Instructions

Arithmetic operations

2.5+3	C 2 · 5 + 3 =
5 – 1	C 5 – 1 =
12÷4	C 9 × 4 =
9×4	C 1 2 ÷ 4 =

Scientific Functions

200 !	C 2 0 0 !
$\sqrt{3}$	C 3 √
sin(10)	C 1 0 sin
cos(30)	C 3 0 cos
tan(45)	C 4 5 tan
ln(1.5)	C 1 · 5 ln
$e^{1.75}$	C 1 · 7 5 e^x
50^2	C 5 0 x^n 2 =
π	π (replaces display contents)

Non Volatile Memory

This memory can contain a single element. It is persistent and it is saved into a file as part of the state of the calculator.

MS	Store display content
MR	Copy memory content to display
MC	Clear memory
M+	Cumulate display content into memory

Degrees / Radians

Trigonometric functions can be calculated with angles in degrees or in radians. The default settings is degrees. To check which is the current settings one should invoke the option menu. The first menu shows the setting to be changed to, that is, the opposite setting.

Usage: Select option menu "Degrees" (if the current setting is radians) or "Radians" (if the current setting is degrees).

Volatile Memory

This memory can contain any number of elements. It is volatile and it is not systematically saved into file but its contents can be saved and restored explicitly. It works as a stack where the last element saved remains on the top. The goal of this memory is to store values temporarily such as in complex calculations. It allows replacing the use of parenthesis by operating in polish notation order. This requirement is compensated by simplicity of use and unlimited calculation depth.

Usage: Select option menu "Memory" and choose one of the following options:

Save: Copy display contents to the top of stack

Remove: Transfer the top of stack to the display

Get last saved element: Copy the top of stack to the display

Get at position in display: Copy to the display the element at the position currently shown in the display

Clear at position in display: Delete the element at the position currently shown in the display

Save all elements to file: Save the stack into a file

Get all elements from file: Restore the stack from saved file

Clear: Delete the stack

Precision Settings

MaxCalc is a multi-precision calculator. This means that it is able to calculate with arbitrary precision. However, due to an unexpected behavior in single line display in Android, maximum precision is limited to 817. This limitation also allows limiting processing time. MaxCalc starts with a default precision of 300. To change this setting follow the instructions below.

Usage: Select option menu "Precision" and choose one of the following options:

Set to default value: Set precision to 300

Set to value in display: Set to the precision shown in display

Set to 100: Set precision to 100

Set to 200: Set precision to 200

Set to 500: Set precision to 500

Set to 817: Set precision to 817 (the maximum)

Numerical Algorithms

In MaxCalc, the square root is calculated through Newton method, standard first estimation techniques and a custom method to optimize calculating roots of numbers between 0 and 1. The other transcendental functions are calculated with custom optimized Taylor series, except for exponential that uses known standard optimizations of the series as well as of domain reduction. MaxCalc uses 3 more decimal digits precision in the computation of transcendental functions using half-up rounding mode, the default rounding. The result is then rounded with the rounding mode set through the Rounding Option Menu (see Rounding mode Settings below). In arithmetic operations this later is the rounding mode applied with the normal precision (see "Precision Settings") .

Rounding Mode Settings

This is a feature not commonly found in calculators. In general, the precision is fixed and the rounding mode is unknown as well as the numerical algorithms used.

Usage: Select option menu "Rounding" and choose one of the following options:

Default (half_up):
Set to default rounding mode, half_up (see below)

Ceiling (plus infinity):
Rounding mode to round towards positive infinity. If the result is positive, behaves as for "Up" rounding mode; if negative, behaves as for "Down" rounding mode. This rounding mode never decreases the calculated value. The effect is the same as in "ceiling" function.

Floor (minus infinity):
Set Rounding mode to round towards negative infinity. If the result is positive, behave as for "Down" rounding mode; if negative, behave as for "Up" rounding mode. This rounding mode never increases the calculated value. The effect is the same as in "floor" function.

Up (away from zero):
Rounding mode to round away from zero, that is, the inverse of "towards zero" rounding (see below). Always increments the digit prior to a non-zero discarded fraction. Note that this rounding mode never decreases the magnitude of the calculated value.

