

Computer Algebra
and
Mathematics
with
the HP40GS

Renée De Graeve
Version 3.0

March 10, 2011

Acknowledgments

It was not believed possible to write an efficient program for computer algebra all on one's own. But one bright person by the name of Bernard Parisse didn't know that...and did it!!!

This is his program for computer algebra (called ERABLE), built for the second time into an HP calculator.

The development of this calculator has led Bernard Parisse to modify his program somewhat so that the computer algebra functions could be edited and cause the appropriate results to be displayed in the Equation Editor.

Explore all the capabilities of this calculator, as set out in the following pages.

I would like to thank:

- Bernard Parisse for his invaluable counsel, his remarks on the text, his reviews, and for his ability to provide functions on demand both efficiently and graciously.
- Jean Tavenas for the concern shown towards the completion of the first version of this guide.

© 2005, Renée De Graeve

The reproduction, distribution and/or the modification of this document is authorised according to the terms of the GNU Free Documentation License, Version 1.1 or later, published by the Free Software Foundation, with no Invariant Sections, with no Front-Cover Texts, and with no Back-Cover Texts.

A copy of this license exists under the section entitled :
GNU Free Documentation License (chapter 9, p. 189).

Preface

The HP40G marked a new stage in the democratisation of the use of symbolic calculation, on the one hand by its competitive price, and on the other hand by making it possible to execute, step-by-step, the principle algorithms taught in mathematics at secondary schools and in the first years of University.

But it was still necessary to add adequate documentation, preferably written by a teacher of mathematics. That is what you find in this guide, written by Renée De Graeve. It contains, naturally, a complete reference of the functions for symbolic calculation, but also demonstrates, using examples taken from study for both certificate and diploma, how to take smart advantage of the calculating power of the HP40GS. The guide ends with two chapters dedicated to programming: the first for learning to program, and the second to illustrate the application of algorithmic language to arithmetic programs as taught in French tertiary schools.

This guide is an update for the HP40GS calculator, with some documentation improvements in addition to the HP40GS software and speed improvements.

Bernard Parisse

Lecturer at the University of Grenoble I

Chapter 1

To Begin

1.1 General Information

1.1.1 Starting the Calculator

Press the ON button.

The HOME screen is displayed.

As you work, you can use the ON button to cancel the current operation. In other words, it has the same function as CANCEL.

To turn the calculator off, press SHIFT plus ON (OFF).

If the calculator fails to respond after several presses of the ON (CANCEL) button, you can press ON and F3 simultaneously to reinitialise the calculator.

1.1.2 What You See

>From top to bottom:

1. The HOME screen

1.a The calculator's status

1.b A horizontal line

1.c A menu bar of commands

2. The keyboard

1. The HOME screen :

1.a The calculator's status gives the HOME screen's current mode:

- RAD, DEG, or GRD when you are working with radians, degrees, or gradians.
- {FUNCTION} to indicate the name of the Aplet currently selected here, the Function Aplet.
- ▲ to indicate that you can use the up-arrow to move back through the history.

1.b Horizontal line :

- Above the horizontal line is found an overview of the calculations carried out in the HOME screen.

PRINCIPLE : On the screen, the requested calculation is written on the left, and the result is written on the right.

- Beneath the horizontal line is the command line.

Using the up-arrow, you can move back through the history and, using COPY on the menu bar, copy a command or a preceding result into the command line.

1.c Menu Bar:

The commands on the menu bar are accessible through the six unlabelled grey keys, which we refer to here as :

F1 F2 F3 F4 F5 F6.

The menu bar can contain items consisting of groups of commands; they are marked by a folder icon.

To activate a command on the menu bar, it's enough to press the corresponding F*i* key.

In the HOME screen, the menu bar contains two commands:

- STO► which enables you to store a value in a variable, and
- CAS, which enables you to open the Equation Editor to perform computer algebra.

2. The keyboard :

You have already been introduced to:

The ON key for starting the calculator or for cancelling the current operation, and SHIFT ON for turning the calculator off.

Other keys you must know include:

- The four arrow keys (left, right, up, down), which enable you to move the cursor while you're in the Equation Editor, in a menu, and so on,
- The SHIFT key, which enables you to access two functions with one key,

- The ALPHA key for typing text in upper case, and the SHIFT ALPHA keys for typing text in lower case. To remain in alphabetic mode, you must press and hold the ALPHA key.
- X T θ which enable you to type X, T, θ , N directly, depending on the active aplet.
- The ENTER key, which is used to confirm a command.

1.2 Notations

The four arrow keys are represented by four triangles:

$\triangle \triangleleft \triangleright \nabla$

The STO \triangleright on the HOME menu is represented in the program by:

STO \triangleright or \triangleright or $- \triangleright$

In the Equation Editor, the position of the cursor is represented by:

\blacktriangleleft

1.3 Online Help

With this calculator, you're able to get practical and efficient online help in either French or English (cf 5.1.1).

You are shown an alphabetised list of computer algebra functions. As with the drop-down menus, you are able to access the functions by pressing the corresponding letter-keys without having to worry about pressing ALPHA. The help consists of a succinct description of the command, as well as an example and its result. Each example can be tested with ECHO (on the menu) and can be either used as is or modified. You can view help for related commands by means of SEE1 SEE2... on the menu. You have to use EXIT to return, without change, in the Equation Editor.

For more details, refer to the description of SHIFT 2 (SYNTAX), sections 3.3.2 and 3.4.6.

Chapter 2

Aplets

2.1 APLET Key

The APLET key gives you access to the list of available Aplets.

This calculator in effect enables you to work with Aplets.

But what is an Aplet ?

An Aplet is a program stored in the calculator, which enables you to easily obtain three views of a mathematical object (a symbolic view, a numeric view, and a graphic view) and all this is pre-programmed !

The various Aplets enable you to work with mathematical objects such as: functions, sequences, statistical data, and so on...

Certain Aplets illustrate particular classroom lessons.

2.2 The Various aplets

While you're in HOME, you can see the name of the selected aplet by looking at the status line.

Here are the possible choices for the APLET key:

- Sequence
This Aplet enables you to define sequences having the names:
 $U_1, U_2 \dots U_9, U_0$
One defines $U_1(N)$:
 - To be a function of N ,
 - To be a function of $U_1(N-1)$,
 - To be a function of $U_1(N-1)$ and $U_1(N-2)$.

You define (for example):

$$U1(N) = N*N+1$$

and then the values of $U1(1)$ and $U1(2)$ are automatically calculated and displayed.

To display the values of $U1(N)$, choose $U1$, then press NUM.

You can find other examples that use the `Sequence` Applet in the following section, such as the calculation of the GCD of two numbers (cf 2.3.2), and the calculation of the coefficients of Bézout's Identity (cf 2.3.3).

- **Function**

This Applet enables you to define functions having the names:

$$F1(X), F2(X) \dots F9(X), F0(X)$$

One defines $F1(X)$:

- To be an expression of a function of X :

For example, the formula:

$$F1(X) = X*LN(X) \text{ defines the function:}$$

$$f1(x) = x \cdot \ln(x)$$

- To use Booleans ($X(0)$ etc.) if the function is defined in parts :

For example, a formula of the form:

$$F1(X) = X(X(0)+2(X(X>0))) \text{ defines the function:}$$

$$f1(x) = x \text{ if } x < 0 \text{ and}$$

$$f1(x) = 2 \cdot x \text{ if } x \geq 0$$

- **Parametric** for tracing curves in parametric coordinates.
- **Polar** for tracing curves in polar coordinates.
- **Solve** for solving numeric equations.
- **Statistics** for working with statistics.
- **Inference** for working with inferential statistics.
- **Quad Explorer** for exploring quadratic functions.
- **Trig Explorer** for exploring trigonometric functions.

2.3 Examples using the `Sequence` Applet

2.3.1 Notation in Base b

Given a and b , produce the series q_n ($n \geq 1$) and r_n ($n \geq 2$) from the quotients and the remainders of the division of q_i by b , defined by:

$$q_1 = a$$

$$q_1 = b \cdot q_2 + r_2 \quad (0 \leq r_2 < b)$$

$$q_2 = b \cdot q_3 + r_3 \quad (0 \leq r_3 < b)$$

.....

$$q_{n-1} = b \cdot q_n + r_n \quad (0 \leq r_n < b)$$

Note that if $r_{n+1} = 0$, the number $r_n r_{n-1} \dots r_3 r_2$ is notated in base b of a , while one assumes $2 \leq b \leq 10$.

Put into B the value of the base, for example:

7 STO \triangleright B

Put into A the number to write in base B (for example, 1789 STO \triangleright A)

Define the following two series:

$$U1(1) = A$$

$$U1(2) = \text{FLOOR}(A/B)$$

$$U1(N) = \text{FLOOR}(U1(N-1)/B)$$

and

$$U2(1) = 0$$

$$U2(2) = A \text{ MOD } B$$

$$U2(N) = U1(N-1) \text{ MOD } B$$

Therefore, $q_n = U1(N)$ and $r_n = U2(N)$.

We obtain :

$$U2(2) = 4 \quad U2(3) = 3 \quad U2(4) = 1 \quad U2(5) = 5 \quad U2(6) = 0$$

and so the notation in base 7 of 1789 is: 5134.

2.3.2 Calculating the GCD

This is an application of Euclid's Algorithm on the HP40GS.

Here is the description of this algorithm:

If one performs the successive Euclidean divisions:

$$A = B \times Q_1 + R_1 \quad 0 \leq R_1 < B$$

$$B = R_1 \times Q_2 + R_2 \quad 0 \leq R_2 < R_1$$

$$R_1 = R_2 \times Q_3 + R_3 \quad 0 \leq R_3 < R_2$$

.....

$$R_{n-2} = R_{n-1} \times Q_n + R_n \quad 0 \leq R_n < R_{n-1}$$

then after a finite number of steps (in excess of B), there exists a whole number n such that: $R_n = 0$.

We have then:

$$GCD(A, B) = GCD(B, R_1) = \dots$$

$$GCD(R_{n-1}, R_n) = GCD(R_{n-1}, 0) = R_{n-1}$$

Using a sequence, one then writes the sequences of remainders.

With the HP40GS, you use the Sequence Aplet (APLET key, then select Sequence, then START on the menu bar).

To determine the $GCD(78,56)$, you define the sequence:

$$U1(1) = 78$$

$$U1(2) = 56$$

$$U1(N) = U1(N - 2) \text{ MOD } U1(N - 1)$$

Press NUM to get the numerical list of $U1(N)$ that is, the list of the remainders of the successive divisions...

The final non-zero remainder is 2, so the $GCD(78, 56) = 2$.

REMARK:

In HOME, you can use the variables A and B to store the two numbers, and then make $U1(1) = A$ and $U1(2) = B$.

It's also important to note that $A \text{ MOD } 0 = A$.

2.3.3 Calculating the Coefficients of Bézout's Identity

Euclid's Algorithm enables you to find a pair U, V such that:

$$A \times U + B \times V = PGCD(A, B)$$

Using the idea of sequence:

Define "the sequence of remainder" R_n and two sequences U_n and V_n , such that at each step one has:

$$R_n = U_n \times A + V_n \times B.$$

Seeing that one has:

$R_n = R_{n-2} - Q_n \times R_{n-1}$, U_n and V_n serve to satisfy the same recurrence relation (where Q_n is the whole-number quotient of R_{n-2} divided by R_{n-1}).

One then has (from the beginning):

$$R_1 = A \quad R_2 = B$$

$$U_1 = 1 \quad U_2 = 0 \quad \text{since } A = 1 \times A + 0 \times B$$

$$V_1 = 0 \quad V_2 = 1 \quad \text{since } B = 0 \times A + 1 \times B$$

With the HP40GS, using the Sequence Aplet, you then define the sequence of remainders U1 and the sequences U2 and U3 such that for all N one has:

$$U1(N) = A * U2(N) + B * U3(N).$$

For this, you need the sequence of quotients, which you put into U4.

The sequences U1, U2 and U3 satisfy the same recurrence relation:

$$U_n = U_{n-2} - Q_n \times U_{n-1} \quad \text{with}$$

$$Q_n = U4(N) = \text{FLOOR}(U1(N - 2) / U1(N - 1))$$

On définit donc :

$$U1(1) = A$$

$$U1(2) = B$$

$$U1(N) = U1(N - 2) - U4(N) * U1(N - 1)$$

$$U2(1) = 1$$

$$U2(2) = 0$$

$$U2(N) = U2(N - 2) - U4(N) * U2(N - 1)$$

$$U3(1) = 0$$

$$U3(2) = 1$$

$$U3(N) = U3(N - 2) - U4(N) * U3(N - 1)$$

$$U4(1) = 0$$

$$U4(2) = 0$$

$$U4(N) = \text{FLOOR}(U1(N - 2)/U1(N - 1))$$

It's important to note that you use $U4(N)$ only for $N > 2$; you have therefore defined the two first values (which are useless!) as zero.

NUM then displays the values of these various sequences, and on the line of the final non-zero remainder you can read the GCD and the coefficients of Bézout's Identity.

2.4 The SYMB, NUM, PLOT and MODES Keys

In general, an Aplet can be viewed in three different ways:

A symbolic view, which corresponds to the SYMB key

A numeric view, which corresponds to the NUM key

A graphic view, which corresponds to the PLOT key

When these keys are SHIFTed (SETUP), this corresponds to choosing the various available parameters (choosing the parameters of the graphic window, the step size for the table etc...).

The SHIFT HOME (MODES) key pressed in HOME screen or in Aplets, corresponds to the choice between Radians, Degrees or Grads.

Chapter 3

CAS user's guide

3.1 Terms and Conditions

Use of the CAS Software requires from the user an appropriate mathematical knowledge. There is no warranty for the CAS Software, to the extent permitted by applicable law. Except when otherwise stated in writing the copyright holder provides the CAS Software As Is without warranty of any kind, either expressed or implied, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. The entire risk as to the quality and performance of the CAS Software is with you. Should the CAS Software prove defective, you assume the cost of all necessary servicing, repair or correction.

In no event unless required by applicable law will any copyright holder be liable to you for damages, including any general, special, incidental or consequential damages arising out of the use or inability to use the CAS Software (including but not limited to loss of data or data being rendered inaccurate or losses sustained by you or third parties or a failure of the CAS Software to operate with any other programs), even if such holder or other party has been advised of the possibility of such damages. If required by applicable law the maximum amount payable for damages by the copyright holder shall not exceed the royalty amount paid by Hewlett-Packard to the copyright holder for the CAS Software

Note that a large part of the HP40GS CAS is (c) Bernard Parisse, Institut Fourier, Université de Grenoble I, France. This part is also licensed under

the LGPL License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

For more details, see

<http://www-fourier.ujf-grenoble.fr/~parisse>

3.2 Introduction

The word CAS is a shortcut for Computer Algebra System. This chapter gives a brief overview on the HP40GS CAS.

3.2.1 What is CAS?

The CAS enables you to perform exact or symbolic calculations. Make sure you understand the difference between:

- exact or symbolic calculations, which are performed by means of the CAS functions. You work in `exact` mode, with infinite precision, and
- numeric calculations, which are performed by means of the MATH key's MTH menu, either in the HOME screen or in `Aplets` or programs. You work in `approximate` mode, with a precision of 10^{-12} .

Example:

If you're working in the HOME screen:

$1/2+1/6$ returns 0.6666666666667

whereas in CAS:

$1/2+1/6$ will return $2/3$.

Each mode has advantages and drawbacks, in exact mode there is no round-off error, but some computations require much more time and memory than in numeric mode. For example computing the factorial of 300 is slightly faster in numeric mode than in exact mode.

3.2.2 How do you perform a symbolic calculation?

The HP40GS has been designed to use computer algebra functions from a special environment: the Equation Editor. You can, however, perform computer algebra in the HOME screen, as long as you take certain precautions (cf 3.4), and it is mandatory if you want do symbolic linear algebra, since the Equation Editor does not handle vectors and matrices.

To open the Equation Editor, press CAS on the menu bar of the HOME screen.

To leave the Equation Editor, press HOME to return to the HOME screen.

Section 3.3 describes how to use the CAS functions from the Equation Editor, section 3.4 how to use the CAS functions from the HOME screen and chapter 5 describes the CAS functions.

3.2.3 Variables

When you use the symbolic calculation functions, you are working with symbolic variables (variables that do not contain a permanent value). In the HOME screen, a variable of this kind must have a name like $S1 \dots S5$, $s1 \dots s5$, $n1 \dots n5$, but not X which is assigned to a real value (by default X is assigned to 0) and to store for symbolic expressions, the variables are $E0, E1 \dots E9$. In the Equation Editor, all variables may, or may not be, assigned: for example X is not assigned to a real value by default, computing $X+X$ will return $2X$. Moreover, Equation Editor variables may have long names, like XY or ABC , unlike in HOME where implied multiplication is assumed (for example ABC becomes $A*B*C$ in HOME). For these reasons, variables used in the Equation Editor cannot be used in HOME, and vice versa. Using the command $PUSH$, you can transfer expressions from the HOME screen history to the CAS history (cf section 3.4.4) and using the command POP , you can also transfer expressions from the CAS history to the HOME screen history (cf section 3.4.5).

3.2.4 The current variable

In the Equation Editor, the current variable is the name of the symbolic variable contained in VX , it is almost always X .

The current variable is always $S1$ in the HOME screen.

Some CAS functions depend on a current variable, for example, the function $DERVX$ calculates the derivative with respect to the current variable.

Hence in the Equation Editor,

$DERVX(2*X+Y)$ returns 2 if $VX=X$, but

$DERVX(2*X+Y)$ returns 1 if $VX=Y$ and

in the HOME screen,

$DERVX(2*S1+S2)=2$, but

$DERIV(2*S1+S2, S2)=1$.

3.3 Using the CAS in the Equation Editor

The Equation Editor enables you to type expressions that you want to simplify, factor, differentiate, integrate, and so on, and then work them through as if on paper.

The CAS key on the HOME menu bar takes you into the Equation Editor, and the HOME key takes you back out.

This section tells you how to write an expression in the Equation Editor using the menus (section 3.3.1) and the keyboard (section 3.3.2), how to select a sub-expression (section 3.3.3), how to call the CAS functions on a sub-expression (section 3.3.3) and how to store values in the Equation Editor variables (section 3.3.5). Chapter 5 explains all the symbolic calculation functions contained in the various menus, together with examples of use.

You can consult the online help with SHIFT 2 (SYNTAX) (cf 3.3.2), to get help for the other available functions, and you can use SHIFT MATH (CMDS) (cf 3.3.2), to type them.

3.3.1 The Equation editor menu

The editor is supplied with a menu bar of menus:

1. The TOOL menu contains the commands:
 - `Cursor` mode enables you to go into cursor mode (cf 3.3.3).
 - `Edit expr.` enables you to edit (modify) the highlighted expression.
 - `Change font` enables you to choose to type using large or small characters (you can make this choice at any time).
 - `Cut` copies the selection into the buffer, then erases the selection.
 - `Copy` copies the selection into the buffer.
 - `Paste` copies the buffer to the location of the cursor. The buffer contains, either whatever `Copy` or `Cut` has selected the last time, or the highlighted level when you hit `COPY` in the CAS history toolbar.
2. The ALGB menu contains functions that enable you to perform algebra: factoring, expansion, simplification, substitution...
3. The DIFF menu contains functions that enable you to perform differential calculus: differentiation, integration, series expansion, limits...

4. The REWRI menu contains functions that enable you to rewrite an expression in another form.
5. The SOLV menu contains functions that enable you to solve equations, linear systems, and differential equations.
6. The TRIG menu contains functions that enable you to transform trigonometric expressions.

3.3.2 The keyboard in the Equation Editor

The keys mentioned in this section have different functions depending on whether they are pressed in the Equation Editor or in the HOME screen. For the functionality of these keys outside the Equation Editor, refer to section 3.4.6, or consult the User's Guide.

MATH key

The MATH key, if pressed in the Equation Editor, displays the functions used in symbolic calculation. These functions are contained in menus:

- The five preceding menus (cf 3.3.1):
Algebra (ALGB), Diff&Int (DIFF), Rewrite (REWRI), Solve (SOLV), Trig. (TRIG).
- The Complex menu containing functions that enable you to work with complex numbers
- The Constant menu containing $i \infty \pi$.
- The Hyperb. menu containing hyperbolic functions
- The Integer menu containing functions that enable you to perform integer arithmetic.
- The Modular menu containing functions that enable you to perform calculations in Z/pZ or $Z/pZ[X]$, p being the value contained in the variable MODULO.
- The Polynom. menu containing functions that enable you to perform calculations with polynomials.

- The `Tests` menu containing:
 - `ASSUME UNASSUME` (to make hypotheses about the parameters, and to modify the variable `REALASSUME` cf 3.3.5)
 - `> ≥ < ≤ == ≠ AND OR NOT`
 - `IFTE` (to write an algebraic function having the same result as an `IF THEN ELSE`)

Refer to section 5.1.9, for the list of the functions contained in the various menus.

SHIFT MATH (CMDS) keys

This key combination opens the list of all the CAS commands available in the `Equation Editor`.

In this way, functions that are not presented elsewhere can be called from this menu, so you don't have to type them in ALPHA mode.

VARs key

This key, if pressed while you're in the `Equation Editor`, displays the names of the variables defined in CAS.

Take special note of `namVX`, which contains the name of the current variable.

To see the contents of a variable, all you have to do is highlight its name and press F2 for `VIEW` on the menu bar.

To change the contents of a variable, highlight its name and press F3 for `EDIT` on the menu bar.

Note also on the menu bar:

`PURGE`, which enables you to destroy an existing variable,

`RENAME`, which enables you to change the name of an existing variable,

`NEW`, which enables you to define a new variable: just enter the contents (object), then the (name).

For more details, refer to section 3.3.5.

SHIFT 2 (SYNTAX) keys

While you're in the `Equation Editor`, the key combination `SHIFT 2 (SYNTAX)` opens the `CAS HELP ON` menu.

To get help in French, choose `Français` on the `CFG` menu, which enables you to change your configuration (cf 5.1.1).

If there is no CAS function selected in the editor, the menu shows a list of

functions available in the Equation Editor. Just highlight a function and press OK to see the help for that function.

If there is a CAS function selected in the editor, for example:

FACTOR(45), the CAS HELP ON menu directly opens the help topic for FACTOR. The help consists of a short description of the command, as well as an example and its result. Each example can be copied into the Equation Editor by means of ECHO on the menu bar, where it can be used as is or modified.

Note that in the help examples, $VX=X$ is used as the current variable. If that is not the case for you, the example will be automatically transformed, taking your value of VX into account, when you transfer it with ECHO.

You can also go directly to see the help of a command pointed to by See : with SEE1, SEE2... on the menu bar.

SYMB key

Pressing the SYMB key in the Equation Editor enables you to access the CAS history.

The history of the calculations performed in CAS differs from the history of the calculations performed in HOME.

As in the HOME screen history, the requested calculations are written on the left, and the results are written on the right. Using the up-arrow, you can move back through the history.

Press COPY to copy the current CAS history level in the buffer in order to paste it later with TOOL->Paste in the Equation Editor.

Press ENTER, or ECHO on the menu bar, to replace the current Equation Editor selection by the current CAS history level.

Press ON to leave the CAS history without any changes.

SHIFT SYMB or SHIFT HOME keys

While you are working in the Equation Editor, the key combination:

SHIFT SYMB (SETUP) or SHIFT HOME (MODES) open the CAS MODES screen to define the CAS configuration and it is similar to CFG menu (the first item in the ALGB menu on the menu bar, cf 5.1.1).

This enables you to specify:

- The name of the variable contained in VX , by typing its name next to `Indep var .`,
- The value of MODULO, by typing its value next to `Modulo`,
- Whether you want to work in exact mode (or in approximate mode, if you've chosen Approx with CHK on the menu bar)

- Whether you want to work in real mode (or in complex mode, if you've chosen Complex with CHK on the menu bar)
- Whether you want to work in Direct mode (or in Step by Step mode, if you've chosen Step/Step with CHK on the menu bar)
- Whether you want polynomials to be written in decreasing order according to exponent (or increasing order, if you've chosen Incr Pow with CHK on the menu bar)
- Whether you want numerical factors suppressed (or not, if you've chosen Num. Factor with CHK on the menu bar)
- Whether you want to work in non-rigorous mode (or in rigorous mode, if you've chosen Rigorous with CHK on the menu bar so as not to neglect the absolute values!)
- Whether you want to simplify non-rational expressions if you've chosen Simp Non-Rational (or not, if you clear the selection by pressing CHK on the menu bar).

Use OK or ENTER to confirm your choices.

SHIFT , key

While you are working in the Equation Editor, the key combination:

SHIFT , (MEMORY) plays the role of an "undo" key.

This is very useful when you've made a mistake, because it enables you to cancel the last command.

PLOT key

When you press PLOT in the Equation Editor, a dialog box asks you if you want to graph a function, a parametric curve, or a polar curve.

Depending on what you choose, the highlighted expression is copied into the appropriate applet, to the location that you have specified as the destination.

NOTE : This supposes that the current variable is also the variable of the function or curve you want to graph: when the expression is copied, it is evaluated, and the current variable (variable contained in VX) is changed to X, T, or θ , depending on the type of plot.

NOTE : If the function depends on a parameter, it is preferable to give the parameter a value before pressing PLOT. If, however, you want the parameterised expression to be copied with its parameter, then the name of the parameter must consist of a single letter other than X, T, or θ , so that there is no confusion.

If the highlighted expression has real values :

the `Function Aplet` or `Polar Aplet` can be chosen, and the graph will be of `Function` or `Polar` type.

If the highlighted expression has complex values :

the `Parametric Aplet` must be chosen, and the graph will be of `Parametric` type.

If you choose:

- the `Function Aplet`, the highlighted expression is copied into the chosen function F_i , and the current variable is changed to X during the copy.
- the `Parametric Aplet`, the real part and the imaginary part of the highlighted expression are copied into the chosen functions X_i , Y_i , and the current variable is changed to T during the copy.
- the `Polar Aplet`, the highlighted expression is copied into the chosen function R_i , and the current variable is changed to θ during the copy.

NUM key

Pressing the `NUM` key in the Equation Editor causes the highlighted expression to be replaced by a numeric approximation.

`NUM` puts the calculator into approximate mode.

SHIFT NUM key

Pressing the `SHIFT NUM` key in the Equation Editor causes the highlighted expression to be replaced by a rational number.

`SHIFT NUM` puts the calculator into exact mode.

VIEWS key

Pressing the `VIEWS` key in the Equation Editor enables you to move the cursor with the \triangleright and \triangleleft arrows to see the entire highlighted expression.

Then press `OK` on the menu bar to return in the Equation Editor.

Short-cut keys

Note: In the Equation Editor, the following short-cut keys are available on the keyboard:

`SHIFT 0` for ∞

SHIFT 1 for i
 SHIFT 2 to open the CAS HELP ON
 SHIFT 3 for π
 SHIFT 5 for $<$
 SHIFT 6 for $>$
 SHIFT 8 for \leq
 SHIFT 9 for \geq
 SHIFT NUM to put the calculator into exact mode
 NUM to put the calculator into approximate mode.

3.3.3 Expressions, subexpressions, selection

When you type expressions in the Equation Editor, the operator that you are typing always carries over to the adjacent or selected expression.

You don't have to preoccupy yourself with where the parentheses go... just select!

You need to view a mathematical expression as a binary tree, and the four arrow keys as enabling you move through the tree in a natural fashion:

The right and left arrow keys enable you to move from one branch to another,

The up and down arrow keys enable you to go up and down a particular tree,

The SHIFT-up and SHIFT-down arrow keys enable you to make multiple selection (cf page 26 example 3).

How do you select?

There are two ways of going into selection mode :

- The up-arrow Δ takes you into selection mode and selects the element adjacent to the cursor.

Example:

$$1 + 2 + 3 + 4 \Delta$$

selects 4, then Δ selects the entire tree $1+2+3+4$.

- The right-arrow \triangleright takes you into selection mode and selects the branch adjacent to the cursor.

Pressing \triangleright again augments the selection, adding the next branch to the right.

Example:

$$1 + 2 + 3 + 4 \triangleright$$

selects $3 + 4$, then \triangleright selects $2+3+4$, then \triangleright selects $1+2+3+4$.

- NOTE: If you are typing a templated function with multiple arguments (such as \sum , \int , SUBST, or the like), the right arrow \triangleright enables you to move through the template by changing the location of the cursor. In effect, the left and right arrow, keys \triangleleft and \triangleright enable you to move from one argument to another. In this case, you always use the up arrow \triangle to select (cf 3.3.4).

Examples of the way this Equation Editor works:

Press CAS on the menu bar to open the Equation Editor, then type the expressions in the examples.

- Example 1

Typing:

$$2 + X \times 3 - X$$

returns:

$$2 + X \cdot 3 - X$$

Press $\triangleright \triangleright \triangleright$ to select the expression,
then the ENTER key to produce:

$$2 + 2 \cdot X$$

Typing:

$$2 + X \triangleright \times 3 - X$$

returns:

$$(2 + X) \cdot 3 - X$$

Press $\triangleright \triangleright$ to select the expression,
then the ENTER key to produce:

$$6 + 2 \cdot X$$

Typing:

$$2 + X \triangleright \times 3 \triangle - X$$

returns:

$$(2 + X) \cdot (3 - X)$$

Press $\triangleright \triangleright \triangleright$ to select the expression,
then the ENTER key to produce:

$$-(X^2 - X - 6)$$

- Example 2
To express:

$$X^2 - 3 \cdot X + 1$$

type:

$$X \ x^y \ 2 \ \triangleright \ - \ 3 \ X \ + \ 1$$

To express:

$$-X^2 - 3 \cdot X + 1$$

type:

$$(-) \ X \ x^y \ 2 \ \triangleright \triangleright \ - \ 3 \ X \ + \ 1$$

In effect, you must select $-X^2$ before typing the rest.

- Example 3
To express:

$$\frac{1}{2} + \frac{1}{3} + \frac{1}{4} + \frac{1}{5}$$

Here, the tree ends in a +, and there are four branches; each of these branches ends in a \div and has two leaves.

Press CAS on the menu bar to open the Equation Editor, then type the first branch:

$$1 \div 2$$

then select this branch with

\triangleright

Then, type

+

and the second branch:

$$1 \div 3$$

then select this branch with

\triangleright

Then, type

$$+$$

and the third branch:

$$1 \div 4$$

then select this branch with

$$\triangleright$$

Then, type

$$+$$

and the fourth branch:

$$1 \div 5$$

then select this branch with

$$\triangleright$$

At this point, the desired expression :

$$\frac{1}{2} + \frac{1}{3} + \frac{1}{4} + \frac{1}{5}$$

is in the Equation Editor, and $\frac{1}{5}$ is selected.

To move back through the tree and select:

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

type

$$\triangleleft \triangleleft$$

to select the $\frac{1}{3}$, then press

$$\text{SHIFT} \triangleright$$

which enables you to select two contiguous branches, the one already selected and the one to the right of it, like this:

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

Advantage : If you want, you can calculate the selected part by pressing ENTER.

This produces:

$$\frac{1}{2} + \frac{7}{12} + \frac{1}{5}$$

with the $\frac{7}{12}$ selected.

If you want to perform the partial calculation

$$\frac{1}{2} + \frac{1}{5}$$

you must first perform a permutation so that the $\frac{1}{2}$ and the $\frac{1}{5}$ are side by side. To do this, type:

SHIFT<

which exchanges the selected element with its neighbour to the left.

This produces:

$$\frac{7}{12} + \frac{1}{2} + \frac{1}{5}$$

with the $\frac{7}{12}$ still selected. Press :

▷SHIFT▷

to select :

$$\frac{1}{2} + \frac{1}{5}$$

Pressing ENTER then produces the result.

- Summing up: SHIFT▷ enables you to select the selected element and its neighbour to the right. SHIFT< enables you to exchange the selected element with its neighbour to the left. The selected element remains selected, even if you move it.

How to modify an expression

If you're typing an expression, the DEL key enables you to erase what you've typed.

If you're selecting, you can:

- Cancel the selection without deleting the expression, by typing :

DEL

The cursor moves to the end of the deselected portion.

- Replace the selection with an expression, just by typing the desired expression
- Transform the selected expression by applying a CAS function to it: you

call the function via one of the CAS menu options.

- Delete the selected expression by typing:

ALPHA SHIFT DEL (ALPHA CLEAR)

- Delete a selected unary operator "the top of the tree" by typing:

SHIFT DEL (CLEAR)

For example, to replace $\text{SIN}(\text{expr})$ with $\text{COS}(\text{expr})$, select $\text{SIN}(\text{expr})$, then press SHIFT DEL and press COS.

- Delete a binary infix operator and one of its argument by selecting the argument you want delete and typing:

SHIFT DEL (CLEAR)

For example : if you have the expression $1+2$, you select 1, then SHIFT DEL deletes 1+ and it leaves only 2.

So, to delete $F(X) =$ of the expression $F(X) = X^2 - X + 1$ you select $F(X)$, then press SHIFT DEL, and this produces $X = X^2 - X + 1$ (you have deleted the unary operator F), then SHIFT DEL, produces $X^2 - X + 1$ (you have deleted X and the operator $=$).

- Delete a binary operator by editing the expression, you select :

Edit expr.

from the TOOL menu on the menu bar, and then make the correction.

- Copy an element from the CAS history. You access to the CAS history by pressing SYMB. In the history, pressing ENTER or selecting ECHO on the menu bar inserts the copy where the cursor is, or in place of the selection. You can also use COPY, on the history menu bar, to copy the selection in the buffer, or the commands Cut, Copy and Paste from the TOOL menu on the Equation Editor menu bar to delete, copy and paste expressions as you would with any text editor (cf.3.3.1).

The cursor mode

The cursor mode enables you to select a large expression quickly. To go into cursor mode, select:

Cursor mode in the TOOL menu,

then use the arrow keys to include your selection in a box (when you release the arrow key, the expression pointed to by the cursor is enclosed).

Then, press ENTER to select the contents of the box.

To see everything

By selecting `Change font` from the `TOOL` menu on the menu bar, you can increase or decrease the font size of the expression. This enables you to view a large expression in its entirety when you need to.

If this is insufficient to see the whole expression, then you'll need to go into cursor mode:

select `Cursor mode` in the `TOOL` menu, then use the arrow key \triangleright , and then use `ENTER` to quit cursor mode,

or you can also press:

the `VIEWS` key, then use the arrow key \triangleright , and press `OK` on the menu bar to quit `VIEWS` screen.

3.3.4 Accessing the CAS functions

While you are in the Equation Editor, you can access all CAS functions, and you can access them in various ways.

General principle:

When you have written an expression in the Equation Editor, all you have to do is press `ENTER` to evaluate the selection (or the entire expression, if nothing is selected).

