

**MCalc**

<b>COLLABORATORS</b>
----------------------

	<i>TITLE :</i> MCalc		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY		July 22, 2024	

<b>REVISION HISTORY</b>
-------------------------

NUMBER	DATE	DESCRIPTION	NAME

# Contents

<b>1</b>	<b>MCalc</b>	<b>1</b>
1.1	MCalc.guide	1
1.2	MCalc.guide/Copyright	2
1.3	MCalc.guide/Guarantee	2
1.4	MCalc.guide/Evolution	2
1.5	MCalc.guide/Installation	3
1.6	MCalc.guide/Usage	3
1.7	MCalc.guide/InputOutput	4
1.8	MCalc.guide/FuncTables	4
1.9	MCalc.guide/Inputs	4
1.10	MCalc.guide/StdDeviation	6
1.11	MCalc.guide/LinRegression	7
1.12	MCalc.guide/SpecNotes	8
1.13	MCalc.guide/Menus	9
1.14	MCalc.guide/Project	9
1.15	MCalc.guide/PickHistory	9
1.16	MCalc.guide/Bases	10
1.17	MCalc.guide/Size	10
1.18	MCalc.guide/Sign	11
1.19	MCalc.guide/Angle	11
1.20	MCalc.guide/Windows	11
1.21	MCalc.guide/ARexx	12
1.22	MCalc.guide/CALC	12
1.23	MCalc.guide/CALCTEX	13
1.24	MCalc.guide/GETOUTPUT	13
1.25	MCalc.guide/GETTEXOUTPUT	14
1.26	MCalc.guide/GETINPUT	15
1.27	MCalc.guide/GETTEXINPUT	15
1.28	MCalc.guide/FORMATTEX	16
1.29	MCalc.guide/SETMODE	16

1.30 MCalc.guide/DELETEHISTORY . . . . . 17

1.31 MCalc.guide/Prefs . . . . . 18

1.32 MCalc.guide/Layout . . . . . 19

1.33 MCalc.guide/Output . . . . . 19

1.34 MCalc.guide/History . . . . . 19

1.35 MCalc.guide/Misc . . . . . 20

1.36 MCalc.guide/Thanks . . . . . 20

1.37 MCalc.guide/UpDates . . . . . 20

1.38 MCalc.guide/Address . . . . . 21

1.39 MCalc.guide/PrgHistory . . . . . 21

# Chapter 1

## MCalc

### 1.1 MCalc.guide

Welcome to MUIProCalc v1.5. This is a short description on how to use MUIProCalc and what possibilities it offers. Anyway, it is a quite powerful calculator, also suited for programmers who have to deal with different computer specific calculations and values.

MUIProCalc doesn't stand for MUI'Professional'Calc. It is meant to stand for MUI'Programmer'Calc...

MUIProCalc (MCalc) is (C) 1993 Kai Iske, GiftWare

The icon for MUIProCalc was taken from the ToolManager 2.0 distribution and was designed by Michael 'Mick' Hohmann.

MUI is Copyright (C) Stefan Stuntz

NOTE that you better set the stack size to at least 8192 bytes

Copyright	Copyright notice and status
Guarantee	What I guarantee
Evolution	Why did I write MUIProCalc
Installation	How to install MUIProCalc
Using MUIProCalc	Well, how to use it ?
Preferences	How to set up MUIProCalc
Thanks	Whom I must say 'thank you'
How to obtain updates	What am 'I' supposed to do
How to reach the author	This is my address
History	Program History

## 1.2 MCalc.guide/Copyright

### Copyright

\*\*\*\*\*

You are allowed to copy and spread this program but only if you do not make any commercial profit out of the program itself and all supported files. The source of MUIProCalc is considered teachware and the author claims the right to be the only one to produce and release new versions of MUIProCalc. Public Domain vendors are not allowed to sell MUIProCalc for more than the price of a regular disk, which should not be higher than 5 Dollars.

