

# An Alternative HP-42S/Free42 Manual

Version 0.7 – January 2010



MoHPC

Author: José Lauro Strapasson, Brazil  
[jstrapasson@uol.com.br](mailto:jstrapasson@uol.com.br) <http://joselauro.com/42s.pdf>

With contributions by Russ Jones, Manhattan Beach, California

Copyright (C) 2010 José Lauro Strapasson.  
Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".

For more information visit the Free Software Foundation at <http://www.fsf.org>

# Contents

1	Introduction.....	3
2	Basic Operations.....	4
2.1	RPN.....	4
2.2	Turn ON/OFF.....	5
2.3	Setting the display contrast.....	5
2.4	Training RPN using HP-42S.....	5
2.5	Menus.....	5
2.6	DISP Menu.....	6
2.6.1	The FIX function.....	6
2.6.2	The ALL function.....	6
2.6.3	The SCI function.....	6
2.6.4	The ENG function.....	6
2.6.5	RDX. And RDX, functions.....	7
2.7	MODES Menu.....	7
2.8	The Stack.....	7
2.9	Getting used to some keys of the keyboard.....	8
3	Memory.....	10
3.1	The ■ CATALOG menu.....	11
3.2	More on the ■ CLEAR menu.....	11
3.3	The ■ CUSTOM menu.....	12
4	Probability.....	12
5	Complex Numbers.....	13
5.1	Complex numbers in rectangular coordinates.....	13
5.2	Complex numbers in polar coordinates.....	14
6	Programming.....	14
6.1	Basic programming.....	14
6.2	More than one program in the memory.....	16
6.3	The X?0 and X?Y sub-menus.....	17
6.4	Real program examples.....	18
7	Using the Solver.....	19
8	Numeric Integration.....	20
9	Statistics.....	21
9.1	The sub-menu CFIT.....	22
9.2	The second line: ALLΣ, LINΣ, ΣREG and ΣRG? Functions.....	22
10	Matrices.....	23
11	Other Bases.....	26
12	Flags.....	26
13	Free42-Specific Features.....	27
13.1	Keyboard Interface (Windows, Linux and Mac Desktops and Laptops).....	27
13.2	Program Import and Export.....	28
13.3	Printing.....	28
14	Comprehensive Command List.....	28
	GNU Free Documentation License.....	35

# 1 Introduction

Since HP-42S was a very nice calculator, and its official manual is no longer freely available and there were many people looking for its manual, seemed good to me to write my own HP-42S manual. I personally don't have a HP-42S (more than US\$300 on ebay). I have a HP-33S and had a HP-48G, but my brother has one and I also use Free42 simulator for PalmOS.

This manual will be of interest to people who:

- a) Have a HP-42S calculator and lost its manual.
- b) Got the Free42 simulator and want to know how to use it.
- c) Have a palmtop with PalmOS and want a nice scientific calculator (get Free42)
- d) Just want to have an idea how 42S was.
- e) Have the official manual but don't want to read more than 300 pages!

Why HP-42S? Because it was a very, very nice calculator and also a powerful one. I know some other HP models from the past and the present like 48G, 49G, 28S, 33S, 20S, 6S Solar, 15C, and even a TI-36X Solar, etc, but 42S is my favorite. And because there is a free simulator (Free42) that works on Palm OS, Windows and Linux and there are also some emulators (at the moment emulators are only useful for who has a real calculator since HP-42S roms are not freely available). This calculator played an unique position among HP calculators! Being a scientific programmable 100% RPN calculator, it also had some graphing abilities but was pocketed sized and non RPL (some people as me like RPN, but dislike RPL).

It is important to say that this manual is not complete and I don't want it to be. Two things I really don't want to see here are PRINTING and HP-41 compatibility. This because I suppose most owners don't have the printer (and it is not so useful) and also haven't had a HP-41 prior to HP-42S.

If you want to download the fantastic Thomas Okken Free42 program please go to this web site [http://home.planet.nl/~demun000/thomas\\_projects/free42/](http://home.planet.nl/~demun000/thomas_projects/free42/)

In my opinion Free42 is even better than the real HP-42S. Try  $\text{asin}(\text{acos}(\text{atan}(\text{tan}(\text{cos}(\text{sin}(6^\circ))))))$ . For more information about HP-42S please see

<http://www.hp42s.com> <http://www.hpmuseum.org/hp42s.htm>

Here you can find emulators for HP-42S <http://privat.swol.de/ChristophGiesselink> (very nice)

<http://www.geocities.com/hrastprogrammer/HP42X/index.htm>

I would like to finish this introduction saying that it would be nice to have the HP-42S back to life again and even better to have a model (both real and in simulator/emulator form) based on HP-42S but with some of the 33S features like more memory, an equation editor, fractions, program lines starting with letters, physical constants, units conversion, less useless functions, etc. And it also would be nice to have HP-42S ROM images for free just like what happened to HP-48G and other models and keeping PDF versions of the manuals of retired models to download would be nice too. Perhaps someone will listen to me! ☺

A quick note on notation: throughout this manual, for the most part, keys that are to be pressed are denoted by putting them in a box, e.g. **ENTER**, except when the keys are numbers or arithmetic operators. Keys that are “2<sup>nd</sup> functions” denoted by orange lettering on the calculator are denoted in orange with an orange box preceding it, e.g. **■ ALPHA**. Functions that are accessed through the menus are generally denoted by shading in grey, such as in **FCN**.

## 2 Basic Operations

### 2.1 RPN

The HP-42S, like most old HP calculators, is a RPN calculator. RPN comes from “Reverse Polish Notation”. In RPN we first enter data and then we enter the mathematical operations.

Example: To make a simple operation like  $2+2$  in a normal algebraic calculator we do “ $2 + 2 =$ ” which give to us 4. To make this same calculation using a RPN calculator we do “2 **ENTER** 2 +” As we can see in RPN mode we first enter the data pressing the **ENTER** key after every data (except for the last in HP's RPN) and then we enter the operations.

Let's now consider the following calculation

$$4 + (2 \times 79)$$

In a RPN calculator we do

$$2 \text{ **ENTER** } 79 \times 4 +$$

But how could one do this in an algebraic calculator? If the calculator has “(“ and “)” keys we enter

$$4 + ( 2 \times 79 ) =$$

But if there are no parenthesis keys we might do this in a good calculator by doing

$$4 + 2 \times 79 =$$

By a “good” calculator we mean a calculator which knows that “ $\times$ ” and “ $/$ ” have precedence over “ $+$ ” and “ $-$ ”. In a bad algebraic calculator which does not know this we have to do

$$2 \times 79 =$$

and

$$+ 4 =$$

Or

$$2 \times 79 + 4 =$$

What about to calculate  $\sin(33^\circ)$ ? In a RPN calculator we enter

$$33 \text{ **sin** }$$

or if you prefer

$$33 \text{ **ENTER** **sin** }$$

(in this case we don't need to press enter key)

But in an algebraic calculator we have two ways. In the classic old models it is like RPN and we do

$$33 \text{ **sin** }$$

but in some modern models (which typically allow you to edit entered data using cursors) we do

$$\boxed{\sin} 33 =$$

So algebraic calculators are ambiguous because the many ways they work. RPN calculators are more standard and so less ambiguous. The main key to understand how to use RPN in more complex calculus is to realize that in RPN we make calculations from “inside” to “outside” instead of from left to right. For example:

$$8 \times \ln [5 + \sin(40^\circ)]$$

in RPN this is accomplished by

$$40 \boxed{\sin} 5 + \boxed{\ln} 8 \times$$

In RPN calculators, there is no operator precedence — operators are executed immediately and the order of the calculations determines precedence. There is never any need for parentheses. In RPN we can make any calculation we could do in algebraic devices and this is not only more elegant but also more effective since there are less ambiguities and we use less key strokes. For example, my HP-33S, which is both algebraic and RPN, is always in RPN mode. (Just to insert equations I think algebraic mode is better) For more information on RPN, please see <http://www.hpmuseum.org/rpn.htm>

## 2.2 Turn ON/OFF

To turn your HP-42 on press ON. The ON key is the same  $\boxed{\text{EXIT}}$  key. To turn your HP-42S off press  $\blacksquare$   $\boxed{\text{OFF}}$ . OFF is in the same key as  $\boxed{\text{EXIT}}$  and ON, and by  $\blacksquare$   $\boxed{\text{OFF}}$  we mean you have to press the orange key before pressing the  $\boxed{\text{EXIT}}$  key (which has “OFF” in orange above it). The orange  $\blacksquare$  key is what in some other calculators is called “second function”. When you press this all keys turn into what is written in orange above them.

Actually  $\blacksquare$   $\boxed{\text{OFF}}$  is a redundancy since OFF can be only accessed by pressing  $\blacksquare$  first. But (as in the HP-42S official manual) we will do this just to remember when we have to press  $\blacksquare$  or not. If you press this key a second time all keys go back to the normal function.

## 2.3 Setting the display contrast

HP-42S, as most HP calculators, can set the display contrast by pressing at the same time ON and  $\boxed{+}$  or  $\boxed{-}$ .

## 2.4 Training RPN using HP-42S

Now that you have your 42S on try to do the following calculations:

Calculation	Keystrokes
$6 \times (4 + 3)$	4 $\boxed{\text{ENTER}}$ 3 + 6 $\times$
$6 + \{8 \times [2 + (4/3)]\}$	4 $\boxed{\text{ENTER}}$ 3 / 2 + 8 $\times$ 6 +

IMPORTANT: For sake of simplicity sometimes we will use / instead of  $\div$ .

## 2.5 Menus

Not all functions of HP-42S are visible above the keys. It has menus with access to many more functions. The menus are

ALPHA	MODES	DISP	CLEAR
SOLVER	$\int f(x)$	MATRIX	STAT
BASE	CONVERT	FLAGS	PROB
CUSTOM	PGM.FCN	PRINT	TOP.FCN
CATALOG			

## 2.6 DISP Menu

The DISP menu is the first menu we have to see. It is above E key. So start by pressing  $\blacksquare$  **DISP**. When you do this the DISP menu appears in the first line with the following functions.

**FIX**   **SCI**   **ENG**   **ALL**   **RDX**   **RDX**

These functions appear just above the top row of keys  $\Sigma+$ ,  $1/x$ ,  $\sqrt{x}$ , **LOG**, **LN** and **XEQ**. Now with the DISP menu active those keys don't represent their original functions but those of the DISP menu. The same happens with all menus.

### 2.6.1 The FIX function

The FIX “function” is not a function in the mathematical sense, but a calculator function. By using FIX function the display becomes with a fixed number of digits after decimal point. Ok, press FIX. (I mean  $\Sigma+$  with DISP menu active) When you do this what appears is FIX \_\_ Then you have to enter a number up to 11. For example FIX 0 4 sets the calculator to have 4 digits of precision after the decimal point. A number like  $\pi$  will appear as 3.1416 and  $\sqrt{2}$  will appear as 1.4142. (You can verify this by doing  $\blacksquare$   $\pi$  and  $2 \sqrt{x}$  respectively)

If you put FIX 0 9 than those numbers will appear as 3.141592654 and 1.414213562. It is important to say that this is not the actual precision the calculator will have but just the display precision. To see all calculator precision you have to press ALL in DISP menu (above LOG key). By doing so those numbers will appear as 3.14159265359 and 1.41421356237. As you can see the numbers are not truncated but rounded.

Not all numbers can be seen with a fixed decimal precision. If you put 4 digits for fixed precision the number  $\pi$  will appear as 3.1416 but if one calculates  $10^8$  (do this by doing  $8 \blacksquare$   $10^x$  or by entering 1E8) what you are going to see is 100,000,000.000 with 3 decimal digits. This happens because the calculator cannot show more than 12 digits at a same line.

### 2.6.2 The ALL function

We already talked about the ALL function. It makes the calculator to show all of its precision.