How to type \int and \sum

\sum is found on the keyboard, all you have to do is type:

$$\text{SHIFT} + (\sum)$$

The symbol \int is also found on the keyboard, it can be produced by typing:

$$\text{SHIFT d/dX} (\int)$$

The symbols \int and \sum are treated as prefix functions with multiple arguments.

\int and \sum are automatically placed before the selected element, if there is one (hence "prefix functions").

You can move the cursor with the arrow keys:

$$\triangleright \triangleleft$$

Enter the expressions according to the rules of selection explained earlier, but you must first go into selection mode by pressing Δ .

NOTE: Do not use the index i to define a summation, because i designates the complex-number solution of $x^2 + 1 = 0$.

\sum performs exact calculations if its argument has a discrete primitive else \sum performs approximate calculations, even in exact mode. For example in exact, or numeric mode :

$$\sum_{k=0}^4 \frac{1}{k!} = 2.70833333334$$

whereas in exact mode :

$$1 + \frac{1}{1!} + \frac{1}{2!} + \frac{1}{3!} + \frac{1}{4!} = \frac{65}{24}$$

the symbol ! is obtained by typing SHIFT ×.

Note that \sum can symbolically calculate summations of rational fractions and hypergeometric series that allow a discrete primitive.

Example :

If you type:

$$\sum_{K=1}^4 \frac{1}{K \cdot (K + 1)}$$

then select the entire expression and press ENTER, you obtain:

$$\frac{4}{5}$$

If you type:

$$\sum_{K=1}^{\infty} \frac{1}{K \cdot (K + 1)}$$

then select the entire expression and press ENTER, you obtain:

$$1$$

How to call infix functions

These functions are typed between their arguments, for example :

AND | MOD , are infix functions.

You can either:

- type them in Alpha mode (using AND MOD), then type the arguments,
- or
- call them by selecting a CAS menu option or by pressing a key, provided

you have already written and selected the first argument.

You move from one argument to the other by using the arrow keys \triangleright \triangleleft .

The comma , enables you to write a complex number:

when you type $1 + 2 \cdot i$ or $(1, 2)$, the parentheses are automatically placed when you type the comma.

If you want to type $(-1, 2)$, you must of course select -1 before you type the comma.

How to call prefix functions

These functions are typed before their arguments (which is the usual case).

To call a prefix function:

you can type the first argument, select it, then call the function using a menu,

or, you can call the function using a menu or a keystroke in Alpha mode, then type the arguments.

The following example illustrates the different ways of calling a prefix function.

Example:

Say you want to factor the expression $x^2 - 4$, then find its value for $x = 4$.

You know that FACTOR is the function for factoring, and that this function is found in the ALGB menu.

You also know that SUBST is the function for substituting a value for a variable in an expression, and that this function is found in the ALGB menu as well.

First possibility: Function call, then arguments

Press the F2 key to activate the ALGB on the menu bar, then highlight FACTOR and press ENTER.

FACTOR(\blacktriangleleft) is displayed in the editor, with the cursor between the parentheses.

Type your expression, using the rules of selection given earlier:

$X X^y 2 \triangleright - 4 \triangleright \triangleright \triangleright$

The following is now selected:

FACTOR($X^2 - 4$)

Pressing ENTER then produces the result:

$(X + 2) \cdot (X - 2)$

The result is selected, and replaces the command.

You do not see this, but after each ENTER, the display is written to the history, so in this case, $\text{FACTOR}(X^2 - 4)$ and the answer $(X + 2) \cdot (X - 2)$ are written to the history (press SYMB to see the history and then press ON to return to the Equation Editor).

At this point, you can erase the preceding result with ALPHA SHIFT DEL (CLEAR), because the result is selected.

Press the key that activates ALGB on the menu bar, then highlight SUBST and press ENTER.

SUBST(◀, ●)

is displayed in the editor, with the cursor between the parentheses at the location of the first argument.

Type your expression, using the rules of selection given earlier:

NOTE: Here, SUBST has two arguments, so you must go into selection mode using Δ :

$X X^y 2 \Delta \Delta - 4 \triangleright X = 4 \triangleright \triangleright$

The following is now selected:

SUBST($X^2 - 4$, $X = 4$)

Pressing ENTER then produces the result:

$4^2 - 4$

The result is selected, and replaces the command. Pressing ENTER then yields the simplified result:

12

Naturally, $\text{SUBST}(X^2 - 4, X = 4)$, $4^2 - 4$ and 12 are all written to the history.

REMARK:

When you call a CAS function that has arguments, you can type it in Alpha mode with its parentheses.

Second possibility: Arguments, then function call

First of all, type the expression and select it using the rules of selection given earlier.

In this case, type:

$X X^y 2 \triangleright - 4 \triangleright \triangleright$

Then call FACTOR:

press the key F2 that activates ALGB on the menu bar, then highlight FACTOR

and press ENTER.

This produces:

$$\text{FACTOR}(X^2 - 4)$$

Pressing ENTER then yields the result:

$$(X + 2) \cdot (X - 2)$$

The result is selected, and replaces the command.

Naturally, $\text{FACTOR}(X^2 - 4)$ and $(X + 2) \cdot (X - 2)$ are both written to the history.

Recall now that because your result is selected, you can apply another command to it.

At this point, then, call SUBST: press the key F2 that activates ALGB on the menu bar, then highlight SUBST and press ENTER.

$$\text{SUBST}((X + 2) \cdot (X - 2), \blacktriangleleft)$$

is displayed in the editor, with your expression as the first argument between the parentheses, and with the cursor at the location of the second argument.

All you have to do then is type:

$X = 4$ then, $\triangleright \triangleright$, followed by ENTER.

This produces:

$$(4 + 2) \cdot (4 - 2)$$

Pressing ENTER then yields:

$$12$$

Naturally, $\text{SUBST}(X^2 - 4, X = 4)$, $(4 + 2) \cdot (4 - 2)$ and 12 are all written to the history.

REMARK:

If you call a CAS function while you're writing an expression, whatever is currently selected is copied into the function's first or "main" argument. If nothing is selected, the cursor is placed at the appropriate location for completing the arguments.

3.3.5 Equation editor variables

You can store objects in variables, then access an object by using the name of its variable.

NOTES:

1- Variables used in CAS cannot be used in HOME, and vice versa.

2- In HOME or in the program editor, use STO> (represented here as STO> or ▷ or ->) to store an object in a variable.

3- In CAS, use the STORE command (cf. 3.3.5) to store a value in a variable.

4- The VARS key displays a menu that contains all the available variables. Pressing this key while you are in HOME displays the names of the variables defined in HOME and in the Aplets.

Pressing this key while you are in the Equation Editor displays the names of the variables defined in CAS.

To store an object in a CAS variable : STORE

In CAS, it is necessary to use the STORE command to store an object in a variable, or to use the VARS key in the Equation Editor (and then choose NEW or EDIT on the menu bar; cf. 3.3.2).

All you need to provide is the name of the variable.

STORE is found in the ALGB menu on the Equation Editor menu bar.

Example:

Type:

$$\text{STORE}(X^2 - 4, \text{ABC})$$

Or, type:

$$X^2 - 4$$

then select it and call STORE,

then type ABC, then, ENTER confirms the definition of the variable ABC.

To destroy the variable, use the VARS key in the Equation Editor (then choose PURGE on the menu bar; cf. 3.3.2), or invoke the UNASSIGN command on the ALGB menu by typing (for example):

$$\text{UNASSIGN}(\text{ABC})$$

Predefined CAS variables

VX contains the name of the current symbolic variable.

Generally this is X, so you should not use X as the name of a numeric variable, or erase the contents of X with the UNASSIGN command in the ALGB menu after having done a symbolic calculation by typing, for example, UNASSIGN(X).

EPS contains the value of epsilon used in the EPSX0 command (cf. 5.11.2).

MODULO contains the value of p for performing symbolic calculations in Z/pZ or in $Z/pZ[X]$. You can change the value of p either with the

MODSTO command in the MODULAR menu, (by typing, for example, MODSTO(13) to give p a value of 13), or use SHIFT SYMB to open the CAS MODES screen and change Modulo.

PERIOD must contain the period of a function before you can find its Fourier coefficients (cf. 5.9.4).

PRIMIT contains the primitive of the last integrated function.

REALASSUME contains a list of the names of the symbolic variables that are considered reals. If you've chosen the Cmplx vars option on the CFG configuration menu, these are by default:

X , Y , t , $S1$, $S2$, as well as any integration variables that are in use.

Of course, if you've chosen the Real vars option on the CFG configuration menu, all symbolic variables are considered reals (cf. 5.1.1). You can also use an assumption to define a variable such as $X>1$.

In a case like this, you use the ASSUME($X>1$) command to make REALASSUME contain $X>1$. The command UNASSUME(X) destroys all the assumptions we've previously made about X .

To see or to modify all these variables, as well as those that you've defined in CAS, press VARS in the Equation Editor, then on EDIT on the menu bar (cf. 3.3.2).

3.4 CAS in the HOME screen

You can use many computer algebra functions directly in the HOME screen, as long as you take certain precautions. CAS functions that take matrices arguments work only from HOME. For the description of these functions see chapter 4.

3.4.1 HOME variables

You can not use the same variable names in HOME as in the Equation Editor. For example X can not be used as a symbolic variable, since the $A..Z$ variables names are numeric variables, and always contain a real value (0 by default).

The names of the symbolic HOME variables are :

$s1..s5$, $s1..s5$, $n1..n5$.

Unlike with Equation Editor variables, it is not possible to assign a value to a HOME symbolic variable: note that $s1..s5$ and $n1..n5$ are sometimes returned by QUAD and ISOLATE respectively instead of a sign (it should be replaced by ± 1 to get both solutions in QUAD, for instance,

QUAD(S1*S1-9, S1) returns $s1*6/2$ or instead of an integer (e.g. for isolation inside a trigonometric function, for instance, ISOLATE(SIN(S1), S1) returns $3.14...+n1$).

If you want to store symbolic expressions in HOME you must use the symbolic expression variables E0 . . E9. If you want to store lists of symbolic expressions in HOME you must use the variables L0 . . L9.

3.4.2 Storing an object in a HOME variable : STO>

STO> enables you to store an object in a HOME variable. STO> is obtained by pressing F1 on the HOME menu bar or on program editor menu bar,. A variable can store only one kind of object, this kind is determined by the variable name (see above).

For example, type:

$S1^2 + 2 \times S1$ STO> E1, then,

DERIV(E1, S1)

you get :

$2 * S1 + 2$.

3.4.3 Some tricks and examples.

CAS functions may be accessed from HOME, either from the CAS submenu of the MATH key's menu bar (F4 key), or directly (type the function name in alpha mode). Note that:

- Certain calculations will be performed in approximate mode because numbers are interpreted as reals instead of integers in HOME. To do exact calculations, you should use the command XQ, it converts an approximate argument into an exact argument. For example, if you work with Radians:

$$\text{ARG}(\text{XQ}(1 + i)) = \frac{\pi}{4}$$

$$\text{ARG}(1 + i) = 0.7853\dots$$

$$\text{FACTOR}(\text{XQ}(45)) = 3^2 * 5$$

$$\text{FACTOR}(45) = 45$$

- The symbolic HOME variable S1 serves as the current variable for the relevant CAS functions in HOME. For example :

$$\text{DERVX}(S1^2 + 2 \times S1) = 2 * S1 + 2$$

The result $2 * S1 + 2$ does not depend on the Equation Editor variable VX). Note that a few CAS function can not work inside HOME because they require to change the current variable.

- Remember that you must use :
 $S1, S2, \dots, S5, s1, s2, \dots, s5, n1, n2, \dots, n5$ for symbolic variables and $E0, E1, \dots, E9$ to store symbolic expressions. For example, if you type :

$$S1^2 - 4 \times S2 \text{ STO } \triangleright E1$$

then you have :

$$\text{DERVX}(E1) = S1 * 2$$

$$\text{DERIV}(E1, S2) = -4$$

$$\text{INTVX}(E1) = 1/3 * S1^3 - 4 * (S2 * S1)$$

- If you want to work with symbolic matrices, you must store them in $L0, L1 \dots L9$, because symbolic matrices are interpreted in the same these matrices are interpreted in the same way as lists of lists (whereas numeric matrices are stored in $M0, M1 \dots M9$). The CAS linear algebra instructions accept list of lists as input. For example, if you type in HOME:

$$\text{XQ}(\{\{S2 + 1, 1\}, \{\sqrt{2}, 1\}\}) \text{ STO } \triangleright L1$$

then you have :

$$\text{TRAN}(L1) = \{\{S2 + 1, \sqrt{2}\}, \{1, 1\}\}$$

Some numeric linear algebra instructions do not work on list of lists, but will after a conversion by AXL , like:

$$\text{DET}(\text{AXL}(L1)) \text{ STO } \triangleright E1$$

you get :

$$S2 - (-1 + \sqrt{2})$$

Section 4 describes CAS instructions that work only from HOME, including some CAS symbolic linear algebra instructions. For more information you can also use the on-line help (HELP or e.g. HELPWITH TRAN) or look at the HP49G, HP49G+ or HP48GII CAS documentation available from:

<http://www-fourier.ujf-grenoble.fr/~parisse>

3.4.4 Send expressions from HOME into the CAS history : PUSH

In the HOME screen, you can use the PUSH command to send expressions into the CAS history.

In the HOME screen, you type:

```
PUSH(S1+1)
```

and S1+1 is written to the CAS history.

3.4.5 Retrieve expressions written to the CAS history in HOME : POP

In the HOME screen, you can use the POP command to retrieve the last expression written to the CAS history.

In the HOME screen, you type:

```
POP
```

and, for example, S1+1 is written to the HOME screen history and S1+1 is removed from the CAS history.

3.4.6 The keyboard in the HOME screen

MATH key

This opens the menu of mathematical functions.

This key, if pressed in the HOME screen, opens a window containing the mathematical (numeric) functions grouped by theme, since the MTH option on the menu bar (F1 key) is selected by default.

If you choose CAS on this window's menu bar (F4 key), you'll find the same menus as when you press the MATH key in the Equation Editor: this gives you access to all available computer algebra functions, grouped by theme, from the HOME screen.

SHIFT F6 keys

The key combination SHIFT F6 (CAS on the menu bar) opens the CAS MODES screen, which enables you to change the CAS configuration from the HOME screen (cf 3.3.2).

SHIFT 2 (SYNTAX) keys

The key combination SHIFT 2 (SYNTAX) places HELPWITH in the command line. All you need to do then is complete the line with the name of the command or CAS function for which you want help. You can enter the name of a CAS function with MATH and CAS on this window's menu bar (F4 key), but take care to omit the parentheses.

For example: HELPWITH DERVX opens the CAS help topic on DERVX.

If you want general CAS help in the HOME screen, press HELP, then ENTER. This gives you help on the CAS functions available in the HOME screen.

To get help in French, choose Français in the CFG menu, which enables you to change your configuration (cf 5.1.1).

Each example can be copied into the HOME screen history by means of ECHO on the menu bar, where it can be used as is or modified, and naturally, the variable X of the example is replaced automatically by S1.

In addition, the examples are sometimes modified automatically to change in HOME screen, reals to integers using the XQ function.

For example, you type in the HOME screen:

HELPWITH PROPFRAC

or

HELP then you select PROPFRAC in CAS HELP ON

You obtain the example for PROPFRAC :

$$\text{PROPFRAC}(\text{XQ}(\frac{43}{12})) = 3 + \frac{7}{12}$$

whereas if, in the equation editor, you select PROPFRAC on the CAS HELP ON, the example is :

$$\text{PROPFRAC}(\frac{43}{12})$$

but typing in HOME :

$$\text{PROPFRAC}(\frac{43}{12})$$

gives

3.5833..

Chapter 4

The CAS functions only valid in HOME screen

These functions have as arguments matrix or vectors and return matrix or vectors.

4.1 Linear systems

In this paragraph, we call "augmented matrix" of the system $A \cdot X = B$ (or matrix "representing" the system $A \cdot X = B$), the matrix obtained augmenting the matrix A to the right with the column vector B .

4.1.1 Resolution of a linear system of equations : REF

REF solves a linear system of equations written in matrix form:

$$A \cdot X = B$$

The argument of REF is the augmented matrix of the system (the matrix obtained augmenting matrix A to the right with the column vector B).

The result is a matrix $[A1, B1]$: $A1$ has zeros under its principal diagonal, and the solutions of:

$$A1 \cdot X = B1$$

are the same as:

$$A \cdot X = B$$

For example, solve the system:

$$\begin{cases} 3 \cdot x + y & = & -2 \\ 3 \cdot x + 2 \cdot y & = & 2 \end{cases}$$

Typing:

$$\text{REF}([[3, 1, -2][3, 2, 2]])$$

gives:

$$\begin{bmatrix} 1 & \frac{1}{3} & \frac{-2}{3} \\ 0 & 1 & 4 \end{bmatrix}$$

So, $y = 4$ and $x = -2/3 - 4/3 = -2$

4.1.2 Resolution of a linear system of equations : `rref`

`rref` solves a linear system of equations written in matrix form:

$$A \cdot X = B$$

The argument of `rref` is the augmented matrix of the system (the matrix obtained augmenting matrix A to the right with the column vector B).

The result is a list containing the list of pivot elements used by the command and a matrix $[A1, B1]$: A1 has zeros both above and under its principal diagonal, and the solutions of:

$$A1 \cdot X = B1$$

are the same as:

$$A \cdot X = B$$

It is interesting to use `rref` in step-by-step mode.

For example, to solve the system:

$$\begin{cases} 3 \cdot x + y & = & -2 \\ 3 \cdot x + 2 \cdot y & = & 2 \end{cases}$$

Typing

$$\text{rref}([[3, 1, -2][3, 2, 2]])$$

gives :

$$\left\{ \{1 \ 1\} \begin{bmatrix} 3 & 0 & -6 \\ 0 & 1 & 4 \end{bmatrix} \right\}$$

Typing

$$\text{rref}([[1, S1][S1, 1]])$$

gives :

$$\left\{ \{S1^2 - 1, 1\} \left[\begin{array}{cc} S1^2 - 1 & 0 \\ 0 & -(S1^2 - 1) \end{array} \right] \right\}$$

4.2 Polynomials

4.2.1 List of factors : FACTORS

FACTORS has either a polynomial or a list of polynomials as argument. FACTORS returns a list containing the factors of the polynomial and their exponents.

Typing:

$$\text{FACTORS}(S1^2 - 2 * S1 + 1)$$

gives:

$$\{S1 - 1, 2.\}$$

Typing:

$$\text{FACTORS}(S1^4 - 2 * S1^2 + 1)$$

gives:

$$\{S1 - 1, 2, S1 + 1, 2\}$$

Typing:

$$\text{FACTORS}(\{S1^3 - 2 * S1^2 + 1, S1^2 - S1\})$$

gives:

$$\{\{S1 - 1, 1, 2 * S1 + -1 + \sqrt{5}, 1, 2 * S1 - (1 + \sqrt{5}), 1, 4, -1\}, \\ \{S1, 1, S1 - 1, 1\}\}$$

4.2.2 Rational function given by its roots and poles :FCOEF

FCOEF has as argument a vector whose components are the roots and poles of a rational function $F[x]$, followed by their multiplicity. FCOEF returns the rational function $F[x]$.

Typing:

$$\text{FCOEF}([1, 2, 0, 3, 2, -1])$$

gives:

$$S1^3 * (S1 - 1)^2 / (S1 - 2)$$

4.2.3 Roots and poles of a rational function : FROOTS

FROOTS has a rational function $F[x]$ as argument.

FROOTS returns a vector whose components are the roots and the poles of $F[x]$, followed by their multiplicity.

Typing:

$$\text{FROOTS}\left(\frac{S1^5 - 2 \cdot S1^4 + S1^3}{S1 - 2}\right)$$

gives:

$$[2, -1, 0, 3, 1, 2]$$

The result means that: 2 is a pole of order 1, 0 is a triple root, and 1 is a double root of $F[x] = \frac{x^5 - 2 \cdot x^4 + x^3}{x - 2}$ since $(x - 1)^2 \cdot x^3 = x^5 - 2 \cdot x^4 + x^3$.

4.2.4 Gröbner basis : GBASIS

GBASIS has two arguments : a vector of polynomials with several variables and the vector of names of these variables.

GBASIS returns a Gröbner basis of the polynomial ideal built by the polynomials which are in the first argument.

We choose to write the polynomial variables with the order of the last argument and to write the polynomials in decreasing power. If I is an ideal and if $(G_k)_{k \in K}$ is a Gröbner basis of this ideal I then if F is a non zero polynomial in I , the greatest power term of F is divisible by the greatest power term of a G_k .

Property : If you do an euclidian division of F by its correspondent G_k , then if you continue with the remainder of this division, we have at end, a null remainder.

Typing :

$$\text{GBASIS}([2 \cdot S1 \cdot S2 - S2^2, S1^2 - 2 \cdot S1 \cdot S2], [S1, S2])$$

gives :

$$[S1^2 - S2^2, 2 \cdot S1 \cdot S2 - S2^2, S2^3]$$

4.2.5 Gröbner reduction : GREDUCE

GREDUCE has three arguments : a polynomial with several variables, a vector made with polynomials which are a Gröbner basis and a vector of names of these variables.

GREDUCE returns the reduction (+ a constant) of the polynomial given in

the first argument with respect to the Gröbner basis given in the second argument.

Typing :

```
GREDUCE(S1 * S2 - 1, [S12 - S22, 2 * S1 * S2 - S22, S23], [S1, S2])
```

gives :

$$(S2^2 - 2)/2$$

that is to say $Y^2 - 2$ is the reduce polynomial of $2 \cdot X \cdot Y - 2$ ($Y^2 - 2$ is the remainder of $2 \cdot (X \cdot Y - 1)$ by $G_2 = 2 \cdot X \cdot Y - Y^2$).

4.2.6 Lagrange's polynomial : LAGRANGE

LAGRANGE has as argument a matrix with two rows and n columns:

the first row corresponds to the abscissa values x_i , and the second row corresponds to ordinate values y_i ($i = 1..n$).

LAGRANGE returns the polynomial P of degree $n - 1$, so that $P(x_i) = y_i$.

Typing:

```
LAGRANGE([[1, 3], [0, 1]])
```

gives:

$$\frac{S1 - 1}{2}$$

in fact $\frac{x-1}{2} = 0$ for $x = 1$ and $\frac{x-1}{2} = 1$ for $x = 3$

4.2.7 GCD of a list : LGCD

LGCD denotes the gcd (greatest common divisor) of a list of whole numbers or of a list of polynomials.

LGCD returns the GCD of all whole numbers of the list or the GCD of all polynomials of the list.

Typing:

```
LGCD({125, 45, 35})
```

gives:

$$5$$

Typing:

```
LGCD({S12 + 2 * S1 + 1, S13 + 1, S12 - 1, S12 + S1})
```

gives :

$$S1 + 1$$

4.2.8 resultant of two polynomials: RESULTANT

RESULTANT has two polynomials as arguments.

RESULTANT returns the resultant of the two polynomials. The resultant of two polynomials is non zero if and only if the polynomials are prime together. It is the determinant of their Sylvester matrix S .

The Sylvester matrix S of two polynomials $A[X] = \sum_{i=0}^{i=n} a_i X^i$ and $B[X] = \sum_{i=0}^{i=m} a_i X^i$ is a square matrix with $m + n$ rows and columns; its first m rows are made from the coefficients of $A[X]$:

$$\begin{pmatrix} s_{11} = a_n & s_{12} = a_{n-1} & \cdots & s_{1(n+1)} = a_0 & 0 & \cdots & 0 \\ s_{21} = 0 & s_{22} = a_n & \cdots & s_{2(n+1)} = a_1 & s_{2(n+2)} = a_0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ s_{m1} = 0 & s_{m2} = 0 & \cdots & s_{m(n+1)} = a_{m-1} & s_{m(n+2)} = a_{m-2} & \cdots & a_0 \end{pmatrix}$$

and the following n rows are made in the same way from the coefficients of $B[X]$:

$$\begin{pmatrix} s_{(m+1)1} = b_m & s_{(m+1)2} = b_{m-1} & \cdots & s_{(m+1)(m+1)} = b_0 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \end{pmatrix}$$

For example, typing:

$$\text{RESULTANT}(S1^3 - S2 * S1 + S3, 3 * S1^2 - S2)$$

gives:

$$-(4 * S2^3 - 27 * S3^2)$$

4.2.9 Division by the PGCD :SIMP2

SIMP2 has two whole numbers (or two lists of whole numbers with the same length) as arguments. These two whole numbers are considered as representing a fraction.

SIMP2 returns the simplified fraction, represented as a list of two whole numbers.

Typing:

$$\text{SIMP2}(18, 15)$$

gives:

$$\{6, 5\}$$

Typing:

$$\text{SIMP2}(\{18, 28\}, \{15, 21\})$$

gives:

$$\{6, 5, 4, 3\}$$

SIMP2 has two polynomials (or two lists of polynomials with the same length) as arguments. These two polynomials are considered as representing a rational fraction.

SIMP2 returns the simplified rational fraction, represented as a list of two polynomials.

Typing:

$$\text{SIMP2}(S1^3 - 1, S1^2 - 1)$$

gives:

$$\{S1^2 + S1 + 1, S1 + 1\}$$

4.2.10 Sturm's sequences : STURM

STURM has a polynomial P as argument.

STURM returns a list containing the Sturm's sequences of P and their multiplicities.

The Sturm sequence R_1, R_2, \dots can be obtained from a square-free factor F of P as follows:

R_1 is the opposite of the remainder of the euclidean division of F by F' ; then, R_2 is the opposite of the remainder of the euclidean division of F' by R_1 ,

and so on, until $R_k = 0$.

For example, type:

$$\text{STURM}(S1^3 + 1)$$

You obtain:

$$\{\{1\}, -1, [S1^3 + 1, -3 \cdot S1^2 - 1], 1.\}$$

The first element of the list denotes that the denominator of P (that is, the element with -1 power), is 1.

4.2.11 Zeros of a polynomial : ZEROS

ZEROS has two arguments: a polynomial P and a variable name.

ZEROS returns a list containing the zeros of P with respect to the given variable, WITHOUT their multiplicity.

Type:

$$\text{ZEROS}(S1^4 - 1, S1)$$

gives:

- in real mode (change the mode with `SHIFT F6` and `UNCHK Complex`)

$$\{-1, 1\}$$

- in complex mode (change the mode with `SHIFT F6` and `CHK Complex`)

$$\{i, -1, -i, 1\}$$

4.3 Rational fractions

4.3.1 Numerator and denominator of a fraction : FXND

FXND has a rational fraction as argument, and returns a list containing the simplified numerator and denominator of this fraction.

Typing:

$$\text{FXND}(3/6)$$

gives :

$$\{1, 2\}$$

Type:

$$\text{FXND}((S1^2 - 1)/(S1 - 1))$$

You obtain:

$$\{S1^2 - 1, S1 - 1\}$$

4.4 Modular calculations

4.4.1 Resolution of a linear system of equations : RREFMOD

RREFMOD solves a system of linear equations $AX = B$ in Z/pZ .

The argument is the matrix A augmented with the vector B as its rightmost column. The result is a matrix composed of $A1$ and $B1$, where $A1$ has zeros both above and under its principal diagonal, and the system $A1X = B1$ is equivalent to $AX = B$.

To solve, in $Z/13Z$:

$$\begin{cases} x + 2 \cdot y & = & 9 \\ 3 \cdot x + 10 \cdot y & = & 0 \end{cases}$$

Typing :

$$\text{MODSTO}(13)$$

$$\text{RREFMOD}([1, 2, 9][3, 10, 0])$$

gives :

$$[[2, 0, 6], [0, 4, -1]]$$

that is, $2X = 6$ and $4Y = -1$ or, which is the same, $X = 3$ $Y = 3$ (since $-4 * 3 = 1 \pmod{13}$).

4.5 Variables

4.5.1 The symbolic variables of an expression : LNAME

LNAME has an expression as argument, and returns a vector whose components are the symbolic variable names the expression contains.

Typing :

$$\text{LNAME}(S1 * S2 * \text{SIN}(S1))$$

gives :

$$[S1, S2]$$

4.5.2 The variables of an expression : LVAR

LVAR has an expression as argument, and returns a vector such that the original expression is a rational fraction with respect to the components of this vector.

Typing :

$$\text{LVAR}(S1 * S2 * \text{SIN}(S1))$$

gives :

$$[\text{SIN}(S1), S1, S2]$$

4.6 Numeric calculus

4.6.1 Omit the values smaller than EPS : EPSX0

EPSX0 has as argument an expression; in the expression, it replaces all numeric values whose magnitude is smaller than EPS with zero and returns the result.

Type:

$$\text{EPSX0}(0.001 + S1)$$

You obtain (when $\text{EPS}=0.01$) :

$$0 + S1$$

You obtain (when $\text{EPS}=0.0001$) :

$$.001 + S1$$

4.6.2 Solve numerically a system : MSLV

MSLV solves numerically a system of non-polynomial equations.

MSLV has three vectors as arguments: a vector containing the equations, a vector containing the system's variables, and a vector containing an initial guess for the solution.

MSLV returns a vector containing an approximate solution of the given system of equations.

While the command is running, the first display line shows the last estimate \vec{V} , and the second line shows the modulo of $\Delta \vec{V}$

For example, type:

$$\text{MSLV}([\text{SIN}(S1) + S2, S1 + \text{SIN}(S2) - 1], [S1, S2], [0, 0])$$

You obtain:

$$[1.82384112611, -.968154636174]$$

List of values : TABVAL

TABVAL has an expression and a list of numbers as arguments.

TABVAL returns a list. This list holds two lists : the list of numbers given as input and the list of the value the expression where the variable S1 is replaced by the list of numbers given as input.

For example, typing:

$$\text{TABVAL}(S1^2 + S1, \{1, 2, 3, XQ(\sqrt{5})\})$$

gives:

$$\{\{1, 2, 3, \sqrt{5}\}, \{2, 6, 12, 5 + \sqrt{5}\}\}$$

4.6.3 Time for evaluation of the answer : TEVAL

TEVAL has a command as argument.

TEVAL returns the list of the answer and the time for the evaluation of this

answer (unit=second).

For example, typing:

$$\text{TEVAL}(\text{FACTOR}(S1^2 - 1))$$

gives:

$$\{(S1 - 1) * (S1 + 1), 0.2991\}$$

4.6.4 Numeric approximation : XNUM

XNUM has either an expression or an array as argument.

XNUM enables approximate mode and returns the numeric approximation of its argument.

Type:

$$\text{XNUM}(\sqrt{2})$$

You obtain:

$$1.41421356237$$

4.6.5 Exact value : XQ

XQ has a real numeric expression as argument.

XQ enables exact mode and returns either a rational or a real approximation of the expression.

Typing:

$$\text{XQ}(1.41421)$$

gives:

$$\frac{66441}{46981}$$

Typing:

$$\text{XQ}(1.414213562)$$

gives:

$$\sqrt{2}$$

4.7 The lists

4.7.1 Concatenation : AUGMENT

AUGMENT has as arguments two vectors, or two lists, or a list and an element.

AUGMENT concatenates its arguments.

Typing:

$$\text{AUGMENT}(\{1, 2\}, 3)$$

gives:

$$\{1, 2, 3\}$$

Typing:

$$\text{AUGMENT}(\{1, 2\}, \{3, 4\})$$

gives:

$$\{1, 2, 3, 4\}$$

4.7.2 Map a function to a list : MAP

MAP has two arguments: a list and a function defining an holomorphic operator.

MAP applies the operator to the list, and returns the result.

Typing:

$$\text{MAP}(1, 0, 2, \text{EXP}(S1))$$

gives:

$$\{\text{EXP}(1), \text{EXP}(0), \text{EXP}(2)\}$$

4.8 Quadratic forms

4.8.1 Matrice to quadratic form : AXQ

AXQ has two arguments: a symmetric matrix A representing a quadratic form q and a vector whose components are the quadratic form's variables.

AXQ returns the quadratic form q .

Typing :

$$\text{AXQ}(\{[0, 1], [1, 0]\}, [S1, S2])$$

gives :

$$2 * S2 * S1$$

4.8.2 Gauss decomposition : GAUSS

GAUSS has two arguments: a quadratic form q and a vector whose components are the quadratic form's variables.

GAUSS returns q expressed as a sum of squares.

Typing:

$$\text{GAUSS}(2 \cdot S1 \cdot S2, [S1, S2])$$

gives:

$$-2 \cdot \left(\frac{S2 - S1}{2}\right)^2 + \frac{1}{2} \cdot (S2 + S1)^2$$

4.8.3 Matrix of a quadratic form : QXA

QXA has two arguments: a quadratic form q and a vector whose components are the form's variables.

QXA returns the matrix A associated with q .

Typing :

$$\text{QXA}(2 * S1 * S2, [S1, S2])$$

gives :

$$[[0, 1], [1, 0]]$$

and with SHOW

$$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

4.8.4 Sylvester decomposition : SYLVESTER

SYLVESTER has one argument: a symmetric matrix representing a quadratic form q .

SYLVESTER returns a list of two elements: the diagonal elements of the diagonal matrix B (obtained expressing q as a sum of squares) and the matrix of change of base Q .

We have:

$${}^tQ \cdot B \cdot Q = A$$

Typing:

$$\text{SYLVESTER}([[0, 1], [1, 0]])$$

gives :

$$\left\{ \left[\frac{1}{2}, -2 \right], \begin{bmatrix} 1 & 1 \\ -\frac{1}{2} & \frac{1}{2} \end{bmatrix} \right\}$$

4.9 Functions of multiple variables

4.9.1 The rotor : CURL

Here $n = 3$.

CURL has two arguments: the value $[E1, E2, E3]$ of a vectorial function (application from R^3 in R^3) and a vector of R^3 denoting the variable names.

CURL returns the rotor of $[E1, E2, E3]$.

$$\text{CURL}([E1, E2, E3], [S1, S2, S3]) = \left[\frac{\partial E3}{\partial S2} - \frac{\partial E2}{\partial S3}, \frac{\partial E1}{\partial S3} - \frac{\partial E3}{\partial S1}, \frac{\partial E2}{\partial S1} - \frac{\partial E1}{\partial S2} \right]$$

Type:

$$\text{CURL}([S1 * S3, -S2^2, 2 * S1 * S2], [S1, S2, S3])$$

You obtain:

$$[2 * S1, S1 - S2 * 2, 0]$$

4.9.2 The gradient : DERIV

DERIV has two arguments: the value of an application $E1$ from R^n in R and a vector of R^n denoting the variable names.

DERIV returns the gradient of $E1$ ($[\frac{\partial E1}{\partial S1}, \frac{\partial E1}{\partial S2}, \frac{\partial E1}{\partial S3}]$ if $n = 3$).

Type:

$$\text{DERIV}(2 * S1^2 * S2 - S1 * S3^3, [S1, S2, S3])$$

After simplification, you obtain:

$$[4 * S2 * S1 - S3^3, 2 * S1^2, -(3 * S1 * S3^2)]$$

4.9.3 The divergence : DIV

DIV has two arguments: the value $[E1, E2, E3]$ of a vectorial function (application from R^n in R^n) and a vector of R^n denoting the variable names.