MUIProCalc is GiftWare, so if you like it and you want to express your feelings, feel free to send me anything you think of being suited. Postcards, cookies, money, simply anything ;)

## 1.3 MCalc.guide/Guarantee

### Guarantee

\*\*\*\*\*

The author does not give any guarantee that the program MUIProCalc works perfectly. The program was tested and it is used every day, anyway nobody is perfect and that is why I can not take the responsibility for any damages occurred during the usage of MUIProCalc.

## 1.4 MCalc.guide/Evolution

### Evolution

\*\*\*\*\*

Due to the fact that I was still using Calc 3.0, written by Jimmy Yang, which wasn't that compatible to newer Kickstarts, I thought about writing a new one, since I couldn't find any calculator suited my needs. Not that Calc 3.0 was/is buggy, but it's just the fact that it isn't font adaptive. Furthermore I wanted to know how powerful MUI really was, so I thought it'd be a good idea to write it using MUI.

Taking Calc 3.0, I tried to implement the functionality of this calculator, putting it into a MUI-GUI. What is still missing is the plotter. This may change in a later release.

One word to the parser: The grammar might not be perfect. There's only a small amount of syntax checking, in order to minimize parser size. Float value error handling isn't perfect either, yet. I just wanted to have a calculator as soon as possible. All this will change in future releases, if there are enough of replies.

---

## 1.5 MCalc.guide/Installation

Installing MUIProCalc

\*\*\*\*\*

Installing MUIProCalc isn't that hard. First of all you should get hold of MUI (MagicUserInteface) and install it. MUIProCalc relies on MUI, since the complete User Interface was written using this great system.

Having installed MUI (if you haven't done so already), you may copy the program MCalc and it's icon MCalc.info to a place you like.

If you want to use the online help, you have to have AmigaGuide(C) installed. If so, move the supplied MCalc.guide file to the same directory you moved the main program to.

That's it. In order to get online help, simply position the mouse over the object you want to get help for and MUIProCalc will pop up a window containing the help text.

One word to the menu shortcuts : They might not be Style Guide compliant, but I think it'll work just fine, anyway ;)

NOTE that you better set the stack size to at least 8192 bytes

## 1.6 MCalc.guide/Usage

Using MUIProCalc

\*\*\*\*\*

Well, it isn't that hard to use MUIProCalc. The main input window is divided into three areas, which are

Input/Output	Where you type and where you read
Function tables	Available functions
Input fields	Which numbers and characters to use ↔
Calculating standard deviation	Special feature to calculate std. ↔
Calculating linear regression	Special feature to calculate lin. ↔
Special notes on usage	Some special notes on usage
Menus	MUIProCalcs menus
ARexx	ARexx features

Within the window's titlebar you'll see the currently selected output modes. The first char is for base, the second for size, the third for sign and the

fourth entry is for angle settings.

## 1.7 MCalc.guide/InputOutput

The input/output area  
=====

Within this area (the upper) you see a String gadget and a ListView gadget. The String gadget is used to enter the expressions you want to calculate.

Within the ListView you see a history of all entered expressions and results. Per default the left column of the ListView is used to display the expression, whilst the right one reflects the result. You may set the number of lines bufferd using the Preferences. Furthermore you may tell MUIProCalc how to format the entry, i.e. you may have four different output types. See History settings, to find out more about history settings.

## 1.8 MCalc.guide/FuncTables

Function tables  
=====

Within this area you'll find all the functions supported by MUIProCalc. Simply click on the function you want to be added to the expression. Depending on your MUI Preferences you, will either see a Cycle gadget, or a set of registers. Using either of those, you may switch between five function tables. The first set is a more mathematical set, whilst the second might only be of use to programmers. There are functions for bit-manipulation etc.pp. The third contains a set of memories, you may store temporary results in. The fourth and the fifth are for standard deviation and linear regression calculations respectively.

This Group may be hidden, thus leading to more space. See Window layout, to find out more about that.