### 2.6.3 The SCI function

The SCI function works just like FIX one but puts the calculator in “scientific” mode. The numbers will be shown as a decimal number between 1 and 10 times a power of 10. For example 1000 will be represented as 1.00E3 with you put the calculator in scientific mode with 2 digits. 1.00E3 means  $1.00 \times 10^3$ . The  $\pi$  number will appear as 3.14E0.

Actually even when in FIX mode, the calculator will convert some answers to scientific notation. For example if you calculate  $1.0001 - 1$  with FIX 3 you are not going to get 0.000 but 1.000E-4. This means that the calculator is “smart” and shows the result in the best way as possible.

Exercise. Show that  $1.0001 - 1$  gives 1.000E-4 in FIX 3 mode.

Answer: First we put the calculator in FIX 3 mode by doing  $\blacksquare$  **DISP** **FIX** 0 3. Then we do  $1.0001$  **ENTER**  $1 -$  and we get the answer.

As you can see, when you are in FIX mode a sign  $\blacksquare$  appears on the right side of the **FIX** name in the DISP menu. This means FIX mode is active. The same happens with SCI, ALL, etc.

### 2.6.4 The ENG function

The ENG function puts the calculator in engineering notation. It looks like scientific notation but now the first number does not need to be between 0 and 1 but can be between 0 and 1000 and the power will be always 3 manifold (corresponding to the magnitude prefixes such as milli-, micro-, kilo-, mega-, etc. used in engineering units). For example: 100 will be represented by 100.E0 in ENG 2

mode while 1000 will be 1.00E3 in the same mode. Why do we get 100.E0 for 100 instead of 100.00E2 in ENG 2 mode? Because the calculator shows in engineering mode the same number of digits it shows in scientific mode.

### 2.6.5 RDX. And RDX, functions

In some countries like Brazil we use ',' for the decimal point instead of '.' and also '.' instead of ',' for thousands separators. For example  $\pi$  is written here (Brazil) as 3,141 etc and not as 3.141 etc. In FIX 3 mode one million is written here as 1.000.000,000 and not as 1,000,000.000 as in English use. By pressing RDX, you make the calculator to use ',' for the decimal point and by pressing RDX. we make it use '.' for decimal point. Again the active mode is followed by a ■ sign. Here, in this manual, I assume the calculator is using '.' for decimal point.

## 2.7 MODES Menu

To access MODES menu just press ■ **MODES**. (MODES is above  $\boxed{+/-}$  key).

DEG actives degree mode for trigonometric functions. In this mode a circumference has  $360^\circ$ . RAD actives radian mode and in this mode a circumference has  $2\pi$  radians or just  $2\pi$ . GRAD is not so useful and correspond to 400 degrades for a circumference. For example: In degrees we have  $\sin(90^\circ)=1$  and in radians we have  $\sin(\pi/2)=1$ .

Try this: ■  $\boxed{\pi} 2 / \boxed{\text{COS}}$  in radians mode. Why the result is not exactly zero?

Answer: Because the number that calculator entered was not exactly  $\pi$  but 3.14159265359.

REC actives rectangular mode (x,y) and POLAR actives polar mode (r, $\theta$ ). We will see this more in detail when study complex numbers.

The MODES menu has another line but we will discuss this later. We will discuss the others menus later too.

## 2.8 The Stack

The stack is intimately related to the way the calculator uses RPN to perform calculations so it's a good idea to understand the concept and behavior of the stack. On the HP42S, the stack consists of 4 registers named X, y, z and t, and normally the values of x and y (or just x if a menu is active) are displayed.

No calculator can store an infinite amount of data. In algebraic calculators the "( )" are limited to a given number depending on the model. The same happens in RPN calculators. In some models like HP-48 or HP-49 the amount of input data is limited only by available memory. But in other models like 32SII, 33S (in RPN mode) and 42S the input data have to fit in a "stack" of four lines. There are four lines labeled x, y, z and t. So the stack is something like

t: 0.0000  
z: 0.0000  
y: 0.0000  
x: 0.0000

But since the calculator's display has only two lines just x and y lines are visible. When you enter a number (say 2  $\boxed{\text{ENTER}}$ ) what happens is the following.

- i) The content of lines t and z are lost.
- ii) The content of line y goes to line t.
- iii) The content of line x goes to line z.
- iv) The content just entered goes to line y and line x.



So what you just entered appears twice. So if you do  $2 \overline{\text{ENTER}} +$  you will have 4 as answer. This is a feature, a bad feature I think, of the HP RPN style used by the 42S (also in the 33S, 12C, etc but not in the HP48 or 49). In my opinion we could have a simpler RPN style. Anyway there is another way to enter data in RPN, namely you just type the number and then press the desired function key. For example, if you do  $2 \overline{1/x}$ , the calculator makes an automatic ENTER before the 1/x function but in this case the content just entered appears only once. So if you do  $2 \overline{1/x}$  or another example  $9 \overline{\sqrt{x}}$  what you will have will be

- i) Only the content of the t register will be lost.
- ii) The content of the z register goes into the t register.
- iii) The content of y goes into z.
- iv) The content of x goes into y.
- v) Your result will be in the x register.

This second way to enter data looks more intuitive to me and I think it should be always like this. But it is not! (So to do  $2+3$  we have to do  $2 \overline{\text{ENTER}} 3 +$  (and not  $2 \overline{\text{ENTER}} 3 \overline{\text{ENTER}} +$ ). (Actually one can also use  $\overline{\text{EXIT}}$  to enter a number without duplication). If you just press  $\overline{\text{ENTER}}$  you duplicate what is in register x. When making a calculation one should never forget about the limitation of the 4 lines of the stack. The lines of the stack cannot contain only numbers but also matrices, complex numbers, etc.

Two basic operations with the stack are:  $\overline{x \leftrightarrow y}$  and  $\overline{R \downarrow}$ . The first exchanges the value in register x with the value in register y. The second makes the stack “roll down” (t goes to z, z goes to y, y goes to x, and x rolls around to t).

In the CLEAR Menu there are some interesting functions: CLST which clears all the stack (something missing in HP-33S). CLX clears the line x in the same way of pressing  $\overline{\leftarrow}$ . The  $\overline{\leftarrow}$  is more used to correct a number when typing it. Another useful function is  $\blacksquare \overline{\text{LASTx}}$  which gives the last calculated result.

## 2.9 Getting used to some keys of the keyboard

Let's discuss some basic keys of the calculator. We will start from upper left side.  $\overline{\Sigma+}$  and  $\blacksquare \overline{\Sigma-}$ : These are statistical functions. We will discuss this later.

$\overline{1/x}$  and  $\blacksquare \overline{y^x}$ : The  $\overline{1/x}$  key just calculates the inverse of a number which is in register x.  $\blacksquare \overline{y^x}$  is the potential function. To calculate  $5^3 = 5 \times 5 \times 5$  we do  $5 \overline{\text{ENTER}} 3 \blacksquare \overline{y^x}$ .

$\overline{\sqrt{x}}$  and  $\blacksquare \overline{x^2}$ : These functions just calculate the square root and the square of a number in x. When studying complex numbers we will see that unlike the HP-33S, in HP-42S the number in square root can be negative.

$\overline{\text{LOG}}$  and  $\blacksquare \overline{10^x}$ : These functions calculate the base 10 logarithm and its inverse.

$\overline{\text{LN}}$  and  $\blacksquare \overline{e^x}$ : These functions calculate the natural (base  $e=2.71828\dots$ ) logarithm and its inverse. If we want a logarithm in another base, we can use the relation  $\log_z y = \log y / \log z$  where z is any other base. If we take  $z = e = 2.71828$  then we have  $\log_x y = \ln y / \ln x$ .

Example: Calculate  $\log_2 8$

Answer:  $8 \overline{\text{LN}} 2 \overline{\text{LN}}$  / which gives us 3 because  $2^3 = 8$ .

$\overline{\text{XEQ}}$  and  $\blacksquare \overline{\text{GTO}}$ : These are related to programming and we shall discuss this later. XEQ will also be discussed in ALPHA menu part.



**STO** and **COMPLEX**: These are related to the memories and complex numbers. We will discuss this later.

**RCL** and **%**: **RCL** is related to memories and we will discuss later. **%** is the percentage function. To calculate 10% of 300 we do 300 **ENTER** 10 **%** which gives 30 as the answer. Note that 300 remains in line y, so if you want to calculate 300 plus 10% you do 300 **ENTER** 10 **%** +

**R↓** and **π**: We already discussed these. The first “rolls down” the stack and the other returns  $\pi=3.14\dots$

**SIN** and **ASIN**: These are the sine trigonometric function and inverse. The angle type is set up as said before in the MODES menu. The default is degrees. **ASIN** is the inverse usually called arcsine or sometimes  $\sin^{-1}$  (not to be confused with cosecant which is  $1/\sin$ ). It is important to remember that **ASIN** is not a real function since there is no single result. For example  $\sin(135^\circ)=\sin(45^\circ)=\sqrt{2}/2$  but the calculator gives always  $\text{ASIN}(\sqrt{2}/2)=45^\circ$ . HP-42S will give a complex number if the input of an arcsinus is bigger than 1 or smaller than -1.

**COS** and **ACOS**: These are the cosine trigonometric function and inverse.

**TAN** and **ATAN**: These are the tangent trigonometric function and inverse. Not all numbers can have a result for tangent. For example  $\tan(90^\circ)$  goes to infinity. The HP-42S gives a big number instead.

**ENTER** and **ALPHA**: The **ENTER** key does not need any comment. **ALPHA** is the alphanumeric menu used to enter letters instead of numbers. When you press **ALPHA** what appears is

ABCDE FGHI JKLM NOPQ RSTUV WXYZ

These are sub-menus. If you press now ABCDE what you will have is

A B C D E

Then just pick the letter you want. But above you can see this symbol  $\blacktriangledown\blacktriangle$ . This symbol means the menu has more than one line. You can access the other lines by pressing  $\blacktriangle$  or  $\blacktriangledown$ . In this case there is just one more line with  $\text{Å}$ ,  $\text{Ä}$  and  $\text{Æ}$ . If you press **FGHI** you will have F G H I, etc. Among all calculators I know this is in my opinion the best way to enter letters! The main **ALPHA** menu also has a  $\blacktriangledown\blacktriangle$  symbol. The other line has the following submenus.

{ } ← ↑ ↓ <=> MATH PUNC MISC

Much more than one will ever need! If you are inside a submenu and want to go back to the main menu just press **EXIT**. Why is the **ALPHA** menu useful? Of course it is useful to label programs and data in memory, but it is also useful to enter commands using the **XEQ** key! For example **XEQ** “SIN” is the same of pressing the **SIN** key. The “” are called automatically when pressing **ALPHA** and **ENTER**. **XEQ** “SINH” calculates the hyperbolic sine while **XEQ** “OFF” turns the calculator off. Finally we must say that **ALPHA** is not always needed! In some cases like **XEQ** and **GTO** (we will see this later) a simple **ENTER** will do.

Entering alphabetic text is even easier with Free42. Free42 allows you to just type on the native keyboard when the **ALPHA** menu is activated.

**x<>y** and **LASTx**: We already talked about these.

$\pm/\text{-}$ : This just changes the sign of a number.

$\text{E}$  and  $\blacksquare$  **DISP**: We already talked about DISP menu. The E is the character meaning the power of 10 in scientific notation. For example, to enter  $5.2 \times 10^{22}$  we do 5.2  $\text{E}$  22  $\text{ENTER}$ .

$\leftarrow$  and  $\blacksquare$  **CLEAR**: As said before,  $\leftarrow$  clears line x and if you are entering a number you can delete the last character. We already talked a little about CLEAR menu and we will discuss it again later.

$\blacktriangle$  or  $\blacktriangledown$ : As said before we use this to change the line in a multi line menu. We will see  $\blacksquare$  **BST** and  $\blacksquare$  **SST** later.

The keys from 0 to 9 have obvious functions.