DIV returns the divergence of $[E1, E2, E3]$.

$$\text{DIV}([E1, E2, E3], [S1, S2, S3]) = \frac{\partial E1}{\partial S1} + \frac{\partial E2}{\partial S2} + \frac{\partial E3}{\partial S3} \text{ (here } n = 3\text{)}$$

Typing:

$$\text{DIV}([S1 * S3, -S2^2, 2 * S1 * S2], [S1, S2, S3])$$

gives:

$$S3 - 2 * S2$$

4.9.4 The hessian : HESS

HESS has two arguments: the value of an application $E1$ from R^n in R and a vector of R^n denoting the variable names.

HESS returns a list containing the hessian of $E1$, the gradient of $E1$ and the vector of the variable names.

Typing :

$$\text{HESS}(2 * S1^2 * S2 - S1 * S3, [S1, S2, S3])$$

gives :

$$\left\{ \begin{bmatrix} 2 \cdot 2 \cdot S2 & 2 \cdot 2 \cdot S1 & -1 \\ 2 \cdot 2 \cdot S1 & 0 & 0 \\ -1 & 0 & 0 \end{bmatrix}, [4 \cdot S1 \cdot S2 - S3, 2 \cdot S1^2, -S1], [S1, S2, S3] \right\}$$

Now, to obtain the critical points of $E1$, you must type:

$$\text{SOLVE}([4 \cdot S1 \cdot S2 - S3, 2 \cdot S1^2, -S1], [S1, S2, S3])$$
4.9.5 The laplacian : LAPL

LAPL has two arguments: the value of an application $E1$ from R^n in R and a vector of R^n denoting the variable names.

LAPL returns the laplacian of $E1$ ($\frac{\partial^2 E1}{\partial S1^2} + \frac{\partial^2 E1}{\partial S2^2} + \frac{\partial^2 E1}{\partial S3^2}$ if $n = 3$).

Type:

$$\text{LAPL}(2 * S1^2 * S2 - S1 * S3^3, [S1, S2, S3])$$

You obtain:

$$4 * S2 - 6 * S1 * S3$$
4.10 Equations**4.10.1 Members of an equation : EXLR**

EXLR has an equation as argument (you have to type ==).

EXLR returns a list containing the left and right hand sides of the equation.

Type:

$$\text{EXLR}(XQ(2 \cdot S1 + 1 == 3/2))$$

You obtain:

$$\{2 \cdot S1 + 1, 3/2\}$$

4.11 Matrices

4.11.1 Matrix to list : AXL

When AXL is given a matrix as argument, it returns the same matrix rewritten as a list of lists. Vice versa, AXL transforms a list of lists into a matrix.

Typing:

$$\text{AXL}(\text{XQ}([[1, 1/2], [3, 4]]))$$

gives:

$$\{\{1, 1/2\}, \{3, 4\}\}$$

Typing:

$$\text{AXL}(\text{XQ}(\{\{1, 1/2\}, \{3, 4\}\}))$$

gives:

$$[[1, 1/2], [3, 4]]$$

But typing:

$$\text{AXL}([[1, 1/2], [3, 4]])$$

gives:

$$\{\{1, 0.5\}, \{3, 4\}\}$$

and typing

$$\text{AXL}(\{\{1, 1/2\}, \{3, 4\}\})$$

gives:

$$[[1, 0.5], [3, 4]]$$

4.11.2 Symbolic matrix to numeric matrix : AXM

If given a symbolic matrix as argument, AXM returns an equivalent (but approximate) numeric matrix.

Type:

$$\text{AXM}([[1/2, 2], [3, 4]])$$

You obtain:

$$[[0.5, 2], [3, 4]]$$

4.11.3 Basis vectors : BASIS

BASIS has as argument a list of vectors, defining a vectorial subspace of R^n .

BASIS returns a list containing the vectors of a basis of the vectorial subspace.

For example, type:

```
BASIS({[1, 2, 3], [1, 1, 1], [2, 3, 4]})
```

You obtain:

```
{[1, 0, -1], [0, 1, 2]}
```

4.11.4 Cholesky matrix : CHOLESKY

CHOLESKY has as argument a square matrix M, positive by definition.

CHOLESKY returns an upper triangular matrix P so that:

${}^tP * P = M$

Typing:

```
CHOLESKY([[1, 1], [1, 5]])
```

gives :

$$\begin{bmatrix} 1 & 1 \\ 0 & 2 \end{bmatrix}$$
4.11.5 Map an operator to a diagonalizable matrix : DIAGMAP

DIAGMAP has two arguments: a diagonalizable matrix and an expression defining an holomorphic operator.

DIAGMAP applies the operator to the matrix, and returns the result.

Typing:

```
DIAGMAP([[1, 1], [0, 2]], EXP(S1))
```

gives:

$$\begin{bmatrix} \text{EXP}(1) & \text{EXP}(2) - \text{EXP}(1) \\ 0 & \text{EXP}(2) \end{bmatrix}$$
4.11.6 Hadamard product : HADAMARD

HADAMARD has two matrices A and B, with the same size, as arguments.

HADAMARD returns the element-by-element product between A and B.

Type:

```
HADAMARD(XQ([[1/2, 2], [3, 4]]), XQ([[5, 6], [7, 8]]))
```

or

HADAMARD(XQ({{1/2, 2}, {3, 4}}), XQ({{5, 6}, {7, 8}}))

You obtain:

$[[5/2, 12], [21, 32]]$

but

HADAMARD([[1/2, 2], [3, 4]], [[5, 6], [7, 8]])

or

HADAMARD({{1/2, 2}, {3, 4}}, {{5, 6}, {7, 8}})

You obtain:

$[[2.5, 12], [21, 32]]$

4.11.7 Hilbert matrix : HILBERT

HILBERT has an integer n as argument.

HILBERT returns the square Hilbert matrix of order n whose elements are given by:

$$a_{i,j} = \frac{1}{i + j - 1}$$

Type:

HILBERT(4)

You obtain, pressing ENTER, then \uparrow , and then SHOW :

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

4.11.8 Basis of an intersection : IBASIS

IBASIS has as arguments two lists of vectors, defining two vectorial spaces.

IBASIS returns a list containing the vectors of a basis of the intersection between these two vectorial spaces.

Typing:

IBASIS({[1, 2]}, {[2, 4]})

gives:

{[1, 2]}

4.11.9 basis of the image : IMAGE

IMAGE has as argument a matrix representing a linear application f in terms of the standard basis.

IMAGE returns a list of vectors; they are a basis of the image of f .

Typing:

$$\text{IMAGE}([[1, 2, 3], [2, 1, 3], [3, 1, 4]])$$

gives :

$$\{[1, 0, -1/3], [0, 1, 5/3]\}$$
4.11.10 Jordan matrix : JORDAN

JORDAN has a matrix A , of order n , as argument.

JORDAN returns a list composed by the minimal polynomial M of A , the characteristic polynomial P of A , the list of the eigenvectors and characteristic vectors (each vector is preceded by its characteristic value), and the vector of the n eigenvalues of A .

Type:

$$\text{JORDAN}([[4, 1, -2], [1, 2, -1], [2, 1, 0]])$$

or

$$\text{JORDAN}(\{\{4, 1, -2\}, \{1, 2, -1\}, \{2, 1, 0\}\})$$

You obtain pressing ENTER, then \uparrow , and then SHOW :

$$\{S1^3 - 6 \cdot S1^2 + 12 \cdot S1 - 8, S1^3 - 6 \cdot S1^2 + 12 \cdot S1 - 8, \\ \{[1, 0, 0], [2, 1, 2], [1, 0, 1]\}, [2, 2, 2]\}$$
4.11.11 Basis of the kernel : KER

KER has as argument a matrix representing a linear application f in terms of the standard basis.

KER returns a list of vectors; they are a basis of the kernel of f .

Typing:

$$\text{KER}([[1, 2, 3], [2, 1, 3], [3, 1, 4]])$$

gives:

$$\{[1, 1, -1]\}$$

4.11.12 Make a matrix : LCXM

LCXM has as arguments two integers, n and p , and an expression of S1 (a row number) and of S2 (a column number) and yielding the value of $a_{i,j}$. LCXM returns a $n \cdot p$ matrix having coefficients $a_{i,j}$. Type:

$$\text{LCXM}(2, 3, \text{S1} + \text{S2})$$

You obtain:

$$[[1 + 1, 1 + 2, 1 + 3], [2 + 1, 2 + 2, 2 + 3]]$$

and then after simplification :

$$[[2, 3, 4], [3, 4, 5]]$$

4.11.13 List of several elements : MAD

MAD has a square matrix A , of order n , as argument.

MAD returns a list containing the determinant of A , the inverse of A , a list containing the matrix coefficients of a polynomial Q , and of the characteristic polynomial P of A .

We have:

$$P(x) = (-1)^n \cdot \det(A - x \cdot I)$$

The polynomial with matrix coefficients $P(A) - P(x) \cdot I$ is divisible by $A - x \cdot I$ (since its value is zero for $x = A$). Let $Q(x)$ be their quotient.

Since $P(A) = 0$, we have $P(A) - P(x) \cdot I = -P(x) \cdot I = (A - x \cdot I) \cdot Q(x)$.

Therefore, $Q(x)$ is also the co-matrix of $A - x \cdot I$ and the following holds: $Q(x) = I \cdot x^{n-1} + \dots + B_0$, where B_0 is the co-matrix of A (with the sign exchanged if n is even!).

Type:

$$\text{MAD}([[1, 1/2], [1, 3]])$$

or

$$\text{MAD}(\{\{1, 1/2\}, \{1, 3\}\})$$

You obtain:

$$\{5/2, [[6/5, -1/5], [-2/5, 2/5]], \{[1, 0], [0, 1]\}, [[-3, 1/2], [1, -1]], \text{S1}^2 - 4 * \text{S1} + 5/2\}$$

or pressing ENTER, then \uparrow , and then SHOW gives:

$$\left\{ \frac{5}{2}, \left[\begin{array}{cc} \frac{6}{5} & -\frac{1}{5} \\ -\frac{2}{5} & \frac{2}{5} \end{array} \right], \left\{ \left[\begin{array}{cc} 1 & 0 \\ 0 & 1 \end{array} \right], \left[\begin{array}{cc} -3 & 1/2 \\ 1 & -1 \end{array} \right] \right\}, \text{S1}^2 - 4 \cdot \text{S1} + \frac{5}{2} \right\}$$

4.11.14 Characteristic polynomial : PCAR

PCAR has a matrix A , of order n , as argument.

PCAR returns the characteristic polynomial P of A ($P[x] = (-1)^n \cdot \det(A - x \cdot I)$) Type:

PCAR([[4, 1, -2], [1, 2, -1], [2, 1, 0]])

or

PCAR({{4, 1, -2}, {1, 2, -1}, {2, 1, 0}})

You obtain:

$$S1^3 - 6 * S1^2 + 12 * S1 - 8$$

4.11.15 Minimal polynomial : PMINI

PMINI has a matrix A as argument.

PMINI returns another matrix, whose first "zero row" is ended by the minimal polynomial of A .

For example, type:

PMINI([[1, 0], [0, 1]])

In step-by-step mode, you obtain:

L2=L2-L1

$$\begin{bmatrix} 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 & S1 \\ 1 & 0 & 0 & 1 & S1^2 \end{bmatrix}$$

L3=L3-L1

$$\begin{bmatrix} 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & S1 - 1 \\ 1 & 0 & 0 & 1 & S1^2 \end{bmatrix}$$

Reduction result

$$\begin{bmatrix} 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & S1 - 1 \\ 0 & 0 & 0 & 0 & S1^2 - 1 \end{bmatrix}$$

The row :

$0, 0, 0, 0, S1-1$ gives the minimal polynomial of A . So, the minimal

polynomial of A : $\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$ is:

$$S1 - 1$$

4.11.16 Q*R decomposition : qr

qr has a square matrix as argument.

qr factors the matrix as Q*R, where Q is an orthogonal matrix and R is a triangular matrix.

Typing:

$$\text{qr}(\left[\left[\begin{array}{cc} 3 & 5 \\ 4 & 5 \end{array}\right]\right])$$

gives:

$$\left\{ \left[\begin{array}{cc} \frac{3}{5} & \frac{4}{5} \\ -\frac{4}{5} & \frac{3}{5} \end{array} \right], \left[\begin{array}{cc} 5 & 7 \\ 0 & 1 \end{array} \right] \right\}$$

4.11.17 System to matrix : SYST2MAT

SYST2MAT has two arguments: a vector containing a system of linear equations (= or == is not allowed and so =0 is omitted), and a vector whose elements are the system's variables.

SYST2MAT rewrites the system in matrix notation, and returns the matrix. For example, type:

$$\text{SYST2MAT}(\left[\left[S1 + S2, S1 - S2 - 2\right], \left[S1, S2\right]\right])$$

You obtain:

$$\left[\begin{array}{ccc} 1 & 1 & 0 \\ 1 & -1 & -2 \end{array} \right]$$

4.11.18 Transposed matrix : TRAN

TRAN has a matrix A as argument.

TRAN returns the input matrix A transposed.

Typing :

$$\text{TRAN}(\text{XQ}(\left[\left[\begin{array}{cc} 1 & 1/2 \\ 3 & 4 \end{array}\right]\right]))$$

gives:

$$\left[\left[\begin{array}{cc} 1 & 3 \\ 1/2 & 4 \end{array}\right]\right]$$

Typing :

$$\text{TRAN}(\text{XQ}(\left\{\left\{\begin{array}{cc} 1 & 1/2 \\ 3 & 4 \end{array}\right\}\right\}))$$

gives:

$$\left\{\left\{\begin{array}{cc} 1 & 3 \\ 1/2 & 4 \end{array}\right\}\right\}$$

But typing :

$$\text{TRAN}(\left[\left[\begin{array}{cc} 1 & 1/2 \\ 3 & 4 \end{array}\right]\right])$$

gives:

$$[[1, 3], [0.5, 4]]$$

4.11.19 Vandermonde matrix : VANDERMONDE

VANDERMONDE has as argument a vector whose components are denoted by x_i for $i = 0..n - 1$.

VANDERMONDE returns the corresponding Vandermonde matrix (the k -th row of the matrix is the vector whose components are x_i^k for $i = 0..n - 1$ and $k = 0..n - 1$).

Type:

$$\text{VANDERMONDE}([1, 2, 3])$$

You obtain:

$$[[1, 1, 1], [1, 2, 4], [1, 3, 9]]$$

Suppose that $E1=6$, $E2=2*S1+S1*S2$ and then type:

$$\text{VANDERMONDE}([E1, E2, E3])$$

You obtain pressing ENTER, then \uparrow , and then SHOW:

$$\begin{bmatrix} 1 & 6 & 36 \\ 1 & 2 \cdot S1 + S1 \cdot S2 & (2 \cdot S1 + S1 \cdot S2)^2 \\ 1 & E3 & E3^2 \end{bmatrix}$$

4.12 The permutations

Permutations are defined with the image list $\{P(1), P(2) \dots P(n)\}$.

For example, defining the permutation P as $P = \{3, 2, 1\}$ means that:

$P(1) = 3$, $P(2) = 2$, $P(3) = 1$.

A cycle is denoted with a list containing the images of an element through the cycle; for example, defining a cycle C as $C = \{3, 2, 1\}$ means that:

$C(3) = 2$, $C(2) = 1$, $C(1) = 3$.

Accordingly, a decomposition into cycles is denoted with a list of lists.

4.12.1 Cycles to permutation : C2P

C2P has a list of cycles as argument.

C2P returns the permutation having the input list as decomposition into

cycles (see also P2C).

Typing:

$$\text{C2P}(\{\{1, 3, 5\}, \{2, 4\}\})$$

gives:

$$\{3, 4, 5, 2, 1\}$$

4.12.2 Composition : CIRC

CIRC has two permutations as arguments.

CIRC returns the permutation obtained by composition: (1stargument \circ 2ndargument).

Typing:

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

gives :

$$\{4, 3, 2, 5, 1\}$$

4.12.3 Decomposition into cycles P2C

P2C has a permutation as argument.

P2C returns its decomposition into cycles and its signature.

Typing:

$$\text{P2C}(\{3, 4, 5, 2, 1\})$$

gives :

$$\{\{\{1, 3, 5\}, \{2, 4\}\}, -1\}$$

4.13 The isometries

4.13.1 Isometry's characteristics elements : ISOM

ISOM has as argument a matrix representing a two or three-dimensional linear isometry.

ISOM returns the list of the isometry's characteristics elements and either +1 (for direct isometries) or -1 (for indirect isometries).

For example, type:

$$\text{ISOM}(\left[\left[\begin{array}{ccc} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{array}\right]\right])$$

You obtain:

$$\{[1, 0, -1], -1\}$$

This result means that the isometry is a symmetry with respect to the plane $x - z = 0$.

If you type:

$$\text{ISOM}\left(\frac{\sqrt{2}}{2} \cdot \begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix}\right)$$

You obtain:

$$\{\text{ACOS}(\sqrt{1/2}), 1\}$$

therefore, this isometry is a rotation of $\frac{\pi}{4}$ radians.

4.13.2 Isometry's matrix : MKISOM

In a three-dimensional space, MKISOM has the list of the characteristics elements of an isometry, and either +1 (denoting a direct isometry) or -1 (denoting an indirect isometry) as arguments,

In a two-dimensional space, MKISOM has the characteristic element of an isometry (either an angle or a vector) and either +1 (denoting a direct isometry) or -1 (denoting an indirect isometry) as arguments.

MKISOM returns the matrix representing the given isometry.

For example, type:

$$\text{MKISOM}(\{-1, 2, -1\}, \pi, 1)$$

You simplify the answer typing :

XQ(then ↑ and COPY, then) ENTER

this gives the matrix of a rotation with axis $[-1, 2, -1]$ and angle π :

$$\begin{bmatrix} \frac{-2}{3} & \frac{-2}{3} & \frac{1}{3} \\ \frac{-2}{3} & \frac{1}{3} & \frac{-2}{3} \\ \frac{1}{3} & \frac{-2}{3} & \frac{-2}{3} \end{bmatrix}$$

For example, type:

$$\text{MKISOM}(\{\pi\}, -1)$$

You obtain the matrix of a symmetry with respect to the origin:

$$[[-1, 0, 0], [0, -1, 0], [0, 0, -1]]$$

Type:

$$\text{MKISOM}(\{[1, 1, 1], \frac{\pi}{3}\}, -1)$$

You obtain (after simplification) the matrix of a rotation with axis $[1, 1, 1]$ and angle $\frac{\pi}{3}$ combined with a symmetry with respect to the plane $x+y+z=0$:

$$\begin{bmatrix} 0 & -1 & 0 \\ 0 & 0 & -1 \\ -1 & 0 & 0 \end{bmatrix}$$

Type:

$$\text{MKISOM}\left(\frac{\pi}{2}, 1\right)$$

You obtain the matrix of a rotation of $\frac{\pi}{2}$ radians in two dimensions:

$$[[0, -1], [1, 0]]$$

(because $\text{COS}(1/2 * \pi) = 0$ and $\text{SIN}(1/2 * \pi) = 1$)

Type:

$$\text{MKISOM}([1, 1], -1)$$

You obtain the matrix of a symmetry with respect to $y = x$ in two dimensions:

$$[[0, -1], [-1, 0]]$$

Chapter 5

CAS Reference from the Equation Editor

When you press on CAS in the HOME menu bar, you open the Equation Editor in which you can do symbolic calculus. In this chapter, we describe the computer algebra functions as used from the Equation Editor (see section 3.4 and chapter 4 if you want to use CAS functions from HOME).

5.1 CAS menu bar

The Equation Editor has a menu bar : it is the CAS menu bar. TOOL, ALGB, DIFF, REWRI, SOLV and TRIG compose the CAS menu bar.

Only the TOOL menu contains commands, the other menus contain the algebraic functions that can be written in Alpha mode and have on its first line the sub-menu CFG enable you to change your configuration.

5.1.1 Change your configuration : CFG

All the menus, except TOOL display the state of your configuration and enable you to change it.

For example, say that you see the following on the first line of a menu:

CFG : R = X S

This means that :

- (1) you are in exact-real mode,
- (2) X is the current variable, and
- (3) you are working in Step by step mode (S).

If you highlight CFG and press OK, a menu is displayed with this at its head:

CFG : R = STEP ↑ X 13 ||

This means that :

- (1) you are in exact-real mode,
- (2) Step by step mode is selected,
- (3) polynomials are written with their terms in ascending order by exponent,
- (4) X is the current variable,
- (5) modular calculations are carried out in $\mathbb{Z}/13\mathbb{Z}$ ($p = 13$), and
- (6) you are working in Rigorous mode (that is, using absolute values).

You can change this configuration by selecting any of the following:

Quit `cfg` (when you're finished making changes)

Complex (or Real)

Approx (or Exact)

Direct (or Step/Step if you want to work in Step by Step mode)

$1 + x + x^2 \dots$ (or $\dots x^2 + x + 1$) how polynomials will appear,

Sloppy (or Rigorous, if you want to work in absolute values)

Num. factor (or Symb factor)

Cmplx vars (or Real vars if you want all symbolic variables to be treated as reals cf. 3.3.5)

English (or Français if you want the line-based help to be in French)

Default `cfg` (configuration R = STEP ↓ X 13 ||).

Press OK to validate each of your choices.

Pressing CANCEL takes you out of the CFG menu (as does choosing `quit cfg` and confirming it with OK).

The name of the current variable contained in VX, as well as the value of the variable MODULO, can be changed by means of the SHIFT SYMB (SETUP) keystroke, or by using the VARS key (see 3.3.2 and 3.3.2).

REMARKS :

- You can also change your configuration with the SHIFT SYMB (SETUP) (or SHIFT HOME (MODES)) keystrokes pressed in the Equation Editor (see section 3.3.2).

- In CAS, angles are always expressed in Radians.

When you are in HOME screen, you can use the MODES menu (the SHIFT HOME keystrokes) to open the HOME MODES screen and to change ANGLE

MEASURE or open the CAS MODES screen with the SHIFT F6 (CAS on the menu bar) to change your CAS configuration.

5.1.2 Step/Step Mode

You choose Step by Step mode (Step/Step, abbreviated S) when you want to see the details of the calculations.

The details of the calculations are displayed on the screen and you can view the next step by pressing OK.

When the screen is not big enough to display all the information, directional arrows ▼ and ▲ appear on the edge of the screen. You can then scroll the screen to see more information by using the arrow keys ▽ △.

If you do not need to see the details of the calculations, choose Direct mode (abbreviated D).

5.1.3 TOOL menu

The functions contained in the TOOL menu are described in section 3.3.1.

Cursor mode

Edit expr.

Change font

Cut

Copy

Paste

5.1.4 ALGB menu

COLLECT

DEF

EXPAND

FACTOR

PARTFRAC

QUOTE

STORE

|

SUBST

TEXPAND

UNASSIGN

5.1.5 DIFF menu

DERIV
DERVX
DIVPC
FOURIER
IBP
INTVX
lim
PREVAL
RISCH
SERIES
TABVAR
TAYLOR0
TRUNC

5.1.6 REWRI menu

DISTRIB
EPSX0
EXPLN
EXP2POW
FDISTRIB
LIN
LNCOLLECT
POWEXPAND
SINCOS
SIMPLIFY
XNUM
XQ

5.1.7 SOLV menu

DESOLVE
ISOLATE
LDEC
LINSOLVE
SOLVE
SOLVEVX

5.1.8 TRIG menu

ACOS2S
 ASIN2C
 ASIN2T
 ATAN2S
 HALFTAN
 SINCOS
 TAN2CS2
 TAN2SC
 TAN2SC2
 TCOLLECT
 TEXPAND
 TLIN
 TRIG
 TRIGCOS
 TRIGSIN
 TRIGTAN

5.1.9 The MATH key

It contains, in addition to the previously described menus (Algebra, Diff&Int, Rewrite, Trig., Solve) we have :

Complex (i ABS ARG CONJ DROITE IM - RE SIGN)
 Constant (e i $\infty\pi$)
 Hyperb. (ACOSH ASINH ATANH COSH SINH TANH)
 Integer (DIVIS EULER FACTOR GCD IDIV2 IEGCD IQUOT IREMAINDER ISPRIME? LCM MOD NEXTPRIME PREVPRIME)
 Modular (ADDTMOD DIVMOD EXPANMOD FACTORMOD GCDMOD INVMOD MODSTO MULTMOD POWMOD SUBTMOD)
 Polynom. (EGCD FACTOR GCD HERMITE LCM LEGENDRE PARTFRAC PROPFRAC PTAYL QUOT REMAINDER TCHEBYCHEFF)
 Reals (CEILING FLOOR FRAC INT MAX MIN)
 Tests (ASSUME UNASSUME > \geq < \leq == \neq AND OR NOT IFTE)
 Refer to sections [3.3.1](#) and [3.3.2](#) for descriptions of these menus.

5.2 Integers (and Gaussian Integers)

All the functions in this section are found in the MATH key's Integer menu, except IABCUV ICHINREM PA2B2, which is in the SHIFT MATH

(CMDS) key's menu.

For certain functions, you can use Gaussian integers (numbers of the form $a + ib$, where a and b are integers) in the place of integers.

5.2.1 Unlimited precision

The calculator can manage integers with unlimited precision, such as the following:

$$100!$$

The symbol ! is obtained by typing SHIFT ×

The decimal value of 100! is very large, but you can view it by using the VIEWS key.

5.2.2 The divisors of a number : DIVIS

DIVIS gives the divisors of a number.

Typing :

$$\text{DIVIS}(12)$$

gives:

$$12 \text{ OR } 6 \text{ OR } 3 \text{ OR } 4 \text{ OR } 2 \text{ OR } 1$$

NOTE : $\text{DIVIS}(0) = 0 \text{ OR } 1$

5.2.3 The Euler index : EULER

EULER returns the Euler index for a whole number.

$\text{EULER}(n)$ is equal to the number of whole numbers less than n and prime with n .

Typing:

$$\text{EULER}(21)$$

gives:

$$12$$

In other words:

$E = \{2, 4, 5, 7, 8, 10, 11, 13, 15, 16, 17, 19\}$ is the set of whole numbers less than 21 and prime with 21. There are 12 members of the set, so $\text{Cardinal}(E) = 12$.

5.2.4 Decomposition into prime factors : FACTOR

FACTOR decomposes an integer into its prime factors.

Typing:

$$\text{FACTOR}(90)$$

gives:

$$2 \cdot 3^2 \cdot 5$$
5.2.5 The greatest common divisor : GCD

GCD returns the greatest common divisor of two integers.

Typing:

$$\text{GCD}(18, 15)$$

gives:

$$3$$

In Step by step mode, typing:

$$\text{GCD}(78, 24)$$

gives:

$$78 \bmod 24 = 6$$

$$24 \bmod 6 = 0$$

Result 6

Pressing ENTER then causes 6 to be written to the Equation Editor.

5.2.6 Resolution of $au + bv = c$: IABCUV

IABCUV(A, B, C) returns U AND V so that AU+BV=C.

It is of course necessary that C must be a multiple of GCD(A, B) to obtain a solution.

Typing :

$$\text{IABCUV}(48, 30, 18)$$

gives :

$$6 \text{ AND } -9$$

5.2.7 Chinese remainders : ICHINREM

ICHINREM(A AND P , B AND Q) returns C AND R.

The numbers $X = C + k \cdot R$ where $k \in \mathbb{Z}$ are such that :

$$X = A \pmod{P} \text{ and } X = B \pmod{Q}.$$

A solution X always exists when P and Q are mutually prime, ($\text{GCD}(P, Q) = 1$) and in this case, all the solutions are congruent modulo $R = P \cdot Q$.

Typing :

$$\text{ICHINREM}(7 \text{ AND } 10, 12 \text{ AND } 15)$$

gives :

$$-3 \text{ AND } 30$$

that is to say that the solutions are such that :

$$X = -3 \pmod{30}$$

EXAMPLE :

Gives the solutions of :

$$\begin{cases} X = 3 \pmod{5} \\ X = 9 \pmod{13} \end{cases}$$

Typing :

$$\text{ICHINREM}(3 \text{ AND } 5, 9 \text{ AND } 13)$$

gives :

$$-147 \text{ AND } 65$$

that is to say that the solutions are such that :

$$X = -147 \pmod{65}$$

5.2.8 Quotient and remainder of two integers : IDIV2

IDIV2 returns the quotient AND the remainder of the Euclidean division between two integers.

Typing:

$$\text{IDIV2}(148, 5)$$

gives:

$$29 \text{ AND } 3$$

In step-by-step mode, the calculator shows the division process like it is taught at school. The division is carried out as if in longhand, using the so-called "gallows" algorithm.

5.2.9 Bézout's Identity : IEGCD

IEGCD(A,B) returns the value of Bézout's Identity for two integers.

In other words, IEGCD(A,B) returns U AND V = D, with U, V, D such that:

AU+BV=D and D=GCD(A,B).

Typing:

$$\text{IEGCD}(48, 30)$$

gives

$$2 \text{ AND } -3 = 6$$

In other words: $2 \cdot 48 + (-3) \cdot 30 = 6$ and $\text{gcd}(48,30)=6$

In Step by step mode, we get:

[z,u,v]:z=u*48+v*30

[48,1,0]

[30,0,1]*-1

[18,1,-1]*-1

[12,-1,2]*-1

[6,2,-3]*-2

Result : [6,2,-3]

puis ENTER ou OK,

$$2 \text{ AND } -3 = 6$$

to be written to the Equation Editor.

5.2.10 The integer Euclidean quotient : IQUOT

IQUOT returns the integer quotient of the Euclidean division of two integers.

Typing:

$$\text{IQUOT}(148, 5)$$

gives:

$$29$$

In Step by step mode, the division is carried out as if in longhand:

$$\begin{array}{r|l} 148 & 5 \\ 48 & \text{---} \\ 3 & 29 \end{array}$$

Press OK to execute the division step by step, then press ENTER to write the result 29 to the Equation Editor.

5.2.11 The integer remainder : IREMAINDER

IREMAINDER returns the integer remainder from the Euclidean division of two integers.

IREMAINDER is found in the MATH key's Integer menu.

Typing:

$$\text{IREMAINDER}(148, 5)$$

gives:

$$3$$

IREMAINDER works with integers or with Gaussian integers, which is what distinguishes it from MOD.

EXAMPLE

$$\text{IREMAINDER}(2 + 3 \cdot i, 1 + i)$$

returns i

Try calculating:

$$\text{IREMAINDER}(148!, 5! + 2)$$

(The symbol ! is obtained by typing SHIFT ×).

In Step by step mode, the division is carried out as if in longhand, using the so-called "gallows" algorithm (see 5.2.10 for an example).

5.2.12 Test of pseudo-primality : ISPRIME?

ISPRIME?(N) returns 1. (TRUE) if N is a pseudo-prime, and 0. (FALSE) if N is not prime.

DEFINITION: For numbers less than 10^{14} , pseudo-prime and prime mean the same thing. But for numbers greater than 10^{14} , a pseudo-prime is a number with a large probability of being prime (cf. Rabin's Algorithm, section 7.6).

Typing:

$$\text{ISPRIME?}(13)$$

gives:

$$1.$$

Typing:

$$\text{ISPRIME?}(14)$$

gives:

$$0.$$

5.2.13 The least common multiple : LCM

LCM returns the least common multiple of two integers.

Typing:

$$\text{LCM}(18, 15)$$

gives:

$$90$$
5.2.14 Remainder of the Euclidean division : MOD

MOD is an infix function that has two integers as arguments.

MOD returns the remainder of the Euclidean division of the arguments.

Typing:

$$3 \text{ MOD } 2$$

produces the result:

$$1$$

MOD accepts real numbers ($7.5 \text{ mod } 2.1 = 1.2$), but not Gaussian integers (cf section 5.5.3).

5.2.15 The smallest pseudo-prime greater than N : NEXTPRIME

NEXTPRIME (N) returns the smallest pseudo-prime greater than N.

Typing:

$$\text{NEXTPRIME}(75)$$

gives:

$$79$$
5.2.16 Resolution of $a^2 + b^2 = p$ with integers: PA2B2

PA2B2 decompose a prime integer p congruent to 1 modulo 4, as follows:

$p = a^2 + b^2$.

The calculator gives the result as $a + b \cdot i$

Typing :

$$\text{PA2B2}(17)$$

gives :

$$4 + i$$

that is, $17 = 4^2 + 1^2$

Typing :

PA2B2(29)

gives :

$5 + 2 \cdot i$

that is, $29 = 5^2 + 2^2$

5.2.17 The greatest pseudo-prime less than N : PREVPRIME

PREVPRIME(N) returns the greatest pseudo-prime less than N.

Typing:

PREVPRIME(75)

gives:

73

5.3 Modular Calculations

All the functions in this section are found in the MATH key's Modular menu except DIV2MOD, which is in the SHIFT MATH (CMDS) key's menu.

You can carry out calculations in modulo p that is, in $\mathbb{Z}/p\mathbb{Z}$ or in $\mathbb{Z}/p\mathbb{Z}[X]$.

NOTE: For some commands, p must be prime.

ALL THE EXAMPLES IN THIS SECTION USE A VALUE FOR p OF 13.

It's assumed that you have already typed:

MODSTO(13)

or that you have used the SHIFT SYMB (SETUP) keystroke to switch MODULO to 13. The representation is the symmetric representation (-5 instead of 8 modulo 13).

5.3.1 Addition in $\mathbb{Z}/p\mathbb{Z}$ or in $\mathbb{Z}/p\mathbb{Z}[X]$: ADDTMOD

ADDTMOD performs an addition in $\mathbb{Z}/p\mathbb{Z}$.

Typing:

ADDTMOD(2, 18)

gives:

-6

ADDTMOD performs an addition in $Z/pZ[X]$.

Typing:

$$\text{ADDTMOD}(11X + 5, 8X + 6)$$

gives:

$$6X - 2$$

5.3.2 Quotient and remainder in $Z/pZ[X]$: DIV2MOD

The arguments of DIV2MOD are two polynomials $A[X]$ and $B[X]$. The result is a AND containing both the quotient and the remainder of the euclidean division of $A[X]$ by $B[X]$ in $Z/pZ[X]$.

Typing:

$$\text{DIV2MOD}(X^3 + X^2 + 1, 2 \cdot X^2 + 4)$$

you obtain in $Z/13Z$:

$$-(6X + 6) \text{ AND } -(2X + 1)$$

since $X^3 + X^2 + 1 = (2 \cdot X^2 + 4) \cdot \frac{X+1}{2} + \frac{5 \cdot X - 4}{4}$

$2 * (-6) = 1 \pmod{13}$ and

$4 * (-3) = 1 \pmod{13}$

5.3.3 Division in Z/pZ or in $Z/pZ[X]$: DIVMOD

The arguments are two integers A and B . When B has an inverse in Z/pZ , the result is $\frac{A}{B}$ simplified as Z/pZ , else there is an error message.

