [0]:No

使用正在从发送方传输的程序取代当前接收方中的程序，按 **[1]** (Yes)。如果不想取代接收方存储器中的文件，则按 **[0]** (No)。此操作将跳过相应程序的传输，开始传输下一个程序。

存储器管理器 (MEMORY)

存储器管理器是一个用于删除计算器存储器中存储的数据的工具。

注意

此处，术语“删除”具有以下含义。

- 初始化（清零）和设定 Alpha Memory
- 删除所有其它数据类型和存储器文件

执行本节中的示例操作，首先选择 MEMORY (**MODE** **▼** **[2]**) 作为模式。

- 此操作将显示“存储器管理器”屏幕，可提供存储器中不同类型数据的菜单。



可删除数据类型和支持的删除操作

下面列出可以使用存储器管理器删除的数据类型。

数据类型	数据名称	支持的删除操作
程序	<PROGRAM>	全部删除或特定删除
用户公式	<FORMULA>	全部删除或特定删除
公式变量	<FMLA Variable>	全部删除或特定删除
矩阵数据	<MATRIX>	全部删除或特定删除
设定	Setup	全部删除
变量	Alpha Memory	全部删除
额外变量	DimZ Memory	全部删除
统计样本数据	STAT	全部删除
递归数据	Recursion	全部删除
表数据	Table	全部删除
方程式数据	Equation	全部删除

- 上表中“数据名称”列显示在“存储器管理器”菜单中显示的数据名称。
- 数据名两边的尖括号 (< >) 表示同时支持全部删除和特定删除的数据文件夹。全部删除可以删除适用文件夹中的所有数据，而特定删除则删除所选的特定数据项。

■ 使用存储器管理器

◆ 选择需要删除的数据

1. 使用  和  将高亮显示移到需要删除的数据或数据文件夹的名称处。
2. 按 **[1]** (SEL)。

- 这将在该名称的左侧显示一个“▶”标记，表示已选中并进行删除。



```
Memory Manager
▶ Setup
Alpha Memory
[1]:SEL [0]:DEL
```

- 每次按 **[1]** (SEL) 均将会切换名称旁边的“▶”标记显示的开（选中）关（不选中）。
 - 尖括号 (< >) 括起来的数据名称，表示数据文件夹。此处选择数据文件夹将删除该文件夹中的所有数据。
3. 按需要重复第 1 步和第 2 步以选择所需的所有数据名称。

◆ 选择文件夹中的特定数据项

1. 使用  和  将高亮显示移到包含要删除的数据的数据文件夹名称。



```
Memory Manager
▶ <MATRIX>
Setup
[1]:SEL [0]:DEL
```

2. 按 **[EXE]** 进入该文件夹。
- 此操作将显示含该文件夹中包含的所有数据项的菜单。



```
Memory Manager
▶
Mat A
Mat B
[1]:SEL [0]:DEL
```

3. 使用  和  将高亮显示移到要删除的数据的名称，然后按 **[1]** (SEL)。
- 这将在该名称的左侧显示一个“▶”标记，表示已选中它进行删除。
4. 按需要重复第 3 步以选择所需的所有数据名称。
5. 要退出该文件夹并返回“存储器管理器”屏幕，请按 **[EXIT]**。

注意

- 如果在“存储器管理器”屏幕上选中了一个文件夹名称（其前面使用“▶”表示），则当在上述过程的第 2 步进入该文件夹时，该文件夹中的所有数据都将选中以供删除。
- 当在上述过程的第 2 步中进入一个文件夹时，所有当前选中的数据（除文件夹本身）都将自动取消选中。
- 当退出文件夹时，在该文件夹中所选的数据将被取消选中。