## 1.9 MCalc.guide/Inputs

Input fields  
=====

These are buttons probably known from any other calculator. Simply click on the digit/operator you want to add to your expression. The characters 'A-F' are used to enter hexadecimal values. The other characters 'I-Z' are memories. These can be set using an expression of

---



the form

$$x = \text{sqrt} (2*3)$$

This expression will set memory 'X' to the result of 'sqrt (2\*3)'. You may now use this value during other operations. You may even set a memory relative to another one, like

$$x = y + \text{sqrt} (2*3)$$

MUIProCalc supports four different value formats, these are :

#### Decimal

Decimal values are entered with no additional characters, like '2.12322'. You may even enter doubles using scientific notation, like '2.123e-2'

#### Hexadecimal

These values are preceded by a '\$'-sign or by '0x'

#### Octal

These values are preceded by a '\'-sign

#### Binary

These values are preceded by a '%'-sign

The '!' operator is used to calculate the factorial of a value. This one is limited to a value of !170.

The 'Exec' button may be used to 'start' the calculator. This can also be done, by simply hittin RETURN from within the String gadget

For percentage calculations there are several modes and functions:

\* X + Y%

Will add Y% of X to X

\* X - Y%

Will subtract Y% of X from X

\* X \* Y%

Will calculate the Y percentage of X

\* X %ch Y

Will return the percentual changes between X and Y

\* X %t Y

Y is which percentage of X? Eg. 250 %t 10 will return 4.

Here is a list of functions, you probably can't make anything out of, on first sight:

---

\* (n k)

Usage in calucaltor: `n_expression ' (n k)' k_expression`. This will calculate the expression '`n over k`' ( $N!/(N-K)!$ )

\* YRoot

Usage in calculator: `y_root 'YRoot' x_value`. This will calculate the yth root of x.

\* REZ

Usage in calculator: `'REZ' x_value`. This will calculate the reciprocal of x.

\* EXG

Usage in calculator: `mem_1 'EXG' mem_2`. This function will swap contents of the specifed memories (`mem_1` and `mem_2`)

`'CLR'` clears the current display.

`'ACLR'` will clear the display and all memories plus standard deviation values and linear regression value pairs.

This Group may be hidden, thus leading to more space. See Window layout, to find out more about that.

## 1.10 MCalc.guide/StdDeviation

Calculating standard deviation

=====

For calculating statistical functions, such a the standard deviation or the linear regression, MUIProCalc offers a somewhat different approach for entering the values. I will now explain the actions you have to take, in order to calculate standard deviation for a set of values.

First, you have to initialize the list of values, this is accomplished through the SD command. It expects, enclosed in brackets, a list of values, each seperated by a comma.

`SD(120, 145, 80, 65, 120)`

Would set up five values for standard deviation calculations. You may enter expressions for each value, like '`120*2/X`', of course.

Now, according to the values entered, MUIProCalc offers functions for calculation, which are:

`SD_AVE`

This function will return the average value of the values entered.

**SD\_PDEV**

Will calculate the standard deviation of the population for the values entered.

**SD\_SDEV**

Will calculate the standard deviation of the sample for the list of values entered.

**SD\_QSUM**

Will return the sum of all values, each value powered by 2, ie.  $\text{SIGMA } X^2$

**SD\_SUM**

Will return the sum of all values, ie.  $\text{SIGMA } X$

**SD\_NUM**

Will return the number of elements in the list

## 1.11 MCalc.guide/LinRegression

### Calculating linear regression

=====

This set of functions is nearly the same as the one described for standard deviation. Again you first have to enter some values, this time pairs of values, for the calculator to work with. As linear regression works with pairs of values, you somehow have to bring two values into some sort of relationship. This is accomplished, for example, by:

`LR(10;1003, 15;1005, 20;1010, 25;1011, 30;1014)`

This will set up a list of pairs of values. This list of X values will be set to 10, 15, 20, 25, and 30, while the list of Y values will be set to 1003, 1005, 1010, 1011, 1014. As you can see, a pair is set up by the X value, followed by a semicolon and the Y value. Each pair is, again, separated through a comma.