Typing:

$$\text{DIVMOD}(5, 3)$$

gives:

$$6$$

The arguments are two polynomials $A[X]$ and $B[X]$. The result is a rationale fraction $\frac{A[X]}{B[X]}$ simplified as $Z/pZ[X]$.

Typing:

$$\text{DIVMOD}(2X^2 + 5, 5X^2 + 2X - 3)$$

gives:

$$\frac{4X + 5}{3X + 3}$$

5.3.4 To expand and simplify expressions in Z/pZ or in $Z/pZ[X]$: EXPANDMOD

EXPANDMOD has as an argument a integer expression.

EXPANDMOD expands this expression in Z/pZ .

Typing:

$$\text{EXPANDMOD}(2 * 3 + 5 * 4)$$

gives:

$$0$$

EXPANDMOD has as an argument a polynomial expression.

EXPANDMOD expands this expression in $Z/pZ[X]$.

Typing:

$$\text{EXPANDMOD}((2X^2 + 12).(5X - 4))$$

gives:

$$-(3X^3 - 5X^2 + 5X - 4)$$

5.3.5 To factor polynomials in $Z/pZ[X]$: FACTORMOD

FACTORMOD has as an argument a polynomial.

FACTORMOD factors this polynomial in $Z/pZ[X]$, providing that $p \leq 97$ and p is prime.

Typing:

$$\text{FACTORMOD}(-(3X^3 - 5X^2 + 5X - 4))$$

gives:

$$-((3X - 5)(X^2 + 6))$$

5.3.6 GCD of the two polynomials in $Z/pZ[X]$: GCDMOD

GCDMOD has two polynomials as arguments.

GCDMOD calculates the GCD of the two polynomials in $Z/pZ[X]$.

Typing:

$$\text{GCDMOD}(2X^2 + 5, 5X^2 + 2X - 3)$$

gives:

$$-(6X - 1)$$

5.3.7 Inverse in $\mathbf{Z}/p\mathbf{Z}$: INVMOD

INVMOD has as an argument an integer.

INVMOD calculates the inverse of the integer in $\mathbf{Z}/p\mathbf{Z}$.

Typing:

$$\text{INVMOD}(5)$$

gives (since $5 \times -5 = -25 = 1 \pmod{13}$):

$$-5$$
5.3.8 To put a value in the MODULO variable : MODSTO

You use the MODSTO command to set the value of the MODULO variable p .

THE EXAMPLES IN THIS SECTION ALL USE A VALUE FOR p OF 13 that is, they assume that you have already typed:

$$\text{MODSTO}(13)$$
5.3.9 Multiplication in $\mathbf{Z}/p\mathbf{Z}$ or $\mathbf{Z}/p\mathbf{Z}[X]$: MULTMOD

MULTMOD performs a multiplication in $\mathbf{Z}/p\mathbf{Z}$.

Typing:

$$\text{MULTMOD}(11, 8)$$

gives:

$$-3$$

MULTMOD performs a multiplication in $\mathbf{Z}/p\mathbf{Z}[X]$.

Typing:

$$\text{MULTMOD}(11X + 5, 8X + 6)$$

gives:

$$-(3X^2 - 2X - 4)$$
5.3.10 Power in $\mathbf{Z}/p\mathbf{Z}$ or in $\mathbf{Z}/p\mathbf{Z}[X]$: POWMOD

POWMOD(A, N) calculates A to the power of N in $\mathbf{Z}/p\mathbf{Z}[X]$, and POWMOD($A(X), N$)

calculates $A(X)$ to the power of N in $\mathbf{Z}/p\mathbf{Z}[X]$.

The MODULO variable p must be a prime number less than 100.

Typing:

$$\text{POWMOD}(11, 195)$$

gives if $p = 13$:

$$5$$

In effect: $11^{12} = 1 \pmod{13}$ so $11^{195} = 11^3 = 5 \pmod{13}$

Typing:

$$\text{POWMOD}(2X + 1, 5)$$

gives:

$$6 \cdot X^5 + 2 \cdot X^4 + 2 \cdot X^3 + X^2 - 3 \cdot X + 1$$

since:

$$10 = -3 \pmod{13} \quad 40 = 1 \pmod{13} \quad 80 = 2 \pmod{13}$$

$$32 = 6 \pmod{13}.$$

5.3.11 Subtraction in Z/pZ or in $Z/pZ[X]$: SUBTMOD

SUBTMOD performs a subtraction in Z/pZ .

Typing:

$$\text{SUBTMOD}(29, 8)$$

gives:

$$-5$$

SUBTMOD performs a subtraction in $Z/pZ[X]$.

Typing:

$$\text{SUBTMOD}(11X + 5, 8X + 6)$$

gives:

$$3X - 1$$

5.4 Rational Numbers

Calculate:

$$\frac{123}{12} + \frac{57}{21}$$

After you press ENTER, you get the result:

$$\frac{363}{28}$$

If you now invoke the XNUM function on the REWRITE menu, or if you press the NUM key, you get the following result:

$$12.9642857143$$

If you mix the two representations, for example:

$$\frac{1}{2} + 0.5$$

the calculator asks to go into approx mode to do the calculation. After responding yes, you get:

$$1.$$

You'll then need to go back into exact mode (CFG, etc... or by typing SHIFT NUM).

5.4.1 The n -th Bernoulli's number : IBERNOULLI

IBERNOULLI has as an integer n as argument.

IBERNOULLI returns the n^{th} Bernoulli's number $B(n)$.

We have :

$$\frac{t}{e^t - 1} = \sum_{n=0}^{+\infty} \frac{B(n)}{n!} t^n$$

Remember that the Bernoulli's polynomials B_k are defined as :

$$B_0 = 1$$

$$B_k'(x) = kB_{k-1}(x)$$

$$\int_0^1 B_k(x) dx = 0$$

Then, the Bernoulli's numbers are defined as:

$$B(n) = B_n(0)$$

Typing :

$$\text{IBERNOULLI}(6)$$

gives :

$$\frac{1}{42}$$

5.4.2 Whole-number and fractionnal part : PROPFRAC

PROPFRAC is found in the MATH key's POLYNOMIAL menu.

PROPFRAC($\frac{A}{B}$) writes the fraction $\frac{A}{B}$ in the form:

$$Q + \frac{R}{B} \text{ avec } 0 \leq R < B$$

Typing:

$$\text{PROPFrac}\left(\frac{43}{12}\right)$$

gives:

$$3 + \frac{7}{12}$$

5.4.3 Symbolic summation : \sum

To calculate symbolic summations of rational fractions and hypergeometric series that allow a discrete primitive, you can use the symbol \sum which can be produced by typing : SHIFT + (see also 3.3.4).

Example:

If you type :

$$\sum_{K=1}^{\infty} \frac{1}{K \cdot (K + 1)}$$

then select the entire expression and press ENTER, you obtain:

$$1$$

If you type:

$$\sum_{K=1}^5 \frac{1}{K \cdot (K + 1)}$$

then select the entire expression and press ENTER, you obtain:

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

5.5 Real Numbers

Calculate:

$$\text{EXP}(\pi \times \sqrt{20})$$

When you press the ENTER key, the response is:

$$\text{EXP}(2 \times \sqrt{5} \times \pi)$$

If you then invoke the XNUM function on the REWRITE menu or press the NUM key, the result is:

$$1263794.7537$$

In the MATH key's Real menu, you'll find the FLOOR function and in the MATH key's Integer menu, you'll find the MOD function.

5.5.1 The whole part : FLOOR

FLOOR has as an argument a real number, and returns its whole part.

Typing:

$$\text{FLOOR}(3.53)$$

gives:

$$3$$
5.5.2 The Γ function : GAMMA

GAMMA returns the values of the Γ function at the given point.

The Γ function is defined as:

$$\Gamma(x) = \int_0^{+\infty} e^{-t} t^{x-1} dt$$

We have :

$$\Gamma(1) = 1$$

$$\Gamma(x + 1) = x \cdot \Gamma(x)$$

Typing :

$$\text{GAMMA}(5)$$

gives :

$$24$$

Typing :

$$\text{GAMMA}\left(\frac{1}{2}\right)$$

gives :

$$\sqrt{\pi}$$
5.5.3 Remainder of the Euclidean division : MOD

MOD is an infix function that has two integers or reals as arguments a and b . MOD returns the remainder of the Euclidean division of the arguments a and b or the real r such that $a = b * q + r$ with q integer and $0 \leq r < b$ or $b < r \leq 0$.

Typing:

$$3 \text{ MOD } 2$$

produces the result:

$$1$$

Typing :

$$3.2 \text{ MOD } 2.1$$

produces the result :

$$1.1$$

because we have $3.2 = 2.1 + 1.1$ and $0 \leq 1.1 < 2.1$

Typing:

$$3.2 \text{ MOD } -2.1$$

produces the result:

$$-1.$$

because we have $3.2 = (-2.1) * (-2) + 1.$ and $-2.1 < -1. \leq 0$

Typing :

$$7.5 \text{ MOD } 2.15$$

produces the result :

$$1.05$$

because we have $7.5 = 2.15 * 3 + 1.05$ and $0 \leq 1.05 < 2.15$

5.5.4 The derivatives of the Digamma function :PSI

PSI has a number a and an integer n as arguments .

PSI returns the value of the n -th derivative of the Digamma function at a .

The function Digamma is the derivative of $\ln(\Gamma(x))$.

Typing :

$$\text{PSI}(3, 1)$$

gives :

$$-\frac{5}{4} + \frac{1}{6} \cdot \pi^2$$

5.5.5 The Digamma function : Psi

Psi has as argument a number a .

Psi returns the value of the Digamma function at a .

The Digamma function is defined as the derivative of $\ln(\Gamma(x))$, so we have

:

$$\text{PSI}(a, 0) = \text{Psi}(a).$$

Typing :

$$\text{Psi}(3)$$

gives (press on NUM) :

$$.922784335098$$

5.6 Complex Numbers

NOTE : Complex numbers of the form $a + b \cdot i$, where a and b are real numbers, can be notated (a, b) or $a + b \cdot i$.

The operators $+$, $-$, $*$, $/$, $^$ can be used for complex numbers.

Type:

$$(1 + 2 \cdot i)^2$$

Then press ENTER.

If you are not in complex mode, the calculator asks to change modes. After responding YES, you get:

$$-(3 - 4 \cdot i)$$

It's worth noting that this expression will not be simplified further: in exact mode, the result is always notated such that the real part of the complex number is positive.

In the MATH key's Complex menu you will find the following functions, all of which have complex numbers as parameters:

- ABS to determine the modulus of the parameter,
- ARG to determine the argument of the parameter,
- CONJ to determine the conjugate of the parameter,
- DROITE takes two complex numbers z_1 , z_2 as parameters, and returns the equation of the line through the Cartesian points, z_1 , z_2 .
- IM to determine the imaginary part of the parameter,
- to determine the opposite of the parameter,
- RE to determine the real part of the parameter,
- SIGN to determine the quotient of the parameter divided by its modulus.

5.6.1 Argument of a complex number : ARG

Typing:

$$\text{ARG}(3 + 4 \cdot i)$$

produces (given that in CAS, you're working in Radians):

$$\text{ATAN}\left(\frac{4}{3}\right)$$

REMARK:

You can do the same calculation in HOME, but you get a numeric result (0.64250 . . ., if you're working in Radians)

In HOME, typing:

$$\text{ARG}(\text{XQ}(3 + 4 \cdot i))$$

produces:

$$\text{ATAN}\left(\frac{4}{3}\right)$$

5.6.2 Conjugate of a complex number : CONJ

Typing:

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

produces the result:

$$1 - 2 \cdot i$$

NOTE: If you chose `Real vars` from the CFG configuration menu, then $\text{CONJ}(Z)=Z$; if you chose `Cmplx vars`, $\text{CONJ}(Z)$ will be different from Z as long as Z is not in the list that contains the variable `REALASSUME`. It is often preferable to write the expression as a quoted expression: $\text{QUOTE}(\text{expression})$, to avoid having to rewrite it. For example, if you select `Real vars` and then type:

$$\text{SUBST}(\text{QUOTE}(\text{CONJ}(Z)), Z = 1 + i)$$

you get:

$$\text{CONJ}(1 + i)$$

whereas:

$$\text{SUBST}(\text{CONJ}(Z), Z = 1 + i)$$

produces:

$$1 + i$$

Of course, if you have selected `Cmplx vars`, and Z is not in the list that contains the variable `REALASSUME`, then typing:

$$\text{SUBST}(\text{CONJ}(Z), Z = 1 + i)$$

gives:

$$\text{CONJ}(1 + i)$$

5.6.3 Equation of the line : DROITE

`DROITE` takes two complex numbers z_1, z_2 as parameters.

`DROITE` returns the equation of the line through the Cartesian points, z_1, z_2 .

Typing:

$$\text{DROITE}((1, 2), (0, 1))$$

or:

$$\text{DROITE}(1 + 2 \cdot i, i)$$

returns:

$$Y = X - 1 + 2$$

Then, ENTER produces:

$$Y = X + 1$$

5.7 Algebraic Expressions

All functions in this section can be found in the ALGB menu on the menu bar.

5.7.1 Factors over the integers : COLLECT

COLLECT has an expression as a parameter.

COLLECT combines like terms, and factors the expression over the integers.

Examples :

To factor over the integers:

$$x^2 - 4$$

type:

$$\text{COLLECT}(X^2 - 4)$$

which gives in real mode :

$$(X + 2) \cdot (X - 2)$$

To factor over the integers:

$$x^2 - 2$$

type:

$$\text{COLLECT}(X^2 - 2)$$

which gives:

$$X^2 - 2$$

5.7.2 Distributivity : EXPAND

EXPAND has an expression as a parameter.

EXPAND expands and simplifies this expression.

Typing:

$$\text{EXPAND}((X^2 + \sqrt{2} \cdot X + 1) \cdot (X^2 - \sqrt{2} \cdot X + 1))$$

gives:

$$X^4 + 1$$

5.7.3 Factorization : FACTOR

FACTOR has an expression as a parameter.

FACTOR factors this expression.

Example:

To factor :

$$x^4 + 1$$

Key in:

$$\text{FACTOR}(X^4 + 1)$$

FACTOR can be found in the ALGB menu.

In real mode, the result is:

$$(X^2 + \sqrt{2} \cdot X + 1) \cdot (X^2 - \sqrt{2} \cdot X + 1)$$

In complex mode (using CFG), the result is:

$$\frac{(2 \cdot X + (1 + i) \cdot \sqrt{2}) \cdot (2 \cdot X - (1 + i) \cdot \sqrt{2}) \cdot (2 \cdot X + (1 - i) \cdot \sqrt{2}) \cdot (2 \cdot X - (1 - i) \cdot \sqrt{2})}{16}$$

5.7.4 Change the order of variables : REORDER

REORDER has two arguments: an expression and the names of variables separated with AND.

REORDER reorders the input expression following the order of variables given by its second argument.

Typing:

$$\text{REORDER}(X^2 + 2 \cdot X \cdot A + A^2 + Z^2 - X \cdot Z, A \text{ AND } X \text{ AND } Z)$$

gives:

$$A^2 + 2 \cdot X \cdot A + X^2 - Z \cdot X + Z^2$$

5.7.5 Partial simplifications : SEVAL

SEVAL has an expression as argument.

SEVAL simplifies the expression, operating on all but the top-level operator of the expression.

Typing :

$$\text{SEVAL}(\text{SIN}(3 \cdot X - X) + \text{SIN}(X + X))$$

gives :

$$\text{SIN}(2 \cdot X) + \text{SIN}(2 \cdot X)$$

5.7.6 Substitute a value for a variable : |

| is an infix operator used to substitute a value for a variable in an expression (a bit like the following function SUBST).

| has two parameters: an expression dependent on a parameter, and an equality (parameter=substitute value).

| substitutes the specified value for the variable in the expression.

Typing:

$$X^2 - 1 |_{X = 2}$$

gives:

$$2^2 - 1$$

5.7.7 Substitute a value for a variable : SUBST

SUBST has two parameters: an expression dependent on a parameter, and an equality (parameter=substitute value).

SUBST substitutes the specified value for the variable in the expression.

Typing:

$$\text{SUBST}(A^2 + 1, A = 2)$$

gives:

$$2^2 + 1$$

5.8 Polynomials

All functions (except DEGREE) in this section can be found in the MATH key's Polynom. menu.

5.8.1 Resolution of $A[X]U[X] + B[X]V[X] = C[X] : ABCUV$

This command applies the Bézout identity like EGCD but, now, the arguments are three polynomials, A , B , C (C must be a multiple of $\text{GCD}(A, B)$): $ABCUV(A[X], B[X], C[X])$ returns $U[X]$ AND $V[X]$, where U, V satisfy the following:

$$C[X] = U[X] * A[X] + V[X] * B[X]$$

Typing:

$$ABCUV(X^2 + 2 \cdot X + 1, X^2 - 1, X + 1)$$

gives:

$$\frac{1}{2} \text{ AND } \frac{-1}{2}$$

Typing:

$$ABCUV(X^2 + 2 \cdot X + 1, X^2 - 1, X^3 + 1)$$

gives:

$$\frac{X^2 - X + 1}{2} \text{ AND } -\frac{X^2 - X + 1}{2}$$

5.8.2 Chinese remainders : CHINREM

CHINREM has two AND as arguments : each AND has two polynomials as components.

$CHINREM(A[X] \text{ AND } R[X], B[X] \text{ AND } Q[X])$ returns an AND with two polynomials as components $P[X]$ and $S[X]$. This polynomials $P[X]$ and $S[X]$ satisfying the following relations when $\text{GCD}(R[X], Q[X]) = 1$:

$$S[X] = R[X] \cdot Q[X],$$

$$P[X] = A[X](\text{mod}R[X]) \text{ and } P[X] = B[X](\text{mod}Q[X]).$$

There is always a solution $P[X]$, if $R[X]$ and $Q[X]$ are mutually primes, and all solutions are congruent modulo $S[X] = R[X] \cdot Q[X]$.

Find the solutions $P[X]$ of:

$$\begin{cases} P[X] = X \pmod{X^2 + 1} \\ P[X] = X - 1 \pmod{X^2 - 1} \end{cases}$$

Typing:

$$CHINREM(X \text{ AND } X^2 + 1, X - 1 \text{ AND } X^2 - 1)$$

gives:

$$-\frac{X^2 - 2 \cdot X + 1}{2} \text{ AND } -\frac{X^4 - 1}{2}$$

that is, $P[X] = -\frac{X^2 - 2 \cdot X + 1}{2} \pmod{-\frac{X^4 - 1}{2}}$

5.8.3 Cyclotomic polynomial : CYCLOTOMIC

CYCLOTOMIC has an integer n as argument.

CYCLOTOMIC returns the cyclotomic polynomial of order n . This is a polynomial having the n -th primitive roots of the unity as zeros.

For example, when $n = 4$ the fourth roots of the unity are: $\{1, i, -1, -i\}$; among them, the primitive roots are: $\{i, -i\}$.

Therefore, the cyclotomic polynomial of order 4 is $(X - i).(X + i) = X^2 + 1$.

Another example, typing:

```
CYCLOTOMIC(20)
```

gives :

$$X^8 - X^6 + X^4 - X^2 + 1$$

5.8.4 degree of a polynomial : DEGREE

DEGREE has as an argument a polynomial in the current variable.

DEGREE returns the degree of this polynomial.

NOTE: The degree of a null polynomial is -1 .

The DEGREE command must be entered using the ALPHA keys.

Typing:

```
DEGREE(X2 + X + 1)
```

returns:

2

5.8.5 Quotient and remainder : DIV2

The arguments of DIV2 are two polynomials $A[X]$ and $B[X]$.

DIV2 returns the quotient AND the remainder of the euclidean division between the two polynomials $A[X]$ and $B[X]$.

Typing:

```
DIV2(X2 + 2 · X + 1, X)
```

gives:

X + 2 AND 1

The step-by-step mode can be of interest here, because it displays the intermediate steps of the division process.

5.8.6 Bézout's Identity : EGCD

This function returns Bézout's Identity, the EGCD (Extended Greatest Common Divisor). In other words, $\text{EGCD}(A(X), B(X))$ returns $U(X)$ AND $V(X) = D(X)$, with D, U, V such that:

$$D(X) = U(X) \cdot A(X) + V(X) \cdot B(X)$$

Typing:

$$\text{EGCD}(X^2 + 2 \cdot X + 1, X^2 - 1)$$

gives:

$$1 \text{ AND } -1 = 2 \cdot X + 2$$

Typing:

$$\text{EGCD}(X^2 + 2 \cdot X + 1, X^3 + 1)$$

gives:

$$-(X - 2) \text{ AND } 1 = 3 \cdot X + 3$$

5.8.7 Factorize a polynomial : FACTOR

FACTOR has a polynomial as an argument.

FACTOR factors this polynomial.

Typing:

$$\text{FACTOR}(X^2 - 2)$$

gives:

$$(X + \sqrt{2}) \cdot (X - \sqrt{2})$$

Typing:

$$\text{FACTOR}(X^2 + 2 \cdot X + 1)$$

gives:

$$(X + 1)^2$$

Typing:

$$\text{FACTOR}(X^4 - 2 \cdot X^2 + 1)$$

gives:

$$(X - 1)^2 \cdot (X + 1)^2$$

Typing:

$$\text{FACTOR}(X^3 - 2 \cdot X^2 + 1)$$

gives:

$$\frac{(X - 1) \cdot (2 \cdot X - 1 + \sqrt{5}) \cdot (2 \cdot X - (1 + \sqrt{5}))}{4}$$

5.8.8 GCD of two polynomials : GCD

GCD returns the GCD (Greatest Common Divisor) of two polynomials.

Typing:

$$\text{GCD}(X^2 + 2 \cdot X + 1, X^2 - 1)$$

gives:

$$X + 1$$

5.8.9 Hermite polynomial : HERMITE

HERMITE has as an argument a whole number n .

HERMITE returns the Hermite polynomial of degree n . This is a polynomial of the following type:

$$H_n(x) = (-1)^n \cdot e^{\frac{x^2}{2}} \frac{d^n}{dx^n} e^{-\frac{x^2}{2}}$$

That means that :

for $n \geq 0$

$$H_n''(x) - xH_n'(x) + nH_n(x) = 0$$

and for $n \geq 1$

$$H_{n+1}(x) - xH_n(x) + nH_{n-1}(x) = 0$$

$$H_n'(x) = nH_{n-1}(x)$$

Typing:

$$\text{HERMITE}(6)$$

produces the result:

$$64 \cdot X^6 - 480 \cdot X^4 + 720 \cdot X^2 - 120$$

5.8.10 To evaluate a polynomial : HORNER

HORNER has two arguments : a polynomial $P[X]$ and a number a .

HORNER($P[X]$, a) returns the polynomial $Q[X]$ (quotient of $P[X]$ divided by $X-a$) AND the value $P[a]$.

Typing :

$$\text{HORNER}(X^3 + 1, 2)$$

gives :

$$X^2 + 2 \cdot X + 4 \text{ AND } 9$$

Typing :

$$\text{HORNER}(X^4 + 2 \cdot X^3 - 3 \cdot X^2 + X - 2, 1)$$

gives :

$$X^3 + 3 \cdot X^2 + 1 \text{ AND } -1$$

5.8.11 LCM of two polynomials : LCM

LCM returns the LCM (Least Common Multiple) of two polynomials.

Typing:

$$\text{LCM}(X^2 + 2 \cdot X + 1, X^2 - 1)$$

gives:

$$(X^2 + 2 \cdot X + 1) \cdot (X - 1)$$

5.8.12 Legendre polynomial : LEGENDRE

LEGENDRE has as an argument a whole number n .

LEGENDRE returns the polynomial L_n , a non-null solution of the differential equation:

$$(x^2 - 1) \cdot y'' - 2 \cdot x \cdot y' - n(n + 1) \cdot y = 0$$

For $n \geq 0$, we have the Rodriguès Formula:

$$L_n(x) = \frac{1}{n!2^n} \frac{d^n}{dx^n} (x^2 - 1)^n$$

and for $n \geq 1$, we have:

$$(n + 1)L_{n+1}(x) = (2n + 1)xL_n(x) - nL_{n-1}(x)$$

Typing:

$$\text{LEGENDRE}(4)$$

gives:

$$\frac{35 \cdot X^4 - 30 \cdot X^2 + 3}{8}$$

5.8.13 Partial fraction expansion : PARTFRAC

PARTFRAC has as an argument a rational fraction.

PARTFRAC returns the partial fraction decomposition of this rational fraction.

Example :

To perform a partial fraction decomposition of a rational function, for example:

$$\frac{x^5 - 2 \times x^3 + 1}{x^4 - 2 \times x^3 + 2 \times x^2 - 2 \times x + 1}$$

you use the PARTFRAC command.

Type:

$$\text{PARTFRAC}\left(\frac{X^5 - 2 * X^3 + 1}{X^4 - 2 * X^3 + 2 * X^2 - 2 * X + 1}\right)$$

In real and direct mode, this produces:

$$X + 2 + \frac{X - 3}{2 \cdot X^2 + 2} + \frac{-1}{2 \cdot X - 2}$$

In complex mode, this produces:

$$X + 2 + \frac{1-3 \cdot i}{X + i} + \frac{-1}{X - 1} + \frac{1+3 \cdot i}{X - i}$$

5.8.14 Whole-number and fractional part : PROPFRAC

PROPFRAC has as an argument a rational fraction.

PROPFRAC rewrites this rational fraction so as to bring out its whole-number part.

PROPFRAC(A(X) / B(X)) writes the rational fraction $\frac{A[X]}{B[X]}$ in the form:

$$Q[X] + \frac{R[X]}{B[X]}$$

where $R[X] = 0$, or $0 \leq \text{deg}(R[X]) < \text{deg}(B[X])$.

Typing:

$$\text{PROPFRAC}\left(\frac{(5 \cdot X + 3) \cdot (X - 1)}{X + 2}\right)$$

gives:

$$5 \cdot X - 12 + \frac{21}{X + 2}$$

5.8.15 Rewrite in order of the powers of $(X - a)$: PTAYL

PTAYL takes two parameters: a polynomial $P[X]$ and a number a .

PTAYL rewrites the polynomial $P[X]$ in order of its powers of $X - a$.

Typing:

$$\text{PTAYL}(X^2 + 2 \cdot X + 1, 2)$$

produces the polynomial $Q[X]$:

$$X^2 + 6 \cdot X + 9$$

NOTE THAT:

$$P(X) = Q(X - 2)$$

5.8.16 Euclidean quotient: QUOT

The arguments of QUOT are two polynomials $A[X]$ and $B[X]$.

QUOT returns the quotient of the two polynomials $A[X]$ and $B[X]$ divided in decreasing order by exponent.

Typing:

$$\text{QUOT}(X^2 + 2 \cdot X + 1, X)$$

gives:

$$X + 2$$

5.8.17 Euclidean remainder : REMAINDER

The arguments of REMAINDER are two polynomials $A[X]$ and $B[X]$.

REMAINDER returns the remainder from the division of the two polynomials $A[X]$ and $B[X]$ divided in decreasing order by exponent.

Typing:

$$\text{REMAINDER}(X^3 - 1, X^2 - 1)$$

gives:

$$X - 1$$

5.8.18 Number of zeros in $[a, b[$: STURMAB

STURMAB has three arguments: a polynomial $P(X)$ and two numbers a and b .

STURMAB returns the number of zeros of P in $[a, b[$.

For example, typing:

$$\text{STURMAB}(X^2 \cdot (X^3 + 2), -2, 0)$$

gives:

1

Typing :

STURMAB($X^2 \cdot (X^3 + 2)$, -2, 1)

gives:

3

5.8.19 First and 2nd-species Tchebycheff polynomials : TCHEBYCHEFF

TCHEBYCHEFF has as an argument an integer n .

If $n > 0$, TCHEBYCHEFF returns the polynomial T_n such that:

$$T_n[x] = \cos(n \cdot \arccos(x))$$

For $n \geq 0$, we have:

$$T_n(x) = \sum_{k=0}^{\lfloor n/2 \rfloor} C_n^{2k} (x^2 - 1)^k x^{n-2k}$$

For $n \geq 0$ we also have:

$$(1 - x^2)T_n''(x) - xT_n'(x) + n^2T_n(x) = 0$$

For $n \geq 1$, we have:

$$T_{n+1}(x) = 2xT_n(x) - T_{n-1}(x)$$

If $n < 0$, TCHEBYCHEFF returns the 2nd-species Tchebycheff polynomial:

$$T_n[x] = \frac{\sin(n \cdot \arccos(x))}{\sin(\arccos(x))}$$

Typing:

TCHEBYCHEFF(4)

gives:

$8 \cdot X^4 - 8 \cdot X^2 + 1$

In effect:

$$\cos(4 \cdot x) = \operatorname{Re}((\cos(x) + i \cdot \sin(x))^4)$$

$\cos(4 \cdot x) = \cos(x)^4 - 6 \cdot \cos(x)^2 \cdot (1 - \cos(x)^2) + ((1 - \cos(x)^2)^2)$
 $\cos(4 \cdot x) = T_4(\cos(x))$ Typing:

TCHEBYCHEFF(-4)

gives:

$$8 \cdot X^3 - 4 \cdot X$$

In effect:

$$\sin(4 \cdot x) = \sin(x) \cdot (8 \cdot \cos(x)^3 - 4 \cdot \cos(x)).$$

5.9 Functions

All the functions in this section can be found in the DIFF menu on the menu bar, except DEF, which is in the ALGB menu, and IFTE, which is in the MATH key's Tests menu.

5.9.1 Define a function : DEF

DEF has as argument an equality between (1) the name of a function (with parentheses containing the variable), and (2) an expression defining the function.

DEF defines this function and returns the equality.

Typing:

$$\text{DEF}(U(N) = 2^N + 1)$$

produces the result:

$$U(N) = 2^N + 1$$

Typing:

$$U(3)$$

then returns:

$$9$$

Another DEF example

Calculate the first six Fermat numbers $F_1..F_6$ and say whether they're prime.

So, you want to calculate $F(k) = 2^{2^k} + 1$ for $k = 1..6$.

Typing the formula :

$$2^{2^2} + 1$$

gives a result of 17. You can then invoke the `ISPRIME?()` command, which is found in the `MATH` key's Integer menu.

The response is `1.`, which means `TRUE`. Using the history (which you access by pressing the `SYMB` key), you put the expression $2^{2^2} + 1$ into the Equation Editor with `ECHO`, and change it to:

$$2^{2^3} + 1$$

Or better, define a function `F(K)` by selecting `DEF` from the `ALGB` menu (on the menu bar), and typing:

$$\text{DEF}(F(K) = 2^{2^K} + 1)$$

The response is $2^{2^K} + 1$, and `F` is now listed amongst the variables (which you can verify using the `VAR`s key.)

For $K = 5$, you then type:

$$F(5)$$

which gives:

$$4294967297$$

You can factor $F(5)$ with `FACTOR`, which you'll find in the `ALGB` menu on the menu bar.

Typing:

$$\text{FACTOR}(F(5))$$

gives

$$641 \cdot 6700417$$

Typing:

$$F(6)$$

gives:

$$18446744073709551617$$

Using `FACTOR` to factor it then yields:

$$274177 \cdot 67280421310721$$

NOTE: Pay careful attention to the position of the dot in :

2.5 the approximate representation of $\frac{5}{2}$

versus

$$2 \cdot 5 = 10$$

Define a function with conditions : IFTE

IFTE has three arguments: a Boolean (note the use of $=$ to test for equality), and two expressions $expr1$, $expr2$.

IFTE evaluates the condition, then returns $expr1$ if the condition is true, or $expr2$ if the condition is false.

Typing:

$$\text{STORE}(2, N)$$

$$\text{IFTE}(N == 0, 1, \frac{N + 1}{N})$$

produces the result:

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

It is easy to define functions using IFTE. For example:

$$\text{DEF}(F(X) = \text{IFTE}(X == 0, 1, \frac{\text{SIN}(X)}{X}))$$

defines the function f such that:

$$f(x) = \begin{cases} 1 & \text{si } x = 0 \\ \frac{\sin(x)}{x} & \text{si } x \neq 0 \end{cases}$$

5.9.2 Derivative and partial derivative : DERIV

DERIV has two arguments: an expression (or a function) and a variable.

DERIV returns the derivative of the expression (or the function) with respect to the variable given as the second parameter (used for calculating partial derivatives).

Example:

Calculate:

$$\frac{\partial(x \cdot y^2 \cdot z^3 + x \cdot y)}{\partial z}$$

Typing:

$$\text{DERIV}(X \cdot Y^2 \cdot Z^3 + X \cdot Y, Z)$$

gives:

$$3 \cdot X \cdot Y^2 \cdot Z^2$$

5.9.3 Derivative : DERVX

DERVX has one argument: an expression.

DERVX calculates the derivative of the expression with respect to the variable stored in VX.

For example, given:

$$f(x) = \frac{x}{x^2 - 1} + \ln\left(\frac{x+1}{x-1}\right)$$

Calculate the derivative of f .

Type:

$$\text{DERVX}\left(\frac{X}{X^2 - 1} + \text{LN}\left(\frac{X+1}{X-1}\right)\right)$$

or, if you have stored the definition of $f(x)$ in F that is, if you have typed:

$$\text{STORE}\left(\frac{X}{X^2 - 1} + \text{LN}\left(\frac{X+1}{X-1}\right), F\right)$$

then type:

$$\text{DERVX}(F)$$

or, if you have defined $F(X)$ using DEF that is, if you have typed:

$$\text{DEF}(F(X) = \frac{X}{X^2 - 1} + \text{LN}\left(\frac{X+1}{X-1}\right))$$

then type:

$$\text{DERVX}(F(X))$$

The result is a complicated expression. Pressing ENTER simplifies it, giving:

$$-\frac{3 \cdot X^2 - 1}{X^4 - 2 \cdot X^2 + 1}$$

5.9.4 Fourier coefficients : FOURIER

FOURIER has two parameters: an expression $f(x)$ and a whole number N .

FOURIER returns the Fourier coefficient c_N of $f(x)$, considered to be a function defined over interval $[0, T]$ and with period T (T being equal to

the contents of the variable PERIOD).

If $f(x)$ is a discrete series, then:

$$f(x) = \sum_{N=-\infty}^{+\infty} c_N e^{\frac{2iNx\pi}{T}}$$

Example:

Determine the Fourier coefficients of a periodic function f with period $2 \cdot \pi$ and defined over interval $[0, 2 \cdot \pi[$ by $f(x) = x^2$.