Down (towards zero):
Set to "down" rounding, that is, towards zero rounding. This rounding approaches the rounded value to the nearest value on the left if the value is positive or on the right if the value is negative.

Half_even (nearest):
Rounding mode to round towards the "nearest neighbor" unless both neighbors are equidistant, in which case, round towards the even neighbor. Behaves as for half_up if the digit to the left of the discarded fraction is odd; behaves as for half_down if it's even. Note that this is the rounding mode that statistically minimizes cumulative error when applied repeatedly over a sequence of calculations. It is sometimes known as "Banker's rounding," and is chiefly used in the USA. This rounding mode is analogous to the rounding policy used for float and double arithmetic in Java.

Half_up (nearest up):
Rounding mode to round towards "nearest neighbor" unless both neighbors are equidistant, in which case round up. Behaves as for the rounding mode "up" if the discarded fraction is greater or equal to 0.5; otherwise, behaves as for rounding mode "down". Note that this is the rounding mode commonly taught at school.

Half_down (nearest down):
Rounding mode to round towards "nearest neighbor" unless both neighbors are equidistant, in which case round down. Behaves as for rounding mode "up" if the discarded fraction is greater than 0.5; otherwise, behaves as for rounding mode "down".

Unnecessary:
Rounding mode to assert that the requested operation has an exact result, hence no rounding is necessary. If this rounding mode is specified on an operation that yields an inexact result an error is notified.

SD Storage

MaxCalc allows saving all values in the calculator to a text file in the SD card storage. It also allows reading from a list of text files to set the corresponding values. Since the SD storage is accessible by a PC, the text files can be read, edited or even created by hand. For this reason these files have a particular format. While reading them MaxCalc applies a meticulous error checking which allows interrupting the reading process once one of the values is not correct. The name of the files obey the following format:

File name format: save<n>.txt

Where *<n>* is a number identifying the file. When saving, the file names are created in this format starting with *n = 0*. If other files exist the value of *n* is equal to the number of files in *MaxCalc* directory obeying the above format. If this directory does not exist it will be created. When reading, the files will be searched in this directory and all files obeying the above format are presented to the user in a menu. Once a particular file is chosen it will be read. Since only the files that fit this format are presented in the list, files with other name formats can exist in this directory without being noticed by MaxCalc. If the SD card is being used (e.g. being accessed through an USB cable in a PC) it will be inaccessible during the time it is blocked. An error message will appear to let the user aware of the situation. Once it is released, files can again be read and written without the need of exiting MaxCalc. To be able to read or to create these files, it is important to understand their internal structure. Each line is a string representing a particular value. This means that every string must end by the character '\n' (new line) which is interpreted in common text editors as the end of the line. Notice that Windows "Notepad" does not behave in this way since it also requires a carriage return character at the end of each line. Therefore, MaxCalc text files cannot be read neither created using "Notepad". Alternatively, one should use Windows "Wordpad", "Notepad++" or any other cross-platform text editor. The internal file structure is given below:

File structure:

< display value > '\n'
< value in accumulator > '\n'
< value in non volatile memory > '\n'
< operation: add, sub, mul, div, x^n > '\n'
< rounding mode: UP, DW, CE, FL, HU, HD, HE, UN > '\n'
< precision > '\n'
< number of elements in volatile memory > '\n'
< first element in volatile memory > '\n'
< second element in volatile memory > '\n'
...
< last element in volatile memory > '\n'

If a particular line is empty, that is, it has only a new line character, the corresponding value in MaxCalc is not changed. Incomplete files are not considered wrong, as far as all the values supplied are given correctly. The only effect is that values expected to be changed by the inexistent lines will remain the same as before the file was read.

Usage: Select option menu "Save to SD", to save values to a text file, or "Get from SD", to read a file from a list of files in a submenu that will open.

Back Key ⮌

This key implements an undo functionality or offers to exit the application. The undo functionality will most often be used after an error occurs. It allows recovering the values just before the error, so the origin of the error can hopefully be corrected. It can also be used to obtain different operations with previous operation and accumulator values by modifying the display value. It also allows to obtain the value for which a scientific function has been called.