$\text{.}$  and  $\blacksquare$  **SHOW**: The '.' is just the decimal point and  $\blacksquare$  **SHOW** is used to show a number for an instant with all precision. For example: If you have  $\pi$  in the first line and you are using the display in FIX 4 you have 3.1416 but pressing  $\blacksquare$  **SHOW** you will see 3.14159265359 for an instant.

### 3 Memory

The real HP-42S has about 7200 bytes of memory while Free42 can have much more depending on the available memory in the computer/handheld. In fact, 7200 bytes is a lot of memory for the HP-42S! A program of 10 lines uses about 15 bytes of memory. This means that, while in some other models like the HP-20S you would be able to program just 99 lines, with 42S you would be able to create programs with thousands of lines!

This available memory is shared with everything including programs, variables, etc. Let's start from the basic. To store a number which is in register x of the stack we use the STO function. The HP-42S has by default 25 positions in the memory from  $R_{00}$  to  $R_{24}$ . To store the number  $\pi$  in  $R_{10}$  just do the following:  $\blacksquare$   $\pi$   $\text{STO}$  10 To get it back it is just do this:  $\text{RCL}$  10.

If you want to make an operation you can use  $\text{STO+}$ ,  $\text{STO-}$ ,  $\text{STO}\times$ ,  $\text{STO}\div$ . Any of these operations can be entered by pressing the STO key followed by the operator key, followed by a register number or name. For example, 6  $\text{STO}$   $\text{-}$  05 subtracts 6 from the number in  $R_{05}$ . 2  $\text{STO}$   $\text{\div}$  10 divides the number in  $R_{10}$  by 2. You can also use  $\text{RCL+}$ ,  $\text{RCL-}$ ,  $\text{RCL}\times$ ,  $\text{RCL}\div$ , but it is not so fun. This gives the result of the calculation but does not change the number in the memory.

If 25 positions in the memory is not enough for you, you can change this number by using the SIZE function (which is in the second line of the MODES menu). For example  $\blacksquare$  **MODES**  $\blacktriangledown$  SIZE 0100 changes to have 100 positions, from  $R_{00}$  to  $R_{99}$ . Although it is possible, I suggest you should not use more than 100 positions. These positions are stored in a normal matrix called REGS (we, the poor owners of the HP-33S for example, just have 26 memory positions, from A to Z).

But this kind of memory position only accept real numbers! What about if you want to store other things? Matrices, complex numbers or even other real numbers? To do this HP-42S has an arbitrary number of positions, limited only by the memory available, which use letters to label the positions instead of numbers. We had stored the  $\pi$  number in  $R_{10}$  but we can create a variable called, for example, "PI" to store it. To do so we just do  $\blacksquare$   $\pi$   $\text{ENTER}$   $\text{STO}$   $\blacksquare$  **ALPHA** "PI"  $\text{ENTER}$ .

Actually is not just PI you type but **NO PQ P FGHI I** but we wrote that for simplicity. Now to get this number back it is just type  $\text{RCL}$  "PI". When you type  $\text{RCL}$  the "PI" should appear for you to select it. More generally, the STO and RCL functions automatically bring up a menu of previously

defined variables currently active in the calculator, and you can use the arrow keys if there are more than will fit on one screen.

You can also use STO+, STO-, STO× and STO÷ even in this case since the types of the things you are operating are the same.

We can deal with the four registers of the stack as we deal with the memory positions. In this case the lines of the stack are called ST X, ST Y, ST Z and ST T respectively. To access this we press '!' before the name of the register. For example: 5 **STO** **ST X** puts 5 in line x of the stack. The submenu that is displayed when we press '.' Actually has two other items, ST L and IND. ST L refers to the LASTx register, and IND is used for indirect parameters.

As the content of the stack can change easily I don't think "STO ." is a good thing. But I cannot say the same of "RCL ." which may be very useful to get the content especially of registers z and t. You can also use STO and RCL with +, -, x and ÷ and '!' to work with the content of the registers of the stack. For example: 5 **STO** **ST Z** divides register z by 5.

We can use an indirect parameter by pressing **IND** when using STO or RCL or any other calculator function that happens to allow indirect parameters. With indirect addressing, we specify a location where the actual parameter is stored, rather than the parameter itself. That location could be a named variable, one of the numbered storage registers, or a stack register. For example, to assign the value 125 to the register specified in the variable ABC:

10 <b>STO</b> "ABC"	sets variable ABC to the value 10
125 <b>STO</b> <b>IND</b> "ABC"	stores 125 in the register pointed to indirectly by "ABC"
<b>RCL</b> 10	returns the value 125 to the x-register

### 3.1 The **CATALOG** menu

The **CATALOG** menu has the following submenus:

**FCN** **PGM** **REAL** **CPX** **MAT** **MEM**

**FCN**: It shows all the functions available in HP-42S calculator. It has many lines and one must use the ▼ and ▲ to navigate through the lines. Here you are going to find important functions we don't see in the keyboard including hyperbolic functions (SINH, COSH, etc), functions to work with integer and real numbers like IP (integer part) and FP (fraction part), programming functions, etc. Don't forget you can also use XEQ "function name".

**PGM**: It shows all variables with programs in the memory.

**REAL**: It shows all variables with real numbers in the memory. (But does not show numbers in the numbered registers R<sub>00</sub>, etc)

**CPX**: It shows all variables with complex numbers.

**MAT**: It shows all variables with matrices. The REGS matrix always appears. It contains the numeric memories R<sub>00</sub>, R<sub>01</sub>, etc.

**MEM**: It shows all available memory.

### 3.2 More on the **CLEAR** menu

We already saw some of the CLEAR menu functions, but there are also:

**CLV**: Clears variables we had stored using STO “name”.

**CLRG**: Clears the  $R_{00}, R_{01}, \dots$  memories known as registers.

**CLLCD**: Clears the LCD display (may be useful when plotting)

**CLALL**: Clears all the memory of the calculator.

### 3.3 The **CUSTOM** menu

This is not really related to memory, but as we have just discovered the FCN menu within the CATALOG menu, now is a convenient place to talk about it.

The HP-42S calculator has a lot of functions. So many, in fact, that it is inconvenient to find the function you want every time in the FCN menu or to use XEQ “function name” every time. To solve this problem HP-42S has the CUSTOM menu which can contain functions or user-written programs you personally select. To do this we use **ASSIGN**. When you call this you can select a function from FCN and also some other things. For now we are interested in functions so press FCN. Now you find the function you want and then you press the position you want it to appear in the CUSTOM menu.

Example: Let's put ABS (absolute value) in the first position of CUSTOM menu.

**ASSIGN** FCN ABS

In the display you are going to see:

ASSIGN “ABS” TO \_

Then you pick a position, for example initially the CUSTOM menu is empty and you have

■ ■ ■ ■ ■ ■

and you press the first ■ your CUSTOM menu will become

ABS ■ ■ ■ ■ ■

As you can see the CUSTOM menu has also the  $\blacktriangledown\blacktriangle$  symbol which means there are more than one line. There are three lines you can use when calling ASSIGN function which means 18 available positions.

(I would like to use this space to make a complaint) There are some HP models with more than 2000 functions! Many functions does not always mean power but does always mean complexity!

## 4 Probability

Probability functions are in **PROB** menu (over the  $\times$  key). They are COMB, PERM, N!, GAM, RAN and SEED.

**COMB**: This calculates the number of combinations of  $N$  things taken  $r$  at a time (mathematically notated as  $C_r^N$ ). The order does not matter. A thing cannot appear more than one time.

Example: If we have the five letters  $a, e, i, o$  and  $u$  the possible combinations taken one at a time are  $\{a, e, i, o, u\}$  or 5 combinations.

Taken two at a time: {ae, ai, ao, au, ei, eo, eu, io, iu, ou} or 10 combinations.

Taken four at a time {aeio, aeiu, aeou, aiou, eiou} or 5 combinations.

The number of combinations C is given by

$$C_r^N = \frac{N!}{r!(N-r)!}$$

where  $N! = N \times (N-1) \times (N-2) \times \dots \times 2 \times 1$ . To calculate this using 42S just enter  $N$ , press **ENTER**, enter  $r$  and press **PROB** **COMB**.

**PERM**: This calculates the number of permutations of  $N$  things taken  $r$  at a time (mathematically notated as  $P_r^N$ ). A thing cannot appear more than one time but now the order matters.

Example: Five cars are in a race. Their colors are red, blue, green, white and cyan. What are the possible results for the first, second, and third place winners?

Solution: For the first position we have five possibilities. For the second position we have four possibilities, and three possibilities for the third position. So we have  $5 \times 4 \times 3 = 60$  different arrangements. To see this using 42S just enter 5, press **ENTER**, enter 3 and press **PROB** **PERM**. It is simple to realize that the number of permutations is given by

$$P_r^N = \frac{N!}{(N-r)!}$$

In particular if  $r = N$  (all the things are taken) then the number of permutation is  $N!$ .

Example: In how many ways we can re-arrange the letters of the word "love".

Solution:  $4! = 24$ .

**N!**: This just calculates the factorial of  $N$  given by  $N! = N \cdot (N-1) \cdot \dots \cdot 1$  for a number (non-negative integer). The biggest number allowed is HP-42S is 253 and in Free42 is 170.

**GAM**: This is the *Gamma function* which is defined by

$$\Gamma(a) = \int_0^{\infty} x^{a-1} e^{-x} dx$$

For a integer number we have  $\Gamma(n) = (n-1)!$  and  $\Gamma(n+1) = n!$  but the argument of the gamma function can be a non-integer (but must be real). In this point HP-42S is different from the 33S which has only one function for both things.

**RAN**: This is the *random number generator* which gives a pseudo-random number in  $0 \leq x \leq 1$ .

**SEED**: A sequence of pseudo-random numbers always starts with a seed. If you repeat the seed the sequence repeats. To enter a new seed just enter a number and press SEED. If the seed is zero the calculator will generate another seed.

## 5 Complex Numbers

### 5.1 Complex numbers in rectangular coordinates.

Unlike the HP-33S (and its ancestor HP-32SII) complex numbers are straightforwardly supported and used in the HP-42S. There is almost nothing special to say. Just enter  $-1$  and press **√x**, what are you going to have is  $x: 0.0000 \text{ i} 1.0000$  which means  $i$ . (Just for comparison, to do the same in HP-33S we have to do **ENTER**  $1$  **+/-** **ENTER**  $0$  **ENTER**  $.5$  **CMPLX** **y<sup>x</sup>** and we will have  $0$  and  $1$  meaning  $i$ )

Yes it is possible, but who wants to calculate the square root of  $-1$  every time, to have  $i$ ?

We can use the **COMPLEX** function to take register  $y$  and register  $x$  of the stack and create a complex number  $y + ix$ . Again unlike HP-33S almost all the functions of the HP-42S fully support complex numbers.

Example: Show that  $i^2$  is  $\sqrt{-1}$ .

Solution: 0 ENTER 1 **COMPLEX** **x<sup>2</sup>** which gives -1.0000 i0.0000 (means -1).

## 5.2 Complex numbers in polar coordinates

When representing a point in  $R^2$  we can use any kind of coordinate system. The most commonly used are the rectangular (or Cartesian system) which use the usual coordinates  $x$  and  $y$  and the polar system which use the coordinates  $r$  and  $\theta$ . The relationship between them is  $x = r \cos \theta$ ,  $y = r \sin \theta$  and  $r = (x^2 + y^2)^{1/2}$ ,  $\theta = \tan^{-1} y/x$ . When dealing with complex numbers we can think of the real axis as being the  $x$  axis and the imaginary axis as being the  $y$  axis in Cartesian coordinates, or we can use also polar coordinates. In this case  $i$  will be  $r = 1$  and  $\theta = \pi/2$  ( $90^\circ$ ). To change between rectangular or polar modes use RECT and POLAR in the **MODES** menu.