Typing:

STORE($2 \cdot \pi$, PERIOD)

FOURIER(X^2 , N)

The calculator do not know that N is a whole number, so you have to replace EXP($2 * i * N * \pi$) with 1 and then simplify the expression. We get:

$$\frac{2 \cdot i \cdot N \cdot \pi + 2}{N^2}$$

So if $N \neq 0$, then:

$$c_N = \frac{2 \cdot i \cdot N \cdot \pi + 2}{N^2}$$

Typing:

FOURIER(X^2 , 0)

gives:

$$\frac{4 \cdot \pi^2}{3}$$

so if $N = 0$, then:

$$c_0 = \frac{4 \cdot \pi^2}{3}$$

5.9.5 Partial integration : IBP

IBP has two parameters: an expression of the form $u(x) \cdot v'(x)$ and $v(x)$. IBP returns the AND of $u(x) \cdot v(x)$ and of $-v(x) \cdot u'(x)$ that is, the terms that one must calculate when one performs a partial integration.

It remains then to calculate the integral of the second term of the AND, then add it to the first term of the AND to obtain a primitive of $u(x) \cdot v'(x)$.

Typing:

IBP(LN(X), X)

gives:

$$X \cdot \text{LN}(X) \text{ AND } - 1$$

One completes the integration by calling INTVX:

$$\text{INTVX}(X \cdot \text{LN}(X) \text{ AND } - 1)$$

which produces the result:

$$X \cdot \text{LN}(X) - X$$

REMARK: If the first IBP (or INTVX) parameter is an AND of two elements, IBP concerns itself only with the second element of the AND, adding the integrated term to the first element of the AND (so that you can perform multiple IBP in succession).

5.9.6 Integrals : \int

To calculate an integral, you can use the symbol \int which is found on the keyboard and it can be produced by typing : SHIFT d/dX.

You have just to enter the arguments (you can move the cursor with the arrow keys \triangleright and \triangleleft). For more details cf. 3.3.4.

You can put $+\infty$ or $-\infty$ as limit of the integral.

EXAMPLE

For the following expression, determine the limit when a approaches $+\infty$:

$$\int_2^a \left(\frac{x}{x^2 - 1} + \ln\left(\frac{x+1}{x-1}\right) \right) dx$$

In the Equation Editor, type:

$$\int_2^{+\infty} \left(\frac{X}{X^2 - 1} + \text{LN}\left(\frac{X+1}{X-1}\right) \right) dX$$

NOTE: To obtain the symbol $+\infty$, type:

$$(-) (-) \infty \text{ (SHIFT 0)}$$

This produces:

$$+\infty - \frac{7 \cdot \text{LN}(3)}{2}$$

and, after simplification:

$$+\infty$$

5.9.7 Primitive and defined integral : INTVX

INTVX has one argument : une expression.

INTVX calculates a primitive of its argument with respect to the variable stored in VX.

Exercise 1

Calculate a primitive of $\sin(x) \times \cos(x)$.

Typing:

$$\text{INTVX}(\text{SIN}(X) \cdot \text{COS}(X))$$

gives, in Step by step mode:

$$\text{COS}(X) \cdot \text{SIN}(X)$$

$$\text{Int}[u' * F(u)] \text{ with } u = \text{SIN}(X)$$

Pressing OK then sends the result to the Equation Editor:

$$\frac{\text{SIN}(X)^2}{2}$$

Exercise 2

Given:

$$f(x) = \frac{x}{x^2 - 1} + \ln\left(\frac{x+1}{x-1}\right)$$

Calculate a primitive of f .

Type:

$$\text{INTVX}\left(\frac{X}{X^2 - 1} + \text{LN}\left(\frac{X+1}{X-1}\right)\right)$$

Or, if you have stored the definition of $f(x)$ in F, that is, if you have already typed:

$$\text{DEF}(F(X) = \frac{X}{X^2 - 1} + \text{LN}\left(\frac{X+1}{X-1}\right))$$

then type:

$$\text{INTVX}(F)$$

Or, if you have used DEF to define F (X), that is, if you have already typed:

$$\text{DEF}(F(X) = \frac{X}{X^2 - 1} + \text{LN}\left(\frac{X+1}{X-1}\right))$$

then type:

$$\text{INTVX}(F(X))$$

The result in all cases is equivalent to:

$$X \cdot \text{LN}\left(\frac{X+1}{X-1}\right) + \frac{3}{2} \cdot \text{LN}(|X-1|) + \frac{3}{2} \cdot \text{LN}(|X+1|)$$

We obtain absolute values only in Rigorous mode.

Exercise 3

Calculate:

$$\int \frac{2}{x^6 + 2 \cdot x^4 + x^2} dx$$

Typing:

$$\text{INTVX}\left(\frac{2}{X^6 + 2 \cdot X^4 + X^2}\right)$$

gives a primitive:

$$-3 \cdot \text{ATAN}(X) - \frac{2}{X} - \frac{X}{X^2 + 1}$$

REMARK:

You can also type:

$$\int_1^X \frac{2}{X^6 + 2 \cdot X^4 + X^2} dX$$

which gives the primitive which is zero for $X = 1$:

$$-3 \cdot \text{ATAN}(X) - \frac{2}{X} - \frac{X}{X^2 + 1} + \frac{3 \cdot \pi + 10}{4}$$

Exercise 4

Calculate:

$$\int \frac{1}{\sin(x) + \sin(2 \cdot x)} dx$$

Typing:

$$\text{INTVX}\left(\frac{1}{\text{SIN}(X) + \text{SIN}(2 \cdot X)}\right)$$

gives the result:

$$\frac{1}{6} \cdot \text{LN}(|\text{COS}(X) - 1|) + \frac{1}{2} \cdot \text{LN}(|\text{COS}(X) + 1|) +$$

$$\frac{-2}{3} \cdot \text{LN}(|2 \cdot \text{COS}(X) + 1|)$$

REMARK: If the argument to INTVX is the AND of two elements, INTVX concerns itself only with the second element of the AND, and adds the result to the first argument.

5.9.8 To calculate limits : LIMIT or lim

LIMIT or lim has two arguments: an expression dependent on a variable, and an equality (a variable = the value to which you want to calculate the limit).

You can omit the name of the variable and the sign =, when this name is in VX).

It is often preferable to use a quoted expression:

QUOTE(expression), to avoid rewriting the expression in normal form (i.e., not to have a rational simplification of the arguments) during the execution of the LIMIT command.

Typing, for example:

$$\text{LIMIT}(\text{QUOTE}((2X - 1) \cdot \text{EXP}(\frac{1}{X - 1})), X = +\infty)$$

gives:

$$+\infty$$

To find a right limit you type, for example:

$$\text{LIMIT}(\frac{1}{X - 1}, \text{QUOTE}(1 + 0))$$

gives (if X is the current variable) :

$$+\infty$$

To find a left limit you type, for example:

$$\text{LIMIT}(\frac{1}{X - 1}, \text{QUOTE}(1 - 0))$$

gives (if X is the current variable) :

$$-\infty$$

It is not necessary to quote the second argument, when it is written with = for example :

$$\text{LIMIT}(\frac{1}{X - 1}, X = 1 + 0)$$

gives :

$$+\infty$$

EXERCISES

For $n > 2$ in the following expression, find the limit as x approaches 0:

$$\frac{n \cdot \tan(x) - \tan(n \cdot x)}{\sin(n \cdot x) - n \cdot \sin(x)}$$

You can use the `LIMIT` command to do this.

Typing:

$$\text{LIMIT} \left(\frac{N \cdot \text{TAN}(X) - \text{TAN}(N \cdot X)}{\text{SIN}(N \cdot X) - N \cdot \text{SIN}(X)}, 0 \right)$$

gives:

2

NOTE To find the limit as x approaches a^+ (resp a^-), the second argument is written :

$X = A + 0$ (resp $X = A - 0$), see also [5.9.8](#).

For the following expression, find the limit as x approaches $+\infty$:

$$\sqrt{x + \sqrt{x + \sqrt{x}}} - \sqrt{x}$$

Typing:

$$\text{LIMIT}(\sqrt{X + \sqrt{X + \sqrt{X}}} - \sqrt{X}, +\infty)$$

produces (after a short wait):

$\frac{1}{2}$

NOTE:

The symbol ∞ is obtained by typing `SHIFT 0`.

To obtain $-\infty$ type:

`(-) ∞`

To obtain $+\infty$ type:

`(-) (-) ∞`

You can also find the symbol ∞ in the `MATH` key's Constant menu.

5.9.9 Plot of an expression : PLOT

PLOT has an expression as argument.

PLOT asks you to select an Aplet and then stores this expression in an Aplet variable.

For example, type:

$$\text{PLOT}(X^2 + X)$$

You select the Aplet Function, and you choose to save the expression $X^2 + X$ in F2. In the history the answer is :

$$X^2 + X$$

To obtain the plot of the expression stored in F2, you have to quit the CAS (press HOME) and then open the Aplet Function, select the function F2, then press on PLOT

5.9.10 Add the plot of an expression :PLOTADD

PLOTADD has an expression as argument.

PLOTADD asks you to select an Aplet and then stores this expression in an Aplet-variable.

a Typing:

$$\text{PLOTADD}(X^2 - X)$$

You select the Aplet Function, and you choose to save the expression $X^2 - X$ in F3. In the history the answer is :

$$X^2 - X$$

To obtain the plot of the expressions stored in F2 and F3, you have to quit the CAS (press HOME), to open the Aplet Function, then select the functions F2 and F3, and press on PLOT.

5.9.11 Evaluate a primitive : PREVAL

PREVAL has three parameters: an expression $F(VX)$ dependent on the variable contained in VX , and two expressions A and B .

For example if VX contains X , and if F is a function, $\text{PREVAL}(F(X), A, B)$ returns $F(B) - F(A)$.

PREVAL is used for calculating an integral defined from a primitive: it evaluates this primitive between the two limits of the integral.

Typing:

$$\text{PREVAL}(X^2 + X, 2, 3)$$

gives:

6

5.9.12 Primitive and defined int egral : RISCH

RISCH has two parameters: an expression and the name of a variable. RISCH returns a primitive of the first parameter with respect to the variable specified in the second parameter.

Typing:

$$\text{RISCH}((2 \cdot X^2 + 1) \cdot \text{EXP}(X^2 + 1), X)$$

gives:

$$X \cdot \text{EXP}(X^2 + 1)$$

REMARK: If the RISCH parameter is the AND of two elements, RISCH concerns itself only with the second element of the AND, and adds the result to the first argument..

5.9.13 Discrete antiderivative : SIGMA

SIGMA has two arguments: the first argument is a function $f(x)$ of a variable x given as the second argument. SIGMA returns the discrete antiderivative of the input function, that is, the function G that satisfies the relation: $G(x + 1) - G(x) = f(x)$.

For example, typing :

$$\text{SIGMA}(X \cdot X!, X)$$

gives :

$$X!$$

because $(X + 1)! - X! = X \cdot X!$

5.9.14 Discrete antiderivative : SIGMAVX

SIGMAVX has as argument a function f of the current variable VX. SIGMAVX returns the discrete antiderivative of the input function, that is a function G that satisfies the relation: $G(x + 1) - G(x) = f(x)$.

For example, type:

$$\text{SIGMAVX}(X^2)$$

You obtain:

$$\frac{2 \cdot X^3 - 3 \cdot X^2 + X}{6}$$

because $2(X + 1)^3 - 3(X + 1)^2 + X + 1 - 2X^3 + 3X^2 - X = 6X^2$

5.9.15 Variation table : TABVAR

TABVAR has as a parameter an expression with a rational derivative.
 TABVAR returns the variation table for the expression in terms of the current variable.

Typing:

$$\text{TABVAR}(\text{LN}(X) + X)$$

In Step by step mode, this gives:

$$F =: (\text{LN}(X) + X)$$

$$F' =: \left(\frac{1}{X} + 1\right)$$

$$\rightarrow: \frac{X+1}{X}$$

Variation table:

$$\left[\begin{array}{cccccc} -\infty & ? & 0 & + & +\infty & X \\ ? & ? & -\infty & \uparrow & +\infty & F \end{array} \right]$$

5.10 Taylor and asymptotic Expansions

All functions in this section can be found in the DIFF menu on the menu bar.

It is customary to write the expansions in ascending order by exponent of the variable; you set this up by choosing $1 + x + x^2 \dots$ in CFG.

5.10.1 Division in increasing order by exponent : DIVPC

DIVPC has three arguments: two polynomials $A(X)$ and $B(X)$ (where $B(0) \neq 0$), and a whole number n .

DIVPC returns the quotient $Q(X)$ of the division of $A(X)$ by $B(X)$, in increasing order by exponent, and with $\deg(Q) \leq n$ or $Q = 0$.

$Q[X]$ is then the limited n th-order expansion of $\frac{A[X]}{B[X]}$ in the vicinity of $X = 0$.

Typing:

$$\text{DIVPC}(1 + X^2 + X^3, 1 + X^2, 5)$$

gives:

$$1 + X^3 - X^5$$

NOTE: When the calculator asks to go into "increasing powers" mode, respond yes.

5.10.2 Limited n th-order expansion : SERIES

SERIES has three arguments : an expression dependent on a variable, an equality (the variable $x =$ the value a to which you want to calculate the expansion) and a whole number (the order n of the limited expansion).

You can omit the name of the variable and the sign $=$, when this name is in VX)

SERIES returns the limited n th-order expansion of the expression in the vicinity of $x = a$.

- Expansion in the vicinity of $x = a$

Example:

Give a limited 4th-order expansion of $\cos(2 \times x)^2$ in the vicinity of $x = \frac{\pi}{6}$.

For this you use the SERIES command.

Typing:

$$\text{SERIES}(\text{COS}(2 \cdot X)^2, X = \frac{\pi}{6}, 4)$$

gives:

$$\left(\frac{1}{4} - \sqrt{3}h + 2h^2 + \frac{8\sqrt{3}}{3}h^3 - \frac{8}{3}h^4 + O\left(\frac{h^5}{4}\right)\right) \Big|_{h = X - \frac{\pi}{6}}$$

- Expansion in the vicinity of $x=+\infty$ or $x=-\infty$

Example 1:

Give a 5th-order expansion of $\arctan(x)$ in the vicinity of $x=+\infty$, taking as infinitely small $h = \frac{1}{x}$.

Typing:

$$\text{SERIES}(\text{ATAN}(X), X = +\infty, 5)$$

gives:

$$\left(\frac{\pi}{2} - h + \frac{h^3}{3} - \frac{h^5}{5} + O\left(\frac{\pi \cdot h^6}{2}\right)\right) \Big|_{h = \frac{1}{X}}$$

Example 2:

Give a 2nd-order expansion of $(2x-1)e^{\frac{1}{x-1}}$ in the vicinity of $x=+\infty$, taking as infinitely small $h = \frac{1}{x}$.

Typing:

$$\text{SERIES}((2X - 1) \cdot \text{EXP}\left(\frac{1}{X - 1}\right), X = +\infty, 3)$$

gives:

$$\left(\frac{12 + 6h + 12h^2 + 17h^3}{6 \cdot h} + 0(2 \cdot h^3) \right) \Big|_{h = \frac{1}{X}}$$

Example 3:

Give a 2nd-order expansion of $(2x-1)e^{\frac{1}{x-1}}$ in the vicinity of $x=-\infty$, taking as infinitely small $h = -\frac{1}{x}$.

Typing:

$$\text{SERIES}((2X - 1) \cdot \text{EXP}\left(\frac{1}{X - 1}\right), X = -\infty, 3)$$

gives:

$$\left(\frac{-12 + 6h - 12h^2 + 17h^3}{6h} + 0(-(2h^3)) \right) \Big|_{h = -\frac{1}{X}}$$

- Unidirectional expansion

You must be in *Rigorous* mode (Press **DIFF**, then **CFG**, then choose *Rigorous* and then **Quit config**).

To perform an expansion in the vicinity of $x = a$ where $x > a$, use a positive real (such as 4. or 4.0) for the order; to perform an expansion in the vicinity of $x = a$ where $x < a$, use a negative real (such as -4.0 or -4.) for the order

Example 1:

Give a 3rd-order expansion of $\frac{(1+X)^{\frac{1}{X}}}{X^3}$ in the vicinity of $X = 0^+$.

Typing:

$$\text{SERIES}(\sqrt{X^2 + X^3}, X = 0, 3.0)$$

gives:

$$\frac{1}{16} \cdot h^4 + \frac{-1}{8} \cdot h^3 + \frac{1}{2} \cdot h^2 + h + 0(h^5) \Big|_{h = X}$$

Example 2:

Give a 3rd-order expansion of $\frac{(1+X)^{\frac{1}{X}}}{X^3}$ in the vicinity of $X = 0^-$.

Typing:

$$\text{SERIES}(\sqrt{X^2 + X^3}, X = 0, -3.0)$$

gives:

$$\left(\frac{-1}{16} \cdot h^4 + \frac{-1}{8} \cdot h^3 + \frac{-1}{2} \cdot h^2 + h + 0(h^5) \right) \Big|_{h = -X}$$

Example 3:

Give a 3rd-order expansion of $\frac{(1+X)^{\frac{1}{X}}}{X^3}$ in the vicinity of $X = 0$.

Typing:

```
SERIES(sqrt(X^2 + X^3), X = 0, 3)
```

gives:

```
SERIESError : Unable to find sign
```

Note :

If you are in `Sloppy` mode the three previous examples give the same answer than in the vicinity of $X = 0+$

5.10.3 Limited expansion in the vicinity of 0 : TAYLOR0

TAYLOR0 has a single argument: the function of x to expand. It returns the function's limited 4th-relative-order expansion in the vicinity of $x = 0$ (if x is the current variable).

Typing:

```
TAYLOR0((TAN(P * X) - SIN(P * X)) / (TAN(Q * X) - SIN(Q * X)))
```

gives:

$$\frac{P^3}{Q^3} + \frac{P^5 - Q^2 \cdot P^3}{4 \cdot Q^3} \cdot X^2$$

NOTE: "th-order" means that the numerator and the denominator are expanded to the 4th relative order (here, the 5th absolute order for the numerator, and for the denominator, which is given in the end, the 2nd order (5-3), seeing that the exponent of the denominator is 3).

5.10.4 Truncate at order $n - 1$: TRUNC

TRUNC enables you to truncate a polynomial at a given order (used to perform limited expansions).

TRUNC has two arguments: a polynomial and X^n .

TRUNC returns the polynomial truncated at order $n - 1$; that is, the returned polynomial has no terms with exponents $\geq n$.

Typing:

```
TRUNC((1 + X + 1/2 * X^2)^3, X^4)
```

gives:

$$1 + 3 \cdot X + \frac{9}{2} \cdot X^2 + 4 \cdot X^3$$

5.11 Conversion Functions

All functions in this section can be found in the REWRITE menu on the menu bar.

5.11.1 Distributivity of multiplication : DISTRIB

DISTRIB enables you to apply the distributivity of multiplication in respect to addition in a single instance.

DISTRIB enables you, when you apply it several times, to carry out the distributivity step by step.

Typing:

$$\text{DISTRIB}((X + 1) \cdot (X + 2) \cdot (X + 3))$$

gives:

$$X \cdot (X + 2) \cdot (X + 3) + 1 \cdot (X + 2) \cdot (X + 3)$$

5.11.2 Disregard small values : EPSX0

EPSX0 has as a parameter an expression in X, and returns the same expression with the values less than EPS replaced by zeroes.

Typing:

$$\text{EPSX0}(0.001 + X)$$

gives, if EPS=0.01:

$$0 + X$$

or, if EPS=0.0001:

$$.001 + X$$

5.11.3 Transform $\exp(n \cdot \ln(x))$ as a power of x : EXP2POW

EXP2POW transforms an expression of the form $\exp(n \times \ln(x))$, rewriting it as a power of x .

Typing:

$$\text{EXP2POW}(\text{EXP}(N \cdot \text{LN}(X)))$$

gives:

$$X^N$$

Take careful note of the difference between this function and LNCOLLECT, as shown in the following examples:

$$\text{LNCOLLECT}(\text{EXP}(N \cdot \text{LN}(X))) = \text{EXP}(N \cdot \text{LN}(X))$$

$$\begin{aligned} \text{LNCOLLECT}(\text{EXP}(\text{LN}(X)/3)) &= \text{EXP}(\text{LN}(X)/3) \\ \text{EXP2POW}(\text{EXP}(\text{LN}(X)/3)) &= \sqrt[3]{X} \end{aligned}$$

5.11.4 Transform a trigonometric expression into complex exponentials : EXPLN

EXPLN takes as an argument a trigonometric expression.

EXPLN transforms the trigonometric function into exponentials and logarithms without linearising it.

EXPLN puts the calculator into complex mode.

Typing:

$$\text{EXPLN}(\text{SIN}(X))$$

gives:

$$\frac{\text{EXP}(i \cdot X) - \frac{1}{\text{EXP}(i \cdot X)}}{2 \cdot i}$$

5.11.5 Distributivity : FDISTRIB

FDISTRIB has an expression as argument.

FDISTRIB enables you to apply the distributivity of multiplication with respect to addition all at once.

Typing:

$$\text{FDISTRIB}((X + 1) \cdot (X + 2) \cdot (X + 3))$$

gives:

$$X \cdot X \cdot X + 3 \cdot X \cdot X + X \cdot 2 \cdot X + 3 \cdot 2 \cdot X + X \cdot X \cdot 1 + 3 \cdot X \cdot 1 + X \cdot 2 \cdot 1 + 3 \cdot 2 \cdot 1$$

after simplification (ENTER) :

$$X^3 + 6 \cdot X^2 + 11 \cdot X + 6$$

5.11.6 Linearise the exponentials : LIN

LIN has as an argument an expression containing exponentials and trigonometric functions.

LIN converts trigonometric expressions to exponentials and then linearises the complex exponentials (in terms of $\exp(n \cdot x)$).

LIN puts the calculator into complex mode when dealing with trigonometric functions.

Example 1

Typing:

$$\text{LIN}((\text{EXP}(X) + 1)^3)$$

gives:

$$3 \cdot \text{EXP}(X) + 1 + 3 \cdot \text{EXP}(2 \cdot X) + \text{EXP}(3 \cdot X)$$

Example 2

Typing:

$$\text{LIN}(\text{SIN}(X))$$

gives:

$$-\left(\frac{i}{2} \cdot \text{EXP}(i \cdot X)\right) + \frac{i}{2} \cdot \text{EXP}(-i \cdot X)$$

Example 3

Typing:

$$\text{LIN}(\text{COS}(X)^2)$$

gives:

$$\frac{1}{4} \cdot \text{EXP}(-2 \cdot i \cdot X) + \frac{1}{2} + \frac{1}{4} \cdot \text{EXP}(2 \cdot i \cdot X)$$

5.11.7 Regroup the logarithms : LNCOLLECT

LNCOLLECT has as an argument an expression containing logarithms. LNCOLLECT regroups the terms in the logarithms. It is therefore preferable to use an expression that has already been factored (using FACTOR).

Typing:

$$\text{LNCOLLECT}(\text{LN}(X + 1) + \text{LN}(X - 1))$$

gives:

$$\text{LN}((X + 1)(X - 1))$$

5.11.8 Transform a power : POWEXPAND

POWEXPAND writes a power in the form of a product.

Typing:

$$\text{POWEXPAND}((X + 1)^3)$$

gives:

$$(X + 1) \cdot (X + 1) \cdot (X + 1)$$

This allow you to do the developpement of $(x + 1)^3$ in step by step, using DISTRIB several times on the precedent result.

5.11.9 Simplify : SIMPLIFY

SIMPLIFY simplifies an expression automatically.

As with all automated simplification routines, however, one should not expect miracles.

Typing:

$$\text{SIMPLIFY}\left(\frac{\text{SIN}(3 \cdot X) + \text{SIN}(7 \cdot X)}{\text{SIN}(5 \cdot X)}\right)$$

gives, after simplification:

$$4 \cdot \text{COS}(X)^2 - 2$$

5.11.10 Evaluation of reals : XNUM

XNUM has an expression as a parameter.

XNUM puts the calculator into approximate mode and returns the numeric value of the expression.

Typing:

$$\text{XNUM}(\sqrt{2})$$

gives:

$$1.41421356237$$

5.11.11 Rational approximation : XQ

XQ has a real numeric expression as a parameter.

XQ puts the calculator into exact mode and gives a rational or real approximation of the expression.

Typing:

$$\text{XQ}(1.41421)$$

gives:

$$\frac{66441}{46981}$$

Typing:

$$\text{XQ}(1.414213562)$$

gives:

$$\sqrt{2}$$

5.12 Equations

All the functions in this section are found in the SOLV menu on the menu bar.

5.12.1 The zeros of an expression : ISOLATE

ISOLATE returns the values that are the zeros of an expression or an equation.

ISOLATE has two parameters: an expression or equation, and the name of the variable to isolate (ignoring REALASSUME).

Typing:

$$\text{ISOLATE}(X^4 - 1 = 3, X)$$

gives in real mode:

$$(X = \sqrt{2}) \text{ OR } (X = -\sqrt{2})$$

and in complex mode:

$$(X = \sqrt{2} \cdot i) \text{ OR } (X = -\sqrt{2}) \text{ OR } (X = -(\sqrt{2} \cdot i)) \text{ OR } (X = \sqrt{2})$$

5.12.2 Solve equations : SOLVE

SOLVE has as two parameters:

(1) either an equality between two expressions, or a single expression (in which case = 0 is implied), and

(2) the name of a variable.

SOLVE solves the equation in \mathbb{R} in real mode and in \mathbb{C} in complex mode (ignoring REALASSUME).

Typing:

$$\text{SOLVE}(X^4 - 1 = 3, X)$$

gives, in real mode:

$$(X = -\sqrt{2}) \text{ OR } (X = \sqrt{2})$$

or, in complex mode:

$$(X = -\sqrt{2}) \text{ OR } (X = \sqrt{2}) \text{ OR } (X = -i \cdot \sqrt{2}) \text{ OR } (X = i\sqrt{2})$$

5.12.3 Solve equations : SOLVEVX

SOLVEVX has as a parameter either

(1) an equality between two expressions in the variable contained in VX,

or

(2) a single such expression (in which case = 0 is implied).

SOLVEVX solves the equation.

Example 1

Typing:

$$\text{SOLVEVX}(X^4 - 1 = 3)$$

gives, in real mode:

$$(X = -\sqrt{2}) \text{ OR } (X = \sqrt{2})$$

or, in complex mode, even if you have chosen X as real :

$$(X = -\sqrt{2}) \text{ OR } (X = \sqrt{2}) \text{ OR } (X = -i \cdot \sqrt{2}) \text{ OR } (X = i\sqrt{2})$$

Example 2

Typing:

$$\text{SOLVEVX}((X - 2) \cdot \text{SIN}(X))$$

gives, in real mode:

$$(X = -2 \cdot \pi \cdot n_1) \text{ OR } (X = 2 \cdot \pi \cdot n_1) \text{ OR } (X = 2)$$

5.13 Systems

All the functions in this section are found in the SOLV menu on the menu bar.

5.13.1 Solve linear system : LINSOLVE

LINSOLVE enables you to solve a system of linear equations.

It's assumed that the various equations are of the form *expression* = 0.

LINSOLVE has two arguments: the first members of the various equations separated by AND, and the names of the various variables separated by AND.

Example 1

Typing:

$$\text{LINSOLVE}(X + Y + 3 \text{ AND } X - Y + 1, X \text{ AND } Y)$$

gives:

$$(X = -2) \text{ AND } (Y = -1)$$

or, in Step by step mode (CFG etc...):

L2=L2-L1

$$\begin{bmatrix} 1 & 1 & 3 \\ 1 & -1 & 1 \end{bmatrix}$$

ENTER

L1=2L1+L2

$$\begin{bmatrix} 1 & 1 & 3 \\ 0 & -2 & -2 \end{bmatrix}$$

ENTER

Reduction Result

$$\begin{bmatrix} 2 & 0 & 4 \\ 0 & -2 & -2 \end{bmatrix}$$

ENTER

The following is then written to the Equation Editor:

$$(X = -2) \text{ AND } (Y = -1)$$

Example 2

Type:

$$(2 \cdot X + Y + Z = 1) \text{ AND } (X + Y + 2 \cdot Z = 1) \text{ AND } (X + 2 \cdot Y + Z = 4)$$

Then, invoke LINSOLVE

and type the unknowns:

$$X \text{ AND } Y \text{ AND } Z$$

and press the ENTER key.

The following result is produced if you're in Step by step mode (CFG etc...):

L2=2L2-L1

$$\begin{bmatrix} 2 & 1 & 1 & -1 \\ 1 & 1 & 2 & -1 \\ 1 & 2 & 1 & -4 \end{bmatrix}$$

puis ok

L3=2L3-L1

$$\begin{bmatrix} 2 & 1 & 1 & -1 \\ 0 & 1 & 3 & -1 \\ 1 & 2 & 1 & -4 \end{bmatrix}$$

...and so on until, finally:

Reduction Result

$$\begin{bmatrix} 8 & 0 & 0 & 4 \\ 0 & 8 & 0 & -20 \\ 0 & 0 & -8 & -4 \end{bmatrix}$$

then press ENTER and

$$(X = -\frac{1}{2}) \text{ AND } (Y = \frac{5}{2}) \text{ AND } (Z = -\frac{1}{2})$$

is written to the Equation Editor.

5.13.2 Solve non linear system : SOLVE

SOLVE accepts non-linear equations if they are polynomial.

Otherwise, MSLV should be used in HOME to get a numerical solution.

It's assumed that the various equations are of the form *expression* = 0.

SOLVE has as arguments, the first members of the various equations separated by AND, and the names of the various variables separated by AND.

Typing:

$$\text{SOLVE}(X^2 + Y^2 - 3 \text{ AND } X - Y^2 + 1, X \text{ AND } Y)$$

gives:

$$(X = 1) \text{ AND } (Y = -\sqrt{2}) \text{ OR } (X = 1) \text{ AND } (Y = \sqrt{2})$$

5.14 Differential Equations

All the functions in this section are found in the SOLV menu on the Equation Editor menu bar.

5.14.1 Solve of differential equations : DESOLVE

DESOLVE enables you to solve differential equations (for linear differential equations having constant coefficients, it is better to use LDEC).

DESOLVE has two arguments : the differential equation where y' is written d1Y(X) (or the differential equation and the initial conditions separated by AND), and the unknown Y(X).

You have to be in real mode.

Example 1

Solve:

$$y'' + y = \cos(x)$$

$$y(0) = c_0 \quad y'(0) = c_1$$

Typing:

$$\text{DESOLVE}(d1d1Y(X) + Y(X) = \text{COS}(X), Y(X))$$

gives:

$$Y(X) = cC0 \cdot \text{COS}(X) + \frac{X + 2 \cdot cC1}{2} \cdot \text{SIN}(X)$$

$cC0$ and $cC1$ are integration constants ($y(0) = cC0$ $y'(0) = cC1$).

You can then assign values to the constants using the `SUBST` command.

To produce the solutions for $y(0) = 1$, type:

$$\text{SUBST}(Y(X) = cC0 \cdot \text{COS}(X) + \frac{X + 2 \cdot cC1}{2} \cdot \text{SIN}(X), cC0 = 1)$$

which gives:

$$Y(X) = \frac{2 \cdot \text{COS}(X) + (X + 2 \cdot cC1) \cdot \text{SIN}(X)}{2}$$

Example 2 :

Solve:

$$y'' + y = \cos(x)$$

$$y(0) = 1 \quad y'(0) = 1$$

It is possible to solve for the constants from the outset.

Typing

$$\text{DESOLVE}((d1d1Y(X) + Y(X) = \text{COS}(X))\text{AND}(Y(0) = 1)\text{AND}(d1Y(0) = 1), Y(X))$$

Gives :

$$Y(X) = \text{COS}(X) + \frac{X + 2}{2} \cdot \text{SIN}(X)$$

5.14.2 Laplace transform and inverse Laplace transform : LAP and ILAP

LAP is the Laplace transform of the expression given as argument. We consider this expression as the value of a function of the variable stored in `VX`.

ILAP is the inverse Laplace transform of the expression given as argument. We consider this expression as the value of a function of the variable stored in VX.

Laplace transform (LAP) and inverse Laplace transform (ILAP) are useful to solve linear differential equations with constant coefficients, for example:

$$y'' + p \cdot y' + q \cdot y = f(x)$$

$$y(0) = a \quad y'(0) = b$$

The following relations hold:

$$\text{LAP}(Y)(X) = \int_0^{+\infty} e^{-X \cdot T} Y(T) dT$$

$$\text{ILAP}(F)(X) = \frac{1}{2 \cdot i \cdot \pi} \int_C e^{Z \cdot X} F(Z) dZ$$

where C is a closed contour enclosing the poles of F.

The following property is used:

$$\text{LAP}(Y')(X) = -Y(0) + X \cdot \text{LAP}(Y)(X)$$

The solution Y of

$Y'' + P \cdot Y' + Q \cdot Y = F(x)$, $Y(0) = A$, $Y'(0) = B$ is then:

$$\text{ILAP}\left(\frac{\text{LAP}(F(X)) + (X + P) \cdot A + B}{X^2 + P \cdot X + Q}\right)$$

Example:

Solve:

$$y'' - 6 \cdot y' + 9 \cdot y = x \cdot e^{3 \cdot x}$$

$$y(0) = a$$

$$y'(0) = b$$

Typing:

$$\text{LAP}(X \cdot \text{EXP}(3 \cdot X)) \text{ ENTER}$$

gives:

$$\frac{1}{X^2 - 6 \cdot X + 9}$$

Typing:

$$\text{ILAP}\left(\frac{\frac{1}{X^2 - 6 \cdot X + 9} + (X - 6) \cdot A + B}{X^2 - 6 \cdot X + 9}\right)$$

gives the solution y :

$$\left(\frac{X^3}{6} - (3 \cdot A - B) \cdot X + A\right) \cdot \text{EXP}(3 \cdot X)$$

5.14.3 Linear differential equations having constant coefficients : LDEC

LDEC enables you to directly solve linear differential equations having constant coefficients.

The parameters are the second member and the characteristic equation.

Solve:

$$y'' - 6 \cdot y' + 9 \cdot y = x \cdot e^{3 \cdot x}$$

Typing:

$$\text{LDEC}(X \cdot \text{EXP}(3 \cdot X), X^2 - 6 \cdot X + 9)$$

gives:

$$-\left(\frac{(18 \cdot X - 6) \cdot cC0 - (6 \cdot X \cdot cC1 + X^3)}{6}\right) \cdot \text{EXP}(3 \cdot X)$$

cC0 and cC1 are integration constants ($y(0) = cC0$ and $y'(0) = cC1$).

5.15 Trigonometric Expressions

All the functions in this section are found in the TRIG menu on the CAS menu bar.

5.15.1 Transform the arccos into arcsin : ACOS2S

ACOS2S has as an argument a trigonometric expression.

ACOS2S transforms the expression by replacing :

$\arccos(x)$ with $\frac{\pi}{2} - \arcsin(x)$.