Here is a list of functions to work with linear regression:

**LR\_XAVE, LR\_YAVE**

Will return the average value for the X or Y values respectively

**LR\_XPDEV, LR\_YPDEV**

Will return the standard deviation of the population for either the X values, or the Y values

**LR\_XSDEV, LR\_YSDEV**

Will calculate the standard deviation of the sample for either the X values, or the Y values

**LR\_XNUM, LR\_YNUM**

This function either returns the number of X values, or the number of Y values. Of course, these numbers will always be the same ;)

LR\_XQSUM, LR\_YQSUM

Will calculate the sum of X values, or Y values powered by 2, ie.  
 $\text{SIGMA } X^2$  or  $\text{SIGMA } Y^2$

LR\_XSUM, LR\_YSUM

Will return the sum of X values, Y values

LR\_XYSUM

Will return the sum of the products of each X,Y value pair, ie.  
 $\text{SIGMA } (X*Y)$

LR\_ALPHA

Will calculate the alpha part of the formula of regression ( $y = \text{alpha} + \text{beta} * x$ )

LR\_BETA

Will calculate the beta part of the formula of regression ( $y = \text{alpha} + \text{beta} * x$ )

LR\_ASSX <expr>

Given a fixed value or an expression for the Y value, this function will try to assess the value of the X value

LR\_ASSY <expr>

Given a fixed value or an expression for the X value, this function will try to assess the value of the Y value

LR\_CORR

Will return the correlation coefficient for the linear regression  
(r)

LR\_CCORR

Will return the critical correlation coefficient for the linear regression ( $r^2$ )

LR\_COVAR

Will return the covariant for the linear regression.

## 1.12 MCalc.guide/SpecNotes

Some special notes on usage

=====

MUIProCalc doesn't really act like a real calculator, you probably own. For example, if you like to calculate the square root of  $9^3$ , you usually follow these steps to calculate it with your calculator:

Enter 9

Press your 'x powered by y' key ( $x^y$ )

Enter 3

(Hit your '=' key), which depends on which calculator you use

Hit your square root key

You get the result (27)

With MUIProCalc you take a different approach. This is almost the same as when writing the formula down a sheet of paper. Usually you would write 'sqrt (9^3)' or 'sqrt 9^3'. This is the same with MUIProCalc: You enter this formula into the Input gadget and simply hit the Exec gadget or hit the RETURN key. MUIProCalc will now return the result (again 27; well, how come?). As you can see, this is somewhat different to normal or rpn calculators. You enter the formula as if you wrote it down, and let MUIProCalc parse and evaluate the expression.

## 1.13 MCalc.guide/Menus

The menus

=====

MUIProCalc offers a set of menus. These are :

Project	The Project menu
History	Picking entries from the history
Display Bases	How to format the output
Size	Output value size
Sign	Signed/Unsigned output
Angle	How to interpret the angle
Windows	Open up several support windows

## 1.14 MCalc.guide/Project

Project

-----

The Project menu offers three entries :

Preferences...

Will open the Preferences editor (see Preferences)

About...

Will display about messages

Quit

Will quit MUIProCalc

## 1.15 MCalc.guide/PickHistory

History

-----

---

If you prefer to use your mouse or the menu shortcuts you may paste an entry from the history to the current cursor position from within this menu :

Pick left entry

Will pick the left entry from the list and paste it to the current cursor position. If only one column is shown, this will be used (see Preferences).

Pick right entry

Will pick the right entry from the list and paste it to the current cursor position. If only one column is shown, this will be used (see Preferences).

Copy left entry

Selecting this item will copy the left entry from the history to the selected clipboard unit. If only one column is show, this one will be used (see Preferences).

Copy right entry

Selecting this item will copy the right entry from the history to the selected clipboard unit. If only one column is show, this one will be used (see Preferences).

Delete entry

This menu item will cause MUIProCalc to delete the selected entry from the history.

## 1.16 MCalc.guide/Bases

### Display Bases

-----

This menu may be used to set the display mode for output. You may have 'Decimal', 'Hexadecimal', 'Octal', or 'Binary' output.

In any other mode than 'Decimal' there will be no difference between 'Signed' and 'Unsigned' modes (see Sign).

The default output type may be set using Output settings.

## 1.17 MCalc.guide/Size

### Size

----

For the 'Hexadecimal', 'Octal', and 'Binary' modes (see Display Bases), you may set an output value size (number of bits). This will reduce/increase maximum numbers.

