章节 5

代数和算术运算

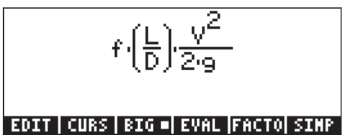
代数对象，或简称代数，是可以根据代数规则进行操作，操作和组合的任何数字，变量名称或代数表达式。 代数对象的示例如下：

- 数字: 12.3, 15.2_m, ‘π’,‘e’,‘i’
- 变量名称: ‘a’, ‘ux’, ‘width’, etc.
- 表达式: ‘p*D^2/4’,’f*(L/D)*(V^2/(2*g))’
- 等式: ‘Q=(Cu/n)*A(y)*R(y)^(2/3)*So^0.5’

输入代数对象

可以通过将单引号之间的对象直接键入堆栈级别1或使用公式 ↱ EQW . 来创建代数对象。 例如，要将代数对象' π * D ^ 2/4'直接输入堆栈级别1，请使用: ' ↰ π × ALPHA D y^x 2 ÷ 4 ENTER . 接下来显示ALG模式（左侧）和RPN模式（右侧）的结果屏幕：

代数对象也可以在Equation Writer中构建，然后发送到堆栈。 方程编写器的操作在章节2中描述。作为练习，在等式编写器中构建以下代数对象：



构建对象后，按下以在堆栈中显示它（ALG和RPN模式如下所示）：

使用代数对象进行简单操作

代数对象可以被加，减，乘，除（除零），提升到幂，用作各种标准函数（指数，对数，三角，双曲线等）的参数，就像你想要的任何真实的或 复数。

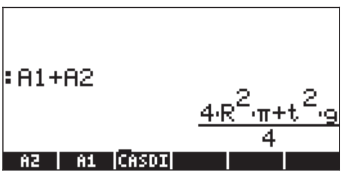
为了演示代数对象的基本操作，让我们创建几个对象，比如' π * R ^ 2'和'g * t ^ 2/4'，并将它们存储在变量A1和A2中（参见章节2以了解如何 创建变量并在其中存储值）。 以下是在ALG模式下存储变量A1的击键：['] [←] π [×] [ALPHA] [R] [Yˣ] [2] [▶] [STO▶] [ALPHA] [A] [1] [ENTER], resulting in:

对应于RPN模式的击键是：
['] [←] π [×] [ALPHA] [R] [Yˣ] [2] [ENTER] [ALPHA] [A] [1] [STO▶]

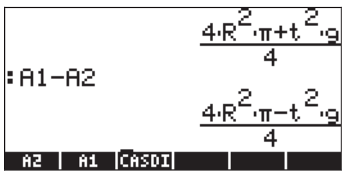
存储变量A2并按下键后，屏幕将显示如下变量：

在ALG模式下，以下按键将显示包含在变量 [A1] and [A2] 中的代数的多个操作（按下 [VAR] 以恢复变量菜单）：

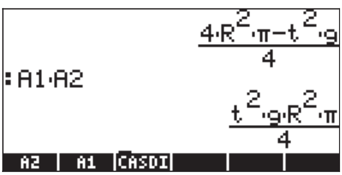
[A1] [+] [A2] [ENTER]



[A1] [−] [A2] [ENTER]

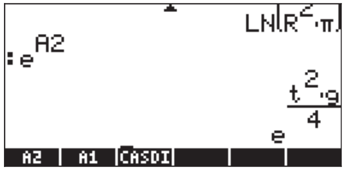


[A1] [×] [A2] [ENTER]



[A1] [÷] [A2] [ENTER]

↱ LN A1 ← eˣ A2



如果使用以下按键，则在RPN模式下获得相同的结果：

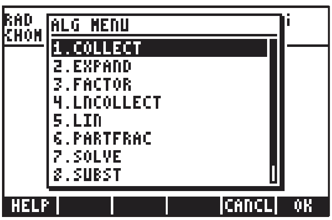
A1 A2 + EVAL A1 A2 − EVAL

A1 A2 × EVAL A1 A2 ÷ EVAL

A1 ↱ LN EVAL A2 ← eˣ EVAL

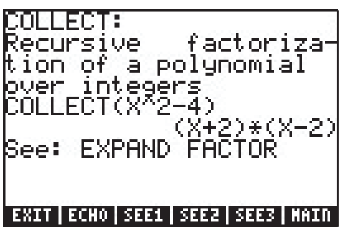
ALG菜单中的功能

使用按键序列 ↱ ALG (associated with the 4 key). 可以使用 ALG（代数）菜单。 系统标志117设置为CHOOSE框，ALG菜单显示以下功能：



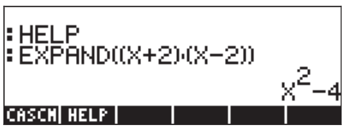
不是列出本手册中每个功能的描述，而是邀请用户使用计算器的帮助工具查找描述: TOOL NXT HELP ENTER . 要查找特定函数，请键入函数的第一个字母。 例如，对于函数COLLECT，我们键入 ALPHA C , 然后使用向上和向下箭头键 ▲ ▼ , 在帮助窗口中找到COLLECT.

要完成操作，请按OK. 这是功能COLLECT的帮助屏幕：



我们注意到，在屏幕底部，行见：EXPAND FACTOR建议链接到其他帮助工具条目，功能EXPAND和FACTOR。 要直接移动到这些条目，请按软键菜单键 SEE1 表示EXPAND，然后按 SEE2 表示FACTOR. 按 SEE1可显示EXPAND的以下信息:

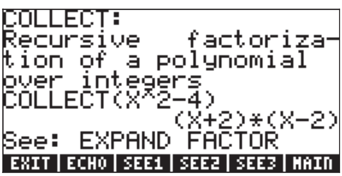
帮助设施
通过TOOL NEXT CASCMD可访问的帮助工具允许您浏览所有CAS命令。 它不仅提供有关每个命令的信息，还提供了其应用程序的示例。 按下 ECHO 软菜单键可将此示例复制到堆栈中。 例如，对于上面显示的EXPAND条目，请按 ECHO 软菜单键将以下示例复制到堆栈中（按 ENTER 执行命令）



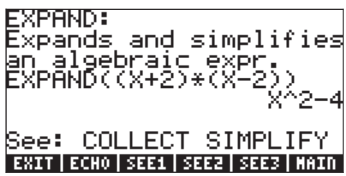
我们留给用户探索可用的CAS功能列表。 以下是几个例子:

帮助工具将显示有关命令的以下信息:

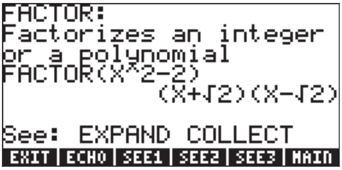
COLLECT:



