GNU Calc Reference Card
(for GNU Emacs version 29)

Starting and Stopping

- `C-x * k` start/stop standard Calc
- `C-x * c` start/stop X keypad Calc
- `C-x * *` stop standard Calc
- `Calc` tutorial
- `run Calc in other window`
- `quick calculation in minibuffer`

Getting Help

- `h f` describe key fully
- `h c` ?
- `h`
- `Getting Help`
- `C-x * q` quick calculation in minibuffer
- `C-x * o` run Calc in other window
- `Calc tutorial`
- `C-x * t` stop standard Calc
- `q`
- `h s` read full Calc summary
- `C-x * i`
- `C-x * s`

In RPN, enter numbers first, separated by `;` if necessary, then type the operator. To enter a calculation in algebraic form, press the apostrophe first.

Examples

- `2 RET 3 +` `1+2=4` in algebraic
- `2 3 ^` `2^3=8`
- `2 3 *` `4*3=12`
- `2 3 /` `2/3=0.67`

Transferring Data

- `C-x * g` grab region from a buffer
- `C-x * t` grab rectangle, summing columns
- `C-x * o` grab rectangle, summing rows
- `C-x * y` yank data to a buffer

Also, try C-k/C-y or X cut and paste.

Error Recovery

- `abort command in progress` C-g
- `display recent error messages` w
- `undo last operation` U
- `redo last operation` D
- `recall last arguments` M-RET ;
- `edit top of stack` h
- `reset Calc to initial state` C-x * 0 (zero)

Display

- `scroll horizontally, vertically` < >, { }
- `home cursor` o
- `line numbers on/off` d l
- `trail display on/off` t d
- `scientific notation` d s
- `fixed-point notation` d f
- `floating-point (normal) notation` d n
- `group digits with commas` d g

For display mode commands, H prefix prevents screen redraw and I prefix temporarily redraws top of stack.

Notations

- `scientific notation` 6.02e23
- `minus sign in numeric entry` 23 or 23 n
- `fractions` 3:4
- `complex numbers` (x, y)
- `polar complex numbers` (r; θ)
- `matrices (or nested vectors)` [1, 2, 3]
- `vectors (commas optional)` [1, 2; 3, 4]
- `error forms (p key)` 100 ±<0.5
- `interval forms` [2 .. 5]
- `modulo forms (M key)` 6 mod 24
- `HMS forms` 50 30' 0"
- `date forms` <Jul 4, 1992>
- `infinity, indeterminate` inf, nan

Scientific Functions

- `ln, log10, logb` L, H, L, B
- `exponential e^x, 10^x` E, H E
- `sin, cos, tan` S, C, T
- `arcsin, arccos, arctan` inverse, hyperbolic prefix keys I, H
- `degrees, radians modes` f T
- `two-argument arctan` m d, m r
- `pi (π)` P
- `factorial, double factorial` !, k d
- `combinations, permutations` k c, H k c
- `prime factorization` k f
- `next prime, previous prime` k n, I k n
- `GCD, LCM` k g, k l
- `random number, shuffle` k r, k h
- `minimum, maximum` f n, f x
- `error functions erf, erfc` f e, I f e
- `gamma, beta functions` f g, f b
- `incomplete gamma, beta functions` f G, f B
- `Bessel Jν, Yν functions` f j, f y
- `complex magnitude, arg, conjugate` A, G, J
- `real, imaginary parts` f r, f i
- `convert polar/rectangular` c p

Units

- `enter with units` M ✴️
- `convert to percentage` c %
- `percentage change` b %
- `present value` b P
- `future value` b F
- `rate of return` b T
- `number of payments` b N
- `size of payments` b H
- `net present value, int. rate of return` b H, I

Above computations assume payments at end of period. Use I prefix for beginning of period, or H for a lump sum investment.