◆ 删除当前选中的数据

使用上述过程选择要删除的数据或文件夹之后，显示“存储器管理器”屏幕，然后按 **[0]** (DEL)。

附录

■ 计算的优先顺序

计算器按照如下优先顺序执行您输入的计算。

- 基本上，计算是从左到右执行。
- 括号中计算优先。

顺序	操作类型	描述
1	带括号的函数	Pol(, Rec($\int()$, $d/dx()$, $d^2/dx^2()$, $\Sigma()$, P(), Q(), R(sin(), cos(), tan(), $\sin^{-1}()$, $\cos^{-1}()$, $\tan^{-1}()$, sinh(cosh(), tanh(), $\sinh^{-1}()$, $\cosh^{-1}()$, $\tanh^{-1}()$ log(), ln(), $e^{\wedge}()$, $10^{\wedge}()$, $\sqrt{\quad}$, $\sqrt[3]{\quad}$ (Arg(), Abs(), ReP(), ImP(), Conjg(Not(), Neg(), Det(), Trn(), Rnd(Int(), Frac(), Intg()
2	前面有值的函数 乘方, 乘方根 标准化变量 百分比 ENG 符号	x^2 , x^{-1} , $x!$, "°, °", °, °, ° $\wedge$, $x\sqrt{\quad}$ (▶t % m, μ , n, p, f, k, M, G, T, P
3	分数	$a^{b/c}$
4	前置符号	(-) (负号) d, h, b, o (基数符号)
5	统计估计值计算	$\hat{x}$, $\hat{y}$, $\hat{x}_1$, $\hat{x}_2$
6	排列, 组合 复数符号	nPr , nCr $\angle$
7	乘, 除 省略的乘号	$\times$, $\div$ 紧邻 π 、变量、科学常量 (2π 、5A、 πA 、3mp、 $2i$ 等) 和带括号的函数 ($2\sqrt{\quad}(3)$ 、Asin(30) 等) 前面的乘号可以自动省略。
8	加, 减	+, -
9	关系运算符	=, $\neq$, >, <, $\geq$, $\leq$
10	逻辑积	and (位运算符) And (逻辑运算命令)
11	逻辑和, 排他逻辑和, 排他 非逻辑和	or (位运算符) Or (逻辑运算命令) xor (位运算符) xnor (位运算符)

注意

- 如果计算包含负值，可能需要将负值括在括号中。例如，如果要求 -2 的平方，需要输入： $(-2)^2$ 。这是因为 x^2 是前面有值的函数（上述优先级 2），其优先级大于负号，负号是前置负号（优先级 4）。

$$\boxed{(-)} \boxed{2} \boxed{x^2} \boxed{=}\quad -2^2 = -4$$

$$\boxed{(} \boxed{(-)} \boxed{2} \boxed{)} \boxed{x^2} \boxed{=}\quad (-2)^2 = 4$$

- 乘除以及省略乘号的乘为同一优先级（优先级 7），所以当在同一计算中混用这两种类型时，运算将从左到右执行。将运算括在括号中可以使其先执行，所以使用括号会产生不同的计算结果。

$$\boxed{1} \boxed{\div} \boxed{2} \boxed{i} \boxed{=}\quad 1 \div 2i = 0.5i$$

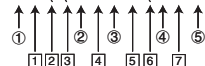
$$\boxed{1} \boxed{\div} \boxed{(} \boxed{2} \boxed{i} \boxed{)} \boxed{=}\quad 1 \div (2i) = -0.5i$$

■ 堆栈限制

本计算器使用称为“堆栈”的存储器区域临时存储计算优先顺序较低的值、命令和函数。“数字堆栈”有 10 层，“命令堆栈”有 26 层，如下图所示。

在 TABLE 和 RECUR 模式中，命令堆栈仅有 25 层（比普通情况下少 1 层）。

$$2 \times ((3 + 4 \times (5 + 4) \div 3) \div 5) + 8 =$$



数字堆栈

①	2
②	3
③	4
④	5
⑤	4
⋮	

命令堆栈

①	×
②	(
③	(
④	+
⑤	×
⑥	(
⑦	+
⋮	

当正在执行的计算超出堆栈容量时，会发生堆栈错误 (Stack ERROR)。

■ 计算范围、位数和精确度

下表显示一般计算范围（值输入和输出范围）、用于内部计算的位数和计算精确度。

计算范围	$\pm 1 \times 10^{-99}$ 到 $\pm 9.999999999 \times 10^{99}$ 和 0
内部计算	15 位
精确度	一般，对于单个计算来说在第 10 位有 ± 1 的误差。计算结果为指数形式的误差为尾数最后一位 ± 1 。在连续计算的情况下，误差会累积。