## 6 Programming

Programming the HP-42S is very simple and very versatile. It does not use the RPL style of the HP-48 or HP-49. You program in the same way you use the calculator and unlike some non-HP cheaper calculators, all the steps are shown in the display and in numbered lines.

### 6.1 Basic programming

Let's imagine you want to make a given calculation. For example: Suppose you want to solve a equation  $x^2 - 5x + 4 = 0$  which is of the form  $ax^2 + bx + c = 0$ . As you know the solution for this kind of equation is

$$x = \frac{-b \pm \sqrt{\Delta}}{2a}$$

where  $\Delta = b^2 - 4ac$ . Let's suppose  $a$ ,  $b$  and  $c$  are in  $R_{00}$ ,  $R_{01}$  and  $R_{02}$  respectively and we are going to use  $R_{03}$  for  $\Delta$ . To solve this equation using HP-42S/Free42 we just do

RCL 01 (This is b)

$x^2$

4

RCL 00 (This is a)

RCL 02 (This is c, keep in mind we have only four lines in the stack)

$\times$

$\times$

$-$

STO 03 (This is  $\Delta$ )

Unlike some other models, say 33S, we don't need to worry whether  $\Delta$  is negative. But we save the square root for later because in  $R_{03}$  the number cannot be complex. (otherwise we would need to store it in a normal memory)

Now we calculate the first root

RCL 01

```

+/-
RCL 03
√x
-
2
RCL 00
×
÷

```

And the second root is given by

```

RCL 01
+/-
RCL 03
√x
+
2
RCL 00
×
÷

```

So what about if you have to solve hundreds of this kind of equation? Only changing the  $a$ ,  $b$  and  $c$  values? It would be better to save all the steps in the calculator's memory and let it do the calculations for you. This is what calculator programming is about.

To enter in the program mode you must do **PRGM** (above the **R/S** key). If the memory has no programs yet, you are going to see:

```
00▶ { 0-Byte PRGM} 01 .END.
```

(If there is a program we can erase it by doing **CLEAR** CLP before entering in program mode). Now just enter the first sequence starting in RCL 01  $x^2$ , etc. Every command will take a line and in the end you will have 08 09▶ STO 03

This means that this part of the program takes 9 lines. You can move through the program lines by using the **▲▼** cursors (which, of course, cannot be programmed). Two important things to say here are:

1. The functions are not always shown in the calculators display as we know them. For example the  $x^2$  function is showed as  $X\uparrow 2$ .
2. We don't need to press **ENTER** after a number, unless it's between two numbers.

Now let's enter the second part of the program which gives to us the first root. (if you used the cursors you must go back where you stopped). After doing so we have

```
17 ×
18▶ ÷
```

Again in the display the functions are not exactly as we know and  $\sqrt{x}$  appears as SQRT.

Unless we store the result in a memory we must find a way to stop the program to see the result. This is doing by the function STOP which is entered by pressing **R/S**. (R/S means "run and stop") So after this we have



18 ÷  
19 ► STOP

Finally we enter the last part of the program and after this we have

27 ×  
28 ► ÷

If you move using the cursors you will find .END. in line 29 (which is the end of the program) and in line zero we find 00 ► { 31-Byte PRGM }. Almost 1 byte per line of program.

As we said the HP-42S has about 7200 bytes of memory. Not bad! Just for comparison, the HP-32S had 390 bytes and spent about 1.5 bytes per line. The HP-20S had only 99 lines/steps and the HP-9G had 400 steps while HP-33S has 31KB (but hardly can take advantage of this due to a limit of 26 memories/labels, which is the same of 32S, and it spends about 3 bytes per line).

After entering the program just press EXIT. Now enter the numbers  $a$ ,  $b$  and  $c$  of the equation into  $R_{00}$ ,  $R_{01}$  and  $R_{02}$ . For example for the equation  $x^2 - 5x + 4 = 0$  we enter 1  $\boxed{\text{STO}}$  00 5  $\boxed{+/-}$   $\boxed{\text{STO}}$  01 4  $\boxed{\text{STO}}$  02. Now we just press  $\boxed{\text{R/S}}$  (to run the program) and we get 1, and pressing it again we have 4.

## 6.2 More than one program in the memory

If we want to have more than one program in the memory we can use more than one program space. To create another program space just press  $\blacksquare$   $\boxed{\text{GTO}}$   $\boxed{.}$   $\boxed{.}$  .

The  $\blacksquare$   $\boxed{\text{GTO}}$  command can be used in two different situations:

1. You are not in the programming mode. In this case you can use  $\blacksquare$   $\boxed{\text{GTO}}$   $\boxed{.}$   $\boxed{.}$  to create another empty program space, but this happens only if the current mode is not already empty.
  - You can use  $\blacksquare$   $\boxed{\text{GTO}}$  "label". (We will see this below)
  - You can also use  $\blacksquare$   $\boxed{\text{GTO}}$  followed by "END" or ".END." etc to move among program spaces. (In this case I must admit 33S is better because the lack of this complication)
2. You are in programming mode. In this case you cannot change the program space.
  - You can use  $\blacksquare$   $\boxed{\text{GTO}}$   $\boxed{.}$   $\boxed{.}$   $\boxed{.}$   $\boxed{.}$  to move to a line where in the "  $\boxed{.}$  "s you put the number of the line where you want to go. (This will happen and it is not programmed)
  - You can use  $\blacksquare$   $\boxed{\text{GTO}}$  "label" (This will be programmed and will cause the program when executed to jump to that label)

But what is a label? A label is a name we give to a position in the program using the LBL command which is available in  $\blacksquare$   $\boxed{\text{PGN.FCN}}$  ("program functions") menu. To create a label you must be in the programming mode ( $\blacksquare$   $\boxed{\text{PRGM}}$ ) and then just press  $\boxed{\text{LBL}}$  and then enter a name (1 to 7 letters). If you use only one letter it is local to the current program and not visible elsewhere, and thus won't appear automatically when you press  $\boxed{\text{XEQ}}$  (just for comparison, in the 33S all labels are just one letter).

Example: In the programming mode  $\blacksquare$   $\boxed{\text{PGN.FCN}}$  LBL AAA creates a label "AAA" which appears as LBL "AAA" in the program. So when the program is running a statement such as GTO "AAA" is encountered (for example), the program will jump to the line which has the LBL "AAA" instruction. (Please note we don't need to press  $\blacksquare$   $\boxed{\text{ALPHA}}$  to access the A, B, C, etc in this case). For example:

01 LBL "AA"  
02 GTO "AA"  
03 .END. (you don't enter this)

This program does nothing. It just runs until you press **EXIT**. By the way, to run it you can use **R/S** when the calculator's "pointer" is over the program or you can use **XEQ** "label". In the present case you would use **XEQ** "AA". The **XEQ** function calls a program (which must have a label) and runs it. You can use the **XEQ** function both in programming mode and also out of programming mode.

When in programming mode the XEQ function is programmed, and when the program finds the XEQ function, it changes to the given program which must finish with the RTN function. So LBL "label" and RTN makes a kind of procedure and after the procedure is run it goes back to the previous position. For example:

```
01 LBL S
02 +
03 RTN
04 LBL A
05 5
06 ENTER
07 7
08 XEQ S
09 1
10
11 .END.
```

This program called "A" creates a procedure "S" which does only a simple addition. In the line 07 the XEQ "S" makes the program to go to the procedure "S" and after that it goes back to the line next line 07 which is of course the line 08. The calculation is 5+7-1 which gives 11.

You can use programmed GTO and XEQ even to call a label in another program space but this is not exactly a good use.

### 6.3 The X?0 and X?Y sub-menus

Up to now we saw nothing about how we could do a IF instruction, like what we have in computer programming languages like BASIC, Pascal or C. In fact there is no IF, THEN, ELSE, ELSEIF, etc in the HP-42S programming language but there are 12 test functions which are:

**X=0?** **X≠0?** **X<0?** **X>0?** **X≤0?** **X≥0?**

and

**X=Y?** **X≠Y?** **X<Y?** **X>Y?** **X≤Y?** **X≥Y?**

The first group of functions involving the number 0 is accessed by the **X?0** sub-menu which is available in the second line of the **PGN.FCN** menu. The second group is accessed by the **X?Y** sub-menu also in the second line of the **PGN.FCN** menu.

How do these functions work? Let's consider the first function "X=0?". If the number in the x register of the stack is zero then the program works normally and it goes to the next line after the "X=0?" instruction. But if the condition is not true then the program jumps the next line and goes to the second line after the instruction. Usually the line after the instruction has a GTO "label" command and this makes the difference in the program flow.

All the other functions involving the 0 work in the same way. If the condition is true the program works normally and if not the program jumps one line. The X?Y functions work in the same way but now the condition is about the registers x and y of the stack and not only about register x.

Example: A kind of “timer”

```
01 LBL A
02 1
03 -
04 X=0?
05 STOP (Enter R/S)
06 GTO A
07 RTN
```

In this program you first enter a big integer number and then press  $\boxed{\text{XEQ}}$  “A”. The program will subtract 1 from this number until it gets to zero. Of course the bigger the number the bigger the time the program will spend. The Emu42 program (yes, I use it too) in my laptop using “Authentic Calculator Speed” option takes about 37s for the number 1000. Without this option, or using Free42, it is too much faster!

#### 6.4 Real program examples

Here is one of my favorite programs. It just see if a number is prime. (With a small change this works in the 33S too).

```
01 LBL “PRIME”
02 STO 00
03 2
04 STO 01
05 MOD (Rmdr in HP-33S)
06 X=0?
07 GTO F
08 3
09 STO 01
10 RCL 00
11 SQRT
12 STO 02
13 LBL B
14 RCL 00
15 RCL 01
16 MOD
17 X=0?
18 GTO F
19 2
20 STO + 01
21 RCL 02
22 RCL 01
23 X≤Y?
24 GTO B
25 RCL 00
26 STO 01
27 LBL F
28 RCL 01
29 RTN
```

Because the HP 42S programming format is a superset of that of the immensely popular HP 41C and HP 41CX calculators, there is a huge library of programs that can be used directly by the HP42S or readily adapted. A big collection can be found at <http://www.hpmuseum.org/software/soft41.htm>.

## 7 Using the Solver

Unfortunately the HP-42S does not have an equation editor like that of the 33S. To use the solver and numeric integration we must enter the equation in a program which must have a global name.

Let's suppose we want to solve the equation  $x^2 - 5x + 4 = 0$ . We are going to enter it in a program. For example:

```
01 LBL "FX" ("FX" is the global name of the program)
02 MVAR "X" (You find this in Solver menu. I will explain this later)
03 RCL "X"
04 X↑2
05 5
06 RCL "X"
07 ×
08 -
09 4
10 +
11 END
```

Well, as you can see we don't enter the equation  $f(x)=0$  but just the function  $f(x)$ . The MVAR function tells the calculator what variables must appear in the solver menu. We suppose all variables are in the memory so we use the RCL function.

Now we leave the program mode and we go to the Solver menu. What should appear is

Select Solve Program

followed by a menu of available Solver programs. Then in our case we select **FX** and we give a start value, for example 8 and we press **X** to enter this value. Again we press **X** to calculate the correct value of  $x$  which gives us  $X=4$ .

But this equation does not have only one solution.  $X=1$  is also a solution. To get it we can enter 2 for example for the start value.

If you want to solve numerically many equations of the form  $ax^2 + bx + c = 0$  you can write a program such as

```
01 LBL "FX"
02 MVAR "A"
03 MVAR "B"
04 MVAR "C"
05 MVAR "X"
06 RCL "X"
07 X↑2
08 RCL "A"
09 ×
10 RCL "B"
11 RCL "X"
12 ×
13 +
14 RCL "C"
15 +
16 END
```

When we leave the program mode and go to the Solver menu again we select FX program and what we are going to see is

A B C X

Now just enter the values of A, B, C and a start value for X and we are done.