---

The default output type may be set using Output settings.

## 1.18 MCalc.guide/Sign

Sign

----

For 'Decimal' output (see Display Bases), you may have signed or unsigned results, when you enter 'Hexadecimal', 'Octal', or 'Binary' values.

The default output type may be set using Output settings.

## 1.19 MCalc.guide/Angle

Angle

-----

For trigonometrical functions you may set the type of angle you've entered. If you set 'RAD', the values will be interpreted as of type radians, if you set 'DEG' angles will be interpreted as of type degree. Finally there is type 'GRA' so that angles are interpreted as of type grads.

The default output type may be set using Output settings.

## 1.20 MCalc.guide/Windows

Windows

-----

Within this menu you will find three entries, each of which will open a support window:

Std.Deviation

This one will open the support window for standard deviation. Within the window you will find a list of all values entered for standard deviation. This is just for validation checks, in order to see, whether you've entered correct values. Basically it's of no use for calculation.

Lin.Regression

Same as for the Std.Deviation window. Within this window you will find the list of pairs of values entered for linear regression. Again, this is just for validation.

Constants

---

This is a more useful window. It will list 31 constants, useful for Physicians, or anyone else who is calculating with one of the well known constants. If you never ever had the nerve to learn all those constants by heart, here they are (well, I guess not all). Simply double click on an entry within the list, and the value will be copied to the Input gadget.

## 1.21 MCalc.guide/ARexx

The ARexx Port

=====

In addition to the default ARexx commands every MUI application knows about, MUIProCalc offers some more, these are :

As with every MUI Application the name of the ARexx Port is constructed from the base name of the Application. For MUIProCalc this will be 'MCALC'.

BTW: If you read something about TeX compatible output, this means you may use these strings directly within a mathematical environment. The current conversion routine isn't quite intelligent I have to admit. Maybe I'll add a smarter one for the next release...

There are two scripts included for use with Cygnused which demonstrate how to use the ARexx Port.

CALC	Calculate expression
CALCTEX	Calculate expression; Result is TeX compatible output
GETOUTPUT	Get output from history
GETTEXOUTPUT	Get output from history; TeX compatible
GETINPUT	Get input from history
GETTEXINPUT	Get input from history; TeX compatible
FORMATTEX	Format an expression to TeX compatible output
SETMODE	Set modes for ARexx calculation
DELETEHISTORY	Delete an entry from history list

## 1.22 MCalc.guide/CALC

CALC - Command

-----

Format:

CALC EXPRESSION

Template:

EXPRESSION/A

Parameter:



Any valid expression as accepted by MUIProCalc

Result:

Will be a string containing the calculated expression

Example:

CALC '1 + 2 + 3'

6.000000

## 1.23 MCalc.guide/CALCTEX

CALCTEX - Command

-----

Format:

CALCTEX EXPRESSION

Template:

EXPRESSION/A

Parameter:

Any valid expression as accepted by MUIProCalc

Result:

Will be a string containing the calculated expression converted to TeX compatible format.

Example:

CALCTEX '1 / 100000'

$1 \cdot 10^{-5}$

## 1.24 MCalc.guide/GETOUTPUT

GETOUTPUT - Command

-----

Format:

GETOUTPUT [ENTRY]

Template:

ENTRY/N

Parameter:

Supply an entry number you want to get from the history's output list. ENTRY may either be

- \* n Any valid entry ranging from 0-n, where n is the last entry in the list. If an illegal value is supplied, MUIProCalc will

return the last entry from the list.

\* -1 For the active entry

\* -2 For the topmost entry

\* -3 For the bottommost entry

Result:

Will be a string containing the expression at the specified entry.

Example:

GETOUTPUT 2

12.0000001

## 1.25 MCalc.guide/GETTEXOUTPUT

GETTEXOUTPUT - Command

-----

Format:

GETTEXOUTPUT [ENTRY]

Template:

ENTRY/N

Parameter:

Supply an entry number you want to get from the history's output list. ENTRY may either be

\* n Any valid entry ranging from 0-n, where n is the last entry in the list. If an illegal value is supplied, MUIProCalc will return the last entry from the list.