■ 函数计算的输入范围和精确度

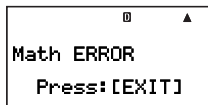
函数	输入范围	
$\sin x$	DEG	$0 \leq x < 9 \times 10^9$
	RAD	$0 \leq x < 157079632.7$
	GRA	$0 \leq x < 1 \times 10^{10}$
$\cos x$	DEG	$0 \leq x < 9 \times 10^9$
	RAD	$0 \leq x < 157079632.7$
	GRA	$0 \leq x < 1 \times 10^{10}$
$\tan x$	DEG	与 $\sin x$ 相同，以下情况除外： $ x = (2n-1) \times 90$ 。
	RAD	与 $\sin x$ 相同，以下情况除外： $ x = (2n-1) \times \pi/2$ 。
	GRA	与 $\sin x$ 相同，以下情况除外： $ x = (2n-1) \times 100$ 。
$\sin^{-1} x$	$0 \leq x \leq 1$	
$\cos^{-1} x$		
$\tan^{-1} x$	$0 \leq x \leq 9.999999999 \times 10^{99}$	
$\sinh x$	$0 \leq x \leq 230.2585092$	
$\cosh x$		
$\sinh^{-1} x$	$0 \leq x \leq 4.999999999 \times 10^{99}$	
$\cosh^{-1} x$	$1 \leq x \leq 4.999999999 \times 10^{99}$	
$\tanh x$	$0 \leq x \leq 9.999999999 \times 10^{99}$	
$\tanh^{-1} x$	$0 \leq x \leq 9.999999999 \times 10^{-1}$	
$\log x / \ln x$	$0 < x \leq 9.999999999 \times 10^{99}$	
10^x	$-9.999999999 \times 10^{99} \leq x \leq 99.99999999$	
e^x	$-9.999999999 \times 10^{99} \leq x \leq 230.2585092$	
$\sqrt{x}$	$0 \leq x < 1 \times 10^{100}$	
x^2	$ x < 1 \times 10^{50}$	
$1/x$	$ x < 1 \times 10^{100}; x \neq 0$	
$\sqrt[3]{x}$	$ x < 1 \times 10^{100}$	
$x!$	$0 \leq x \leq 69$ (x 为整数)	
nPr	$0 \leq n < 1 \times 10^{10}, 0 \leq r \leq n$ (n, r 为整数) $1 \leq \{n!/(n-r)!\} < 1 \times 10^{100}$	
nCr	$0 \leq n < 1 \times 10^{10}, 0 \leq r \leq n$ (n, r 为整数) $1 \leq n!/r! < 1 \times 10^{100}$ 或 $1 \leq n!/(n-r)! < 1 \times 10^{100}$	
$\text{Pol}(x, y)$	$ x , y \leq 9.999999999 \times 10^{99}$ $\sqrt{x^2 + y^2} \leq 9.999999999 \times 10^{99}$	

函数	输入范围
$\text{Rec}(r, \theta)$	$0 \leq r \leq 9.999999999 \times 10^{99}$ θ : 与 $\sin x$
0°''	$ a , b, c < 1 \times 10^{100}$ $0 \leq b, c$
$\leftarrow 0^\circ \text{''}$	$ x < 1 \times 10^{100}$ 10 进制 $\leftrightarrow$ 60 进制变换: $0^\circ 0' 0'' \leq x \leq 9999999^\circ 59' 59''$
$^{\wedge}(x^y)$	$x > 0: -1 \times 10^{100} < y \log x < 100$ $x = 0: y > 0$ $x < 0: y = n, \frac{m}{2n+1}$ (m, n 为整数) 但是: $-1 \times 10^{100} < y \log x < 100$
$x\sqrt[y]{y}$	$y > 0: x \neq 0, -1 \times 10^{100} < 1/x \log y < 100$ $y = 0: x > 0$ $y < 0: x = 2n+1, \frac{2n+1}{m}$ ($m \neq 0; m, n$ 为整数) 但是: $-1 \times 10^{100} < 1/x \log y < 100$
a^b/c	整数、分子和分母的总位数必须等于或小于 10 (包括分隔符)。

- $^{\wedge}(x^y)$, $x\sqrt[y]{y}$, $^3\sqrt{\quad}$, $x!$, nPr , nCr 类型的函数需要的内部计算, 这可能导致在每个单个计算中发生的误差累积起来。
- 误差是累积的, 在靠近函数的连点和拐点处可能误差很大。

■ 错误信息

如果执行的计算超出计算器的限制, 或试图执行不允许的操作, 则在屏幕上将显示错误信息。



错误信息示例

□ 从错误信息恢复

可以通过执行下面描述的键操作而从错误信息恢复, 无论错误类型如何。

- 按 **[EXIT]**、**◀** 或 **▶** 以显示发生错误前刚刚输入的计算表达式的编辑屏幕, 光标指向发生错误的位置。有关更多信息, 请参见第 20 页的“查找错误位置”。
- 按 **[AC/ON]** 可清除出现错误前输入的计算表达式。请注意, 在这种情况下, 原始计算不会保存在计算履历存储器中。