Some interesting things to say are:

1. We can't find complex solutions.
2. In this particular case we are not limited to the case  $a \neq 0$ .
3. For polynomial equations it is more generally useful to write a more complex equation like  $ax^4 + bx^3 + cx^2 + dx + e = 0$ . We can set the coefficients of the higher order terms to zero if we want to solve a lower order polynomial.
4. In any equation we are not limited to find one specific variable, say X, of course we can find any missing variable.
5. We don't need to use the solver only for "complex" hard to find solution equations. We can use the solver just to automate some easy calculations.
6. We ordinarily do not need to enter a starting guess for the variable we are solving for — just press that menu button without keying a value first, and the Solver will solve for it.

Example: Consider the ideal gas equation  $PV=nRT$  where R is 8.3144472 J/mol . K. We can write a program like

```
01 LBL "GAS"  
02 MVAR "P"  
03 MVAR "V"  
04 MVAR "N"  
05 MVAR "T"  
06 RCL "P"  
07 RCL "V"  
08 ×  
09 RCL "N"  
10 RCL "T"  
11 8.3144472  
12 ×  
13 ×  
14  
15 RTN  
16 END
```

So we will have in the solver menu P V N and T. If we want to know how many moles of an ideal gas is inside a container of 1L at a 1000Pa pressure and at 300K all we have to do is 10000 **P** 0.001 **V** 300 **T** and we give a try for N, for example 1 **N** and then pressing **N** again we have 0.0040 moles.

## 8 Numeric Integration

Suppose we want to solve numerically a integral of the form

$$\int_a^b f(x) dx$$

We write the function in the same way we did in the solver case.

Example: Calculate  $\int_0^1 x^2 dx$

First we enter the function as a program

```
01 LBL "FX"  
02 MVAR "X" (also available in ∫f(x) menu)  
03 RCL X  
04 X↑2  
05 RTN  
06 END
```

Then we go to the  $\int f(x)$  menu and what we see is "Select  $\int f(x)$  Program". In our case we select **FX** then what appears is "Set Vars; Select  $\int$  var" and we select **X** (before that we may set some variables/constants if needed) and then we have to set **LLIM** (lower limit), **ULIM** (upper limit) and **ACC** (Accuracy factor). In our case LLIM=0 and ULIM=1 and for ACC we can use 0.001. Now just press  $\int$  and we find 0.33333028. If we set ACC to 0.000001 we will have 0.3333333333. In the Free42 program the results may be little different.

## 9 Statistics

To enter statistical data we use the function  $\Sigma+$ . For one variable statistic just enter the number and press  $\Sigma+$ . For two variables statistic first enter y, press **ENTER** and then enter x (without pressing **ENTER**) and finally press  $\Sigma+$ . To clear statistics data just do **CLEAR** **CLΣ**.

The statistical functions are available in **STAT** menu and they are:

**$\Sigma+$** : The same of pressing  $\Sigma+$ .

**SUM**: Calculates the sum of the statistical data.

**MEAN**: Calculates the mean of the statistical data.

**WMN**: Calculates the weighted mean (x weighted by y).

**SDEV**: Sample standard deviation. (To calculate the Population standard deviation you must calculate the mean, add it to statistical data and then press SDEV)

**CFIT**: This is a sub-menu to fit the data in linear regression, logarithm regression, exponential regression or power regression.

There is also a second line with **ALL**, **LINΣ**, **ΣREG** and **ΣRG?** Functions.

Example: A particle of mass  $m_1 = 1$  kg is at the position  $x_1 = 2$ m while a particle of mass  $m_2 = 3$  kg is at position  $x_2 = 6$ m. What is the position of the center of the mass?

Solution: The position of the center of the mass is given by

$$x = \frac{m_1 x_1 + m_2 x_2}{m_1 + m_2}$$

This is a weighted mean where the physical mass is the statistical weight. To calculate it using HP-42S we do:

**CLEAR** **CLΣ** (clearing statistical data)

1	(This is $m_1$ )
<b>ENTER</b>	
2	(This is $x_1$ )
<b>Σ+</b>	
3	(This is $m_2$ )
<b>ENTER</b>	
6	(This is $x_2$ )
<b>Σ+</b>	
<b>STAT</b> <b>WMN</b>	

Which gives to us 5.

## 9.1 The sub-menu CFIT

The CFIT menu has the following functions:

**FCSTX** **FCSTY** **SLOPE** **YINT** **CORR** **MODL**

Let's start from **MODL** which is a "sub-sub-menu". It has **LINF**, **LOG**, **EXPF**, **PWRF** and **BEST**.

- If the LINF is active the calculator will try to fit the data to a linear model  $y = Mx+B$ .
- If the LOG is active the calculator will fit the data to a logarithmic model  $y = M \ln(x)+B$ .
- If EXPF is active the model is  $y = B \exp\{Mx\}$  and
- If PWRF is active it is  $y = B x^M$ .

How can I select the model? This depends on what kind of data you are using. If you really don't know try the BEST function which will have the calculator select the best model.

The other functions in the **CFIT** sub-menu are quite simple.

**YINT**: Gives the value of B (the name comes from LINF model which is of course the main model).

**SLOPE**: Gives the value of M (again this comes from the LINF model).

**FCSTX**: Just run the model for a given y and returns a estimated x.

**FCSTY**: Returns a estimated y for a given x.

**CORR**: Returns a number between -1 and 1 (the correlation coefficient) which tells us how good is the fit. 1 is the best result.

Of course if you have only two data points the calculator will find LINF to best model and the correlation coefficient will be 1, which does not mean the estimation will be good!

## 9.2 The second line: ALLΣ, LINΣ, ΣREG and ΣRG? Functions

Unlike HP-33S which has special variables for statistic, the HP-42S uses the normal memories from  $R_{11}$  to  $R_{23}$  in the following way.

- $R_{11}$  for  $\Sigma x$ ,  $R_{12}$  for  $\Sigma x^2$ ,  $R_{13}$  for  $\Sigma y$ ,  $R_{14}$  for  $\Sigma y^2$ ,  $R_{15}$  for  $\Sigma xy$  and  $R_{16}$  for n. This is like the HP-41 and if LINEΣ is active it is all we have.
- If ALLΣ is active we have also  $R_{17}$  for  $\Sigma \ln(x)$ ,  $R_{18}$  for  $\Sigma (\ln x)^2$ ,  $R_{19}$  for  $\Sigma \ln(y)$ ,  $R_{20}$  for  $\Sigma (\ln y)^2$ ,  $R_{21}$  for  $\Sigma (\ln x \times \ln y)$ ,  $R_{22}$  for  $\Sigma x \ln(y)$  and  $R_{23}$  for  $\Sigma y \ln(x)$ .



Why should we care? Because now we know how to access statistical data in programming and also we know we cannot put important data in memories from  $R_{11}$  to  $R_{23}$  if we are going to use statistics. (33S is better)

Of course unless you need HP-41 compatibility, you should keep the calculator always in ALLΣ mode.

We said the statistical data start in  $R_{11}$  but you can change this using ΣREG and to view where it is starting just use ΣRG? Function.

Conclusion: Four useless functions! ☹

## 10 Matrices

Despite of its small display one of the nice features of HP-42S is the ability to work with matrices. This is done using the **MATRIX** menu. This multi-line menu has the following items:

```
NEW INV DET TRAN SIMQ EDIT
DOT CROSS UVEC DIM INDEX EDITN
STOIJ RCLIJ STOEL RCLEL PUTM GETM
```

Let's start from the first line.

**NEW**: This function creates a new matrix of a given size. Line y of the stack should contain the number of rows and line x should contain the number of columns. For example:

```
3 [ENTER] 2 [MATRIX] NEW
```

creates a 3x2 matrix. This matrix is empty or null because all elements are zero. (See EDIT)

**INV**: Calculates the inverse of a matrix in line x of the stack. The given matrix must have the same number of rows and columns and a non-zero determinant.

**DET**: Calculates the determinant of a matrix in line x of the stack. The given matrix must have the same number of rows and columns.

**TRAN**: Calculates the transpose of a matrix. The transpose is another matrix with rows changed by columns. If A is a matrix  $n \times m$  and  $a_{ij}$  is an element of it then its transpose will be a matrix B where an element  $b_{ij} = a_{ji}$ .

**SIMQ**: Means "Simultaneous Equations". We are not going to see this here.

**EDIT**: This edits a matrix in line x of the stack. It has the following sub items: In the first line we have

```
← OLD ↑ ↓ GOTO →
```

and there is also a second line we are not going to discuss. When you call EDIT with a matrix in the x register of the stack you are going to see something like

```
1:1=0.0000
```

This means the element (1,1) “line one and column one” of the given matrix is 0.0000 (here we are using FIX 4 for the examples). If you want to change this element just enter the new value. For example: 56 **ENTER** gives us 1:1=56.0000.

To change another element just go to its position (we use **←** or **→** to change the column and **↑** or **↓** to change the line we are editing) and change it. One can also use **GOTO** to go to a specific row and **OLD** to undo an entered element. In fact you don't need to press **ENTER**, just press **→** to move to the next element.

Exercise: Calculate the determinant of this matrix:

First line:	1	2	3
Second line:	-2	3	5
Third line:	0	4	-1

Solution: First we create a 3x3 matrix

3 **ENTER** **MATRIX** **NEW**

We have “x: [ 3x3 Matrix]” in the display Now we press **EDIT** and we have “1:1=0.0000”. Let's enter all elements. (First line)

1 **→** 2 **→** 3

Let's go, for example to (2,1), to enter the second line.

**↓** **←** **←**  
 2 **+/-** **→** 3 **→** 5  
**↓** **←** **←**  
 0 **→** 4 **→** 1 **+/-**

Then we press exit to stop editing. Now to calculate the determinant is just press **DET** which gives -51 (if you are going to use the same matrix again you'd better save it before any calculation).

The size of a matrix is limited by the available memory of the calculator. In my Palm Tungsten E using Free42 I can create a matrix of 90×90 and in my PC the Free42 program can give me a 5000×5000 or bigger while in the real HP-42S the limit is 29×29.

The **EDIT** function is not only useful to enter a matrix but also to see all the elements of a matrix resulting from a calculation.

Talking about matrix calculation, the HP-42S does +, −, × and ÷ of matrices in normal way. Of course, as you know, the operations are not always possible. For example: To sum or subtract matrices they must have the same size, etc.

How can one use matrices to solve linear systems? The HP-42S owner's manual explains it by using the **SIMQ** function. But it would be more profitable to remember a little of linear algebra. If you have  $n \times n$  linear system you can always write it as the matrix equation

$$A X = B$$

where A is a  $n \times n$  matrix called the coefficient matrix, B is a  $n \times 1$  column matrix called the independent terms matrix and X is also a  $n \times 1$  column matrix which contains the unknown variables. By multiplying this equation by the inverse matrix of A we have

$$X = A^{-1} B.$$

So if you are able to perform the inverse of matrix and able to multiply matrices you can solve a linear system without needing to learn another calculator's function.

What about complex matrices? You cannot enter complex numbers in a normal matrix. You have to create a complex matrix first. The procedure to do this is like that of creating a complex number. First you enter the real part, then you enter the imaginary part and then you press **COMPLEX**. As here it is not a normal sum but a matrix sum we have to enter two matrices of the same size. In fact one can create an empty complex matrix and then edit it. For example, to create a 3x3 complex matrix we do (with the MATRIX menu active)

3 **ENTER** **NEW** **ENTER** **COMPLEX**

where '3 **ENTER**' puts the number 3 in register x and register y of the stack. '**NEW**' creates a 3x3 real matrix. '**ENTER**' creates another one and '**COMPLEX**' makes the complex matrix.

We are not going to study the second and the third line of the **MATRIX** menu (too specialized) but there are two functions in the second line that may be useful. They are: DOT and CROSS. As you know vectors can be represented by a single row or a single column matrix. In the HP-42S, vectors will be represented only by single row matrices.