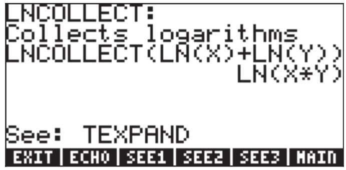
EXPAND:



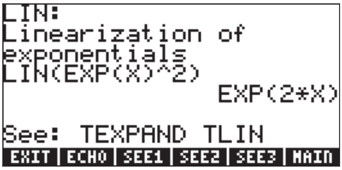
FACTOR:



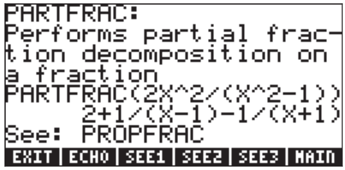
LNCOLLECT:



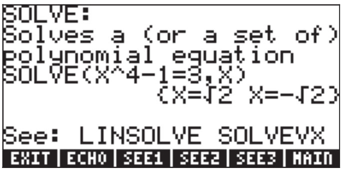
LIN:



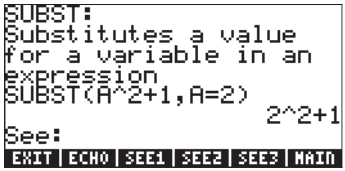
PARTFRAC:



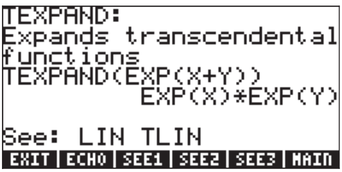
SOLVE:



SUBST:



TEXPAND:



注意：回想一下，要在RPN模式下使用这些或任何其他功能，

你必须首先输入参数，然后输入函数。 例如，

TEXPAND的示例，在RPN模式下将设置为：

['] [←] e^x [+] [ALPHA] [X] [+] [ALPHA] [Y] [ENTER]

此时，从菜单ALG中选择功能TEXPAND（或直接从目录中选择）

[→] CAT), 完成操作.

代数表达式中的其他替代形式

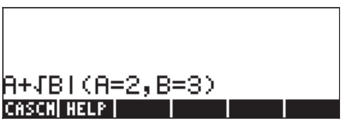
功能SUBST（如上所示）用于替换表达式中的变量。 第二种替换形式可以通过使用 ↱ | （与I键相关联）。 例如，在ALG模式下，将显示以下条目在表达式x + x2中替换值x = 2。 左边的图显示了在按下ENTER之前输入表达式的方法（替换值，x = 2，必须括在括号中）。 按下 ENTER 键后，结果显示在右侧图中：

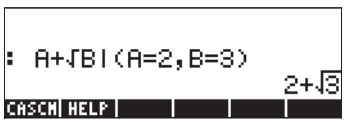
在RPN模式下，这可以通过首先输入表达式where来完成
将执行替换（x + x2），然后是包含替换变量，空格和要替换的值的列表（参见章节8），即{x 2}。 最后一步是按下击键组合：↱ | .

所需的击键如下：

' ALPHA ↰ X + ALPHA ↰ X Yˣ 2 ENTER
↰ {} ALPHA ↰ X SPC 2 ENTER ↱ | ENTER

在ALG模式下，可以替换多个变量，如以下示例所示（在按下ENTER之前和之后显示）

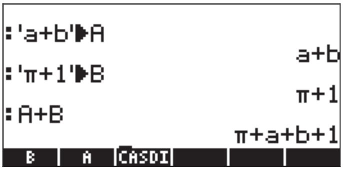




在RPN模式下，也可以一次替换多个变量，如下例所示。 回想一下，RPN模式使用变量名称和值列表进行替换。

另一种替换方法包括在计算器变量中定义替换表达式并将变量的名称放在原始表达式中。 例如，在ALG模式下，存储以下变量：

然后，输入表达式A + B：



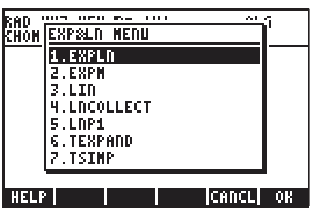
输入的最后一个表达式在按下ENTER键后自动计算，产生上面显示的结果。

具有超越功能的操作

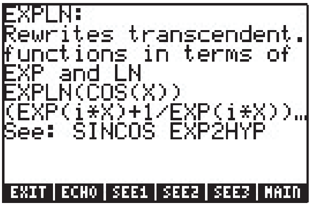
计算器提供了许多函数，可用于根据三角函数或指数函数替换包含对数，指数，三角函数和双曲函数的表达式。 包含替换三角函数的函数的菜单可以通过按下右移键后跟8键直接从键盘获得，即，↱ TRIG． 该键与左移键的组合，即，↱ EXP&LN，生成一个菜单，允许您根据指数或自然对数函数替换表达式。 在接下来的部分中，我们将更详细地介绍这些菜单。

使用log-exp函数进行扩展和分解

↰ EXP&LN 生成以下菜单：

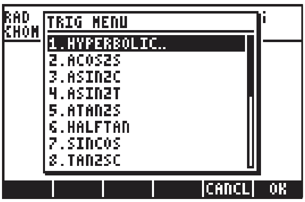


有关这些命令的信息和示例，请参见计算器的帮助工具。 EXP & LN菜单中列出的一些命令，即LIN， LNCOLLECT和TEXPAND也包含在前面提到的ALG菜单中。 功能LNP1和EXPM在菜单HYPERBOLIC中的MTH菜单下引入（参见章节2）。 唯一剩下的功能是EXPLN。 其描述显示在左侧，帮助工具的示例显示在右侧：



使用三角函数进行扩展和分解

TRIG菜单，由使用 (→) TRIG 触发, 显示以下功能：



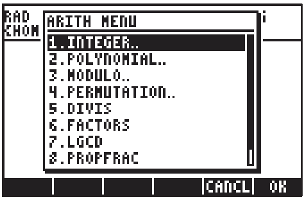
这些函数允许通过替换某些类别来简化表达式另一个的三角函数。 例如，功能ACOS2S允许将函数arccosine（acos（x））替换为其表达式arcsine（asin（x））。

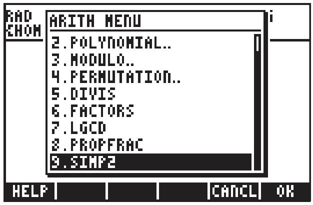
这些命令的描述和它们的应用程序的例子是可在计算器的帮助工具中找到(TOOL NXT HELP). 邀请用户浏览此工具以查找有关TRIG菜单中命令的信息。