Typing:

$$\text{ACOS2S}(\text{ACOS}(X) + \text{ASIN}(X))$$

gives, when simplified :

$$\frac{\pi}{2}$$

5.15.2 Transform the arcsin into arccos : ASIN2C

ASIN2C has as an argument a trigonometric expression.

ASIN2C transforms the expression by replacing :

$\arcsin(x)$ with $\frac{\pi}{2} - \arccos(x)$.

Typing:

$$\text{ASIN2C}(\text{ACOS}(X) + \text{ASIN}(X))$$

gives, when simplified:

$$\frac{\pi}{2}$$

5.15.3 Transform the arcsin into arctan : ASIN2T

ASIN2T has as an argument a trigonometric expression.

ASIN2T transforms the expression by replacing :

$\arcsin(x)$ with $\arctan\left(\frac{x}{\sqrt{1-x^2}}\right)$.

Typing:

$$\text{ASIN2T}(\text{ASIN}(X))$$

gives:

$$\text{ATAN}\left(\frac{X}{\sqrt{1-X^2}}\right)$$

5.15.4 Transform the arctan into arcsin : ATAN2S

ATAN2S has as an argument a trigonometric expression.

ATAN2S transforms the expression by replacing $\arctan(x)$ with $\arcsin\left(\frac{x}{\sqrt{1+x^2}}\right)$.

Typing:

$$\text{ATAN2S}(\text{ATAN}(X))$$

gives:

$$\text{ASIN}\left(\frac{X}{\sqrt{X^2+1}}\right)$$

5.15.5 Transform exponential into hyperbolic functions :

EXP2HYP

EXP2HYP has an expression enclosing exponentials as argument.

EXP2HYP transforms this expression with the relation :

$\exp(a) = \sinh(a) + \cosh(a)$.

Typing :

$$\text{EXP2HYP}(\text{EXP}(A))$$

gives :

$$\text{SINH}(A) + \text{COSH}(A)$$

Typing :

$$\text{EXP2HYP}(\text{EXP}(-A) + \text{EXP}(A))$$

gives :

$$2 \cdot \text{COSH}(A)$$

5.15.6 Transform in terms of $\tan(x/2)$: HALFTAN

HALFTAN has as an argument a trigonometric expression.

HALFTAN transforms $\sin(x)$, $\cos(x)$ and $\tan(x)$ in the expression, rewriting them in terms of $\tan(\frac{x}{2})$.

Typing:

$$\text{HALFTAN}\left(\frac{\text{SIN}(2 \cdot X)}{1 + \text{COS}(2 \cdot X)}\right)$$

gives, after simplification:

$$\text{TAN}(X)$$

Typing:

$$\text{HALFTAN}(\text{SIN}(X)^2 + \text{COS}(X)^2)$$

gives ($\text{SQ}(X) = X^2$):

$$\left(\frac{2 \cdot \text{TAN}\left(\frac{X}{2}\right)}{\text{SQ}\left(\text{TAN}\left(\frac{X}{2}\right)\right) + 1}\right)^2 + \left(\frac{1 - \text{SQ}\left(\text{TAN}\left(\frac{X}{2}\right)\right)}{\text{SQ}\left(\text{TAN}\left(\frac{X}{2}\right)\right) + 1}\right)^2$$

or, after simplification:

$$1$$

5.15.7 Transform the complex exponentials into sin and cos : SINCOS

SINCOS takes as an argument an expression containing complex exponentials.

SINCOS then rewrites this expression in terms of $\sin(x)$ and $\cos(x)$.

Typing:

$$\text{SINCOS}(\text{EXP}(i \cdot X))$$

gives after turning on complex mode, if necessary :

$$\text{COS}(X) + i \cdot \text{SIN}(X)$$

5.15.8 Transform $\tan(x)$ with $\sin(2x)$ and $\cos(2x)$: TAN2CS2

TAN2CS2 takes as an argument a trigonometric expression.

TAN2CS2 transforms this expression by replacing $\tan(x)$ with $\frac{1 - \cos(2 \cdot x)}{\sin(2 \cdot x)}$.

Typing:

$$\text{TAN2CS2}(\text{TAN}(X))$$

gives:

$$\frac{1 - \cos(2 \cdot X)}{\sin(2 \cdot X)}$$

5.15.9 Replace $\tan(x)$ with $\sin(x)/\cos(x)$: TAN2SC

TAN2SC has as an argument a trigonometric expression.

TAN2SC transforms this expression by replacing $\tan(x)$ with $\frac{\sin(x)}{\cos(x)}$.

Typing:

$$\text{TAN2SC}(\text{TAN}(X))$$

gives:

$$\frac{\sin(X)}{\cos(X)}$$

5.15.10 Transform $\tan(x)$ with $\sin(2x)$ and $\cos(2x)$: TAN2SC2

TAN2SC2 has as an argument a trigonometric expression.

TAN2SC2 transforms this expression by replacing $\tan(x)$ with $\frac{\sin(2 \cdot x)}{1 + \cos(2 \cdot x)}$.

Typing:

$$\text{TAN2SC2}(\text{TAN}(X))$$

gives

$$\frac{\sin(2 \cdot X)}{1 + \cos(2 \cdot X)}$$

5.15.11 Reconstruct the sine and the cosine of the same angle : TCOLLECT

TCOLLECT has as an argument a trigonometric expression.

TCOLLECT linearises this expression in terms of $\sin(n \cdot x)$ and $\cos(n \cdot x)$, then (in Real mode) reconstructs the sine and cosine of the same angle.

Typing:

$$\text{TCOLLECT}(\sin(X) + \cos(X))$$

gives:

$$\sqrt{2} \cdot \cos\left(X - \frac{\pi}{4}\right)$$

5.15.12 Develop transcendental expressions : TEXPAND

TEXPAND has as argument a transcendental expression (that is an expression with trigonometric or exponential or logarithmic functions).

TEXPAND develops this expression in terms of $\sin(x)$, $\cos(x)$, $\exp(x)$ or $\ln(x)$.

Example 1

Typing:

$$\text{TEXPAND}(\text{EXP}(X + Y))$$

gives:

$$\text{EXP}(X) \cdot \text{EXP}(Y)$$

Example 2

Typing:

$$\text{TEXPAND}(\text{LN}(X * Y))$$

gives:

$$\text{LN}(Y) \cdot \text{LN}(X)$$

Example 3

Typing:

$$\text{TEXPAND}(\text{COS}(X + Y))$$

gives:

$$\text{COS}(Y) \cdot \text{COS}(X) - \text{SIN}(Y) \cdot \text{SIN}(X)$$

Example 4

Typing:

$$\text{TEXPAND}(\text{COS}(3 \cdot X))$$

gives:

$$4 \cdot \text{COS}(X)^3 - 3 \cdot \text{COS}(X)$$

Example 5

Typing:

$$\text{TEXPAND}\left(\frac{\text{SIN}(3 \cdot X) + \text{SIN}(7 \cdot X)}{\text{SIN}(5 \cdot X)}\right)$$

gives, after pressing ENTER to simplify:

$$4 \cdot \text{COS}(X)^2 - 2$$

5.15.13 Linearise une trigonometric expression : TLIN

TLIN has as an argument a trigonometric expression.

TLIN linearises this expression in terms of $\sin(n \cdot x)$ and $\cos(n \cdot x)$.

Example 1

Typing:

$$\text{TLIN}(\text{COS}(X) \cdot \text{COS}(Y))$$

gives:

$$\frac{1}{2} \cdot \text{COS}(X - Y) + \frac{1}{2} \cdot \text{COS}(X + Y)$$

Example 2

Typing:

$$\text{TLIN}(\text{COS}(X)^3)$$

gives:

$$\frac{1}{4} \cdot \text{COS}(3 \cdot X) + \frac{3}{4} \cdot \text{COS}(X)$$

Example 3

Typing:

$$\text{TLIN}(4 \cdot \text{COS}(X)^2 - 2)$$

gives:

$$2 \cdot \text{COS}(2 \cdot X)$$

5.15.14 Simplify using $\sin(x)^2 + \cos(x)^2 = 1$: TRIG

TRIG has as an argument a trigonometric expression.

TRIG simplifies this expression using the identity $\sin(x)^2 + \cos(x)^2 = 1$.

Typing:

$$\text{TRIG}(\text{SIN}(X)^2 + \text{COS}(X)^2 + 1)$$

gives:

$$2$$

5.15.15 Simplify using the cosines : TRIGCOS

TRIGCOS has as an argument a trigonometric expression.

TRIGCOS simplifies this expression, using the identity $\sin(x)^2 + \cos(x)^2 = 1$ to rewrite it in terms of cosines.

Typing:

$$\text{TRIGCOS}(\text{SIN}(X)^4 + \text{COS}(X)^2 + 1)$$

gives:

$$\text{COS}(X)^4 - \text{COS}(X)^2 + 2$$

5.15.16 Simplify using the sines : TRIGSIN

TRIGSIN has as an argument a trigonometric expression.

TRIGSIN simplifies this expression, using the identity $\sin(x)^2 + \cos(x)^2 = 1$ to rewrite it in terms of sines.

Typing:

$$\text{TRIGSIN}(\text{SIN}(X)^4 + \text{COS}(X)^2 + 1)$$

gives:

$$\text{SIN}(X)^4 - \text{SIN}(X)^2 + 2$$

5.15.17 Simplify using the tangents : TRIGTAN

TRIGTAN has as an argument a trigonometric expression.

TRIGTAN simplifies this expression, using the identity $\sin(x)^2 + \cos(x)^2 = 1$ to rewrite it in terms of tangents.

Typing:

$$\text{TRIGTAN}(\text{SIN}(X)^4 + \text{COS}(X)^2 + 1)$$

gives:

$$\frac{2 \cdot \text{TAN}(X)^4 + 3 \cdot \text{TAN}(X)^2 + 2}{\text{TAN}(X)^4 + 2 \cdot \text{TAN}(X)^2 + 1}$$

5.15.18 Simplification with complex exponentials : TSIMP

TSIMP has an expression as argument.

TSIMP simplifies the expression rewriting it in function of complex exponentials, and then reducing the number of variables (enabling complex mode in the process).

Use TSIMP only as a last resort.

Typing:

$$\text{TSIMP}\left(\frac{\text{SIN}(3 \cdot X) + \text{SIN}(7 \cdot X)}{\text{SIN}(5 \cdot X)}\right)$$

You obtain after simplification (that is, after copying the result 2 times):

$$\frac{\text{EXP}(i \cdot X)^4 + 1}{\text{EXP}(i \cdot X)^2}$$

5.16 Version number of your CAS : VER

VER returns the version number of your CAS.

Typing:

VER

gives:

4.20050219

This result means that you have a version 4 CAS, dated 19 February 2005. Instead, *VERSION* returns the version number of the calculator's ROM as a whole.

Type in HOME screen:

VERSION

You obtain:

Version HP39 – E Copyright HP 19932004

Chapter 6

Worked Exercises with the HP40

6.1 Introduction

Begin by selecting CAS:

to do this, press F6 for CAS on the menu bar.

The various commands used in this chapter are found:

- in the Equation Editor menus:

ALGB (CFG DEF FACTOR SUBST TEXPAND)
DIFF (DERIVX DERIV INTVX LIMIT TABVAR)
REWRITE (DISTRIB LIN POWEXPAND XNUM)
SOLV (LINSOLV)

and in the MATH key's menu:

Complex (DROITE RE IM)
Integer (IEGCD ISPRIME? PROPFRAC).

Next, put the calculator into exact real algebraic mode:

to do this, press ALGB on the menu bar and highlight CFG, then press OK on the menu bar.

It suffices then to choose `Default cfg`, then OK on the menu bar, but you can also choose `Direct mode` or `Step by step mode (Step/step)`, then quit the configuration menu with CANCEL on the menu bar.

Do not forget that you must press ENTER after each command!

In the remainder of this chapter, you will find portions of the 1999 mathematical proof of Brevet d'Amiens, and the 1999 mathematical proof (series

S) of Bac.

The examples illustrate as many features of the HP40GS as possible...

It is worth noting, though, that it is still up to the student to take care to check the calculations and to know what course to follow when carrying out calculations.

6.2 Exercises on Brevet

6.2.1 Exercise 1

Given A :

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

calculate the result of A in the form of an irreducible fraction, showing each step of the calculation.

In the Equation Editor, enter the value of A by typing:

$$3 \div 2 \triangleright - 1 \triangleright \triangleright \div 1 \div 2 \triangleright + 1$$

\triangleright selects the denominator.

ENTER simplifies the denominator, giving:

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

Then, select the numerator using \triangleleft ,

ENTER simplifies the numerator, giving:

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

\triangle selects the entire fraction, and ENTER simplifies the fraction, giving:

$$\frac{1}{3}$$

6.2.2 Exercise 2

Given the number C :

$$C = 2\sqrt{45} + 3\sqrt{12} - \sqrt{20} - 6\sqrt{3}$$

write C in the form $d\sqrt{5}$, where d is a whole number.

In the Equation Editor, we enter the value of C by typing:

$2\sqrt{45} \triangleright \triangleright + 3\sqrt{12} \triangleright \triangleright - \sqrt{20} \triangleright \triangleright - 6\sqrt{3}$

$\triangleright \triangleright \triangleright$ selects $-6\sqrt{3}$ and

\triangleleft selects $-\sqrt{20}$

$\nabla \nabla$ selects $-\sqrt{20}$

Invoke the FACTOR command, which is found in the ALGB menu.

Pressing ENTER then causes 20 to be factored into $2^2 \cdot 5$.

\triangle selects $\sqrt{2^2 \cdot 5}$ and ENTER returns $2\sqrt{5}$

\triangleright selects $-2\sqrt{5}$

SHIFT \triangleleft exchanges $3\sqrt{12}$ with $-2\sqrt{5}$

\triangleleft selects $2\sqrt{45}$

$\nabla \triangleright \nabla$ selects 45.

Invoke the FACTOR command, which is found in the ALGB menu.

Pressing ENTER then causes 45 to be factored into $3^2 \cdot 5$.

\triangle selects $\sqrt{3^2 \cdot 5}$ and ENTER replaces $\sqrt{3^2 \cdot 5}$ with $3\sqrt{5}$,

\triangle selects $2 \cdot 3\sqrt{5}$,

SHIFT \triangleright selects $2 \cdot 3\sqrt{5}$ and $-2\sqrt{5}$, and then ENTER completes the operation and gives:

$4\sqrt{5}$.

It remains to transform $3\sqrt{12}$ and to see that this term is combined with $-6\sqrt{3}$.

The result is:

$C = 4\sqrt{5}$

6.2.3 Exercise 3

Given the expression $D = (3x - 1)^2 - 81$.

1. Expand and reduce D .

2. Factor D .

3. Solve the equation: $(3x - 10)(3x + 8) = 0$.

4. Evaluate D for $x = 5$.

1. First, write D into the Equation Editor.

typing:

$3 X - 1 \triangleright \triangleright x^y 2 \triangleright - 81$

Select $(3X - 1)^2$ ($\triangleright \triangleleft$), then press ENTER to expand the expression.

This gives:

$9X^2 - 6X + 1 - 81$

To do the expansion step by step, press MEMORY (SHIFT ,) to recall the previous expression, then invoke POWEXPAND for $(3.X - 1)^2$, then execute

DISTRIB a couple of times on the result to obtain:

$$9X^2 - 6X + 1$$

Δ selects the entire expression, then pressing ENTER reduces it to:

$$9X^2 - 6X - 80$$

2. Search for D in the history (SYMB key), then highlight D and press ENTER to confirm your choice.

Invoke FACTOR to obtain:

$$(3X + 8)(3X - 10)$$

You can also retrieve the original expression, select 81 to factor it into 3^4 , and work out the difference between the two squares...

3. Invoke the SOLVEX command, then press ENTER to obtain:

$$\left(X = -\frac{8}{3}\right) \text{ OR } \left(X = \frac{10}{3}\right)$$

4. Search for D in the history (SYMB key), then highlight D and press ENTER to confirm your choice.

Invoke SUBST, then complete the second argument:

$$X = -5$$

Then press $\triangleright \triangleright \triangleright$ and ENTER to obtain :

$$(3 \cdot (-5) - 1)^2 - 81$$

Pressing ENTER once more yields the result:

$$175$$

therefore, $D = 175$ when $X = -5$.

6.2.4 Exercise 4

A baker produces two different assortments of biscuits and macaroons.

A packet of the first assortment contains 17 biscuits and 20 macaroons.

A packet of the second assortment contains 10 biscuits and 25 macaroons.

Both packets cost 90 c.

Calculate the price of one biscuit, and the price of one macaroon.

Let x be the price in cents of one biscuit, and y the price of one macaroon.

The problem is then to solve:

$$\begin{cases} 17x + 20y = 90 \\ 10x + 25y = 90 \end{cases}$$

In the Equation Editor, type:

LINSOLVE(17 · X + 20 · Y - 90 AND 10 · X + 25 · Y - 90 , X AND Y)

If you are working in Step by step mode, this produces:

$$L_2 = 17L_2 - 10L_1$$

$$\begin{bmatrix} 17 & 20 & -90 \\ 10 & 25 & -90 \end{bmatrix}$$

$$L_1 = 45L_1 - 4L_2$$

$$\begin{bmatrix} 17 & 20 & -90 \\ 0 & 225 & -630 \end{bmatrix}$$

$$\text{Reduction Result}$$

$$\begin{bmatrix} 765 & 0 & -90 \\ 0 & 225 & -630 \end{bmatrix}$$

Pressing ENTER then produces the result:

$$(X = 2) \text{ AND } (Y = \frac{14}{5})$$

If you select $\frac{14}{5}$, and press the NUM key (or invoke XNUM), you get:

$$(X = 2) \text{ AND } (Y = 2.8)$$

In other words, the price of one biscuit is 2 c, and the price of one macaroon is 2.8 c.

NOTE: If the calculator has gone into Approx mode, put it back into Exact mode using CFG.

6.2.5 Exercise 5

Say that A and B are points having the coordinates:

$A(-1 ; 3)$ and $B(-3 ; -1)$, where the unit of measure is the centimetre.

1/ Find exact length of AB in centimetres.

2/ Determine the equation of the line AB .

First Method

Type:

STORE ((-1 , 3) , A)

accept the change to Complex mode, if necessary.

STORE ((-3 , -1) , B)

The vector \overrightarrow{AB} has coordinates $B - A$.

1/ Typing:

$$\text{ABS}(B - A)$$

gives:

$$2\sqrt{5}$$

2/ Apply the DROITE command, on the MATH key's Complex menu:

$$\text{DROITE}(A, B)$$

gives:

$$Y = 2 \cdot X + 5$$

Pressing ENTER then produces the result:

$$Y = 2 \cdot X + 5$$

Or second Method

1/ Typing:

$$(-3, -1) - (-1, 3)$$

gives:

$$-2 - 4 \cdot i$$

Apply the ABS command to get:

$$\text{ABS}(-2 - 4i)$$

gives:

$$2\sqrt{5}$$

2/ Typing:

$$\text{DROITE}((-1, 3), (-3, -1))$$

gives:

$$Y = 2 \cdot (X - -1) + 3$$

Pressing ENTER then produces the result:

$$Y = 2 \cdot X + 5$$

6.3 Exercises on Bac

6.3.1 Exercise 1

Terms of exercise 1

Let m be a point on the circle C of center O and radius 1. Consider the image M of m defined on their affixes by the transformation $F : z \mapsto \frac{1}{2}z^2 - z$. When m moves on the circle C , M will move on a curve Γ . The object of this exercise is to study and plot Γ .

1. Let $t \in [-\pi, \pi]$ and m the point on C of affix $z = e^{i \cdot t}$. Find the coordinates of M in terms of t .
2. Compare $x(-t)$ with $x(t)$ and $y(-t)$ with $y(t)$. Show that Γ has a symmetry center.
3. Compute $x'(t)$ and find the variations of x over $[0, \pi]$.
4. Same question for y .
5. Show the variations of x and y in the same table.
6. Put the points of Γ corresponding to $t = 0, \pi/3, 2\pi/3, \pi$ and draw the tangent to Γ at these points. Draw the part of Γ corresponding to $t \in [0, \pi]$. Draw Γ .

Correction

The object of this exercise is to trace the curve Γ described by M and given by $\frac{1}{2} \cdot z^2 - z$, when m , given by z describes a circle C with center O and having a radius of 1. Let t be a real number in $[-\pi, \pi]$, and let m be the point of C given by $z = e^{i \cdot t}$.

1. Calculating the coordinates of M :

First, make t the VX variable (SHIFT SYMB (SETUP) keys and then
 Indep var : t)

Then, enter the expression $\frac{1}{2} \cdot z^2 - z$ into the Equation Editor.

In the Equation Editor, type:

ALPHA Z x^y 2 > ÷ 2 > - ALPHA Z > >

The expression $\frac{Z^2}{2} - Z$ is selected.

Since $z = e^{i \cdot t}$, invoke SUBST and complete the second argument by typing:

SUBST($\frac{Z^2}{2} - Z, Z = \text{EXP}(i \times t)$)

which gives:

$$\frac{\text{EXP}(i \cdot t)^2}{2} - \text{EXP}(i \cdot t)$$

Then, you linearise the expression with the command:

LIN

The result (after accepting the switch to Complex mode) is:

$$\frac{1}{2} \cdot \text{EXP}(2 \cdot i \cdot t) + -1 \cdot \text{EXP}(i \cdot t)$$

- Now store the result with the STORE command by typing:

$$\text{STORE}(\frac{1}{2} \cdot \text{EXP}(2 \cdot i \cdot t) + -1 \cdot \text{EXP}(i \cdot t), M)$$

then pressing ENTER.

To calculate the real part of this expression, enter the command:

RE

which returns:

$$\frac{\text{COS}(t \cdot 2) - 2 \cdot \text{COS}(t)}{2}$$

We suppose that VX contains t (SHIFT HOME (MODES) then Indep var : t).

Then, define the function $x(t)$ by invoking DEF:

NOTE: You will need to type = X(t), then exchange X(t) with the expression $\frac{\text{COS}(t \cdot 2) - 2 \cdot \text{COS}(t)}{2}$. To do this, highlight X(t) with \triangleright and type SHIFT \triangleleft . Highlight the entire expression and select the DEF command:

$$\text{DEF}(X(t) = \frac{\text{COS}(t \cdot 2) - 2 \cdot \text{COS}(t)}{2})$$

Then press ENTER.

To calculate the imaginary part of this expression, type the command:

IM(M)

which returns:

$$\frac{\text{SIN}(t \cdot 2) - 2 \cdot \text{SIN}(t)}{2}$$

Then, define the function $y(t)$ (in the same way as you defined $x(t)$):

$$\text{DEF}(Y(t) = \frac{\text{SIN}(t \cdot 2) - 2 \cdot \text{SIN}(t)}{2})$$

Then press ENTER.

2. To find an axis of symmetry for Γ , calculate $x(-t)$ and $y(-t)$ by typing:

X(-t) ENTER

which produces:

$$\frac{\cos(t \cdot 2) - 2 \cdot \cos(t)}{2}$$

In other words: $x(-t) = x(t)$

Then type:

Y(-t) ENTER

which produces:

$$\frac{-\sin(t \cdot 2) + 2 \cdot \sin(t)}{2}$$

In other words: $y(-t) = -y(t)$

If $M_1(x(t), y(t))$ is part of Γ , then $M_2(x(-t), y(-t))$ is also part of Γ .

Since M_1 and M_2 are symmetrical with respect to the x -axis, we can deduce that the x -axis is an axis of symmetry for Γ .

3. Calculate $x'(t)$:

Typing:

DERVX(X(t))

returns:

$$\frac{2 \cdot (-2 \cdot \sin(t \cdot 2) - 2 \cdot (-\sin(t)))}{4}$$

or, after simplification (ENTER), the response is :

$$-\sin(t) \cdot (2 \cdot \cos(t) - 1)$$

You can now define the function $x'(t)$ by invoking DEF:

Note: You will need to type = X1 (t), then exchange X1 (t) with the expression $-\sin(t) \cdot (2 \cdot \cos(t) - 1)$. To do this, highlight X1 (t) (\triangleright) and type SHIFT \triangleleft . This returns:

$$\text{DEF}(X1(t) = -\sin(t) \cdot (2 \cdot \cos(t) - 1))$$

Then press ENTER.

4. Calculate $y'(t)$:

Typing:

DERVX(Y(t))

returns:

$$\frac{2 \cdot (2 \cdot \cos(t \cdot 2) - 2 \cdot \cos(t))}{4}$$

and, after simplification (ENTER), the response is :

$$2 \cdot \cos(t)^2 - \cos(t) - 1$$

Invoke FACTOR to factor the expression:

$$\text{FACTOR}(2 \cdot \cos(t)^2 - \cos(t) - 1)$$

then press ENTER, the response is:

$$(\cos(t) - 1) \cdot (2 \cdot \cos(t) + 1)$$

You can now define the function $y'(t)$ (in the same way as you defined $x'(t)$):

$$\text{DEF}(Y1(t) = (\cos(t) - 1) \cdot (2 \cdot \cos(t) + 1))$$

5. Variations of $x(t)$ and $y(t)$

For this, you trace $x(t)$ and $y(t)$ on the same graph.

We suppose that t is contained in VX (SHIFT SYMB (SETUP))

Type X (t) in the Equation Editor and press ENTER.

Then press the PLOT key.

Select Function using the dialog box, and select F1 as the destination.

Then, do the same thing with Y (t), making F2 the destination.

To graph the functions: quit CAS (using the ON (CANCEL) button), choose the Function aplet, and check F1 and F2. You will have to set the window's parameters (SHIFT PLOT), then press PLOT to see the graphs.

Back in the Equation Editor (press the HOME key, then CAS on the menu bar), we can get exact outputs from the curve Γ :

- Values of $x(t)$ and $y(t)$

Find the values of $x(t)$ and $y(t)$ for $t = 0, \frac{\pi}{3}, \frac{2\pi}{3}, \pi$ by typing in succession (ENTER is pressed twice in most cases for further simplification):

$$X(0) \text{ ENTER}$$

Response: $\frac{-1}{2}$ $X\left(\frac{\pi}{3}\right)$ ENTER ENTERResponse: $\frac{-3}{4}$ $X\left(2 \times \frac{\pi}{3}\right)$ ENTER ENTERResponse: $\frac{1}{4}$ $X(\pi)$ ENTER ENTERResponse: $\frac{3}{2}$ $Y(0)$ ENTER

Response: 0

 $Y\left(\frac{\pi}{3}\right)$ ENTER ENTERResponse: $\frac{-\sqrt{3}}{4}$ $Y\left(2 \times \frac{\pi}{3}\right)$ ENTER ENTERResponse: $\frac{-3 \cdot \sqrt{3}}{4}$ $Y(\pi)$ ENTER ENTER

Response: 0

- Slope of the tangents ($m = \frac{y'(t)}{x'(t)}$)

Find the values of $\frac{y'(t)}{x'(t)}$ for $t = 0, \frac{\pi}{3}, \frac{2 \cdot \pi}{3}, \pi$ by typing in succession:

$$\text{LIMIT}\left(\frac{Y1(t)}{X1(t)}, t = 0\right) \text{ ENTER}$$

Response: 0

$$\text{LIMIT}\left(\frac{Y1(t)}{X1(t)}, t = \pi \div 3\right) \text{ ENTER}$$

Response (answer YES when asked UNSIGNED INF. SOLVE?):

 ∞

$$\text{LIMIT}\left(\frac{Y1(t)}{X1(t)}, t = 2 \times \pi \div 3\right) \text{ ENTER}$$

Response: 0

$$\text{LIMIT}\left(\frac{Y1(t)}{X1(t)}, t = \pi\right) \text{ ENTER}$$

Response (answer YES when asked UNSIGNED INF. SOLVE?):

(

Here, then, are the variations of $x(t)$ and $y(t)$:

t	0	$\frac{\pi}{3}$		$\frac{2\pi}{3}$		π
$x'(t)$	0	-	0	+	$\sqrt{3}$	+ 0
$x(t)$	$\frac{-1}{2}$	↓	$\frac{-3}{4}$	↑	$\frac{1}{4}$	↑ $\frac{3}{2}$
$y(t)$	0	↓	$-\frac{\sqrt{3}}{4}$	↓	$-\frac{3\sqrt{3}}{4}$	↑ 0
$y'(t)$	0	-	-1	-	0	+ 2
m	0		∞		0	∞

- The curve Γ :

Now graph the parametric curve.

In the Equation Editor, type $X(t) + i \times Y(t)$, then press ENTER.

Then press:

PLOT, and select Parametric using the dialog box, specifying X1, Y1 as the destination.

To make the graph of the curve Γ : quit CAS (using the HOME button), then choose the Parametric applet. Check X1(T), Y1(T), and choose default values in PLOT SETUP.

6.3.2 Exercise 2 (specialty)

Terms of exercise 2

For n a strictly positive integer, we define

$$a_n = 4 \times 10^n - 1, \quad b_n = 2 \times 10^n - 1, \quad c_n = 2 \times 10^n + 1$$

1. a/ Compute $a_1, b_1, c_1, a_2, b_2, c_2, a_3, b_3$ and c_3 .
 b/ Find how many digits the decimal representation of a_n and c_n have. Show that a_n and c_n are divisible by 3.
 c/ Using a list of prime smaller than 100, show that b_3 is prime.
 d/ Show that for every integer $n > 0$, $b_n \times c_n = a_{2n}$. Deduce the prime factor decomposition of a_6 .
 e/ Show that $\text{GCD}(b_n, c_n) = \text{GCD}(c_n, 2)$. Deduce that b_n and c_n are prime together.

2. Consider the equation:

$$(1) \quad b_3x + c_3y = 1$$

where the integers x and y are the unknown.

a/ Show that (1) has at least one solution

b/ Apply Euclide's algorithm to c_3 and b_3 and find a solution to (1).

c/ Find all solutions of (1)

Correction

Define the following for a natural whole number:

$$a_n = 4 \times 10^n - 1, \quad b_n = 2 \times 10^n - 1 \quad \text{and} \quad c_n = 2 \times 10^n + 1$$

Begin by typing:

$$\text{DEF(A(N))} = 4 \cdot 10^N - 1$$

$$\text{DEF(B(N))} = 2 \cdot 10^N - 1$$

$$\text{DEF(C(N))} = 2 \cdot 10^N + 1$$

1. Now do the following:

Calculate

- a) Simply type (in succession):

A(1)

Result: 39

B(1)

Result: 19

C(1)

Result: 21

A(2)

Result: 399

B(2)

Result: 199

C(2)

Result: 201

A(3)

Result: 3999

B(3)

Result: 1999

C(3)

C(3)

Result: 2001

- b) Number of digits, and divisibility

In this, the calculator is used only to try out different values of n ...

Show that the whole numbers k such that:

$$10^n \leq k < 10^{n+1}$$

have $(n + 1)$ digits in decimal notation.

We have:

$$10^n < 3 \cdot 10^n < a_n < 4 \cdot 10^n < 10^{n+1}$$

$$10^n < b_n < 2 \cdot 10^n < 10^{n+1}$$

$$10^n < 2 \cdot 10^n < c_n < 3 \cdot 10^n < 10^{n+1}$$

so a_n, b_n, c_n have $(n + 1)$ digits in decimal notation.

Moreover, $d_n = 10^n - 1$ is divisible by 9, since its decimal notation can only end in 9.

We also have:

$$a_n = 3 \cdot 10^n + d_n$$

and

$$c_n = 3 \cdot 10^n - d_n$$

so a_n and c_n are both divisible by 3.

- c) b_3 is a prime number

Typing:

$$\text{ISPRIME?}(B(3))$$

gives:

1.

which means true.

To prove that $b_3 = 1999$ is a prime number, it is necessary to

show that 1999 is not divisible by any of the prime numbers less than or equal to $\sqrt{1999}$.

As $1999 < 2025 = 45^2$, that means testing the divisibility of 1999 by $n = 2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41$.

1999 is not divisible by any of these numbers, so we can conclude that 1999 is prime.

- d) $a_{2n} = b_n \times c_n$

Typing:

$$B(N) \cdot C(N)$$

produces:

$$4 \cdot (10^N)^2 - 1$$

after applying the EXP2POW command. This is the value a_{2n} .

Decomposition of a_6 into its prime factors:

Typing:

$$\text{FACTOR}(A(6))$$

yields:

$$3 \cdot 23 \cdot 29 \cdot 1999$$

- e) b_n and c_n are relatively prime.

In this part, the calculator is useful only for trying out different values of n ...

To show that c_n and b_n are relatively prime, it is enough to remark that

$$c_n = b_n + 2$$

That means that the common divisors of c_n and b_n are the common divisors of b_n and 2, as well as the common divisors of c_n and 2. b_n and 2 are relatively prime because b_n is a prime number different from 2. So:

$$PGCD(c_n, b_n) = PGCD(c_n, 2) = PGCD(b_n, 2) = 1$$

2. Given the equation:

$$b_3 \cdot x + c_3 \cdot y = 1$$

- a) It has at least one solution, as it is actually a form of Bézout's Identity.

In effect, Bézout's Theorem says:

If a and b are relatively prime, there exist x and y such that:

$$a \cdot x + b \cdot y = 1$$

Therefore, the equation:

$$b_3 \cdot x + c_3 \cdot y = 1$$

has at least one solution.

- b) Typing:

$$\text{IEGCD}(B(3), C(3))$$

returns:

$$1000 \text{ AND } -999 = 1$$

In other words:

$$b_3 \times 1000 + c_3 \times (-999) = 1$$

so we have a particular solution:

$$x = 1000, y = -999.$$

The rest can be done on paper:

$$c_3 = b_3 + 2 \text{ and } b_3 = 999 \times 2 + 1$$

so, $b_3 = 999 \times (c_3 - b_3) + 1$, or:

$$b_3 \times 1000 + c_3 \times (-999) = 1$$

- c) In this part, the calculator is not used for finding the general solution.

We have:

$$b_3 \cdot x + c_3 \cdot y = 1$$

and

$$b_3 \times 1000 + c_3 \times (-999) = 1$$

so, by subtraction:

$$b_3 \cdot (x - 1000) + c_3 \cdot (y + 999) = 0$$

or:

$$b_3 \cdot (x - 1000) = -c_3 \cdot (y + 999)$$

According to Gauss's Theorem, c_3 is prime with b_3 , so c_3 is a divisor of $(x - 1000)$.

So, there exists $k \in Z$ such that:

$$(x - 1000) = k \times c_3$$

and

$$-(y + 999) = k \times b_3$$

Solving for x and y , we get:

$$x = 1000 + k \times c_3$$

and

$$y = -999 - k \times b_3$$

for $k \in Z$

This gives us:

$$b_3 \cdot x + c_3 \cdot y = b_3 \times 1000 + c_3 \times (-999) = 1$$

The general solution for all $k \in Z$ is therefore:

$$x = 1000 + k \times c_3$$

$$y = -999 - k \times b_3$$

6.3.3 Exercise 3 (non-specialty)

Terms of exercise 2

For n an integer, define:

$$u_n = \int_0^2 \frac{2x+3}{x+2} e^{\frac{x}{n}} dx$$

1. a/ Define g over $[0, 2]$ by $g(x) = \frac{2x+3}{x+2}$.