Financial Functions

- `straight-line depreciation` b S
- `sum-of-years'-digits` b V
- `double declining balance` b D

© 2023 Free Software Foundation, Inc. Permissions on back.
**Programmer's Functions**

binary, octal, hex display  
\texttt{d 2, d 8, d 6}

decimal, other radix display  
\texttt{d 0, d r}

display leading zeros  
\texttt{d z}

entering non-decimal numbers  
16#7FFF

binary word size  
\texttt{b w}

binary AND, OR, XOR  
\texttt{b a, b o, b x}

binary DIFF, NOT  
\texttt{b d, b n}

left shift  
\texttt{b l}

logical right shift  
\texttt{b r}

arithmetic right shift  
\texttt{b R}

integer quotient, remainder  
\texttt{\%}

integer square root, logarithm  
\texttt{f I}

integer quotient, remainder  
\texttt{d Q}

“Big” display mode  
\texttt{C, Pascal, FORTRAN modes}

TyX, LaTeX, eqn modes  
\texttt{D T, D L, D E}

Maxima  
\texttt{D X}

Unformatted mode  
\texttt{D U}

Normal language mode  
\texttt{D N}

simplify formula  
\texttt{a s}

put formula into rational form  
\texttt{a r}

evaluate numerically  
\texttt{s 1 x=evl}

let variable equal a value in formula  
\texttt{n}

declare properties of variable  
\texttt{s l}

let variable equal a value in formula  
\texttt{n}

evaluate numerically  
\texttt{s 1 x=evl}

let variable equal a value in formula  
\texttt{n}

declare properties of variable  
\texttt{s l}

let variable equal a value in formula  
\texttt{n}

declare properties of variable  
\texttt{s l}

**Variables**

Variable names are single digits or whole words.