❑ 错误信息参考

本节列出计算器显示的所有错误信息，以及它们的原因和避免它们所需要的操作。

错误信息	原因	操作
Math ERROR (数学错误)	- 计算的中间结果或最终结果超出允许的计算范围。 - 输入值超出允许的输入范围。 - 您试图执行非法的数学运算（如除以 0）。 - 将复数输入为函数的自变量。	- 检查输入值并按需要减少位数。 - 当使用独立存储器或变量作为函数自变量时，确保该存储器或变量值在该函数允许的范围内。 - 复数不能用作三角函数和其他函数的自变量。将该自变量更改为实数。
Stack ERROR (堆栈错误)	- 计算已导致超出数字堆栈或命令堆栈的容量。 - 计算包含的矩阵超过 10 个。	- 简化该计算表达式，以便不超出堆栈的容量。 - 尝试将该计算分为 2 个或更多部分。
Syntax ERROR (句法错误)	计算表达式的句法错误。	检查句法并进行所需的更正。
Argument ERROR (自变量错误)	计算正在使用错误的自变量。	检查使用自变量的方式，并进行所需的更正。
Dimension ERROR (维数错误)	- 计算中使用的矩阵无内容。 - 您正在试图对两个维数不同的矩阵执行加或其他运算，或试图执行其他非法矩阵运算。	- 将矩阵更改为包含数据的矩阵。 - 检查矩阵并确保它们不超过正在执行的计算类型的限制。
Can't Solve (无法解)	SOLVE 功能（第 90 页）无法得出解。（示例：$X = X + 1$）	- 检查该方程式以确保其中没有错误。 - 输入接近解的初始值并重试。
Time Out (超时)	执行了积分或微分计算，但解不满足最终条件。	- 增加 <i>tol</i> 值以放宽解的条件并重试。在这种情况下解的精确度将降低。 - 在三角函数微分的情况下，请选择 Rad 作为角度单位。
Memory ERROR (存储器错误)	试图调用不存在的数据（列表、矩阵、额外变量、公式变量等）。	确保试图调用的数据名称正确。
Go ERROR (转移错误)	- 执行的程序中的“Goto <i>n</i>”没有对应的“Lbl <i>n</i>”。 - Prog 命令调用的文件名不存在。	- 为“Goto <i>n</i>”命令添加“Lbl <i>n</i>”或删除相应的“Goto <i>n</i>”命令。 - 确保 Prog 命令后输入的文件名正确。

错误信息	原因	操作
Ne ERROR (嵌套错误)	- 执行的程序中嵌套的 If (第 112 页)、For (第 113 页)、While (第 114 页) 或 Do 语句 (第 114 页) 或子程序 (第 114 页) 超出 10 个。 - 计算中有过多嵌套的函数或括号。	- 将该程序编辑为嵌套的语句或子程序数等于或少于 10 个。 - 将计算划分为多个部分然后分别计算每个部分。
Transmit ERROR, Receive ERROR (传输错误, 接收错误)	LINK 模式的数据通信在发送 (传输错误) 或接收 (接收错误) 期间被中断。	- 确保线已正确连接。 - 确保接收计算器处于待收模式。
Memory Full (存储器满)	操作或存储器存储操作超出剩余的存储器容量。	- 将使用的存储器数限制在当前指定的存储器数之内。 - 简化尝试存储的数据以使其适于可用存储器容量。 - 删除不再需要的数据以为新数据空出位置。
Already Exists (已存在)	在数据通信期间, 正在发送的数据中的程序或文件名已在接收计算器的存储器中使用。	要使用发送的程序或文件覆盖接收计算器存储器中当前的文件, 请按 1 (Yes)。
Range ERROR (范围错误)	在 TABLE 或 RECUR 模式中: - 指定的表范围错误。 - 试图创建多于 199 行的表。	检查指定的范围并进行所需的更正。
No Variable (无变量)	对 SOLVE 功能的计算输入的方程式中没有要解的变量。	将方程式更改为具有要解的变量的方程式。

■ 假设计算器发生故障之前 ...