Find the variations of g over $[0, 2]$. Show that for every real x in

$$[0, 2], \frac{3}{2} \leq g(x) \leq \frac{7}{4}.$$

- b) Show that for every real x in $[0, 2]$: $\frac{3}{2}e^{\frac{x}{n}} \leq g(x)e^{\frac{x}{n}} \leq \frac{7}{4}e^{\frac{x}{n}}$.

c) After integration, show that :

$$\frac{3}{2}n(e^{\frac{2}{n}} - 1) \leq u_n \leq \frac{7}{4}n(e^{\frac{2}{n}} - 1)$$

d) Using $\lim_{h \rightarrow 0} \frac{e^h - 1}{h} = 1$.

show that if u_n has a limit L as $n \rightarrow \infty$, then $3 \leq L \leq \frac{7}{2}$.

2. a) Show that for every x in $[0, 2]$: $\frac{2x+3}{x+2} = 2 - \frac{1}{x+2}$.

Find the value of $I = \int_0^2 \frac{2x+3}{x+2} dx$.

b) Show that for every x in $[0, 2]$: $1 \leq e^{\frac{x}{n}} \leq e^{\frac{2}{n}}$.

Deduce that $1 \leq u_n \leq e^{\frac{2}{n}} I$.

c) Show that (u_n) is convergent and find its limit L .

Correction

Before you begin, check that the calculator is in exact real mode with X as the current variable; if not, select Default cFg in CFG.

Given:

$$u_n = \int_0^2 \frac{2x+3}{x+2} e^{\frac{x}{n}} dx$$

1. Do the following:

2. • Variation of $g(x) = \frac{2x+3}{x+2}$ for $x \in [0, 2]$

Typing:

$$\text{DEF}(G(X) = \frac{2X+3}{X+2})$$

then:

$$\text{TABVAR}(G(X))$$

yields:

$$\begin{array}{cccccc} -\infty & + & -2 & + & +\infty & X \\ 2 & \uparrow & \infty & \uparrow & 2 & F \end{array}$$

The first line gives the sign of $g'(x)$ according to x , and the second line the variations of $g(x)$. Note that for TABVAR the function is always called F.

We can deduce, then, that $g(x)$ increases over $[0, 2]$.

If you were in Step by step mode (for this, choose Step/step and then OK on the CFG menu bar), you would obtain (although the function is labelled F):

$$F =: \frac{2 \cdot X + 3}{X + 2}$$

Press ENTER :

$$F' := \frac{2 \cdot (X + 2) - (2 \cdot X + 3)}{\text{SQ}(X + 2)}$$

Using the down-arrow ∇ , scroll down the screen:

$$\rightarrow \frac{1}{(X + 2)^2}$$

Then press ENTER to obtain the table of variations.

If you are not in Step by step mode, you can also request the calculation of the derivative by typing:

$$\text{DERVX}(G(X))$$

which produces the preceding calculation.

To calculate $g(0)$ and $g(2)$, type:

$$G(0)$$

Response: $\frac{3}{2}$

$$G(2)$$

Response: $\frac{7}{4}$

whence the inequality:

$$\frac{3}{2} \leq g(x) \leq \frac{7}{4} \text{ for } x \in [0, 2]$$

- b) The calculator is not needed here... simply stating that :

$$e^{\frac{x}{n}} \geq 0 \text{ pour } x \in [0, 2]$$

is sufficient to show that, for $x \in [0, 2]$, we have:

$$\frac{3}{2}e^{\frac{x}{n}} \leq g(x)e^{\frac{x}{n}} \leq \frac{7}{4}e^{\frac{x}{n}}$$

- c) To integrate the preceding inequality, type:

$$\int_0^2 e^{\frac{x}{n}} dX$$

which produces:

$$N \cdot e^{\frac{2}{N}} - N$$

We can then deduce that:

$$\frac{3}{2}(ne^{\frac{2}{n}} - n) \leq u_n \leq \frac{7}{4}(ne^{\frac{2}{n}} - n)$$

To justify the preceding calculation, it is necessary to assume that $n \cdot e^{\frac{x}{n}}$ is a primitive of $e^{\frac{x}{n}}$.

If you are not sure, you can type:

$$\text{INTVX}(\text{EXP}(\frac{X}{N}))$$

The simplified result is: $N \cdot e^{\frac{X}{N}}$

- d) To find the limit of $(ne^{\frac{2}{n}} - n)$ when $n \rightarrow +\infty$:

$$\text{LIMIT}(N \cdot \text{EXP}(\frac{2}{N}) - N, N = +\infty)$$

The result is:

$$2$$

NOTE :

The variable VX is set equal to N; use the SHIFT SYMB (SETUP) keys to reset VX to X.

To check the result, we can say that:

$$\lim_{x \rightarrow 0} \frac{e^x - 1}{x} = 1$$

and that therefore:

$$\lim_{n \rightarrow +\infty} \frac{e^{\frac{2}{n}} - 1}{\frac{2}{n}} = 1$$

or, simplifying:

$$\lim_{n \rightarrow +\infty} (e^{\frac{2}{n}} - 1) \cdot n = 2$$

If the limit L of u_n exists as n tends to $+\infty$ in the inequalities in 1b), we get:

$$\frac{3}{2} \cdot 2 \leq L \leq \frac{7}{4} \cdot 2$$

3. • a) Do the following:

$$g(x) = 2 - \frac{1}{x+2} \text{ and the calculation of } I = \int_0^2 g(x) dx$$

Typing:

$$\text{PROPFAC}(G(X))$$

returns:

$$2 - \frac{1}{X + 2}$$

To calculate the integral I , type:

$$\int_0^2 G(X) dX$$

This produces:

$$-(\text{LN}(2) - 4)$$

Working by hand: $2x + 3 = 2(x + 2) - 1$, so :

$$g(x) = 2 - \frac{1}{x + 2}$$

Then, integrating term by term between 0 and 2 produces:

$$\int_0^2 g(x) dx = [2x - \ln(x + 2)]_{x=0}^{x=2}$$

that is, since $\ln 4 = 2 \ln 2$:

$$\int_0^2 g(x) dx = 4 - \ln 2$$

- b) The calculator is not needed here... simply stating that $e^{\frac{x}{n}}$ increases for $x \in [0, 2]$ is sufficient to yield the inequality:

$$1 \leq e^{\frac{x}{n}} \leq e^{\frac{2}{n}}$$

Then, through multiplication, $g(x)$ being positive over $[0, 2]$, we get:

$$g(x) \leq g(x)e^{\frac{x}{n}} \leq g(x)e^{\frac{2}{n}}$$

and then, integrating:

$$I \leq u_n \leq e^{\frac{2}{n}} I$$

- c) Convergence of u_n
Find the limit of $e^{\frac{2}{n}}$ when $n \rightarrow +\infty$:

$$\text{LIMIT}(\text{EXP}(\frac{2}{N}), N = +\infty)$$

This returns:

$$1$$

In effect, $\frac{2}{n}$ tends to 0 as n tends to $+\infty$, so $e^{\frac{2}{n}}$ tends to $e^0 = 1$ as n tends to $+\infty$.

As n tends to $+\infty$, u_n is the portion between I and a quantity that tends to I (cf. inequalities 2b)).

So then, u_n converges, and its limit is I .

We have therefore shown that:

$$L = I = 4 - \ln 2$$

6.4 Conclusion

You can see that, by that making good use of the HP40GS calculator, you can use it for a large proportion of the calculations...

It is necessary to note, however, that arithmetic does require more reasoning. Here, the calculator enables you to check your work...

Chapter 7

Programming

7.1 Implementation

7.1.1 How to edit and save a program

>From the HOME screen, to get access to the catalogue of programs, press the SHIFT 1 (PROGRAM) keys.

A screen then appears containing the list of available programs and a menu bar (EDIT NEW SEND RECV RUN).

To create a new program, press F2 (NEW).

You are asked for the name of the program.

NOTE: If you are not in Alpha mode, press F4 (A . . Z) to go into it!

Type its name, then press F6 (OK).

You enter your program, and your work is automatically saved when you leave the editor by pressing HOME or SHIFT 1 (PROGRAM).

7.1.2 How to correct a program

If the syntax is invalid, the calculator displays:

Invalid Syntax Edit program? Respond by pressing F6 (YES).

The calculator automatically positions the cursor where the compiler has detected the error. All you need do then is correct it!

7.1.3 How to run a program

To run a program, open the catalogue of programs by pressing `SHIFT 1` (`PROGRAM`).

A screen is then displayed containing the list of available programs and the menu bar `EDIT NEW SEND RECV RUN`.

Highlight the name of the program and press `F6` (`RUN`).

7.1.4 How to modify a program

To modify a program (and overwrite the old one), open the catalogue of programs by pressing `SHIFT 1` (`PROGRAM`). A screen is then displayed containing the list of available programs and the menu bar `EDIT NEW SEND RECV RUN`.

Highlight the name of the program and press `F1` (`EDIT`).

If you want to save and edit the old program under a new name, then:

- Open the catalogue of programs (`SHIFT 1` (`PROGRAM`))
- Press `F2` (`NEW`) and type the new name, then press `F6` (`OK`). The Editor opens.
- Press `VAR`S, then the letter `P` to highlight `Program`.
- Using the arrows, highlight the name of the old program, then press `F4` (`VALUE`) (to select `VALUE` on the menu bar), then `F6` (`OK`).

This copies the text of the old program into the editor.

7.2 Comments

It is important to comment your programs.

In writing algorithms, a comment commonly starts with `//` and ends at the end of the line.

In the HP40GS, a comment starts with `@` and ends at the end of the line or at the next `@`, whichever comes first.

NOTE:

Do not forget to put a space after the `@`.

The character `@` is obtained by typing `SHIFT VAR` (`CHARS`), then highlighting the character and choosing `ECHO` on the menu bar (or `ECHO1` `ECHO1` `OK` to have two characters `@`).

7.2.1 SHIFT 1 (PROGRAM) keys

This key combination, if pressed in HOME, displays the screen :

```
PROGRAM CATALOG
```

It shows:

- a list of the programs that you've written, and
- a menu bar containing the commands:

```
EDIT NEW RUN SEND RECV.
```

EDIT enables you to edit the highlighted program,

NEW enables you to create a new program,

RUN enables you to run the highlighted program (cf 7.1),

SEND and RECV are functions that enable your calculator to talk to your computer or another calculator.

For example:

If you press SEND on the menu bar, it asks:

```
HP39/40 (Wire) or Disk Drive
```

You highlight HP39/40 (Wire) to send a program to another HP40GS, or you highlight Disk Drive to send a program to a computer.

Then press OK on the menu bar.

For example, here is how you connect a Linux computer to the HP40GS so as to use the program C-Kermit version 7 (which you can find at the URL www.columbia.edu/kermit, or which you can download via anonymous ftp from the site kermit.columbia.edu):

Connect the calculator to a data transfer cable.

On the computer, type:

```
kermit
```

```
set line /dev/ttyS0 (or S1... depending on the number of your
serial port)
```

```
set speed 9600
```

```
set carrier-watch off
```

```
serv
```

On the HP40GS:

Highlight the program called NAME, then press SEND on the menu bar and highlight Disk drive. Then press OK on the menu bar, and the program called NAME on the HP40GS is copied into your computer.

OR:

Press RECV on the menu bar and highlight Disk drive. Then press OK on the menu bar, and the calculator displays a list of the HP40GS programs on your computer. (Naturally, you have to have already created a directory on your computer where the HP40GS programs are stored).

You then highlight GCD, and the program called GCD on your computer is copied into the HP40GS.

For Windows users, the connectivity program is found on the URL :
<http://www.hp.com/calculators/france>.

To find out more about the use of Kermit with HP calculators, visit the URL: <http://www.columbia.edu/kermit/hp48.html>

7.3 Variables

7.3.1 Their names

These are the containers in which you can store values, numbers, expressions, and other objects.

With the HP40GS, only the 26 letters of the alphabet, and the Greek letter θ are available for storing real numbers.

7.3.2 The concept of local variables

Local variables are used to protect user variables from modifications after a program call. This concept does not exist natively on the HP40GS (only global variables are supported).

For advanced users, it is however possible to mimic the local variables mechanism using the CAS history and PUSH and POP. Say for example that our program use the variables A, B, C. At the beginning we can PUSH A then PUSH B then PUSH C. This will save the values of A, B and C in the CAS history. At the end of the program we restore the value of A, B and C using POP and store the value in C, B and A, it must be done **in reverse order** since the CAS history is a stack container (LIFO=Last In First Out).

7.3.3 The concept of parameters

It is not possible for programs written on the HP40GS to natively pass parameters. Hence you cannot use the HP40GS programming language to write functions having parameters.

For advanced users, it is possible to mimic parameters using the CAS history. The caller PUSH a list of parameters in the CAS history, then runs the program, that will have as first instruction a POP instruction. This mechanism does not work in conjunction with the local variables turnaround above. If you want to combine both concepts, you can choose a list vari-

able for parameter transmission and return value, and use local variables as described above.

7.4 User entry

7.4.1 Algorithmic syntax

So that the user is able to enter a value for the variable A during the course of the program's execution, one writes in algorithms:

```
input A
```

And for entering values into A and B, one writes:

```
input A,B
```

7.4.2 HP40GS syntax

```
INPUT A;"TITRE";"A=";;0:
```

If it annoys you to have to type all the punctuation in the INPUT command, you might prefer to use the PROMPT or PROMPTSTO command instead (thanks to Jean Yves!).

PROMPTSTO(A) or PROMPT A opens a window that asks you to enter the value of A.

In the remainder of this guide, programs that were written before the existence of PROMPT use the subroutine IN, which enables you to enter two values into A and B.

7.5 Output

7.5.1 Algorithmic syntax

In algorithms one writes:

```
display "A=",A
```

7.5.2 HP40GS syntax

```
DISP 3;"A="A: 3 represents the number of the line whereA will be displayed
```

Or:

```
MSGBOX "A="A:
```

7.6 The sequence of statements, or "action"

An action is a sequence of one or more statements.

7.6.1 Algorithmic syntax

In algorithmic language, you use the space or the linebreak to terminate the statement.

7.6.2 HP40GS syntax

: indicates the end of the statement.

7.7 The assignment statement

A value or an expression is stored in a variable by means of an assignment.

7.7.1 Algorithmic syntax

In algorithms, one writes (for example):

```
2*A->B
```

to store 2*A in B.

7.7.2 HP40GS syntax

The arrow is obtained by means of the ST0> key on the menu bar.

One types (for example):

```
2*A-> B that mean :
```

press 2*A then press ST0> on the menu bar then press B.

7.8 Conditional statements

7.8.1 Algorithmic syntax

Conditional statements have two syntax :

```
If condition then
```

```
action
```

```
endif
```

and also :

```
If condition then
action1 else
action2
endif
```

Example:

```
If A = 10 ou A < B then
B-A->B else
A-B->A
endif
```

7.8.2 HP40GS syntax

```
IF condition THEN
action :
```

```
END:
```

and also :

```
IF condition THEN
action1 : ELSE
action2:
```

```
END:
```

NOTE au == to denote the condition of equality.

Example :

```
IF A==10 OR A<B THEN
B-A->B : ELSE
A-B-> A
END:
```

7.9 FOR statement

7.9.1 Algorithmic syntax

```
For I = A to B do action endfor
```

and also :

```
For I = A to B (step P) do action endfor
```

7.9.2 HP40GS syntax

```
FOR I = A TO B STEP 1 ; action : END:
```

and also :

```
FOR I = A TO B STEP P ; action : END:
```

7.10 WHILE statement

7.10.1 Algorithmic syntax

While condition do action endwhile

7.10.2 HP40GS syntax

WHILE *condition* REPEAT *action* : END:

7.11 Boolean Expressions

A condition is a function that has a Boolean as a value, that is, it is either TRUE or FALSE.

7.11.1 Algorithmic syntax

To express a condition, simply use the operators:

= > > ≤ ≥ ≠

7.11.2 HP40GS syntax

NOTE: for the HP40GS calculator, equality is denoted by:

==

Otherwise, the operators are the same.

7.12 Logical Operators

7.12.1 Algorithmic syntax

To express complex conditions, use the logical operators:

or and not

7.12.2 HP40GS syntax

Or, and, not are represented on the HP40GS as OR, AND, NOT

7.13 Lists

7.13.1 Algorithmic syntax

In algorithms, you use { } to define a list.

For example, {} designates a void list, and {1, 2, 3} is a list with three elements.

The + is used to concatenate two lists, or a list and an element, or an element and a list:

{1, 2, 3} -> TAB

TAB + 4 -> TAB (TAB now designates {1, 2, 3, 4})

TAB[2] indicates the second element of TAB, here 2.

7.13.2 HP40GS syntax

The list variables have the following names: L0, L1, L2, ... L9.

You use { } to delimit a list.

For example, {1, 2, 3} is a list with three elements.

But {} does not designate a void list; to clear a list you must use the command CLRVAR, e.g.:

CLRVAR L0

will initialise the list L0 as void. You can also use the command SUB which extract a sub-list (if L1={1, 2, 3, 4} then SUB L2;L1;2;3 make the list L2={2, 3}) and so, SUB L1;L1;2;1 to initialise the list L1 as void.

Following are some useful commands:

MAKELIST(I*I, I, 1, 10, 2) designates a list of the squares of the first five odd whole numbers (2 indicates the step of I).

L1(I) designates the Ith element of the list.

CONCAT (L1, {5}) designates a list having the element 5 in addition to the elements of the list L1.

You can also use:

AUGMENT(L1, 5), which designates a list having the element 5 in addition to the elements of the list L1.

SUB L2; L1; 2; 4 is a command that copies into L2 the elements of L1 having indices from 2 to 4.

NOTE the difference between functions and commands:

Functions return a value, and they have parentheses enclosing their arguments (which are separated by commas), whereas

Commands do not return values, and their arguments are written directly after the name of the command (and are separated by semi-colons).

7.14 Example: The Sieve of Eratosthenes

7.14.1 Description

To find the prime numbers less than or equal to N :

1. Write the numbers from 1 to N in a list.
2. Cross out 1, and let 2 be the first value of P .
If $P * P \leq N$, then taking the numbers from P to N .
3. Cross out all the multiples of P from $P * P$ onward.
4. Augment P by 1, If $P * P$ is less than or equal to N , then taking the non-excluded numbers from P to N .
5. Change P to the smallest non-excluded number between P and N .
6. Repeat steps 3, 4 and 5 until $P * P$ exceeds N .

7.14.2 Algorithmic syntax

```
Function sieve(N)
local TAB PRIMES I P
//TAB and PRIMES are lists
{} ->TAB
{} ->PRIMES
//do steps 1 and 2
for I = 2 to N do
    TAB+I -> TAB
endfor
0 + TAB -> TAB
2-> P
//TAB is the list 0 2 3 4 ..N
//1 has been excluded by replacing it with 0

while P*P ≤ N do

//exclude all multiples of P from P*P onward
for I = P to FLOOR(N/P) do
    //FLOOR(N/P) designates the integer part of N/P
    0 -> TAB[I*P]
endfor
```

```

    P+1-> P
//Find the smallest non-excluded number
//between P and N

    while (P*P ≤ N) and (TAB[P]=0) do

        P+1 -> P
    endwhile
endwhile
//write the result to the list PRIMES
for I = 2 to N do

    If TAB[I] ≠ 0 then

        PRIMES + I -> PRIMES
    endif
endfor
Result: PRIMES

```

7.14.3 HP40GS syntax

Following is the program SIEVE:

The user must input a value for N.

At the end, the list L2 contains the prime numbers less than or equal to N.

```

INPUT N;"CRIBLE";"N=";;10:
ERASE:
MAKELIST(I,I,1,N,1) -> L1:
0 -> L1(1):
2->P:

WHILE P*P ≤ N REPEAT

    FOR I = P TO INT(N/P) STEP 1;
        0->L1(I*P):
    END:
    DISP 3;" "L1:
    P+1->P:

    WHILE P*P ≤ N AND L1(P) == 0 REPEAT

        P+1->P:
    END:
END:

```

```
{2}->L2:  
@ on sait que 2 est premier  
FOR I=3 TO N STEP 1;  
  
  IF L1(I) ≠ 0 THEN  
  
    CONCAT(L2,{I}) ->L2:  
    END:  
  END:  
DISP 3 ;"PREM" L2:  
FREEZE:
```


Chapter 8

Arithmetic Programs

8.1 GCD and Euclid's Algorithm

Given the two whole positive numbers A and B , find their GCD (Greatest Common Divisor).

Euclid's Algorithm is based on the recursive definition of GCD :

$$\begin{aligned}GCD(A, 0) &= A \\GCD(A, B) &= GCD(B, A \bmod B) \text{ if } B \neq 0\end{aligned}$$

where $A \bmod B$ designate the remainder of the Euclidean division of A by B .

Here is the description of this algorithm:

Perform the successive Euclidean divisions:

$$\begin{aligned}A &= B \times Q_1 + R_1 & 0 \leq R_1 < B \\B &= R_1 \times Q_2 + R_2 & 0 \leq R_2 < R_1 \\R_1 &= R_2 \times Q_3 + R_3 & 0 \leq R_3 < R_2 \\&\dots\dots\end{aligned}$$

After a finite number of steps, there exists a whole number n such that:

$$R_n = 0.$$

One then has:

$$\begin{aligned}GCD(A, B) &= GCD(B, R_1) = \dots = \\GCD(R_{n-1}, R_n) &= GCD(R_{n-1}, 0) = R_{n-1}.\end{aligned}$$

8.1.1 Algorithmic syntax

- Iterative version

If $B \neq 0$, calculate $R = A \bmod B$. Then, make A equal to B , and B equal to R , and repeat the calculation until $B = 0$. The *GCD* is then A .

```
Function GCD(A,B)
Local R
While B  $\neq$  0 do
    A mod B ->R
    B ->A
    R ->B
endwhile
Result: A
endfunction
```

- Recursive version

One simply writes the recursive definition given earlier.

```
Function GCD(A,B)
If B  $\neq$  0 then
    Result: GCD(B,A mod B)
Else
    Result: A
endif
endfunction
```

8.1.2 HP40GS syntax

- Iterative version for two whole numbers

First, type the subroutine IN, which enables the user to enter two numbers A and B:

```
INPUT A;"A";;1:
INPUT B;"B";;1:
ERASE:
```

Then type the GCD program:

```
RUN IN:
DISP 3;"PGCD "{A,B}:
```

```
WHILE B  $\neq$  0 REPEAT
```

```
A MOD B ->R:
B ->A:
R ->B:
END:
DISP 4;"PGCD "A:
FREEZE:
```

- Recursive version for two whole numbers A and B
 You cannot create recursive functions on the HP40GS... but you can create the program GCDR :

```
DISP 3;"PGCD "{A,B}:
FREEZE:
```

```
IF B  $\neq$  0 THEN
```

```
A MOD B ->R:
B ->A:
R ->B:
PGCDR:
ELSE
DISP 3;"PGCD "A:
FREEZE:
END:
```

First, the values of A and B are stored.

The program GCDR displays the GCD that it is in the process of calculating. The recursive call GCDR returns you to the GCDR program, which you must execute by pressing RUN on the menu bar.

The GCDR program then displays the intermediate GCD calculations.

You can also replace GCDR in the preceding program with RUN GCDR to avoid having to press RUN on the menu bar, and to suppress the display of the intermediate values, so that you can use this program in a larger program that caters for input and output:

>From the recursive program GCDR, we can derive the recursive program PR :

```
IF B  $\neq$  0 THEN
```

```
A MOD B ->R:
B ->A:
```

```
R ->B:
RUN PR:
END:
```

The program PR can be inserted into a larger program catering for input and output:

```
PROMPT A:
PROMPT B:
RUN PR:
ERASE:
MSGBOX A:
```

- Iterative version for two complex numbers.

If you use the symbolic calculation function IREMAINDER in place of MOD in the preceding programs, GCD (or PR) can then have Gaussian whole numbers as parameters on condition that you replace the names of the variables A, B and R with Z1, Z2 and Z3, and that you change the test for completion. When creating the new version start by recalling the contents of the earlier version, as described in 6.1.4. Here is the iterative version:

```
PROMPT Z1:
PROMPT Z2:
ERASE:
DISP 3;"PGCD "{Z1,Z2}:

WHILE ABS(Z2) ≠ 0 REPEAT

XNUM(IREMAINDER(XQ(Z1),XQ(Z2)) ->Z3:
Z2 ->Z1:
Z3 ->Z2:
END:
DISP 4;"PGCD "Z1:
FREEZE:
```

- Iterative version for two polynomials.

The variables E1, E2, . . . enable you to store expressions, which is what you must do if you deal with polynomials! If you use the symbolic calculation function REMAINDER in place of MOD in the preceding programs, GCD (or PR) can then have polynomials as parameters on condition that you replace the names of the variables A, B and R with E1, E2, E3, and that you change the test for completion. The CAS must be in Exact and Direct modes for this program.

```

PROMPT E1 :
PROMPT E2 :
ERASE :

WHILE DEGREE(E2) ≠ -1 REPEAT
REMAINDER(E1, E2) ->E3 :
E2 ->E1 :
E3 ->E2 :
END :
DISP 4 ; "PGCD "E1 :
FREEZE :

```

You enter (for example):

$E1 = S1^2 - 1$ and $E2 = S1^2 - 2 * S1 + 1$ to find the GCD equal to $2 * S1 - 2$.

8.2 Bézout's Identity

Bézout's function (A,B) returns the list $\{U, V, PGCD(A, B)\}$ (where U and V are such that:

$$A \times U + B \times V = PGCD(A, B).$$

8.2.1 Iterative version without lists

Euclid's Algorithm enables us to find a pair U and V such that:

$$A \times U + B \times V = PGCD(A, B)$$

In effect, if we call A_0 and B_0 the values the A and B at the start, then we have:

$$A = A_0 \times U + B_0 \times V \text{ avec } U = 1 \text{ and } V = 0$$

$$B = A_0 \times W + B_0 \times X \text{ avec } W = 0 \text{ and } X = 1$$

You then derive A, B, U, V, W, X in such a way that the two relations above are always satisfied.

If:

$$A = B \times Q + R \quad 0 \leq R < B \quad (R = A \text{ mod } B \text{ and } Q = E(A/B))$$

then:

$$R = A - B \times Q = A_0 \times (U - W \times Q) + B_0 \times (V - X \times Q) = A_0 \times S + B_0 \times T \text{ avec } S = U - W \times Q \text{ and } T = V - X \times Q$$

It remains then to repeat the process with:

B in the role of A ($B \rightarrow A$ $W \rightarrow U$ $X \rightarrow V$), and

R in the role of B ($R \rightarrow B$ $S \rightarrow W$ $T \rightarrow X$)

whence the algorithm:

```
Function Bezout (A,B)
local U,V,W, X, S, T, Q, R
1->U 0->V 0->W 1->X
```

```
While B  $\neq$  0 do
```

```
  A mod B->R
  FLOOR(A/B)->Q
  //R=A-B*Q
  U-W*Q ->S
  V-X*Q ->T
  B->A
  W->U
  X->V
  R->B
  S->W
  T->X
```

```
endwhile
```

```
Result: {U, V, A}
```

```
endfunction
```

8.2.2 Iterative version with lists

You can simplify the algorithmic syntax below by using fewer variables:

for this, use the lists LA , LB , LR to store the triplets $\{U, V, A\}$, $\{W, X, B\}$ and $\{S, T, R\}$. This is quite easy, as the calculator knows how to add lists of the same length (by adding elements with the same index), as well as how to multiply a list by a number (by multiplying each of the list's elements by the number).

```
Function Bezout (A,B)
local LA LB LR
{1, 0, A}->LA
{0, 1, B}->LB
```

```
While LB[3]  $\neq$  0 do
```

```

LA-LB*E(LA[3]/LB[3])->LR
LB->LA
LR->LB
endwhile
Result : LA
endfunction

```

8.2.3 Recursive version with lists

Bézout's Function can be recursively defined by:

$$\text{Bezout}(A, 0) = \{1, 0, A\}$$

If $B \neq 0$, is it necessary to define $\text{Bezout}(A, B)$ as a function of $\text{Bezout}(B, R)$ when

$$R = A - B \times Q \text{ and } Q = \text{FLOOR}(A/B).$$

We have:

$$\begin{aligned} \text{Bezout}(B, R) = LT = \{W, X, \text{pgcd}(B, R)\} \\ \text{avec } W \times B + X \times R = \text{pgcd}(B, R) \end{aligned}$$

with

$$\begin{aligned} W \times B + X \times (A - B \times Q) &= \text{gcd}(B, R) \text{ or} \\ X \times A + (W - X \times Q) \times B &= \text{gcd}(A, B). \end{aligned}$$

whence, if $B \neq 0$ and if $\text{Bezout}(B, R) = LT$, we have:

$$\text{Bezout}(A, B) = \{LT[2], LT[1] - LT[2] \times Q, LT[3]\}.$$

```

function Bezout (A,B)
local LT Q R

```

```

If B ≠ 0 then

```

```

    FLOOR(A/B) ->Q
    A-B*Q->R
    Bezout(B,R)->LT    Result: {LT[2], LT[1]-LT[2]*Q, LT[3]}
else Result: {1, 0, A}
endif
endfunction

```

8.2.4 Recursive version without lists

If you use global variables for A , B , D , U , V , T , you can view the function Bezour as using A , B to calculate the values for U , V , D ($AU+BV=D$) by means of a local variable Q .

One can then write:

```

program Bezour
local Q
If B  $\neq$  0 then
    FLOOR(A/B) ->Q
    A-B*Q->T
    B->A
    T->B
    Bezour
    U-V*Q->T
    V->U
    T->V
else
    1->U
    0->V
    A->D
endif

```

8.2.5 HP40GS syntax

- Iterative version with lists

Here, you also use the program IN, which enables the user to enter two numbers A and B:

```

INPUT A;"A" ;; ;1:
INPUT B;"B" ;; ;1:
ERASE:

```

Now type the BEZOUT program:

```

RUN IN:
DISP 3;"BEZOUT " {A,B}:
{1,0,A} ->L1:
{0,1,B} ->L2:
WHILE L2(3)  $\neq$  0 REPEAT

```



```

L1-L2*FLOOR(L1(3)/L2(3)) ->L3:
L2 ->L1:
L3 ->L2:
END:
DISP 4;"U V PGCD "L1:
FREEZE:

```

- Recursive version without lists

Type the program BEZOUR, using the commands (thanks to Bernard!!!):

PUSH (the command PUSH(A) stores the contents of A on the CAS stack),
and

POP (the POP command retrieves values stored on the CAS stack)

PUSH and POP can simulate the local variable Q We have replaced in the translation (cf 8.2.4) the variables R and W by the temporary variable T.

```

PROGRAM BEZOUR
IF B ≠ 0 THEN

PUSH (FLOOR(A/B)):
A MOD B->T:
B->A:
T->B:
RUN BEZOUR:
U-V*POP->T:
V->U:
T->V:
ELSE
  1->U:
  0->V:
  A->D:
END:

```

PUSH (FLOOR(A/B)) has the effect of putting the different values of FLOOR(A/B) onto a stack, and POP recovers them.

T is an auxiliary variable.

BEZOUR takes, as user input, the values of the global variables A and B, and assigns values to the global variables U and V such that:

$A \cdot U + B \cdot V = \text{GCD}(A, B)$. We can then write the final program BEZOURT, which caters for the input of A and B and the output of $\{U, V, D\}$:

```

PROGRAM BEZOURT

PROMPT A:
PROMPT B:

```

```
RUN BEZOUR:
ERASE:
MSGBOX {U,V,D}:
```

REMARKS (cf. iterative version for two complex 8.1.2):

If you use the symbolic calculation functions IREMAINDER and IQUOT(XQ(Z1), XQ(Z2)) in place of MOD and FLOOR(A/B) in the preceding programs, then BEZOUT and BEZOUR can take Gaussian integers as parameters, on condition that you replace the names of the variables A, B, R... with Z1, Z2, Z3....

If you use the symbolic calculation function REMAINDER in place of MOD in the preceding programs, then BEZOUT and BEZOUR can take polynomials as parameters, on condition that you replace the names of the variables A, B, R... with E1, E2, E3... and that you change the test for completion.

8.3 Decomposition into prime factors

8.3.1 Algorithmic syntax

- Initial algorithm

Let N be a whole number.

For all numbers D from 2 to N , test the divisibility of N by D .

If D is a divisor of N , then find the divisors of N/D , and so on... with N/D taking the role of N . The process stops when $N = 1$.

As divisors are found, they are stored in the list FACT.

```
function facprimes(N)
local D FACT
2->D
{} -> FACT

While N ≠ 1 do
    If N mod D = 0 then
        FACT + D -> FACT
        N/D -> N
    Else
        D+1 -> D
    endif
endwhile
```

```
Result: FACT
endfunction
```

- First improvement

One only tests the divisors D that are between 2 and $FLOOR(\sqrt{N})$.

In effect, if $N = D1 * D2$, then we can say:

Let $D1 \leq FLOOR(\sqrt{N})$, and let $D2 \leq FLOOR(\sqrt{N})$; otherwise, we would have:

$$D1 * D2 \geq (FLOOR(\sqrt{N}) + 1)^2 > N.$$

```
function facprimes(N)
local D FACT
2-> D
{ } -> FACT
```

```
While D*D ≤ N do
```

```
  If N mod D = 0 then
    FACT + D -> FACT
    N/D -> N
```

```
  Else
    D+1-> D
```

```
  endif
```

```
endwhile
```

```
FACT + N -> FACT
```

```
Result: FACT
```

```
endfunction
```

- Second improvement

One looks to see whether 2 is a divisor of N , then one tests only the odd divisors D that are between 3 and $FLOOR(\sqrt{N})$.

In the list `FACT`, each divisor is written followed by its exponent:

`decomp(12)={2,2,3,1}`.

```
function facprimes(N)
local K D FACT
{ }->FACT
0 -> K
While N mod 2 = 0 do
  K+1 -> K
  N/2 -> N
endwhile
```

```

If K ≠ 0 then
  FACT + {2 K} -> FACT
endif
3-> D

While D*D ≤ N do
  0 -> K
  While N mod D = 0 do
    K+1 -> K
    N/D -> N
  endwhile

  If K ≠ 0 then
    FACT + (D K) -> FACT
  endif
  D+2 -> D
endwhile
{\tt If N $\neq 1$ 1 then}
\begin{verbatim}  FACT + {N 1} -> FACT
endif
Result: FACT
endfunction

```

8.3.2 HP40GS syntax

This is a translation of the last algorithm.

The HP40GS calculator does not understand {}, so to initialise L1 as an empty list you must type:

```

CLRVAR L1
or SUB L1;L1;2;1.