请注意，TRIG菜单中的第一个命令是HYPERBOLIC菜单，其功能在第2节中介绍。

ARITHMETIC菜单中的功能

ARITHMETIC菜单包含许多用于数论中特定应用的子菜单（整数，多项式等），以及适用于一般算术运算的许多函数。 ARITHMETIC菜单通过击键组合 ARITH (associated with the / key). 触发。 系统标志117设置为CHOOSE框， ARITH 显示以下菜单：

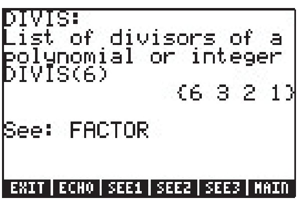




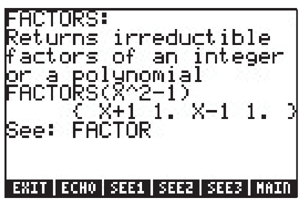
在此菜单列表中，选项5到9（DIVIS，FACTORS，LGCD，PROPFRAC，SIMP2）对应于适用于整数或多项式的常用函数。 其余选项（1. INTEGER，2.POLYNOMIAL，3。MODULO和4. PERMUTATION）实际上是适用于特定数学对象的函数的子菜单。 当系统标志117设置为SOFT菜单时，子菜单（选项1到4）和普通功能（选项5到9）之间的区别变得清晰。 在这些情况下，激活ARITHMETIC菜单 (ARITH), 会产生：

接下来，我们在ARITHMETIC菜单中显示选项5到9的功能的帮助工具条目 (TOOL NXT HELP):

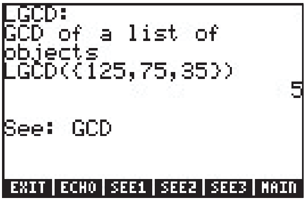
DIVIS:

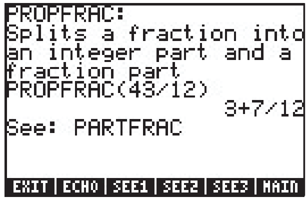


FACTORS:

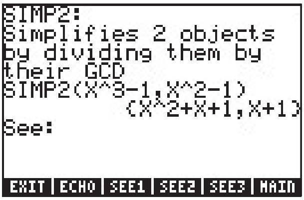


LGCD（最大公分母）：PROPFRAC（适当分数）





SIMP2:



与ARITHMETIC子菜单相关的功能：
INTEGER，POLYNOMIAL，MODULO和PERMUTATION如下：

INTEGER menu

EULER	整数的数量<n，与n的co -prime
IABCUV	求解au + bv = c，a，b，c =整数n-th伯
IBERNOULLI	努利数
ICHINREM	中文提醒整数
IDIV2	欧几里德分为两个整数
IEGCD	返回u，v，这样au + bv = gcd（a，b）
IQUOT	欧几里德两个整数的商
IREMAINDER	欧几里德的两个整数的余数
ISPRIME?	测试整数是否为素数
NEXTPRIME	给定整数的下一个素数
PA2B2	素数作为复数的平方范数
PREVPRIME	给定整数的前一个素数

POLYNOMIAL menu

ABCUV	Bézout多项式方程（au + bv = c）
CHINREM	中文余数为多项式
CYCLOTOMIC	第n个分圆多项式
DIV2	欧几里德划分两个多项式
EGDC	从au + bv = gcd（a，b）返回u，v

FACTOR	将整数或多项式归纳
FCOEF	生成给定根和多重性的分数
FROOTS	给出一个分数返回根和多重性
GCD	2个数或多项式n次Hermite多项式的最大公约数
HERMITE	霍纳对多项式的评价
HORNER	
LAGRANGE	拉格朗日多项式插值
LCM	2个数的最低公倍数或多项式n次勒让德多项式
LEGENDRE	给定部分的部分分数分解
PARTFRAC	(缺少帮助设施条目)
PCOEF	返回Q（x-a）中的Q（x-a）＝P（x），泰勒多项式
PTAYL	两个多项式的欧几里德商
QUOT	2个多项式的Sylvester矩阵的行列式
RESULTANT	
REMAINDER	欧几里德提醒两个多项式
STURM	多项式的Sturm序列
STURMAB	在边界之间签署低边界和零数

MODULO menu

ADDTMOD	添加两个表达式模数电流模数
DIVMOD	将2个多项式除以模电流模量
DIV2MOD	具有模系数的2个多项式的欧几里德除法
EXPANDMOD	扩展/简化多项式模电流模量
FACTORMOD	对2个多项式的多项式模电流模数GCD进行因式分解
GCDMOD	模数电流模数整数模电流模量的倒数
INVMOD	(不在帮助设施中输入)
MOD	将模数设置更改为指定值
MODSTO	两个多项式模电流模量的乘法
MULTMOD	
POWMOD	将多项式提高到功率模电流模量
SUBTMOD	减去2个多项式模电流模量

ARITHMETIC菜单的应用

本节旨在介绍应用ARITHMETIC菜单功能所需的一些背景知识。 接下来给出关于多项式，多项式分数和模运算的主题的定义。 下面给出的示例与计算器设置（ALG或RPN）无关

模块化算术

考虑一个整数计数系统，它周期性地循环自身并重新开始，例如时钟中的小时数。 这种计数系统称为环。 由于环中使用的整数数是有限的，因此该环中的算术称为有限算术。 让我们的有限整数系统由数字0,1,2,3，...，n-1，n组成。 我们也可以将这个计数系统的算法称为模数n的模运算。 在时钟小时的情况下，模数是12.（如果使用时钟中的小时模数运算，我们将不得不使用整数0,1,2,3，...，10， 11，而不是1,2,3，...，11,12）。

模运算中的操作

模数n的模运算中的加法是正整数，遵循以下规则：如果j和k是任意两个非负整数，均小于n，如果j +k≥n，则j + k定义为j + KN。例如，在时钟的情况下，即，对于n = 12,6 + 9“=”3。为了将这种“等式”与无限算术等式区分开，使用符号≡代替等号，并且数字之间的关系被称为一致而不是平等。因此，对于前面的例子，我们将写6 +9≡3（mod 12），并将此表达式读作“六加九与三，十二模一致。”如果数字表示自午夜以来的小时数，例如，同余6 +9≡3（mod 12），可以解释为“午夜过后第九个小时的六个小时将是中午三个小时。”其他可以在模数12算术中定义的总和是：2 +5≡ 7（mod 12）; 2 +10≡0（mod 12）; 7 +5≡0（mod 12）;等等。