\* -1 For the active entry

\* -2 For the topmost entry

\* -3 For the bottommost entry

Result:

Will be a string containing the expression at the specified entry, formatted to TeX compatible output.

Example:

GETTEXOUTPUT 3

$1 \cdot 10^{-5}$

---

## 1.26 MCalc.guide/GETINPUT

GETINPUT - Command

-----

Format:

GETINPUT [ENTRY]

Template:

ENTRY/N

Parameter:

Supply an entry number you want to get from the history's input list. ENTRY may either be

- \* n Any valid entry ranging from 0-n, where n is the last entry in the list. If an illegal value is supplied, MUIProCalc will return the last entry from the list.
- \* -1 For the active entry
- \* -2 For the topmost entry
- \* -3 For the bottommost entry

Result:

Will be a string containing the expression at the specified entry.

Example:

GETINPUT 2

2 + 2 + 3 + 4

## 1.27 MCalc.guide/GETTEXINPUT

GETTEXINPUT - Command

-----

Format:

GETTEXINPUT [ENTRY]

Template:

ENTRY/N

Parameter:

Supply an entry number you want to get from the history's input list. ENTRY may either be

- \* n Any valid entry ranging from 0-n, where n is the last entry in the list. If an illegal value is supplied, MUIProCalc will return the last entry from the list.
  - \* -1 For the active entry
-

\* -2 For the topmost entry

\* -3 For the bottommost entry

Result:

Will be a string containing the expression at the specified entry, formatted to TeX compatible output.

Example:

```
GETTEXINPUT 3
```

```
1\cdot 32
```

## 1.28 MCalc.guide/FORMATTEX

FORMATTEX - Command

-----

Format:

```
FORMATTEX EXPRESSION
```

Template:

```
EXPRESSION/A
```

Parameter:

Supply an expression you would like to have converted to TeX compatible format.

Result:

Will be a string containing the converted expression, formatted to TeX compatible output.

Example:

```
FORMATTEX '1 * 10^-5'
```

```
1\cdot 10^{-5}
```

## 1.29 MCalc.guide/SETMODE

SETMODE - Command

-----

Format:

```
SETMODE [BASE=DEC|HEX|OCT|BIN] [SIZE=8|16|32]  
[SIGN=SIGNED|UNSIGNED] [ANGLE=RAD|DEG|GRA]
```

Template:

```
BASE/K, SIZE/K, SIGN/K, ANGLE/K
```

**Parameter:****BASE**

Specify a new conversion/output base for ARexx calculations.  
You may supply either one of the following :

- \* DEC Decimal output
- \* HEX Hexadecimal output
- \* OCT Octal output
- \* BIN Binary output

**SIZE**

Specify the default conversion size for hexadecimal/octal/binary values. You may supply either one of the following :

- \* 8 8 Bits
- \* 16 16 Bits
- \* 32 32 Bits

**SIGN**

Specify signed or unsigned output. Supply either one of :

- \* SIGNED Signed output
- \* UNSIGNED Unsigned output

**ANGLE**

Specify the default type of supplied angles. This may either be :

- \* RAD Angles of type radians
- \* DEG Angles of type degree
- \* GRA Angles of type grads

**Result:**

This will always be FALSE (0)

**Example:**

```
SETMODE 'BASE=HEX' 'SIZE=8' 'SIGN=SIGNED' 'ANGLE=DEG'
```

## 1.30 MCalc.guide/DELETEHISTORY

DELETEHISTORY - Command

-----

**Format:**

```
DELETEHISTORY [ENTRY]
```

---

**Template:**

ENTRY/N

**Parameter:**

Supply an entry number you want to delete from the history's input list. ENTRY may either be

- \* n Any valid entry ranging from 0-n, where n is the last entry in the list. If an illegal value is supplied, MUIProCalc will not delete anything.
- \* -1 For the active entry
- \* -2 For the topmost entry
- \* -3 For the bottommost entry

**Result:**

This will always be FALSE (0)

**Example:**

GETTEXINPUT 3

This will delete the fourth entry from the list

## 1.31 MCalc.guide/Prefs

**Preferences**

\*\*\*\*\*

Within the Preferences window you may configure some of MUIProCalc's default values. These are :

Window layout	How the main window should be set up
Output settings	Default output modes
History settings	Set up history ListView
Misc	Miscellaneous

Furthermore the active function table, set with the 'Functions' cycle gadget within the main window, will be taken into account.