```

Here is the program FACTPRIMES:

```

INPUT N;"N";;1:
ERASE:
0 ->K:
CLRVAR L1:
WHILE N MOD 2 == 0 REPEAT
1+K -> K:
N/2 -> N:
END:

```

```

IF K ≠ 0 THEN
{2,K} ->L1:
END:
3 ->D:

WHILE D*D ≤ N REPEAT
0 -> K:
WHILE N MOD D == 0 REPEAT
K+1 -> K:
N/D -> N:
END:

IF K ≠ 0 THEN
CONCAT (L1, {D,K}) -> L1:
END:
2+D -> D:
END:

IF N ≠ 1 THEN
CONCAT (L1, {N,1}) -> L1:
END:
DISP 3; "FACT" L1:
FREEZE:

```

8.4 Calculating $A^P \text{ mod } N$

8.4.1 Algorithmic syntax

- First algorithm

This algorithm uses two local variables, POWER and I.

Write an iterative program such that at each stage, POWER represents $A^I \pmod{N}$.

```

function powermod (A, P, N)
local POWER, I
1 -> POWER
for I = 1 to P do
    A*POWER mod N -> POWER
endfor
Result: POWER
endfunction\

```

- Second algorithm

This algorithm uses a single local variable POW, but it varies P such that at each stage of the iteration we have:

$$\text{result} = \text{POW} * A^P \pmod{N}$$

```
function powermod (A, P, N)
local POW
1 -> POW
While P>0 do
  A*POW mod N -> POW
  P-1 ->P
endwhile
Result: POW
endfunction
```

- Third algorithm

One can easily modify this program by taking into account that:

$$A^{2^*P} = (A * A)^P.$$

Therefore when P is even, we have the relation:

$$PUI * A^P = PUI * (A * A)^{P/2} \pmod{N}$$

and when P is odd, we have the relation:

$$PUI * A^P = PUI * A * A^{P-1} \pmod{N}.$$

We are left with a very fast algorithm for $A^P \pmod{N}$.

```
function powermod (A, P, N)
local POW
1->POW
While P<0 do
  If P mod 2=0 then
    P/2->P
    A*A mod N->A
  Else
    A*POW mod N ->POW
    P-1->P
  endif
endwhile
Result: POW
endfunction
```

It goes without saying that if P is odd, then P-1 is even.

One can then write:

```

function powermod (A, P, N)
local POW
1->POW
While P<0 do
  If P mod 2=1 then      A*POW mod N->POW
    P-1->P
  endif
  P/2-(>P
  A*A mod N->A
endwhile
Result: POW
endfunction

```

8.4.2 HP40GS syntax

The calculation of $A^P \bmod N$ is treated in the program on the probability method of Mr Rabin. Please refer to the HP40GS version of that program (cf 8.6).

8.5 The function "isprime"

8.5.1 Algorithmic syntax

- Initial algorithm

Write a Boolean function of N that is equal to `TRUE` when N is prime and `FALSE` when it is non-prime.

For this, find whether N has a divisor $\neq 1$ and $\leq \text{FLOOR}(\sqrt{N})$ (the whole part of the square root of N).

The case where $N=1$ is treated separately!

Here, the Boolean variable `PRIME` is used, which is `TRUE` by default, and which is set to `FALSE` if a divisor of N is found.

```

Function isprime(N)
local PRIME, I, J
FLOOR((√N)- > J
If N = 1 then
  FALSE->PRIME
else
  TRUE->PRIME

```

```

endif
2->I
While PRIME and I ≤ J do
    If N mod I=0 then
        FALSE->PRIME
    else
        I+1->I
    endif
endwhile
Result: PRIME
endfunction

```

- First improvement

Of course, we can test to see if N is even, then look only for odd divisors of N .

```

function isprime(N)
Local PRIME, I, J
FLOOR(√N) → J
If (N = 1) or (N mod 2 = 0) and (N≠2) then
    FALSE->PRIME
else
    TRUE->PRIME
endif
3->I
While PRIME and I ≤ J do
    If N mod I = 0 then
        False -> PRIME}
    else
        I+2->I
    endif
endwhile
Result: PRIME
endfunction

```

- Second improvement

We can test to see if N is divisible by 2 or 3, then look only for divisors of N that are of the form $6 \times k - 1$ or $6 \times k + 1$.


```

function isprime(N)
local PRIME, I, J

FLOOR( $\sqrt{N}$ ) -> J

If (N = 1) or (N mod 2 = 0) or (N mod 3 = 0) then
  FALSE->(PRIME
else
  TRUE->PRIME
endif
If N=2 or N=3 then
  TRUE->PRIME
endif
5->I

While PRIME and I  $\leq$  J do

  If (N mod I = 0) or (N mod (I + 2) = 0) then
    FALSE->PRIME
  else
    I + 6 -> I
  endif
endwhile
Result: PRIME
endfunction
end{verbatim}
\subsection{{\tt HP40GS} syntax}
\begin{verbatim}
INPUT N;"N";;1:
IF N MOD 2== 0 OR N MOD 3==0 OR N==1 THEN
0 ->P:
ELSE
1->P:
END:
IF N==2 OR N==3 THEN
1->P:
END:
5->I:

FLOOR( $\sqrt{N}$ ) -> J:
WHILE I  $\leq$  J AND P REPEAT

```

```

IF N MOD I==0 OR N MOD (I+2)==0 THEN
0 ->P:
ELSE
I+6 ->I:
END:
END:
ERASE:
DISP 5;P:
FREEZE:

```

8.6 Mr Rabin's probability method

If N is prime, then all numbers K less than N are prime with N , so according to Fermat's Little Theorem we have:

$$K^{N-1} = 1 \pmod{N}$$

If N is not prime, the integers K such that:

$$K^{N-1} = 1 \pmod{N}$$

are very few indeed.

A number N such that $K^{N-1} = 1 \pmod{N}$ for 20 random tries of K is called a pseudo-prime number. The probability method of Rabin consists of choosing a random number K ($1 < K < N$) and calculating:

$$K^{N-1} \pmod{N}$$

If $K^{N-1} = 1 \pmod{N}$, another random number is tried, whereas if $K^{N-1} \neq 1 \pmod{N}$, it is certain that N is not prime.

If $K^{N-1} = 1 \pmod{N}$ is obtained for 20 tries of K , one can conclude that N is prime with a very small probability of error.

Naturally, this method is used to test whether large numbers are pseudo-primes, but the command `ISPRIME?` use the Miller-Rabin method. This method use also Fermat's Little Theorem and is also a probabilist method. It conclude that N is prime with a very small probability of error (less than $(0.25)^{20}$ if you make 20 random tries of K or on the order of 10^{-12}).

8.6.1 Algorithmic syntax

Let us suppose that:

`Random(N)` returns a random whole number between 0 and $N - 1$.

Calculation of:

$$K^{N-1} \pmod{N}$$

is carried out using the very fast algorithm developed earlier (cf page 181).

NOTE the statement:

powermod(K, P, N), the function that calculates $K^P \bmod N$.

```
Function isprime(N)
local K, I, P
1->I
1->>P
While P = 1 and I < 20 do
  Random(N-2)+2->K
  powermod(K, N-1, N)->P
  I+1->I
endwhile
If P =1 then
  Result: TRUE
else
  Result: FALSE
endif
endfunction
```

8.6.2 HP40GS syntax

```
PROMPT N:
RANDSEED TIME:
1->I:
1->P:
WHILE I < 20 AND P==1 REPEAT
  FLOOR( RANDOM * (N-2))+2->K:
  N-1->M:
  @ Calcul de K puissance M mod N dans P.
  1->P:
  WHILE 0 < M REPEAT
    IF M MOD 2 == 0 THEN
      M / 2 -> M :
      (K * K) MOD N ->K :
    ELSE
      K*P MOD N -> P:
      M - 1 -> M:
    END:
  END:
  @ P contient K puissance M mod N et M=N-1.
  I+1 ->I:
```

```

END:
ERASE:
IF P==1 THEN
  DISP 3;"PREMIER " N:
ELSE
  DISP 3;"NON PREMIER " N:
END:
FREEZE:

```

REMARK: You can also use the computer algebra function POWMOD, substituting:

```

MODSTO(N):
POWMOD(K,N-1) -> P:

```

for the statements between the comments (@). This results in :

```

PROMPT N:
RANDSEED TIME:
1->I:
1->P:
WHILE I < 20 AND P==1 REPEAT
  FLOOR( RANDOM * (N-2) )+2->K:
  MODSTO(N):
  POWMOD(K,N-1)-> P:
  I+1 ->I:
END:
ERASE:
IF P==1 THEN
  DISP 3;"PREMIER " N:
ELSE
  DISP 3;"NON PREMIER " N:
END:
FREEZE:

```

Chapter 9

GNU Free Documentation License

Version 1.1, March 2000

Copyright (C) 2000 Free Software Foundation, Inc. 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other written document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondly, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you".

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (For example, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, whose contents can be viewed and edited directly and straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup has been designed to thwart or discourage subsequent modification by readers is not Transparent. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, designed for human modification. Opaque formats include PostScript, PDF, proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY

If you publish printed copies of the Document numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a publicly-accessible computer-network location containing a complete Transparent copy of the Document, free of added material, which the general network-using public has access to download anonymously at no charge using public-standard network protocols. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

If it is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

* A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission. * B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has less than five). * C. State on the Title page the name of the publisher of the Modified Version, as the publisher. * D. Preserve all the copyright notices of the Document. * E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices. * F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below. * G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice. * H. Include an unaltered copy of this License. * I. Preserve the section entitled "History", and its title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on its Title Page. If there is no section entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence. * J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission. * K. In any section entitled "Acknowledgements" or "Dedications", preserve the section's title, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein. * L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles. * M. Delete any section entitled "Endorsements". Such a section may not be included in the Modified Version. * N. Do not retitle any existing section as "Endorsements" or to conflict in title with any Invariant Section.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties—for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections entitled "History" in the various original documents, forming one section entitled "History"; likewise combine any sections entitled "Acknowledgements", and any sections entitled "Dedications". You must delete all sections entitled "Endorsements."

6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, does not as a whole count as a Modified Version of the Document, provided no compilation copyright is claimed for the compilation. Such a compilation is called an "aggregate", and this License does not apply to the other self-contained works thus compiled with the Document, on account of their being thus compiled, if they are not themselves derivative works of the Document. If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one quarter of the entire aggregate, the Document's Cover Texts may be placed on covers that surround only the Document within the aggregate. Otherwise they must appear on covers around the whole aggregate.

8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant

Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License provided that you also include the original English version of this License. In case of a disagreement between the translation and the original English version of this License, the original English version will prevail.

9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided for under this License. Any other attempt to copy, modify, sublicense or distribute the Document is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See <http://www.gnu.org/copyleft/>.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation.

How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

Copyright (c) YEAR YOUR NAME. Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.1 or any later version published by the Free Software Foundation; with the Invariant Sections being LIST THEIR TITLES, with the Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST. A copy of the license is included in the section entitled "GNU Free Documentation License".

If you have no Invariant Sections, write "with no Invariant Sections" instead of saying which ones are invariant. If you have no Front-Cover Texts, write "no Front-Cover Texts" instead of "Front-Cover Texts being LIST"; likewise for Back-Cover Texts.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.

Appendix A

Index

Skip 2 pages.

Index

- ◀, 7
- \int , 105
- |, 69, 91
- \sum , 84
- ▷, 7
- , 71, 87
- >, 7
- △ ◁ ▷ ▽, 7

- ABCUV, 92
- ABS, 71, 87
- ACOS2S, 71, 126
- ADDTMOD, 71, 78
- ARG, 71, 87
- ASIN2C, 71, 126
- ASIN2T, 71, 127
- ASSUME, 20, 36, 71
- ATAN2S, 71, 127
- AUGMENT, 51
- AXL, 56
- AXM, 56
- AXQ, 52

- BASIS, 57

- C2P, 63
- CEILING, 71
- CFG, 67
- Change font, 18, 69
- CHINREM, 92
- CHOLESKY, 57
- CIRC, 64

- CLRVAR, 165
- COLLECT, 69, 89
- CONJ, 71, 87, 88
- Copy, 18, 69
- CURL, 54
- Cursor mode, 18, 69
- Cut, 18, 69
- CYCLOTOMIC, 93

- DEF, 69, 100
- DEGREE, 93
- DERIV, 54, 70, 102
- DERVX, 70, 103
- DESOLVE, 70, 123
- DIAGMAP, 57
- DISTRIB, 70, 116
- DIV, 54
- DIV2, 93
- DIV2MOD, 79
- DIVIS, 71, 72
- DIVMOD, 71, 79
- DIVPC, 70, 112
- DROITE, 71, 87, 88, 139

- Edit expr., 18, 69
- EGCD, 71, 94
- EPSX0, 49, 70, 116
- EULER, 71, 72
- EXLR, 55
- EXP2HYP, 127
- EXP2POW, 70, 116
- EXPAND, 69, 90

- EXPANDMOD, 71, 80
 EXPLN, 70, 117
 FACTOR, 69, 71, 73, 90, 94
 FACTORMOD, 71, 80
 FACTORS, 43
 FCOEF, 43
 FDISTRIB, 70, 117
 FLOOR, 71, 85
 FOURIER, 70, 103
 FRAC, 71
 FROOTS, 44
 FXND, 48
 GAMMA, 85
 GAUSS, 53
 GBASIS, 44
 GCD, 71, 73, 95
 GCDMOD, 71, 80
 GREDUCE, 44
 HADAMARD, 57
 HALFTAN, 71, 128
 HELP, 7, 20, 40
 HELPWITH, 40
 HERMITE, 71, 95
 HESS, 55
 HILBERT, 58
 HORNER, 95
 i, 71
 IABCUV, 73
 IBASIS, 58
 IBERNOULLI, 83
 IBP, 70, 104
 ICHINREM, 74
 IDIV2, 71, 74
 IEGCD, 71, 75
 IFTE, 20, 71, 102
 ILAP, 124
 IM, 71, 87
 IMAGE, 59
 INT, 71
 INTVX, 70, 106
 INVMOD, 71, 81
 IQUOT, 71, 75
 IREMAINDER, 71, 76
 ISOLATE, 70, 120
 ISOM, 64
 ISPRIME?, 71, 76, 101
 JORDAN, 59
 KER, 59
 LAGRANGE, 45
 LAP, 124
 LAPL, 55
 LCM, 71, 77, 96
 LCXM, 60
 LDEC, 70, 126
 LEGENDRE, 71, 96
 LGCD, 45
 lim, 108
 LIMIT, 70, 108
 LIN, 70, 117
 LINSOLVE, 70, 121
 LNAME, 49
 LNCOLLECT, 70, 118
 LVAR, 49
 MAD, 60
 MAP, 52
 MAX, 71
 MIN, 71
 MKISOM, 65
 MOD, 71, 77, 85
 MODSTO, 36, 71, 81
 MSLV, 50
 MULTMOD, 71, 81
 NEXTPRIME, 71, 77

- P2C, 64
 PA2B2, 77
 PARTFRAC, 69, 71, 97
 Paste, 18, 69
 PCAR, 61
 PLOT, 110
 PLOTADD, 110
 PMINI, 61
 POP, 39, 160, 177
 POWEXPAND, 70, 118
 POWMOD, 71, 81
 PREVAL, 70, 110
 PREVPRIME, 71, 78
 PROMPT, 161
 PROMPTSTO, 161
 PROPFAC, 71, 83, 97
 PSI, 86
 Psi, 86
 PTAYL, 71, 98
 PUSH, 39, 160, 177

 qr, 62
 QUOT, 71, 98
 QUOTE, 69, 88, 108
 QXA, 53

 RE, 71, 87
 REALASSUME, 20, 36, 88
 REF, 41
 REMAINDER, 71, 98
 REORDER, 90
 RESULTANT, 46
 RISCH, 70, 111
 rref, 42
 RREFMOD, 48

 SERIES, 70, 113
 SEVAL, 91
 SIGMA, 111
 SIGMAVX, 111
 SIGN, 71, 87

 SIMP2, 46
 SIMPLIFY, 70, 119
 SINCOS, 70, 71, 128
 SOLVE, 70, 120, 123
 SOLVEVX, 70, 121
 STO, 37
 STO>, 7
 STORE, 35, 69
 STURM, 47
 STURMAB, 98
 SUBST, 69, 91, 124
 SUBTMOD, 71, 82
 SYLVESTER, 53
 SYST2MAT, 62

 TABVAL, 50
 TABVAR, 70, 112
 TAN2CS2, 71, 128
 TAN2SC, 71, 129
 TAN2SC2, 71, 129
 TAYLOR0, 70, 115
 TCHEBYCHEFF, 71, 99
 TCOLLECT, 71, 129
 TEVAL, 50
 TEXPAND, 69, 71, 130
 TLIN, 71, 131
 TRAN, 62
 TRIG, 71, 131
 TRIGCOS, 71, 131
 TRIGSIN, 71, 132
 TRIGTAN, 71, 132
 TRUNC, 70, 115
 TSIMP, 132

 UNASSIGN, 35, 69
 UNASSUME, 20, 36, 71

 VANDERMONDE, 63
 VER, 133
 VERSION, 133

INDEX

197

XNUM, 51, 70, 82, 84, 119

XQ, 51, 70, 119

ZEROS, 47

Appendix B

Table of contents

Skip 2 pages

Contents

1	To Begin	5
1.1	General Information	5
1.1.1	Starting the Calculator	5
1.1.2	What You See	5
1.2	Notations	7
1.3	Online Help	7
2	Aplets	9
2.1	APLET Key	9
2.2	The Various aplets	9
2.3	Examples using the Sequence Aplet	10
2.3.1	Notation in Base b	10
2.3.2	Calculating the GCD	11
2.3.3	Calculating the Coefficients of Bézout's Identity	12
2.4	The SYMB, NUM, PLOT and MODES Keys	13
3	CAS user's guide	15
3.1	Terms and Conditions	15
3.2	Introduction	16
3.2.1	What is CAS?	16
3.2.2	How do you perform a symbolic calculation?	16
3.2.3	Variables	17
3.2.4	The current variable	17
3.3	Using the CAS in the Equation Editor	18
3.3.1	The Equation editor menu	18
3.3.2	The keyboard in the Equation Editor	19
3.3.3	Expressions, subexpressions, selection	24
3.3.4	Accessing the CAS functions	30

3.3.5	Equation editor variables	34
3.4	CAS in the HOME screen	36
3.4.1	HOME variables	36
3.4.2	Storing an object in a HOME variable : STO>	37
3.4.3	Some tricks and examples.	37
3.4.4	Send expressions from HOME into the CAS history : PUSH	39
3.4.5	Retrieve expressions written to the CAS history in HOME : POP	39
3.4.6	The keyboard in the HOME screen	39
4	The CAS functions only valid in HOME screen	41
4.1	Linear systems	41
4.1.1	Resolution of a linear system of equations : REF	41
4.1.2	Resolution of a linear system of equations : rref	42
4.2	Polynomials	43
4.2.1	List of factors : FACTORS	43
4.2.2	Rational function given by its roots and poles :FCOEF	43
4.2.3	Roots and poles of a rational function : FROOTS	44
4.2.4	Gröbner basis : GBASIS	44
4.2.5	Gröbner reduction : GREDUCE	44
4.2.6	Lagrange's polynomial : LAGRANGE	45
4.2.7	GCD of a list : LGCD	45
4.2.8	resultant of two polynomials: RESULTANT	46
4.2.9	Division by the PGCD :SIMP2	46
4.2.10	Sturm's sequences : STURM	47
4.2.11	Zeros of a polynomial : ZEROS	47
4.3	Rational fractions	48
4.3.1	Numerator and denominator of a fraction : FXND	48
4.4	Modular calculations	48
4.4.1	Resolution of a linear system of equations : RREFMOD	48
4.5	Variables	49
4.5.1	The symbolic variables of an expression : LNAME	49
4.5.2	The variables of an expression : LVAR	49
4.6	Numeric calculus	49
4.6.1	Omit the values smaller than EPS : EPSX0	49
4.6.2	Solve numerically a system : MSLV	50
4.6.3	Time for evaluation of the answer : TEVAL	50
4.6.4	Numeric approximation : XNUM	51
4.6.5	Exact value : XQ	51

4.7	The lists	51
4.7.1	Concatenation : AUGMENT	51
4.7.2	Map a function to a list : MAP	52
4.8	Quadratic forms	52
4.8.1	Matrice to quadratic form : AXQ	52
4.8.2	Gauss decomposition : GAUSS	53
4.8.3	Matrix of a quadratic form : QXA	53
4.8.4	Sylvester decomposition : SYLVESTER	53
4.9	Functions of multiple variables	54
4.9.1	The rotor : CURL	54
4.9.2	The gradient : DERIV	54
4.9.3	The divergence : DIV	54
4.9.4	The hessian : HESS	55
4.9.5	The laplacian : LAPL	55
4.10	Equations	55
4.10.1	Members of an equation : EXLR	55
4.11	Matrices	56
4.11.1	Matrix to list : AXL	56
4.11.2	Symbolic matrix to numeric matrix : AXM	56
4.11.3	Basis vectors : BASIS	57
4.11.4	Cholesky matrix : CHOLESKY	57
4.11.5	Map an operator to a diagonalizable matrix : DIAGMAP	57
4.11.6	Hadamard product : HADAMARD	57
4.11.7	Hilbert matrix : HILBERT	58
4.11.8	Basis of an intersection : IBASIS	58
4.11.9	basis of the image : IMAGE	59
4.11.10	Jordan matrix : JORDAN	59
4.11.11	Basis of the kernel : KER	59
4.11.12	Make a matrix : LCXM	60
4.11.13	List of several elements : MAD	60
4.11.14	Characteristic polynomial : PCAR	61
4.11.15	Minimal polynomial : PMINI	61
4.11.16	Q*R decomposition : qr	62
4.11.17	System to matrix : SYST2MAT	62
4.11.18	Transposed matrix : TRAN	62
4.11.19	Vandermonde matrix : VANDERMONDE	63
4.12	The permutations	63
4.12.1	Cycles to permutation : C2P	63
4.12.2	Composition : CIRC	64
4.12.3	Decomposition into cyclesP2C	64

4.13	The isometries	64
4.13.1	Isometry's characteristics elements : ISOM	64
4.13.2	Isometry's matrix : MKISOM	65
5	CAS Reference from the Equation Editor	67
5.1	CAS menu bar	67
5.1.1	Change your configuration : CFG	67
5.1.2	Step/Step Mode	69
5.1.3	TOOL menu	69
5.1.4	ALGB menu	69
5.1.5	DIFF menu	70
5.1.6	REWRI menu	70
5.1.7	SOLV menu	70
5.1.8	TRIG menu	71
5.1.9	The MATH key	71
5.2	Integers (and Gaussian Integers)	71
5.2.1	Unlimited precision	72
5.2.2	The divisors of a number : DIVIS	72
5.2.3	The Euler index : EULER	72
5.2.4	Decomposition into prime factors : FACTOR	73
5.2.5	The greatest common divisor : GCD	73
5.2.6	Resolution of $au + bv = c$: IABCUV	73
5.2.7	Chinese remainders : ICHINREM	74
5.2.8	Quotient and remainder of two integers : IDIV2	74
5.2.9	Bézout's Identity : IEGCD	75
5.2.10	The integer Euclidean quotient : IQUOT	75
5.2.11	The integer remainder : IREMAINDER	76
5.2.12	Test of pseudo-primality : ISPRIME?	76
5.2.13	The least common multiple : LCM	77
5.2.14	Remainder of the Euclidean division : MOD	77
5.2.15	The smallest pseudo-prime greater than N : NEXTPRIME	77
5.2.16	Resolution of $a^2 + b^2 = p$ with integers: PA2B2	77
5.2.17	The greatest pseudo-prime less than N : PREVPRIME	78
5.3	Modular Calculations	78
5.3.1	Addition in $\mathbb{Z}/p\mathbb{Z}$ or in $\mathbb{Z}/p\mathbb{Z}[X]$: ADDTMOD	78
5.3.2	Quotient and remainder in $\mathbb{Z}/p\mathbb{Z}[X]$: DIV2MOD	79
5.3.3	Division in $\mathbb{Z}/p\mathbb{Z}$ or in $\mathbb{Z}/p\mathbb{Z}[X]$: DIVMOD	79
5.3.4	To expand and simplify expressions in $\mathbb{Z}/p\mathbb{Z}$ or in $\mathbb{Z}/p\mathbb{Z}[X]$: EXPANDMOD	80
5.3.5	To factor polynomials in $\mathbb{Z}/p\mathbb{Z}[X]$: FACTORMOD	80

5.3.6	GCD of the two polynomials in $\mathbb{Z}/p\mathbb{Z}[X]$: GCDMOD	80
5.3.7	Inverse in $\mathbb{Z}/p\mathbb{Z}$: INVMOD	81
5.3.8	To put a value in the MODULO variable : MODSTO	81
5.3.9	Multiplication in $\mathbb{Z}/p\mathbb{Z}$ or $\mathbb{Z}/p\mathbb{Z}[X]$: MULTMOD	81
5.3.10	Power in $\mathbb{Z}/p\mathbb{Z}$ or in $\mathbb{Z}/p\mathbb{Z}[X]$: POWMOD	81
5.3.11	Subtraction in $\mathbb{Z}/p\mathbb{Z}$ or in $\mathbb{Z}/p\mathbb{Z}[X]$: SUBTMOD	82
5.4	Rational Numbers	82
5.4.1	The n -th Bernoulli's number : IBERNOULLI	83
5.4.2	Whole-number and fractionnal part : PROPFRAC	83
5.4.3	Symbolic summation : \sum	84
5.5	Real Numbers	84
5.5.1	The whole part : FLOOR	85
5.5.2	The Γ function : GAMMA	85
5.5.3	Remainder of the Euclidean division : MOD	85
5.5.4	The derivatives of the Digamma function : PSI	86
5.5.5	The Digamma function : Psi	86
5.6	Complex Numbers	87
5.6.1	Argument of a complex number : ARG	87
5.6.2	Conjugate of a complex number : CONJ	88
5.6.3	Equation of the line : DROITE	88
5.7	Algebraic Expressions	89
5.7.1	Factors over the integers : COLLECT	89
5.7.2	Distributivity : EXPAND	90
5.7.3	Factorization : FACTOR	90
5.7.4	Change the order of variables : REORDER	90
5.7.5	Partial simplifications : SEVAL	91
5.7.6	Substitute a value for a variable :	91
5.7.7	Substitute a value for a variable : SUBST	91
5.8	Polynomials	91
5.8.1	Resolution of $A[X]U[X]+B[X]V[X]=C[X]$: ABCUV	92
5.8.2	Chinese remainders : CHINREM	92
5.8.3	Cyclotomic polynomial : CYCLOTOMIC	93
5.8.4	degree of a polynomial : DEGREE	93
5.8.5	Quotient and remainder : DIV2	93
5.8.6	Bézout's Identity : EGCD	94
5.8.7	Factorize a polynomial : FACTOR	94
5.8.8	GCD of two polynomials : GCD	95
5.8.9	Hermite polynomial : HERMITE	95
5.8.10	To evaluate a polynomial : HORNER	95
5.8.11	LCM of two polynomials : LCM	96

5.8.12	Legendre polynomial : LEGENDRE	96
5.8.13	Partial fraction expansion : PARTFRAC	97
5.8.14	Whole-number and fractional part : PROPFRAC	97
5.8.15	Rewrite in order of the powers of $(X - a)$: PTAYL	98
5.8.16	Euclidean quotient: QUOT	98
5.8.17	Euclidean remainder : REMAINDER	98
5.8.18	Number of zeros in $[a, b]$: STURMAB	98
5.8.19	First and 2nd-species Tchebycheff polynomials : TCHEBYCHEFF	99
5.9	Functions	100
5.9.1	Define a function : DEF	100
5.9.2	Derivative and partial derivative : DERIV	102
5.9.3	Derivative : DERVX	103
5.9.4	Fourier coefficients : FOURIER	103
5.9.5	Partial integration : IBP	104
5.9.6	Integrals : \int	105
5.9.7	Primitive and defined integral : INTVX	106
5.9.8	To calculate limits : LIMIT or lim	108
5.9.9	Plot of an expression : PLOT	110
5.9.10	Add the plot of an expression : PLOTADD	110
5.9.11	Evaluate a primitive : PREVAL	110
5.9.12	Primitive and defined intégral : RISCH	111
5.9.13	Discrete antiderivative : SIGMA	111
5.9.14	Discrete antiderivative : SIGMAVX	111
5.9.15	Variation table : TABVAR	112
5.10	Taylor and asymptotic Expansions	112
5.10.1	Division in increasing order by exponent : DIVPC	112
5.10.2	Limited n th-order expansion : SERIES	113
5.10.3	Limited expansion in the vicinity of 0 : TAYLOR0	115
5.10.4	Truncate at order $n - 1$: TRUNC	115
5.11	Conversion Functions	116
5.11.1	Distributivity of multiplication : DISTRIB	116
5.11.2	Disregard small values : EPSX0	116
5.11.3	Transform $\exp(n \cdot \ln(x))$ as a power of x : EXP2POW	116
5.11.4	Transform a trigonometric expression into complex exponentials : EXPLN	117
5.11.5	Distributivity : FDISTRIB	117
5.11.6	Linearise the exponentials : LIN	117
5.11.7	Regroup the logarithms : LNCOLLECT	118
5.11.8	Transform a power : POWEXPAND	118

5.11.9	Simplify : SIMPLIFY	119
5.11.10	Evaluation of reals : XNUM	119
5.11.11	Rational approximation : XQ	119
5.12	Equations	120
5.12.1	The zeros of an expression : ISOLATE	120
5.12.2	Solve equations : SOLVE	120
5.12.3	Solve equations : SOLVEVX	121
5.13	Systems	121
5.13.1	Solve linear system : LINSOLVE	121
5.13.2	Solve non linear system : SOLVE	123
5.14	Differential Equations	123
5.14.1	Solve of differential equations : DESOLVE	123
5.14.2	Laplace transform and inverse Laplace transform : LAP and ILAP	124
5.14.3	Linear differential equations having constant coef- ficients : LDEC	126
5.15	Trigonometric Expressions	126
5.15.1	Transform the arccos into arcsin : ACOS2S	126
5.15.2	Transform the arcsin into arccos : ASIN2C	126
5.15.3	Transform the arcsin into arctan : ASIN2T	127
5.15.4	Transform the arctan into arcsin : ATAN2S	127
5.15.5	Transform exponential into hyperbolic functions : EXP2HYP	127
5.15.6	Transform in terms of $\tan(x/2)$: HALFTAN	128
5.15.7	Transform the complex exponentials into sin and cos : SINCOS	128
5.15.8	Transform $\tan(x)$ with $\sin(2x)$ and $\cos(2x)$: TAN2CS2	128
5.15.9	Replace $\tan(x)$ with $\sin(x)/\cos(x)$: TAN2SC	129
5.15.10	Transform $\tan(x)$ with $\sin(2x)$ and $\cos(2x)$: TAN2SC2	129
5.15.11	Reconstruct the sine and the cosine of the same an- gle : TCOLLECT	129
5.15.12	Develop transcendental expressions : TEXPAND	130
5.15.13	Linearise une trigonometric expression : TLIN	131
5.15.14	Simplify using $\sin(x)^2 + \cos(x)^2 = 1$: TRIG	131
5.15.15	Simplify using the cosines : TRIGCOS	131
5.15.16	Simplify using the sines : TRIGSIN	132
5.15.17	Simplify using the tangents : TRIGTAN	132
5.15.18	Simplification with complex exponentials : TSIMP	132
5.16	Version number of your CAS : VER	133

6	Worked Exercises with the HP40	135
6.1	Introduction	135
6.2	Exercises on Brevet	136
6.2.1	Exercise 1	136
6.2.2	Exercise 2	136
6.2.3	Exercise 3	137
6.2.4	Exercise 4	138
6.2.5	Exercise 5	139
6.3	Exercises on Bac	140
6.3.1	Exercise 1	140
6.3.2	Exercise 2 (specialty)	146
6.3.3	Exercise 3 (non-specialty)	151
6.4	Conclusion	156
7	Programming	157
7.1	Implementation	157
7.1.1	How to edit and save a program	157
7.1.2	How to correct a program	157
7.1.3	How to run a program	158
7.1.4	How to modify a program	158
7.2	Comments	158
7.2.1	SHIFT 1 (PROGRAM) keys	159
7.3	Variables	160
7.3.1	Their names	160
7.3.2	The concept of local variables	160
7.3.3	The concept of parameters	160
7.4	User entry	161
7.4.1	Algorithmic syntax	161
7.4.2	HP40GS syntax	161
7.5	Output	161
7.5.1	Algorithmic syntax	161
7.5.2	HP40GS syntax	161
7.6	The sequence of statements, or "action"	162
7.6.1	Algorithmic syntax	162
7.6.2	HP40GS syntax	162
7.7	The assignment statement	162
7.7.1	Algorithmic syntax	162
7.7.2	HP40GS syntax	162
7.8	Conditional statements	162
7.8.1	Algorithmic syntax	162

7.8.2	HP40GS syntax	163
7.9	FOR statement	163
7.9.1	Algorithmic syntax	163
7.9.2	HP40GS syntax	163
7.10	WHILE statement	164
7.10.1	Algorithmic syntax	164
7.10.2	HP40GS syntax	164
7.11	Boolean Expressions	164
7.11.1	Algorithmic syntax	164
7.11.2	HP40GS syntax	164
7.12	Logical Operators	164
7.12.1	Algorithmic syntax	164
7.12.2	HP40GS syntax	164
7.13	Lists	165
7.13.1	Algorithmic syntax	165
7.13.2	HP40GS syntax	165
7.14	Example: The Sieve of Eratosthenes	166
7.14.1	Description	166
7.14.2	Algorithmic syntax	166
7.14.3	HP40GS syntax	167
8	Arithmetic Programs	169
8.1	GCD and Euclid's Algorithm	169
8.1.1	Algorithmic syntax	170
8.1.2	HP40GS syntax	170
8.2	Bézout's Identity	173
8.2.1	Iterative version without lists	173
8.2.2	Iterative version with lists	174
8.2.3	Recursive version with lists	175
8.2.4	Recursive version without lists	176
8.2.5	HP40GS syntax	176
8.3	Decomposition into prime factors	178
8.3.1	Algorithmic syntax	178
8.3.2	HP40GS syntax	180
8.4	Calculating $A^P \bmod N$	181
8.4.1	Algorithmic syntax	181
8.4.2	HP40GS syntax	183
8.5	The function "isprime"	183
8.5.1	Algorithmic syntax	183
8.6	Mr Rabin's probability method	186

8.6.1	Algorithmic syntax	186
8.6.2	HP40GS syntax	187
9	GNU Free Documentation License	189
A	Index	193
B	Table of contents	195