减法的规则是如果j-k <0，那么j-k被定义为j-k + n。 因此，8-10≡2（mod 12）被读为“8减10与2相同，模12'。模数12算术中减法的其他例子将是10-5≡5（mod 12）; 6-9≡9（mod 12）; 5 - 8≡9（mod 12）; 5-10≡7（mod 12）;等等。
乘法遵循以下规则：如果j k> n，则j k=m n+ r，其中m和r是非负整数，均小于n，则j k≡r（mod n）。

如果j k> n，则在模数n算术中乘以j乘以k的结果实质上是无穷算术中j k/ n的整数余数。 例如，在模数12算术中，我们有7 3= 21 = 12 + 9，（或，7 = 3/12 = 21/12 = 1 + 9/12，即21/12的整数提醒为9）。 我们现在可以写7 3≡9（mod 12），并将后一个结果读作"七次三次与九次一致，十二次模数"。
除法运算可以用乘法定义如下，r /k≡j（mod n），if，j k≡r（mod n）。这意味着r必须是j k/ n的余数。 例如，9 /7≡3（mod 12），因为7 3≡9（mod 12）。 模块化算术中不允许某些分区。 例如，在模数12算术中，您不能定义5/6（mod 12），因为6的乘法表不显示模数12算术中的结果5。 该乘法表如下所示：

6*0 (mod 12)	0		6*6 (mod 12)	0
6*1 (mod 12)	6		6*7 (mod 12)	6
6*2 (mod 12)	0		6*8 (mod 12)	0
6*3 (mod 12)	6		6*9 (mod 12)	6
6*4 (mod 12)	0		6*10 (mod 12)	0
6*5 (mod 12)	6		6*11 (mod 12)	6

有限算术环的形式定义

The expression $a \equiv b \ (mod\ n)$ is interpreted as "<u>*a* is congruent to *b*, modulo *n*</u>," and holds if *(b-a)* is a multiple of *n*. With this definition the rules of arithmetic simplify to the following: 表达式a≡b（mod n）被解释为"a与b一致，模n"，并且如果（b-a）是n的倍数则保持。

If $a \equiv b \ (mod\ n)$ and $c \equiv d \ (mod\ n)$,

then

$$a+c \equiv b+d \ (mod\ n),$$
$$a-c \equiv b - d \ (mod\ n),$$
$$a\times c \equiv b\times d \ (mod\ n).$$

对于分部，请遵循前面提到的规则。 For example, 17 ≡ 5 (mod 6), and 21 ≡ 3 (mod 6). Using these rules, we can write:

17 + 21 ≡ 5 + 3 (mod 6) => 38 ≡ 8 (mod 6) => 38 ≡ 2 (mod 6)
17 – 21 ≡ 5 - 3 (mod 6) => -4 ≡ 2 (mod 6)
17 × 21 ≡ 5 × 3 (mod 6) => 357 ≡ 15 (mod 6) => 357 ≡ 3 (mod 6)

请注意，只要“同余”符号右侧的结果产生的结果大于模数（在这种情况下，n = 6），您总是可以从该结果中减去模数的倍数，将其简化为小于模数的数字。因此，第一种情况8（mod6）中的结果简化为2（mod6），并且第三种情况15（mod6）的结果简化为3（mod6）。混乱？好吧，如果你让计算器处理这些操作，那就不行了。因此，请阅读以下部分以了解如何在计算器中对有限算术环进行操作。

计算器中的有限算术环

我们一直定义了有限的算术运算，结果总是正的。计算器中的模运算系统被设置为模数环n包括数字*-n/2+1, …,-1, 0, 1,…,n/2-1, n/2,*如果n是 偶数， *–(n-1)/2, -(n-3)/2,…,-1,0,1,…,(n-3)/2, (n-1)/2,*如果n是奇数. 例如，对于n = 8（偶数），计算器中的有限算术环包括数字：（- 3，-2，-1,0,1,3,4），而对于n = 7（奇数）， 相应的计算器的有限算术环由（-3，-2，-1,0,1,2,3）给出。

计算器中的模块化算术

要在计算器中启动模块化算术菜单，请选择ARITHMETIC菜单中的MODULO子菜单(← ARITH)． 可用菜单包括功能：ADDTMOD，DIVMOD，DIV2MOD，EXPANDMOD，FACTORMOD，GCDMOD，INVMOD，MOD，MODSTO，MULTMOD，POWMOD和SUBTMOD。前面部分提供了这些功能的简要描述。接下来我们介绍这些功能的一些应用。

设置模数（或**MODULO**）

计算器包含一个名为MODULO的变量，该变量放在{HOME CASDIR}目录，将存储模数运算中使用的模数的大小。

MODULO的默认值为13.要更改MODULO的值，可以将新值直接存储在子目录{HOME CASDIR}中的变量MODULO中。或者，可以使用MODSTO函数存储新的MODULO值。

带数字的模块化算术运算

要使用模运算对函数进行加，减，乘，除和提升，您将使用函数ADDTMOD，SUBTMOD，MULTMOD，DIV2MOD和DIVMOD（用于除法）和POWMOD。在RPN模式下，您需要输入要操作的两个数字，用[ENTER]或[SPC]分隔

输入，然后按相应的模运算功能。 例如，使用12的模数，请尝试以下操作：

ADDTMOD examples

$6+5 \equiv -1 \pmod{12}$ $6+6 \equiv 0 \pmod{12}$ $6+7 \equiv 1 \pmod{12}$

$11+5 \equiv 4 \pmod{12}$ $8+10 \equiv -6 \pmod{12}$

SUBTMOD examples

$5 - 7 \equiv -2 \pmod{12}$ $8 - 4 \equiv 4 \pmod{12}$ $5 - 10 \equiv -5 \pmod{12}$

$11 - 8 \equiv 3 \pmod{12}$ $8 - 12 \equiv -4 \pmod{12}$

MULTMOD examples

$6 \cdot 8 \equiv 0 \pmod{12}$ $9 \cdot 8 \equiv 0 \pmod{12}$ $3 \cdot 2 \equiv 6 \pmod{12}$