store to variable  
\texttt{s t}

store and keep on stack  
\texttt{s s}

recall from variable  
\texttt{s r}

shorthands for digit variables  
\texttt{t, s, n, r, n}

unstore, exchange variable  
\texttt{s u, s x}

edit variable  
\texttt{s e}

**Vector Operations**

vector of 1, 2, ..., \( n \)  
\texttt{v x n}

vector of \( n \) counts from \( a \) by \( b \)  
\texttt{C-u v x}

vector of copies of a value  
\texttt{v b}

concatenate into vector  
\texttt{v l}

pack many stack items into vector  
\texttt{v p}

unpack vector or object  
\texttt{v u}

length of vector (list)  
\texttt{v l}

reverse vector  
\texttt{v v}

sort, grade vector  
\texttt{v S, V G}

histogram of vector data  
\texttt{V H}

extract vector element  
\texttt{v r}

matrix determinant, inverse  
\texttt{V D, \&}

matrix transpose, trace  
\texttt{v t, V T}

cross, dot products  
\texttt{V C, *}

identity matrix  
\texttt{v i}

extract matrix row, column  
\texttt{v r, v c}

intersection, union, diff of sets  
\texttt{V -, V V, V -}

cardinality of set  
\texttt{V #}

add vectors elementwise (i.e., \texttt{map} \texttt{+})  
\texttt{V M +}

sum elements in vector (i.e., \texttt{reduce} \texttt{+})  
\texttt{V R +}

sum rows in matrix  
\texttt{V R - +}

sum columns in matrix  
\texttt{V R : +}

sum elements, accumulate results  
\texttt{V U +}

**Algebra**

enter an algebraic formula  
\texttt{\textasciitilde 2x+3y-2}

enter an equation  
\texttt{\textasciitilde 2x^2+18}

symbolic (vs. numeric) mode  
\texttt{m s}

fractions (vs. float) mode  
\texttt{m f}

suppress evaluation of formulas  
\texttt{m 0}

return to default evaluation rules  
\texttt{m D}

“Big” display mode  
\texttt{C, Pascal, FORTRAN modes}

TyX, LaTeX, eqn modes  
\texttt{D T, D L, D E}

Maxima  
\texttt{D X}

Unformatted mode  
\texttt{D U}

Normal language mode  
\texttt{D N}

simplify formula  
\texttt{a s}

put formula into rational form  
\texttt{a r}

evaluate numerically  
\texttt{s 1 x=evl}

let variable equal a value in formula  
\texttt{n}

declare properties of variable  
\texttt{s l}

let variable equal a value in formula  
\texttt{n}

declare properties of variable  
\texttt{s l}

let variable equal a value in formula  
\texttt{n}

declare properties of variable  
\texttt{s l}

let variable equal a value in formula  
\texttt{n}

declare properties of variable  
\texttt{s l}

**Numerical Computations**

sum formula over a range  
\texttt{a +}

product of formula over a range  
\texttt{a *}

tabulate formula over a range  
\texttt{a T}

integrate numerically over a range  
\texttt{a I}

find zero of formula or equation  
\texttt{a R}

find local min, max of formula  
\texttt{a N, a X}

fit data to line or curve  
\texttt{a F}

mean of data in vector or variable  
\texttt{u M}

median of data  
\texttt{u M}

grouped mean of data  
\texttt{u G}

sum, product of data  
\texttt{u +, u *}

minimum, maximum of data  
\texttt{u N, u X}

sample, pop. standard deviation  
\texttt{u S, I u S}

**Selections**

select subformula under cursor  
\texttt{j a}

select nth subformula  
\texttt{j n}

select more  
\texttt{j a}

unselect this, all formulas  
\texttt{j 0, j c}

copy indicated subformula  
\texttt{J RET}

delete indicated subformula  
\texttt{J DEL}

commute selected terms  
\texttt{j C}

commute term leftward, rightward  
\texttt{j L, j R}

distribute, merge selection  
\texttt{j D, j M}

isolate selected term in equation  
\texttt{j I}

negate, invert term in context  
\texttt{j M, j &}

rewrite selected term  
\texttt{j r}

**Graphics**

graph function or data  
\texttt{g f}

graph 3D function or data  
\texttt{g f}

replot current graph  
\texttt{g p}

print current graph  
\texttt{g a}

add curve to graph  
\texttt{g a}

set number of data points  
\texttt{g n}

set line, point styles  
\texttt{g n, g S}

set log vs. linear \( x, y \) axis  
\texttt{g l, g L}

set range for \( x, y \) axis  
\texttt{g r, g R}

close graphics windows  
\texttt{g q}

**Programming**

begin, end recording a macro  
\texttt{C-x (, C-x )}

replay keyboard macro  
\texttt{X}

read region as written-out macro  
\texttt{C-x * m}

if, else, endif  
\texttt{Z I, Z Z, Z J}

equal to, less than, member of  
\texttt{a <, a <, a \{}

repeat \( n \) times, break from loop  
\texttt{Z C, Z >, Z /}

“for” loop: start, end; body, step  
\texttt{Z :}

save, restore mode settings  
\texttt{Z '}

record mode settings permanently  
\texttt{Z (}

record variable value permanently  
\texttt{Z n}

record user-defined command permanently  
\texttt{Z a}

put finished macro on a key  
\texttt{Z K}

read region as written-out macro  
\texttt{C-x * m}

replay keyboard macro  
\texttt{X}

write region as written-out macro  
\texttt{C-x * m}

if, else, endif  
\texttt{Z I, Z Z, Z J}

equal to, less than, member of  
\texttt{a <, a <, a \{}

repeat \( n \) times, break from loop  
\texttt{Z C, Z >, Z /}

“for” loop: start, end; body, step  
\texttt{Z :}

save, restore mode settings  
\texttt{Z '}

record mode settings permanently  
\texttt{Z (}

record variable value permanently  
\texttt{Z n}

record user-defined command permanently  
\texttt{Z a}

put finished macro on a key  
\texttt{Z K}

define function with formula  
\texttt{Z F}

define editing  
\texttt{Z E}

record user-defined command permanently  
\texttt{Z P}

record variable value permanently  
\texttt{Z P}

record mode settings permanently  
\texttt{Z M}

**GNU Calc Reference Card**

Copyright © 2023 Free Software Foundation, Inc.
designed by Dave Gillespie and Stephen Gildea,
for GNU Emacs Calc.

Released under the terms of the GNU General Public License version 3
or later.

For more Emacs documentation, and the TyX source for this card, see
the Emacs distribution, or [https://www.gnu.org/software/emacs](https://www.gnu.org/software/emacs)