**Use**

will use the current settings and store them temporarily, so if you restart MUIProCalc, these settings will be used again.

**Save**

will save/store the settings on disk.

**Cancel**

will restore the previously used values.

## 1.32 MCalc.guide/Layout

Window layout

=====

Within this group there are two CheckBox gadgets. if the 'No Functions' CheckBox is set, the main window will not contain the function tables. If not the function tables will be displayed. If you set 'No Input' the input fields will not be displayed. If you clear this CheckBox, the lower gadget group will be displayed.

Using the 'In/Out Order' gadget you may control the ordering of the input String gadget and the output ListView gadget. When 'Input then Output' is selected, the String gadget will appear 'on top' of the ListView. Otherwise ('Output then Input') it will appear underneath the ListView.

## 1.33 MCalc.guide/Output

Output settings

=====

You may set the default values for 'Display Bases', 'Size', 'Sign' and 'Angle' here.

See Display Bases Size Sign Angle

## 1.34 MCalc.guide/History

History settings

=====

Using the 'Lines' Slider gadget you may adjust the maximum number of lines to capture during a session.

With the 'Entries' Cycle gadget you may set the output format for the history. You may choose from the following formats :

Input & Output

Within the left column you'll see the expression entered. In the right column there'll be the result for that expression.

Output & Input

Is the opposite to the above

Input

Will only show your input expression

Output

Will only show the result

---

Using the 'ClipUnit' Slider gadget you may set the default clipboard unit for History clipping. Simply select a unit you want to use.

## 1.35 MCalc.guide/Misc

Misc  
====

Set the 'Clear input on exit' CheckBox gadget if you prefer to have the input gadget cleared every time you've entered an expression. Otherwise simply clear it.

The 'Flush memory' CheckBox gadget may be used to tell MUIProCalc to flush all unused libraries (including unused MUI-Classes) from memory on exit, thus freeing all that precious memory used up by MUIProCalc.

The 'Deviation window', 'Regression window' and 'Constants window' tell MUIProCalc whether to open the corresponding windows, or not ('Std.Deviation', 'Lin.Regression' and 'Constants' windows). Set a CheckBox gadget in order to make MUIProCalc open that window on startup.

## 1.36 MCalc.guide/Thanks

Whom I must say 'thank you'  
\*\*\*\*\*

I would like to thank the following people

Dirk Federlein  
For betatesting and suggestions

Stefan Stuntz  
Especially for MUI and for answering all my questions...

Volker Güth  
For being the first (and only one so far) contacting me just because of MUIProCalc...  
  
...and all the others I forgot

## 1.37 MCalc.guide/UpDates

How to obtain updates  
\*\*\*\*\*

---



MUIProCalc will be available on ADS, FRAS, FTP, and Fred Fish disks as soon as there's a new release available.

## 1.38 MCalc.guide/Address

How to contact the author  
\*\*\*\*\*

If you have any questions, suggestions, bug reports or anything else you want to tell me, you may contact me under one of these addresses.

Either use normal snail mail:

Kai Iske  
Brucknerstrasse 18  
63452 Hanau  
Germany  
Tel.: +49-(0) 6181-850181

or reach me using electronical mail

INTERNET: iske@informatik.uni-frankfurt.de  
COMPUSERVE: Kai Iske, 100524,1201  
FIDO: Kai Iske, 2:244/6302.11  
ZNET: KAI@SWEET.RHEIN-MAIN.DE