$5 \cdot 6 \equiv 6 \pmod{12}$ $11 \cdot 3 \equiv -3 \pmod{12}$

DIVMOD examples

$12/3 \equiv 4 \pmod{12}$ $12/8 \pmod{12}$ does not exist

$25/5 \equiv 5 \pmod{12}$ $64/13 \equiv 4 \pmod{12}$

$66/6 \equiv -1 \pmod{12}$

DIV2MOD examples

$2/3 \pmod{12}$ does not exist

$26/12 \pmod{12}$ does not exist

$125/17 \pmod{12}$ ½ 1 with remainder = 0

$68/7$ ½ $-4 \pmod{12}$ with remainder = 0

$7/5$ ½ $-1 \pmod{12}$ with remainder = 0

Note：DIVMOD提供模块化除法j / k（mod n）的商，而DIMV2MOD不仅提供商而且提供模块化除法j / k（mod n）的余数。

POWMOD examples

$2^3 \equiv -4 \pmod{12}$ $3^5 \equiv 3 \pmod{12}$ $5^{10} \equiv 1 \pmod{12}$

$11^8 \equiv 1 \pmod{12}$ $6^2 \equiv 0 \pmod{12}$ $9^9 \equiv -3 \pmod{12}$

在上面所示的模运算操作的例子中，我们使用了不一定属于环的数字，即诸如66,125,17等的数字。计算器将在对它们进行操作之前将这些数字转换为环形数。 您还可以使用EXPANDMOD函数将任意数字转换为振铃数。 例如，

EXPANDMOD(125) ≡ 5 (mod 12)
EXPANDMOD(17) ≡ 5 (mod 12)
EXPANDMOD(6) ≡ 6 (mod 12)

数字的模数逆

设数k属于模数n的有限算术环，则k的模逆，即1 / k（mod n），是数j，使得j k≡1（mod n）。 可以通过使用ARITHMETIC菜单的MODULO子菜单中的函数INVMOD来获得数字的模逆。 例如，在模数12算术中：

1/6 (mod 12) does not exist.
1/7 ≡ -5 (mod 12)
1/11 ≡ -1 (mod 12)
1/5 ≡ 5 (mod 12)
1/3 (mod 12) does not exist.

The MOD operator

MOD运算符用于获得对应于给定整数的给定模数的余数。 在纸上，该操作被写为m mod n = p，并被读作“m modulo n等于p”。 例如，要计算15 mod 8，请输入：

- ALG mode: 1 5 MOD 8 ENTER
- RPN mode: 1 5 ENTER 8 ENTER MOD

The result is 7, i.e., 15 mod 8 = 7. Try the following exercises:
18 mod 11 = 7　　23 mod 2 =1　　40 mod 13 = 1
23 mod 17 = 6　　34 mod 6 = 4

用于编程目的的MOD函数的一个实际应用是确定整数是奇数还是偶数，因为n mod 2 = 0，如果n是偶数，并且n模式2 = 1，如果n是奇数。 它还可以用于确定整数m是否是另一个整数n的倍数，如果是m mod n = 0的情况。

注意：有关其他模块化算法的说明和示例，请参阅计算器中的帮助工具。 许多这些函数适用于多项式。 有关使用多项式的模运算的信息，请参阅数论的教科书。

多项式

多项式是由一个或多个包含给定变量的递减幂的项组成的代数表达式。 例如，'X ^ 3 + 2 * X ^ 2-3 * X + 2'是X中的三阶多项式，而'SIN（X）^ 2-2'是SIN中的二阶多项式（X）。 之前介绍了ARITHMETIC菜单中多项式相关函数的列表。 接下来提供关于多项式的一些一般定义。 在这些定义中，A（X），B（X），C（X），P（X），Q（X），U（X），V（X）等是多项式。

- 多项式分数：分子和分母是多项式的分数，比方说，C(X) = A(X)/B(X)
- 多项式的根或零：values of X for which P(X) = 0
- 分数的极点：分母的根
- 根或极的多重性：根出现的次数，例如P（X）=（X + 1）2（X-3）具有根{-1, 3} 具有多重性{2,1}
- 分圆多项式（Pn（X））：EULER（n）阶的多项式，其根是单位的原始第n个根，例如， $P_2(X) = X+1$, $P_4(X) = X^2+1$
- Bézout的多项式方程: A(X) U(X) + B(X)V(X) = C(X)

接下来提供多项式应用的具体示例。

使用多项式的模运算

与我们在前一节中为数字定义有限算术环的方法相同，我们可以为给定多项式为模的多项式定义有限算术环。 例如，我们可以将某个多项式P（X）写为P（X）= X（mod X2），或另一个多项式Q（X）= X + 1（mod X-2）。

多项式P（X）属于多项式模M（X）的有限算术环，如果存在第三多项式Q（X），则（P（X） - Q（X））是M的倍数 （X）。 然后我们写：P（X）≡ Q（X）（mod M（X））。 后面的表达式被解释为“P（X）与Q（X）一致，模M（X）”。

The CHINREM function

CHINREM代表CHINese REMainder。 在该命令中编码的操作使用中国剩余定理解决了两个同余系统。 此命令可以与多项式一起使用，也可以与整数一起使用

数字（功能ICHINREM）。 输入由两个向量[expression_1，modulo_1]和[expression_2，modulo_2]组成。 输出是包含[expression_3，modulo_3]的向量，其中modulo_3与乘积相关（modulo_1） （modulo_2）。

示例：: CHINREM([X+1, X^2-1],[X+1,X^2]) =[X+1,-(X^4-X^2)]

关于整数的中国剩余定理的陈述

如果m1，m2，...，mr是自然数，每对都是相对素数，并且a1，a2，...，ar是任意整数，那么有一个整数x同时满足同余：x≡a1（mod m1），x≡a2（mod m2），...，x≡ar（mod mr）。 另外，如果x = a是任何解，那么所有其他解都与等于乘积m1·m2·...... mr的模数一致。

The EGCD function

EGCD代表扩展最大公约数。 给定两个多项式A（X）和B（X），函数EGCD产生多项式C（X），U（X）和V（X），因此C(X), U(X), and V(X), so that C(X) = U(X)*A(X) + V(X)*B(X). For example, for A(X) = X^2+1, B(X) = X^2-1, EGCD(A(X),B(X)) = {2, 1, -1}. i.e., 2 = 1*(X^2+1')-1*(X^2-1). Also, EGCD('X^3-2*X+5','X') = { 5,1,-(X^2-2)}, i.e., 5 = – (X^2-2)*X + 1*(X^3-2*X +5).