There is nothing special to say about addition, subtraction or multiplication by a scalar since there is no difference for the case of a matrix. But if you want to calculate the dot product in the calculator you can use DOT function. As you know if we have two vectors A and B, the dot product is  $A_x B_x + A_y B_y + A_z B_z$ . If the number of dimensions is not 3 but N we calculate the dot product in the same way as  $A_1 B_1 + \dots + A_N B_N$ . You probably won't use this because it is faster to do by hand! (we would spend a lot of time just to enter the vector in the calculator).

The cross product which is given by  $i(A_y B_z - A_z B_y) + j(A_z B_x - A_x B_z) + k(A_x B_y - A_y B_x)$  is more interesting and can be calculated using the CROSS function. The cross product is only defined in 3 dimensions.

Example: Calculate  $A \times B$  for  $A = 5i + 3j - 2k$  and  $B = i - 5k$ .

Solution: 'x' usually means cross product while '•' usually means dot product.

Let's enter the vector A.

**MATRIX** 1 **ENTER** 3 **NEW**  
**EDIT** 5 **ENTER** → 3 **ENTER** → 2 **+/-** **ENTER** **EXIT** **STO** "A"

(In fact we don't need the **ENTER**)

Let's enter now the vector B.

1 **ENTER** 3 **NEW**  
**EDIT** 1 **ENTER** → → 5 **+/-** **ENTER** **EXIT** **STO** "B"

We have stored both matrices because when you use the EDIT function, if you press ENTER as we did, what you enter goes onto to the stack and we would lose the first matrix.

Now we do **RCL** A **RCL** B **MATRIX** **CROSS** which gives the answer, which we can see using the EDIT function.

The HP-42S Owner's Manual fails to mention the existence of three additional useful matrix functions:

[MIN] - Gets a column's smallest element.

[MAX] - Get a column's largest element.

[FIND] - Searches a matrix for a given element.

These additional functions were discovered independently by Joseph K. Horn,; his documentation can be found at [http://www.hp42s.com/docs/hidden\\_matrix\\_functions.html](http://www.hp42s.com/docs/hidden_matrix_functions.html).

## 11 Other Bases

To work with other bases we must use the **BASE** menu (over the “4”). This menu has the following functions:

**A...F**: Select hexadecimal mode and show A to F.

**HEXM**: Select hexadecimal mode.

**DECM**: Decimal mode.

**OCTM**: Octal mode.

**BINM**: Binary mode.

**LOGIC**: Show the logic functions **AND**, **OR**, **XOR**, **NOT**, **BIT?**, **ROTXY**.

Example: How do we write 500 in hexadecimal mode?

Solution: In decimal mode enter 500 and then select **HEXM**. You will find 1F4.

## 12 Flags

The HP-42S has 100 flags from 00 to 99. A flag is a “binary memory” which works as an indicator of calculator's status. A flag can be “set” (true) or “clear” (false). Just for comparison the 33S has only 12 flags.

But why do we need flags? Why not set up the calculator status using just the default menus? For me the most interesting reason is related to programming. Using flags you can get the status of the calculator or change it from a program.

To work with flags there are some functions in the **FLAGS** menu (over the “6”) which are **SF**, **CF**, **FS?**, **FC?**, **FS?C**, **FC?C** (again the HP-42S has more functions than we need).

**SF**: **Set flag**. Makes a flag to be set (true).

**CF**: **Clear flag**. Makes a flag to be clear (false).

**FS?**: **Flag set test**. True if the flag is set (true).

**FC?**: **Flag clear test.** True if the flag is clear (false).

**FS?C**: **Flag set test and clear.** True if the flag is set (true), and simultaneously clears the flag.

**FC?C**: **Flag clear test and clear.** The same as FC? and CF together.

Now let's study some of the flags.

**00–10 and 81–99**: These 30 flags are not related to the calculators status but are just “binary memories” you can use for what you want.

**11**: When you turn on the calculator if the flag 11 is set the calculator runs the program where the “calculator's pointer” is stopped and clears the flag 11.

**24**: If this flag is set the calculator will ignore error messages. “Out of Range” will appear as 9.9999999999E499 in the HP-42S and as 1.7977E308 in Free42. This is important if you want to prevent an unwanted stop during a program.

**26**: Turns audio on.

**28**: Use '.' instead of ',' (default).

**29**: If set the calculator separates digits in groups of three. For example: 5000000 becomes 5,000,000. (default)

**36–80**: You cannot change the flags from 36 to 80.

**44**: Always on. If true the calculator does not turn off after about 10 minutes.

**65**: Matrix editor.

**68–71**: Base modes. CCCC=Decimal (default), CCCS=Binary, CSSS=Octal and SSSS=Hexadecimal. It is a shame! Why not only two? :)

**73**: This flag if true makes the calculator to show complex numbers in polar notation.

## 13. Free42-Specific Features

This section describes implementation features of Free42 that are enhancements to the functionality of the original HP-42s.

### 13.1 Keyboard Interface (Windows, Linux and Mac Desktops and Laptops)

On machines with a standard keyboard, Free42 maps the calculator keys onto the keyboard according to the following layout:


OFF ← TOP ROW PROGRAMMABLE KEYS →

EXIT ESC	F1	F2	F3	F4	F5	F6							
ASSIGN CUSTOM PGM.FCN BASE CONVERT FLAGS SOLVER x MATRIX TOP.FCN +													
1	2	3	4	5	6	7	8	9	0	-	CATALOG =		
x <sup>2</sup> √X Q		LASTx X↔Y W	DISP E E	% RCL R	ATAN TAN T		Y	U	I	10 <sup>x</sup> LOG O		P	
Σ- A		ASIN S	π R↓ D	F	G	H	J	K	e <sup>x</sup> LN L		;	'	CLEAR ENTER ENTER
2 <sup>nd</sup> Function SHIFT		GTO Z	ACOS X C	y <sup>x</sup> 1/x V	B	MODES +/- N	COMPLEX STO M	,	.	/	2 <sup>nd</sup> Function SHIFT	▲ ↑	BST
Inaccessible: f(x), PROB												▼ ↓	SST

## 13.2 Program Import and Export

One of the greatest limitations of the real HP 42s calculator is that it has no means of input or output of electronically interchanged files; so if you want to transfer a program from one HP 42s to another, you have to manually key it in (and if you don't happen to have the HP printer, you also have to transcribe it manually).

This limitation is removed with Free42. Free42 allows you to import and export program file binaries, and using suitable translation software, convert raw binaries to/from ASCII text files. On a desktop machine (Windows, Mac or Linux), the Free42 File menu has program import/export options allowing you to specify files in "raw" format for import or export.

On mobile devices such as the iPhone,  **OPTIONS** opens an options menu including "Import and Export Programs". WiFi access is required to use this option, and selecting it causes Free42 to open a special web-enabled file server on the local network and assign it the url <http://new-host.home:8000> which can then be accessed by a browser from any machine connected to the same local network.

The program *txt2raw.pl*, written by Vini Matangrano and available on the Free42 website (<http://thomasokken.com/free42/> or specifically <http://thomasokken.com/free42/txt2raw/txt2raw.html>) converts plain text files into the raw format for import to Free42; and printing to a text file from Free42 and transferring the print file to your computer effectively exports a program from the HP 42s in plain text format.

You can find at <http://thomasokken.com/free42/42progs/index.html> a collection of useful HP 42s programs.

## 13.3 Printing



On desktop/laptop machines (Windows, Mac or Linux), Free42 provides an option to print to either a text file or to a GIF graphics file under the Preferences screen (File→Preferences).

On mobile devices, the print output is stored in the *print.txt* file accessible from the browser interface.

## 14. Comprehensive Command List

The following table lists every function or command supported by the HP-42S in alphabetical order, along with a synopsis of its purpose and behavior.

<i>Name</i>	<i>Description</i>
ABS	<i>Absolute value.</i> Returns $ x $ .
ACOS	<i>Arc cosine.</i> Returns $\cos^{-1} x$ .
ACOSH	<i>Arc hyperbolic cosine.</i> Returns $\cosh^{-1} x$ .
ADV	<i>Advance</i> the printer paper one line
AGRAPH	<i>Alpha graphics.</i> Display a graphics image. Each character in the Alpha register specifies an 8-dot column pattern. The x- and y-registers specify the pixel location of the image.
AIP	<i>Append Integer part</i> of x to the Alpha register.
ALENG	<i>Alpha length.</i> Returns the number of characters in the Alpha register.
ALL	Select the <i>All</i> display format.
ALLΣ	Select ALLΣ (All-statistics) mode, which uses 13 summation coefficients.
AND	<i>Logical AND.</i> Returns $x \text{ AND } y$ .
AOFF	<i>Alpha off.</i> Exit from the ALPHA menu.
AON	<i>Alpha on.</i> Select the ALPHA menu.
ARCL	<i>Alpha recall.</i> Copy data into the Alpha register, appending it to the current contents. Numbers are formatted using the current display format. Parameter: register or

<i>Name</i>	<i>Description</i>
	variable. (indirect allowed)
AROT	<i>Alpha rotate</i> . Rotate the Alpha register by the number of characters specified in the x-register.
ASHF	<i>Alpha shift</i> . Shifts the six left-most characters out of the Alpha register.
ASIN	<i>Arc sine</i> . Returns $\sin^{-1} x$ .
ASINH	<i>Arc hyperbolic sine</i> . Returns $\sinh^{-1} x$ .
ASSIGN	<i>Assign</i> a function, program, or variable to a menu key in the CUSTOM menu. Parameter 1: function name, alpha program label, or variable name. Parameter 2: key number (01–18).
ASTO	<i>Alpha store</i> . Copy the first six characters in the Alpha register into a register or variable. Parameter: register or variable (indirect allowed)
ATAN	<i>Arc tangent</i> . Returns $\tan^{-1} x$ .
ATANH	<i>Arc hyperbolic tangent</i> . Returns $\tanh^{-1} x$ .
ATOX	<i>Alpha to X</i> . Convert the left-most character in the Alpha register to its character code (returned to the x-register) and delete the character.
AVIEW	<i>Alpha view</i> . Display the Alpha register.
BASE+	<i>Base addition</i> . Returns the 36-bit sum of $y + x$ .
BASE-	<i>Base subtraction</i> . Returns the 36-bit difference of $y - x$ .
BASEx	<i>Base multiplication</i> . Returns the 36-bit product of $y \times x$ .
BASE÷	<i>Base division</i> . Returns the 36-bit quotient of $y \div x$ .
BASE+/-	<i>Base change sign</i> . Returns the 36-bit 2's complement of $x$ .
BEEP	Sound a sequence of four tones.
BEST	Select the <i>best</i> curve-fitting model for the current statistical data.
BINM	Select <i>Binary</i> mode (base 2)
BIT?	Test the $x^{\text{th}}$ bit of $y$ . If the bit is set (1), execute the next program line; if the bit is clear (0), skip the next program line.
BST	<i>Back step</i> . Move the program pointer to the previous program line. (Not programmable.)
CF	<i>Clear flag</i> $nn$ ( $00 \leq nn \leq 35$ or $81 \leq nn \leq 99$ ). Parameter: flag number (indirect allowed)
CLA	<i>Clear Alpha register</i> . If Alpha mode is on and character entry is terminated (no cursor displayed), then  also executes the CLA function.
CLALL	<i>Clear all</i> . Clear all stored programs and data.(Not Programmable.)
CLD	<i>Clear display</i> . Clear a message from the display.
CLKEYS	Clear all CUSTOM menu key assignments.
CLLCD	<i>Clear LCD</i> (liquid crystal display). Blanks the entire display.
CLMENU	<i>Clear MENU</i> . Deletes all menu key definitions for the programmable menu.
CLP	<i>Clear a program</i> from memory. Parameter: global label
CLRG	<i>Clear Registers</i> . Clear all of the numbered storage registers to zero.
CLST	<i>Clear Stack</i> . Clear the stack registers to zero.
CLV	<i>Clear a variable</i> from memory. Parameter: variable name (indirect allowed)
CLX	<i>Clear x-register</i> to zero. If digit entry is terminated (no cursor in the display), then  also executes CLX.
CLZ	<i>Clear statistics</i> . Clear the accumulated statistical data in the summation registers.
COMB	<i>Combinations</i> of $y$ items taken $x$ at a time = $y! / [x!(y-x)!]$
COMPLEX	Convert two real numbers (or matrices) into a complex number (or matrix). Converts a complex number (or matrix) into two real numbers (or matrices).
CORR	Returns a correlation coefficient using the current statistical data and curve-fitting model.
COS	<i>Cosine</i> . Returns $\cos(x)$ .
COSH	<i>Hyperbolic cosine</i> . Returns $\cosh(x)$ .
CPXRES	<i>Complex-results</i> . Enable the calculator to return a complex result, even if the inputs are real numbers.