计算期间发生错误或计算结果与预期不同的情况下, 请执行以下步骤。如果一步不能更正问题, 则继续下一步。请注意在执行这些步骤之前, 应对重要数据进行备份。

① 检查计算表达式以确保其中没有错误。

② 确保对试图执行的计算类型使用的是正确的模式。

③ 执行以下步骤。

(1) 按 **MODE** **3** (SYSTEM) 以进入 SYSTEM 模式。

(2) 按 **2** (Reset Setup)。

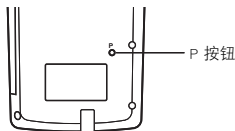
(3) 在出现的确认屏幕上, 按 **EXE** (Yes)。

(4) 按 **EXIT**。

(5) 按 **MODE** 以显示计算模式菜单, 并为要执行的计算类型选择相应的计算模式。

(6) 重新执行计算。

- ④ 使用细尖状物品按计算器背面的 P 按钮以初始化计算器。如果正确执行此过程，计算器将返回上次关闭时的状态。



- ⑤ 如果第 ④ 步不能恢复正常操作，则按 **MODE** $\blacktriangledown$ **3** (SYSTEM) **3** (Reset All) **EXE** (Yes) 以初始化所有模式和设置。

- 有关详细信息，请参见“将计算器复原为初始缺省值设定”（第 1 页）。

重要事项！

- 将删除计算履历数据、存储器数据、统计计算样本数据、程序数据和您输入的所有其他数据。

■ 电池容量低指示器

当电池容量低时，会显示如下信息。发生这种情况时，停止使用并关闭计算器，然后更换电池。

Low Battery
Press: [EXIT]

重要事项！

- 当显示器上显示 Low Battery 信息时，计算器的数据传输功能被禁用。
- 通常换计算器电池时，计算器会将其当前存储器内容存储在闪存中，并在更换电池完成之后恢复它们。如果在出现上述信息后继续使用计算器而不更换电池，最终计算器将不能把数据存储在闪存中。当发生这种情况时，更换电池后计算器将无法恢复存储器内容。

电源要求

本计算器使用一节 AAA 碱性电池 (LR03) 运行。

使用错误的电池类型可能会大大降低电池寿命，并容易使计算器出现故障。

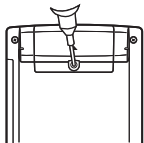
◆ 更换电池

当电池容量低时会出现 Low Battery 信息。发生这种情况时，停止使用并关闭计算器，然后更换电池。您还应至少一年一次定期更换电池，即使计算器使用正常。

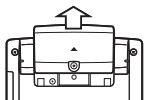
注意

本计算器使用闪存存储数据，使得存储器数据可以在取出电池时不丢失。

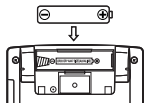
1. 按 **SHIFT** **AC/ON** (OFF) 关闭计算器。
 - 确保在更换电池之前关闭计算器。在计算器打开的情况下更换电池将会删除存储器中的数据。
 - 更换电池后，盖上计算器盖。这将有助于防止不小心打开计算器电源。
2. 在计算器的后面，拧松固定电池盖的螺丝。
 - 该螺丝设计为可以拧松而不会从盖上取下。将螺丝拧松到最大程度。



3. 按箭头指示的方向滑动电池盖以取下。



4. 取出旧的电池。
5. 请参考电池舱中的图示装入新的电池，使其正极 $\oplus$ 和负极 $\ominus$ 方向正确。



6. 将电池盖滑回其原始位置，并使用螺丝固定。
7. 使用细尖状物品按计算器背面的 P 按钮以初始化计算器。**请一定要执行此步骤！不要跳过！**

自动关闭电源

如果不执行任何操作大约 10 分钟，本计算器会自动关闭。如果发生这种情况，按 **AC/ON** 键可重新打开计算器。

规格

电源要求：AAA 电池（碱性）：LR03 × 1

大概电池寿命：1 年（假设每天操作 1 小时）

功耗：0.12W

操作温度：0°C 到 40°C

尺寸：15.1（高）× 81.5（宽）× 163（长）毫米

大概重量：150 克

MEMO

MEMO

MEMO

MEMO



CASIO Europe GmbH
Bornbarch 10, 22848 Norderstedt,
Germany



此标志只适用于 EU 国家。



CASIO®

CASIO COMPUTER CO., LTD.

6-2, Hon-machi 1-chome
Shibuya-ku, Tokyo 151-8543, Japan