The GCD function

函数GCD（最大公分母）可用于获得两个多项式的最大公分母或两个相同长度的多项式列表。 在使用GCD之前，两个多项式或多项式列表将被放置在堆栈级别2和1中。 结果将是多项式或表示两个多项式或每个多项式列表的最大公分母的列表。 在RPN模式下的示例如下（计算器设置为精确模式）：

'X^3-1' ENTER 'X^2-1' ENTER GCD Results in: 'X-1'

{'X^2+2*X+1','X^3+X^2'} ENTER {'X^3+1','X^2+1'} ENTER GCD results in {'X+1' 1}

The HERMITE function

函数HERMITE [HERMI]使用整数k作为参数，并返回k度的Hermite多项式。 Hermite多项式，Hek（x）定义为

$$He_0 = 1, \quad He_n(x) = (-1)^n e^{x^2/2} \frac{d^n}{dx^n}(e^{-x^2/2}), \quad n = 1,2,...$$

Hermite多项式的另一种定义是

$$H_0 * = 1, \quad H_n * (x) = (-1)^n e^{x^2} \frac{d^n}{dx^n}(e^{-x^2}), \quad n = 1,2,...$$

where d^n/dx^n =相对于x的n阶导数。 这是计算器中使用的定义。

示例：3阶和5阶的Hermite多项式由下式给出：

HERMITE(3) = '8*X^3-12*X',

And HERMITE(5) = '32*x^5-160*X^3+120*X'.

The HORNER function

函数HORNER通过因子（X-a）产生多项式P（X）的Horner除法或合成除法。函数的输入是多项式P（X）和数字a。 该函数返回由将P（X）除以（X-a），a的值和P（a）的值得到的商多项式Q（X）。 换句话说，P（X）= Q（X）（X-a）+ P（a）。 例如，
HORNER('X^3+2*X^2-3*X+1',2) = {'X^2+4*X+5', 2, 11}. We could, therefore, write $X^3+2X^2-3X+1 = (X^2+4X+5)(X-2)+11$. A second example:
HORNER('X^6-1',-5)= {'X^5-5*X^4+25*X^3-125*X^2+625*X-3125',-5, 15624}
i.e., $X^6-1 = (X^5-5*X^4+25X^3-125X^2+625X-3125)(X+5)+15624$.

The variable VX

计算器的{HOME CASDIR}目录中存在一个名为VX的变量，该目录默认采用"X"的值。 这是代数和微积分应用的首选自变量的名称。 避免在程序或方程式中使用变量VX，以免与CAS的VX混淆。 例如，如果需要参考速度的x分量，则可以使用vx或Vx。 有关CAS变量的其他信息，请参阅附录C.

The LAGRANGE function

函数LAGRANGE需要具有两行和n的矩阵作为输入列。 矩阵存储形式[[x1，x2，...，xn] [y1，y2，...，yn]]的数据点。 函数LAGRANGE的应用产生从中扩展的多项式

$$p_{n-1}(x) = \sum_{j=1}^{n} \frac{\prod_{k=1,k\neq j}^{n}(x-x_k)}{\prod_{k=1,k\neq j}^{n}(x_j-x_k)} \cdot y_j.$$

For example, for n = 2, we will write:

$$p_1(x) = \frac{x-x_2}{x_1-x_2}\cdot y_1 + \frac{x-x_1}{x_2-x_1}\cdot y_2 = \frac{(y_1-y_2)\cdot x + (y_2\cdot x_1 - y_1\cdot x_2)}{x_1-x_2}$$

Check this result with your calculator:
LAGRANGE([[x1,x2],[y1,y2]]) = '((y1-y2)*X+(y2*x1-y1*x2))/(x1-x2)'.

Other examples: LAGRANGE([[1, 2, 3][2, 8, 15]]) = '(X^2+9*X-6)/2'
LAGRANGE([[0.5,1.5,2.5,3.5,4.5][12.2,13.5,19.2,27.3,32.5]]) =
'-(.1375*X^4+ -.7666666666667*X^3+ - .74375*X^2 +
1.991666666667*X-12.92265625)'.

Note: 矩阵在第10节中介绍。

The LCM function

函数LCM（最小公倍数）获得两个多项式的最小公倍数或相同长度的多项式列表。 Examples:

LCM('2*X^2+4*X+2' ,'X^2-1') = '(2*X^2+4*X+2)*(X-1)'.
LCM('X^3-1','X^2+2*X') = '(X^3-1)*(X^2+2*X)'

The LEGENDRE function

n阶勒让德多项式是求解微分方程的多项式函数

$$(1-x^2)\cdot\frac{d^2y}{dx^2} - 2\cdot x\cdot\frac{dy}{dx} + n\cdot(n+1)\cdot y = 0$$

要获得n阶勒让德多项式，请使用LEGENDRE（n），例如，

LEGENDRE(3) = '(5*X^3-3*X)/2'
LEGENDRE(5) = '(63*X ^5-70*X^3+15*X)/8'

The PCOEF function

给定包含多项式根的数组，函数PCOEF生成包含相应多项式系数的数组。 系数对应于自变量的递减顺序。例如： PCOEF([-2,–1,0,1,1,2]) = [1. –1. –5. 5. 4. –4. 0.], 表示多项式X^6-X^5-5X^4+5X^3+4X^2-4X.

The PROOT function

给定包含多项式系数的数组，按递减顺序，函数PROOT提供多项式的根。 例子，来自

X^2+5X-6 =0, PROOT([1, –5, 6]) = [2. 3.].

The PTAYL function

给定多项式P（X）和数a，函数PTAYL用于获得表达式Q（X-a）= P（X），即，得到（X-a）的幂的多项式。 这也称为泰勒多项式，函数名称Polynomial和TAYLor遵循：

For example, PTAYL('X^3-2*X+2',2) = 'X^3+6*X^2+10*X+6'.

实际上，您应该将此结果解释为意思

'(X-2) ^3+6*(X-2) ^2+10*(X-2) +6'.

让我们使用替换来检查：'X = x - 2'。 我们恢复原始多项式，但就小写x而不是大写x而言。

The QUOT and REMAINDER functions

函数QUOT和REMAINDER分别提供商Q（X）和余数R（X），由分割两个多项式P_1（X）和P_2（X）得到。 换句话说，它们从P_1（X）/ P_2（X）= Q（X）+ R（X）/ P_2（X）提供Q（X）和R（X）的值。 例如，

QUOT(X^3-2*X+2, X-1) = X^2+X-1

REMAINDER(X^3-2*X+2, X-1) = 1.

因此，我们可以写： (X^3-2X+2)/(X-1) = X^2+X-1 + 1/(X-1).

Note: 您可以使用PROPFRAC获得后者的结果：
PROPFRAC('(X^3-2*X+2)/(X-1)') = 'X^2+X-1 + 1/(X-1)'.

EPSX0功能和CAS变量EPS

变量ε（epsilon）通常用于数学教科书中以表示非常小的数字。 当您使用EPSX0功能时，计算器的CAS会创建一个变量EPS，默认值为0.0000000001 = 10^{-10}。 如果您更喜欢EPS的不同值，则可以在创建后更改此值。 当应用于多项式时，函数EPSX0将用零替换绝对值小于EPS的所有系数。 功能EPSX0在ARITHMETIC菜单中不可用，必须从功能目录（N）访问。 例：

EPSX0('X^3-1.2E-12*X^2+1.2E-6*X+6.2E-11)= 'X^3-0*X^2+.0000012*X+0'.
With EVAL: 'X^3+.0000012*X'.