## 1.39 MCalc.guide/PrgHistory

Program history  
\*\*\*\*\*

-----  
Version 1.5 (21.01.95 17:43:49)  
-----

- Added new function to menu/ARexx. DELETEHISTORY will delete an entry from the history list.  
(Requested by : Sorry, I lost you name, but thanks for the suggestions and the translation of the manual)
  - Added support for GRADS  
(Requested by : Sorry, I lost you name, but thanks for the suggestions and the translation of the manual)
  - Introduced exg operator, which exchanges the contents of two memories  
(Requested by : Sorry, I lost you name, but thanks for the suggestions and the translation of the manual)
  - Added great support for physical constants. 31 listed
-

- (Requested by : Sorry, I lost you name, but thanks for the suggestions and the translation of the manual)
- Added full features linear regression support  
(Requested by : Volker Güth)
  - Added standard deviation support  
(Requested by : Volker Güth)
  - Added (n k) n-over-k function, 1/x, and the y-th root of x function.  
(Requested by : Sorry, I lost you name, but thanks for the suggestions and the translation of the manual)
  - Increased number of internal memories from 3 to 18 (I-Z)
  - Changed function group to use Registers/Cycle Gadget, according to your MUI settings.
  - MCalc requested any muimaster.library, even though version 7 (MUI 2.x) was required
  - Some of the input gadgets didn't react on key nor mouse activations.  
(Reported by : Andreas Kirchwitz, and several others)
  - Recompiled scanner using Flex 2.4.7
  - The format identifier for binary values has been changed; I forgot to change the output format identifier as well.

-----  
Version 1.4 (18.08.94 17:35:24)  
-----

- MCalc didn't even care about Percent calculations
- atan() wasn't definied for some ranges (somehow)
- The InputString will be default object now
- Recompiled with Scanner generated by flex 2.4.6

-----  
Version 1.3 (11.02.94 23:35:32)  
-----

- One couldn't use 1e3 for scientific notation of 1000. You had to write 1.0e3. Changed  
(Reported by : Henry Norman)
  - No more error lines will be copied from History  
(Requested by : Stefan Stuntz)
  - "Radiant" was wrong. Changed. Why didn't anybody tell me.
-

(Reported by : Henry Norman)

- Chose wrong case for mathematical "e".  
(Reported by : Henry Norman)

-----  
Version 1.2

(08.12.93 23:48:31)  
-----

- Default number of history lines increased  
(Requested by : Stefan Stuntz)
- No checks were made on the ENTRY argument to an ARexx function
- For calculations via ARexx one can set the type, size, sign and angle flags seperately, now
- Improved error handling from within ARexx commands a bit
- MUIProCalc may output TeX style mathematical expressions when using the ARexx commands. Additionally there is another option to convert expressions to TeX format.  
(Requested by : Volker Güth)
- Added more ARexx functions to the default ones  
(CALC, GETINPUT, GETOUTPUT)  
(Requested by : Volker Güth)
- MUIProCalc didn't handle extremely small values  
(Reported by : Volker Güth)
- Default values weren't set up
- ACLR and CLR gadget have got menu entries too, now  
(Requested by : Volker Güth)
- History entries may be copied to clipboard now  
(Requested by : Volker Güth)
- Recompiled using SAS 6.50
- Bug in ListView display fixed
- You may now tell MCalc to clear the input string after hitting RETURN or EXEC.

-----  
Version 1.1

(17.10.93 15:03:10)  
-----

- Save and Use gadgets weren't arranged as proposed by the AUISG. (Reported by: Stefan Stuntz)
  - Changed layout of Prefs window a bit
-

(Suggested by: Stefan Stuntz)

- NOT operator didn't behave as expected ;) It was a simple negate... (Reported by : Top on IRC)
- Hex-Values may be entered using 'C'-Style now (e.g. 0xff)
- One can use the faktorian '!' behind numbers now, too.  
(Suggested by : Top on IRC)
- One could accidentally change the ordering of the functions and input-gadgets groups by using 'No Functions' and 'No Input' (reported by: Top on IRC)
- Order of Input and Ouput gadgets (String + ListView) may be set now, ie. the input string may show up underneath the ListView and vice versa (Requested by Dirk Federlein)

-----  
Version 1.0

(02.10.93 02:16:34)  
-----

--- Initial release ---

---