<i>Name</i>	<i>Description</i>
CPX?	If the x-register contains a complex number, execute the next program line; if the x-register does not contain a complex number, skip the next program line.
CROSS	Returns the <i>cross product</i> of two vectors (matrices or complex numbers).
DECM	Selects <i>Decimal</i> mode (base 10).
DEG	Select the <i>Degrees</i> angular mode.
DEL	<i>Delete</i> the specified number of lines from the current program. Program-entry mode must be on. (Not programmable.) Parameter: number of lines.
DELAY	Set the <i>print delay</i> time to x seconds.
DELR	<i>Delete row</i> . Delete the current row from the indexed matrix.
DET	<i>Determinant</i> . Returns the determinant of the matrix in the x-register.
DIM	<i>Dimension</i> a matrix to x columns and y rows. If the matrix does not exist, DIM creates it. Parameter: variable name (indirect allowed)
DIM?	Returns the dimensions of the matrix in the x-register (rows to the y-register and columns to the x-register).
DOT	<i>Dot Product</i> . Returns the dot product of two vectors (matrices or complex numbers).
DSE	<i>Decrement, Skip if (less than or) Equal</i> . Given cccccc.fflii in a variable or register, decrements cccccc by ii and skips the next program line if cccccc is now $\leq$ fff. Parameter: register or variable (indirect allowed)
EDIT	<i>Edit</i> a matrix in the x-register.
EDITN	<i>Edit a named matrix</i> . Parameter: variable name (indirect allowed)
END	<i>End</i> of a program.
ENG	Select <i>Engineering</i> display format. Parameter: number of digits (indirect allowed)
ENTER	Separate two numbers keyed in sequentially; copies x into the y-register, y into the z-register, and z into the t-register, and loses t.
EXITALL	<i>Exit all</i> menus.
EXPF	Select the <i>exponential</i> curve-fitting model.
E↑X	<i>Natural exponential</i> . Returns $e^x$ .
E↑X-1	<i>Natural exponential</i> for values of x which are close to zero. Returns $e^x-1$ , which provides a much higher accuracy in the fractional part of the result.
FC?	<i>Flag clear test</i> . If the specified flag is clear, executes the next program line; if the flag is set, skips the next program line. Parameter: flag number (indirect allowed)
FC?C	<i>Flag clear test and clear</i> . If the specified flag is clear, execute the next program line; if the flag is set, skip the next program line. Cleared after the test is complete. (This function can be used only with flags 00 through 35 and 81 through 99.) Parameter: flag number (indirect allowed)
FCSTX	<i>Forecasts an x-value</i> given a y-value.
FCSTY	<i>Forecasts a y-value</i> given an x-value.
FIX	Select <i>Fixed-decimal</i> display format. Parameter: number of digits (indirect allowed)
FNRM	Returns the <i>Frobenius norm</i> of the matrix in the x-register.
FP	Returns the <i>fractional part</i> of x.
FS?	<i>Flag set test</i> . If the specified flag is set, execute the next program line; if the flag is clear, skip the next program line. Parameter: flag number (indirect allowed)
FS?C	<i>Flag set test and clear</i> . If the specified flag is set, execute the next program line; if the flag is clear, skip the next program line. Clear the flag after the test is complete. (This function can be used only with flags 00 through 35 and 81 through 99.) Parameter: flag number (indirect allowed)
GAMMA	<i>Gamma function</i> . Returns $\Gamma(x)$ .
GETKEY	<i>Get key</i> . The calculator waits for you to press a key. When you do, the key number is returned to the x-register. Keys are numbered from 1 through 37 ( $\Sigma+$ through $\div$ ) for normal keys and 38 through 74 ( $\Sigma-$ through <b>CATALOG</b> ) for shifted keys.
GETM	<i>Get matrix</i> . Copy a submatrix into the x-register from the indexed matrix.
GRAD	Select <i>Grads</i> angular mode.
GROW	Select <i>Grow</i> mode. Executing $\rightarrow$ or J+ causes the matrix to grow by one new row if

<i>Name</i>	<i>Description</i>
GTO	the index pointers are at the last (lower-right) element in the matrix. <i>Go to label.</i> From the keyboard, move the program pointer to the specified label. In a running program, cause the program to branch to the specified label. Parameter: local or global label (indirect allowed)
HEXM	Select <i>Hexadecimal</i> mode (base 16).
HMS+	Add x and y using H.MMSSss (hours-minutes-seconds) format.
HMS-	Subtract x from y using H.MMSSss format.
I+	<i>Increment the row pointer</i> in the indexed matrix.
I-	<i>Decrement the row pointer</i> in the indexed matrix.
INDEX	<i>Index</i> a named matrix. Parameter: variable name (indirect allowed)
INPUT	Recall a register or variable to the x-register, display the name of the register or variable along with the contents of the x-register, and halt program execution. Pressing <span style="border: 1px solid black; padding: 0 2px;">R/S</span> (or <span style="border: 1px solid black; padding: 0 2px;">SST</span> ) stores x into the register or variable; pressing <span style="border: 1px solid black; padding: 0 2px;">EXIT</span> cancels. (Used only in programs.) Parameter: register or variable (indirect allowed)
INSR	<i>Insert a row</i> in the indexed matrix.
INTEG	<i>Integrate</i> the selected integration program with respect to the specified variable. Parameter: variable name (indirect allowed)
INVRT	Returns the inverse of the matrix in the x-register.
IP	Returns the integer part of x
ISG	<i>Increment, Skip if Greater.</i> Given cccccc.fffii in a variable or register, increments cccccc by ii and skips the next program line if cccccc is now > fff. Parameter: register or variable (indirect allowed)
J+	<i>Increment the column pointer</i> in the indexed matrix.
J-	<i>Decrement the column pointer</i> in the indexed matrix.
KEYASN	Selects <i>key-assignments</i> mode for the CUSTOM menu.
KEYG	<i>On menu key, go to.</i> Branch to specified label to when a particular menu key is pressed. Parameter 1:: Key number (1 through 9), Parameter 2: program label (global or local)
KEYX	<i>On menu key, execute.</i> Execute (as a subroutine) specified label when a particular menu key is pressed. Parameter 1:: Key number (1 through 9), Parameter 2: program label (global or local)
LASTX	<i>Last x.</i> Recall the last value of x used in a calculation.
LBL	<i>Label.</i> Identify programs and routines for execution and branching. Parameter: local or global label.
LCLBL	<i>Select Local label mode</i> for the CUSTOM menu (to use CUSTOM menu assignments to execute local labels within the current program).
LINF	Select the <i>linear</i> curve-fitting model.
LINΣ	Select <i>Linear statistics</i> mode, which uses six summation coefficients.
LIST	Print a portion of a program listing. (Not programmable.) Parameter: number of lines.
LN	<i>Natural logarithm.</i> Returns ln(x).
LN1+X	<i>Natural logarithm</i> for values close to zero. Returns ln(1 + x), which provides a much higher accuracy in the fractional part of the result.
LOG	<i>Common logarithm.</i> Returns log <sub>10</sub> (x).
LOGF	Select the <i>logarithmic</i> curve-fitting model.
MAN	Select <i>Manual print mode</i> .
MAT?	If the x-register contains a matrix, execute the next program line; if the X-register does not contain a matrix, skip the next program line.
MEAN	<i>Mean.</i> Returns the mean of x-values ( $\Sigma x / n$ ) and the mean of y-values ( $\Sigma y / n$ ).
MENU	Select the programmable menu.
MOD	<i>Modulo.</i> Returns the remainder for y / x.
MVAR	Declare a <i>menu variable</i> in a SOLVER program. Parameter: variable name.
N!	<i>Factorial.</i> Returns x!.
NEWMAT	<i>New matrix.</i> Creates a y × x matrix in the x-register.
NORM	Select <i>Normal print mode</i> , which prints a record of keystrokes.

<i>Name</i>	<i>Description</i>
NOT	<i>Logical NOT</i> . Returns NOT(x).
OCTM	Select <i>Octal</i> mode
OFF	Turn the calculator off (programmable). (Pressing <span style="border: 1px solid black; padding: 0 2px;">OFF</span> does not execute the programmable OFF function.)
OLD	Recall the current element from the indexed matrix. (Equivalent to RCLEL.)
ON	<i>Continuous on</i> . Prevent the calculator from automatically turning off after ten minutes of inactivity.
OR	<i>Logical OR</i> . Returns x OR y.
PERM	<i>Permutations</i> of y items taken x at a time. Returns $y!/(y - x)!$
PGMINT	Select a program to integrate. Parameter: global label (indirect allowed)
PGMSLV	Select a program to solve. Parameter: global label (indirect allowed) ,
PI	Put an approximation of $\pi$ into the x-register (3.14159265359).
PIXEL	Turn on a single pixel (dot) in the display. The location of the pixel is given by the numbers in the x- and y-registers.
POLAR	Select <i>polar</i> coordinate mode for displaying complex numbers.
POSA	<i>Position in Alpha</i> . Searches the Alpha register for the target specified in the x-register. If found, returns the character position; if not found, returns -1.
PRA	Print Alpha register.
PRLCD	<i>Print LCD</i> (liquid crystal display). Prints the entire display.
PROFF	<i>Printing off</i> . Clears flags 21 and 55.
PROMPT	Display the Alpha register and halt program execution.
PRON	<i>Printing on</i> . Sets flags 21 and 55.
PRP	<i>Print program</i> . If a label is not specified, print the current program. (Not programmable.) Parameter: global label (optional)
PRSTK	<i>Print stack</i> . Print the contents of the stack registers (x, y, z and t).
PRUSR	<i>Print user</i> variables and programs.
PRV	<i>Print variable</i> . Parameter: variable name (indirect allowed)
PRX	<i>Print x-register</i> .
PRZ	<i>Print statistics</i> . Prints the contents of the summation registers.
PSE	<i>Pause</i> program execution for about 1 second.
PUTM	<i>Put matrix</i> . Stores the matrix in the X-register into the indexed matrix beginning at the current element.
PWRF	Select the <i>power</i> curve-fitting model.
QUIET	Toggle flag 26 to disable or enable the beeper (Not programmable )
RAD	Select <i>Radians</i> angular mode
RAN	Returns a <i>random</i> number ( $0 \leq x < 1$ )
RCL	<i>Recall</i> data into the x-register. Parameter: register or variable (indirect allowed)
RCL+	<i>Recall addition</i> . Recall data and add it to the contents of the x-register. Parameter: register or variable (indirect allowed)
RCL-	<i>Recall subtraction</i> . Recall data and subtract it from the contents of the x-register. Parameter: register or variable (indirect allowed)
RCL×	<i>Recall multiplication</i> . Recall data and multiply it by the contents of the x-register. Parameter: register or variable (indirect allowed)
RCL÷	<i>Recall division</i> . Recall data and divide it into the contents of the x-register. Parameter: register or variable (indirect allowed)
RCLEL	<i>Recall element</i> . Recalls the current matrix element from the indexed matrix.
RCLIJ	Recall the row- and column-pointer values (I and J) for the indexed matrix.
RDX,	Select a comma to be used as the radix mark (decimal point).
RDX.	Select a period to be used as the radix mark (decimal point).
REALRES	<i>Real-results</i> . Disables the calculator's ability to return a complex result using real-number inputs.
REAL?	If the x-register contains a real number, execute the next program line; if the x-register does not contain a real number, skip the next program line.
RECT	Select <i>Rectangular</i> coordinate mode for displaying complex numbers.