The PEVAL function

函数PEVAL（多项式估计）可用于评估多项式 $p(x) = a_n \cdot x^n + a_{n-1} \cdot x^{n-1} + \ldots + a_2 \cdot x^2 + a_1 \cdot x + a_0$, 给定系数数组[$a_n$, a_{n-1}, … a_2, a_1, a_0] 和值 x_0. 结果是评估 $p(x_0)$. 功能PEVAL在ARITHMETIC菜单中不可用，必须从功能目录(,N).

Example:

PEVAL([1,5,6,1],5) = 281.

The TCHEBYCHEFF function

函数TCHEBYCHEFF（n）生成第一类的Tchebycheff（或Chebyshev）多项式，阶数n，定义为Tn（X） = cos（n arccos（X））。 如果整数n为负（n <0），则函数TCHEBYCHEFF（n）生成第二类Tchebycheff多项式，阶数n，定义为Tn（X） = sin（n arccos（X））/ sin（arccos） （X））。 例子：

TCHEBYCHEFF(3) = 4*X^3-3*X
TCHEBYCHEFF(-3) = 4*X^2-1

Fractions
可以使用ALG菜单（，×）中的函数EXPAND和FACTOR来扩展和分解分数。 For example:

EXPAND('(1+X)^3/((X-1)*(X+3))') = '(X^3+3*X^2+3*X+1)/(X^2+2*X-3)'
EXPAND('(X^2)*(X+Y)/(2*X-X^2)^2)') = '(X+Y)/(X^2-4*X+4)'
EXPAND('X*(X+Y)/(X^2-1)') = '(X^2+Y*X)/(X^2-1)'
EXPAND('4+2*(X-1)+3/((X-2)*(X+3))-5/X^2') =

'(2*X^5+4*X^4-10*X^3-14*X^2-5*X+30)/(X^4+X^3-6*X^2)'

FACTOR('(3*X^3-2*X^2)/(X^2-5*X+6)') = 'X^2*(3*X-2)/((X-2)*(X-3))'
FACTOR('(X^3-9*X)/(X^2-5*X+6)') = 'X*(X+3)/(X-2)'
FACTOR('(X^2-1)/(X^3*Y-Y)') = '(X+1)/((X^2+X+1)*Y)'

The SIMP2 function
函数SIMP2和PROPFRAC分别用于简化分数和生成适当的分数。 函数SIMP2将两个数字或多项式作为参数，表示有理分数的分子和分母，并返回简化的分子和分母。 For example: SIMP2('X^3-1','X^2-4*X+3') = { 'X^2+X+1','X-3'}.

The PROPFRAC function
PROPFRAC函数将有理分数转换为"适当"分数，即，如果可能的话，这种分解可以是添加到分数部分的整数部分。 For example:

PROPFRAC('5/4') = '1+1/4'
PROPFRAC('(x^2+1)/x^2') = '1+1/x^2'

The PARTFRAC function
PARTFRAC函数将有理分数分解为产生原始分数的部分分数。 For example:

PARTFRAC('(2*X^6-14*X^5+29*X^4-37*X^3+41*X^2-16*X+5)/(X^5-7*X^4+11*X^3-7*X^2+10*X)') =

'2*X+(1/2/(X-2)+5/(X-5)+1/2/X+X/(X^2+1))'

这种技术可用于计算有理分数的积分（参见关于微积分的章节）。

如果您激活复数模式，结果将是：

'2*X+(1/2/(X+i)+1/2/(X-2)+5/(X-5)+1/2/X+1/2/(X-i))'

The FCOEF function

在给定分数的根和极点的情况下，函数FCOEF用于获得有理分数。

Note: 如果给出有理分数为F（X）= N（X）/ D（X），则分数的根来自求解方程N（X）= 0，而极点来自求解方程D（X））= 0。

该函数的输入是一个向量，列出了根，然后是它们的多重性（即给定根重复了多少次），并且极点后面跟着它们的多重性表示为负数。 例如，如果我们想要创建一个分数，其根2具有多重性1，0具有多重性3，-5具有多重性2，极数1具有多重性2和-3具有多重性5，则使用：

FCOEF([2, 1, 0, 3, –5, 2, 1, –2, –3, –5]) = '(X--5)^2*X^3*(X-2)/(X+3)^5*(X-1)^2'

If you press EVAL ← ANS ENTER (or, simply EVAL, in RPN mode) you will get:

'(X^6+8*X^5+5*X^4-50*X^3)/(X^7+13*X^6+61*X^5+105*X^4-45*X^3-297*X^2-81*X+243)'

The FROOTS function

函数FROOTS获得分数的根和极点。 例如，将函数FROOTS应用于上面生成的结果将导致：[1-2。 -3 -5。 0 3. 2 1. -5 2.]。 结果显示极点后面是多重性作为负数，而根则是多重性作为正数。 在这种情况下，极点分别是具有多重性（2,5）的（1，-3），并且根分别是具有多重性（3,1,2）的（0,2，-5）。

Another example is: FROOTS('(X^2-5*X+6)/(X^5-X^2)')= [0 –2. 1 –1. 3 1. 2 1.]. i.e., poles = 0 (2), 1(1), and roots = 3(1), 2(1).

如果您选择了复数模式，那么结果将是：
[0 –2. 1 –1. – ((1+i*√3)/2) –1. – ((1–i*√3)/2) –1. 3 1. 2 1.].

使用多项式和分数的逐步操作

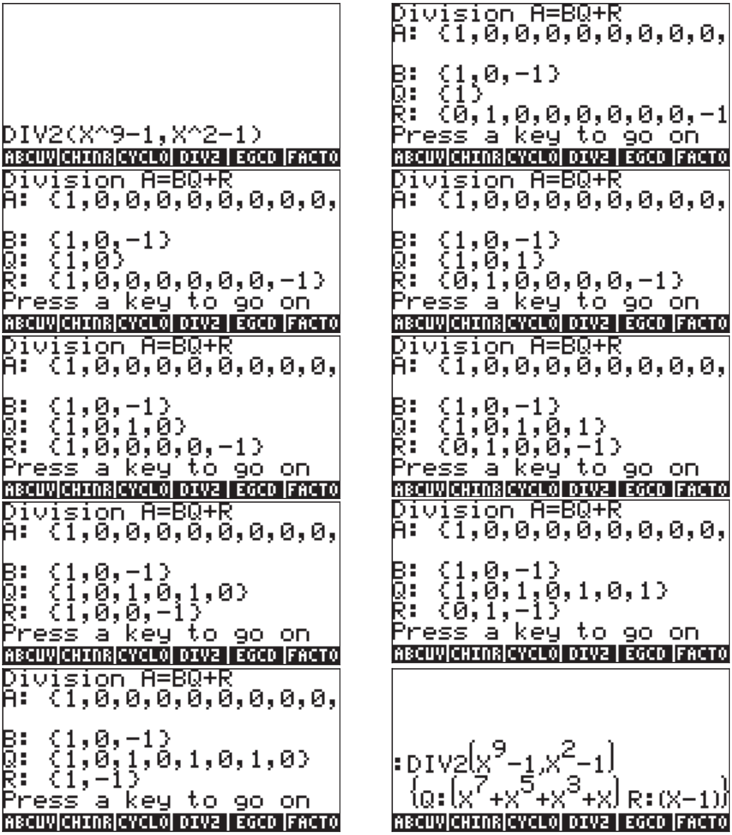
通过将CAS模式设置为Step / step，计算器将逐步显示多项式的分数或运算的简化。 这对于查看合成除法的步骤非常有用。

$$\frac{X^3 - 5X^2 + 3X - 2}{X - 2}$$

附录C中详细显示了除法的示例。以下示例显示了较长的合成除法：

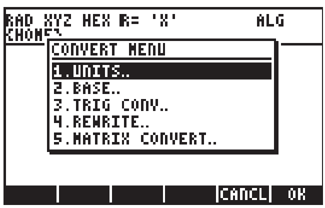
$$\frac{X^9 - 1}{X^2 - 1}$$

请注意，DIV2可从ARITH / POLYNOMIAL菜单中获得。



CONVERT菜单和代数运算

使用 CONVERT 键（6键）激活CONVERT菜单。 此菜单汇总了计算器中的所有转换菜单。 这些菜单的列表如下所示：



接下来显示每个子菜单中可用的功能。

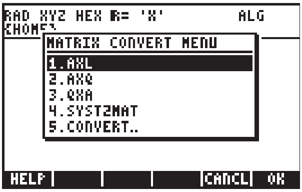
UNITS convert menu (Option 1)

UNITS . 获得的UNITS菜单相同。 该菜单的应用将在第3节中详细讨论。

BASE convert menu (Option 2)
↱ BASE 获得的UNITS菜单相同。 第19章详细讨论了该菜单的应用。

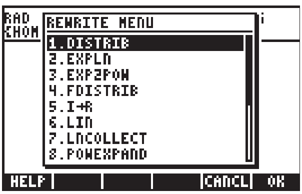
TRIGONOMETRIC convert menu (Option 3)
↱ TRIG 获得的TRIG菜单相同。 本章节详细讨论了该菜单的应用。

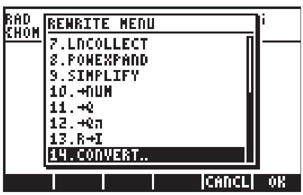
MATRICES convert menu (Option 5)
此菜单包含以下功能:



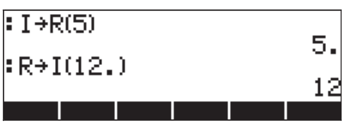
这些功能将在第10节中详细讨论。

REWRITE convert menu (Option 4)
此菜单包含以下功能:





Functions I→R and R→I 用于将数字从整数（I）转换为实数（R），反之亦然。显示整数不带尾随小数点，而表示整数的实数将有一个尾随小数点，例如，

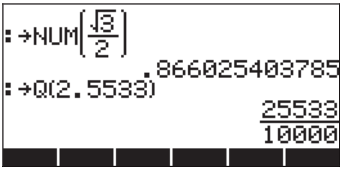


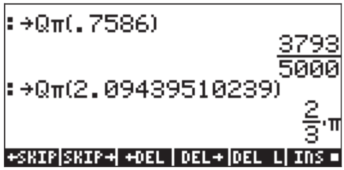
Function →NUM 与击键组合具有相同的效果， ⬚ →NUM (associated with the ENTER key).

Function →NUM 将符号结果转换为其浮点值。

Function →Q将浮点值转换为分数。

Function →Qπ 如果可以找到 π 的一小部分，将浮点值转换为 π 的一部分; 否则，它将数字转换为分数。 接下来显示这三个功能的示例。

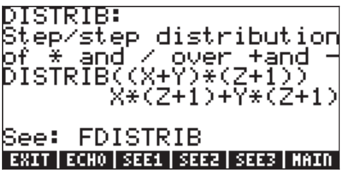




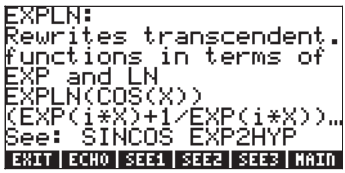
在REWRITE菜单中的函数中，函数

DISTRIB，EXPLN，EXP2POW，FDISTRIB，LIN，LNCOLLECT，POWEREXPAND和SIMPLIFY适用于代数表达式。 其中许多功能都在本章节中介绍。 但是，为了完整起见，我们在此提供这些功能的帮助设施条目。

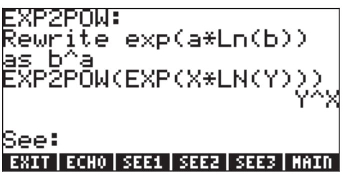
DISTRIB



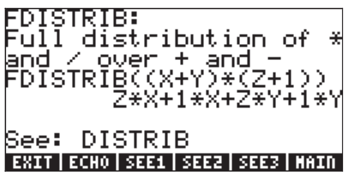
EXPLN



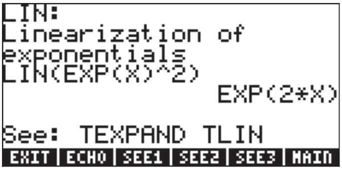
EXP2POW



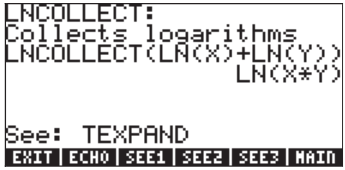
FDISTRIB



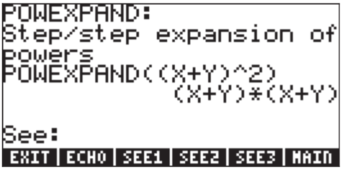
LIN



LNCOLLECT



POWEREXPAND



SIMPLIFY