<i>Name</i>	<i>Description</i>
RND	Round the number in the x-register using the current display format.
RNRM	Return the <i>row norm</i> of the matrix in the x-register.
ROTXY	Rotate the 36-bit number in the y-register by x bits.
RSUM	Return the <i>row sum</i> of each row of the matrix in the x-register and returns the sums in a column matrix.
RTN	<i>Return</i> . In a running program, branches the program pointer back to the line following the most recent XEQ instruction. If there is no matching XEQ instruction, program execution halts. From the keyboard, RTN moves the program pointer to line 00 of the current program.
R↔R	<i>Row swap row</i> . Swaps the elements in rows x and y in the indexed matrix.
R↑	<i>Roll up</i> the contents of the four stack registers one position.
R↓	<i>Roll down</i> the contents of the four stack registers one position.
<span style="border: 1px solid black; padding: 2px;">R/S</span>	<i>Run/stop</i> . Runs a program (beginning at the current program line) or stops a running program. In program-entry mode, inserts a STOP instruction into the program.
SCI	Select <i>scientific</i> notation display format. Parameter: number of digits (indirect allowed)
SDEV	<i>Standard deviation</i> . Returns $s_x$ and $s_y$ using the current statistical data.
SEED	Store a <i>seed</i> for the random number generator.
SF	<i>Set flag nn</i> ( $00 \leq nn \leq 35$ ; $81 \leq nn \leq 99$ ). Parameter: flag number (indirect allowed)
SIGN	<i>Sign</i> . Return 1 for $x \geq 0$ , $-1$ for $x < 0$ , and 0 for non-numbers. Returns the unit vector of a complex number.
SIN	<i>Sine</i> . Returns $\sin(x)$ .
SINH	<i>Hyperbolic sine</i> . Returns $\sinh(x)$ .
SIZE	Set the number of storage registers. Parameter: number of registers.
SLOPE	Return the <i>slope</i> of the linear transformation of the current curve-fitting model.
SOLVE	<i>Solve</i> for an unknown variable. Parameter: variable name (indirect allowed)
SORT	<i>Square root</i> . Returns $\sqrt{x}$ .
SST	<i>Single step</i> . Moves the program pointer to the next program line. (Not programmable.)
STO	<i>Store</i> a copy of x into a destination register or variable. Parameter: register or variable (indirect allowed)
STO+	<i>Store addition</i> . Adds x to an existing register or variable. Parameter: register or variable (indirect allowed)
STO-	<i>Store subtraction</i> . Subtracts x from an existing register or variable. Parameter: register or variable (indirect allowed)
STO×	<i>Store multiplication</i> . Multiplies an existing register or variable by x. Parameter: register or variable (indirect allowed)
STO÷	<i>Store division</i> . Divides an existing register or variable by x. Parameter: register or variable (indirect allowed)
STOEL	<i>Store element</i> . Stores a copy of x into the current element of the indexed matrix.
STOIJ	Moves the row- and column-pointers to $I = x$ and $J = y$ in the indexed matrix.
STOP	<i>Stop</i> program execution. ( <span style="border: 1px solid black; padding: 2px;">R/S</span> in program entry mode).
STR?	If the x-register contains an Alpha string, execute the next program line; if the x-register does not contain an Alpha string, skip the next program line.
SUM	Returns the sums $\Sigma x$ and $\Sigma y$ into the x- and y-registers.
TAN	<i>Tangent</i> . Returns $\tan(x)$ .
TANH	<i>Hyperbolic tangent</i> . Returns $\tanh(x)$ .
TONE	Sounds a <i>tone</i> . Parameter: tone number (0–9) (indirect allowed)
TRACE	Select <i>Trace</i> printing mode, which prints a record of keystrokes and results.
TRANS	Return the <i>transpose</i> of the matrix in the x-register.
UVEC	<i>Unit vector</i> . Return the unit vector for the matrix or complex number in the x-register.
VARMENU	Create a <i>variable menu</i> using MVAR instructions following the specified global label. Parameter: global program label. (indirect allowed)
VIEW	<i>View</i> the contents of a register or variable. Parameter: register or variable (indirect allowed)

<i>Name</i>	<i>Description</i>
WMEAN	<i>Weighted mean.</i> Return the mean of x-values weighted by the y-values $\Sigma xy / \Sigma y$
WRAP	Select <i>Wrap</i> mode, which prevents the indexed matrix from growing.
X<>	Swaps the contents of the x-register with another register or variable. Parameter: register or variable (indirect allowed)
X<>Y	Swaps the contents of the x- and y-registers.
X<O?	<i>X less than zero test.</i> If true, execute the next program line; if false, skip the next program line
X<Y?	<i>X less than y test.</i> If true, execute the next program line; if false, skip the next program line
X<=O?	<i>X less than or equal to zero test.</i> If true, execute the next program line; if false, skip the next program line
X<=Y?	<i>X less than or equal to y test.</i> If true, execute the next program line; if false, skip the next program line
X=O?	<i>X equal to zero test.</i> If true, execute the next program line; if false, skip the next program line
X=Y?	<i>X equal to y test.</i> If true, execute the next program line; if false, skip the next program line
X≠O?	<i>X not equal to zero test.</i> If true, execute the next program line; if false, skip the next program line
X≠Y?	<i>X not equal to y test.</i> If true, execute the next program line; if false, skip the next program line
X>O?	<i>X greater than zero test.</i> If true, execute the next program line; if false, skip the next program line
X>Y?	<i>X greater than y test.</i> If true, execute the next program line; if false, skip the next program line
X≥O?	<i>X greater than or equal to zero test.</i> If true, execute the next program line; if false, skip the next program line
X≥Y?	<i>X greater than or equal to y test.</i> If true, execute the next program line; if false, skip the next program line
XEQ	<i>Execute</i> a function or program. Parameter: function or label (indirect allowed)
XOR	<i>Logical XOR</i> (exclusive OR). Returns x XOR y.
XTOA	<i>X to Alpha.</i> Appends a character (specified by the code in the x-register) to the Alpha register. If the x-register contains an Alpha string, appends the entire string.
X↑2	<i>Square.</i> Returns $x^2$ .
YINT	<i>y-intercept.</i> Returns the y-intercept of the curve fitted to the current statistical data.
Y↑X	<i>Power.</i> Returns $y^x$ .
1/x	<i>Reciprocal.</i> Returns $1/x$ .
10↑X	<i>Common exponential.</i> Returns $10^x$ .
+	<i>Addition.</i> Returns $y + x$ .
−	<i>Subtraction.</i> Returns $y - x$ .
×	<i>Multiplication.</i> Returns $x \times y$ .
÷	<i>Division.</i> Returns $y / x$ .
+/-	<i>Change the sign</i> of the number in the x-register. While entering an exponent, can also be used to change the sign of the exponent.
Σ+	<i>Summation plus.</i> Accumulate a pair of x- and y values into the summation registers.
Σ−	<i>Summation minus.</i> Subtract a pair of x- and y-values from the summation registers.
ΣREG	<i>Summation registers.</i> Defines which storage register begins the block of summation registers. Parameter: register number (indirect allowed)
ΣREG?	Return the register number of the first summation register.
→DEC	<i>To decimal.</i> Converts the octal (base 8) representation of a number to decimal (base 10). Note: This function is included to provide program compatibility with the HP-41 (which uses the function name DEC).
→DEG	<i>To degrees.</i> Convert an angle-value from radians to degrees. Returns $x \times (180/\pi)$ .
→HMS	<i>To hours, minutes, and seconds.</i> Convert x from a decimal fraction to a minutes-

<i>Name</i>	<i>Description</i>
	seconds format.
→HR	<i>To hours.</i> Converts x from a minutes-seconds format to a decimal fraction.
→OCT	<i>To octal.</i> Converts a decimal number to the octal representation. Note: This function is included to provide program compatibility with the HP-41 (which uses the function name OCT).
→POL	<i>To polar.</i> Converts x and y to the corresponding polar coordinates r and $\theta$ . If the x-register contains a complex number, converts the two parts of the number to polar values.
→RAD	<i>To radians.</i> Converts a angle value in degrees to radians. Returns $x \times (\pi/180)$ .
→REC	<i>To rectangular.</i> Converts r (in the x-register) and $\theta$ (in the y-register) to the corresponding rectangular coordinates, x and y. If the X-register contains a complex number, converts the two parts of the number to rectangular values.
◀	<i>Backspace</i> or clear x-register. In Program entry mode, deletes the current program line.
←	<i>Move left</i> one element in the indexed matrix.
↑	<i>Move up</i> one element in the indexed matrix.
↓	<i>Move down</i> one element in the indexed matrix.
→	<i>Move right</i> one element in the indexed matrix.
%	<i>Percent.</i> Returns $(x \times y) / 100$ . (Leaves the y value in the y-register.)
%CH	<i>Percent change.</i> Returns $(x - y) \times (100 / y)$ .

## GNU Free Documentation License

Version 1.3, 3 November 2008

Copyright (C) 2000, 2001, 2002, 2007, 2008 Free Software Foundation, Inc. <<http://fsf.org/>>

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 functional and useful 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, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a

world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you". You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

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. (Thus, 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. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

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 Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document 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, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. 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, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include 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, PostScript or PDF 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.

The "publisher" means any person or entity that distributes copies of the Document to the public.

A section "Entitled XYZ" means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as "Acknowledgements",

"Dedications", "Endorsements", or "History".) To "Preserve the Title" of such a section when you modify the Document means that it remains a section "Entitled XYZ" according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

## 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 (or copies in media that commonly have printed covers) 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 computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. 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.

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 fewer than five), unless they release you from this requirement.
- 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", Preserve 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 the 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. For any section Entitled "Acknowledgements" or "Dedications", Preserve the Title of the section, 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 to be Entitled "Endorsements" or to conflict in title with any Invariant Section.
- O. Preserve any Warranty Disclaimers.

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, and that you preserve all their Warranty Disclaimers.

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, is called an "aggregate" if the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which 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 half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form.

Otherwise they must appear on printed covers that bracket 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, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled "Acknowledgements", "Dedications", or "History", the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

## 9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, or distribute it is void, and will automatically terminate your rights under this License.

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, receipt of a copy of some or all of the same material does not give you any rights to use it.

## 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. If the Document specifies that a proxy can decide which future versions of this License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Document.

## 11. RELICENSING

"Massive Multiauthor Collaboration Site" (or "MMC Site") means any World Wide Web server that publishes copyrightable works and also provides prominent facilities for anybody to edit those works. A public wiki that anybody can edit is an example of such a server. A "Massive Multiauthor Collaboration" (or "MMC") contained in the site means any set of copyrightable works thus published on the MMC site.

"CC-BY-SA" means the Creative Commons Attribution-Share Alike 3.0 license published by Creative Commons Corporation, a not-for-profit corporation with a principal place of business in San Francisco, California, as well as future copyleft versions of that license published by that same organization.

"Incorporate" means to publish or republish a Document, in whole or in part, as part of another Document.

An MMC is "eligible for relicensing" if it is licensed under this License, and if all works that were first published under this License somewhere other than this MMC, and subsequently incorporated in whole or in part into the MMC, (1) had no cover texts or invariant sections, and (2) were thus incorporated prior to November 1, 2008.

The operator of an MMC Site may republish an MMC contained in the site under CC-BY-SA on the same site at any time before August 1, 2009, provided the MMC is eligible for relicensing.