- 1234567890 .

numerics

category: **math** (shares line)

use: **change main variable** (or use as params) to numeric values

list of colour codes

0 black
1 blue
2 green
3 cyan
4 red
5 magenta
6 brown
7 white
8 grey
9 l.blue
10 l.green
11 l.cyan
12 l.red
13 l.magenta
14 yellow
15 b.white

fprint __

fprint *filepath*

category: **output** (shares line)

use: write **main variable** to open file designated by ***filepath***

setting main variable

x (line is now shared)

category: **start line** (shared)

use: **variablenamegoeshere** 5

variablename
p "" arr # make array named p

background colours

0 black
1 blue
2 green
3 cyan
4 red
5 magenta
6 brown
7 white

cls

cls

category: **output** (shares line)

use: clears the screen. currently only affects text screen

timer

timer

category: **input** (shares line)

use: **change main variable** to number of seconds past midnight

arrstdin

arrstdin

category: **input** (shares line)

use: **change main variable** to array containing lines of stdin

textmode

textmode

category: **output -- own line**

use: **suppress graphics**; force graphics commands to use text

lineinput

lineinput

category: **input** (shares line)

use: **change main variable** to string input from keyboard

flineinput __

flineinput *filepath*

category: **input** (shares line)

use: **change main variable** to string of line from open file

highlight __

highlight *colourcode*

category: **output** (shares line)

use: change background colour of upcoming text output to ***colourcode*** 0-15 (8-15 are a repeat of 0-7)

time

time

category: **input** (shares line)

use: **change main variable** to string of current time: hh:mm:ss

arropen __

arropen *filepath*

category: **input** (shares line)

use: **change main variable** to array of file lines in ***filepath***

pset __ __ __

pset *x y c*

category: **output** (shares line)

use: draw dot at location (***x***, ***y***) in colourcode ***c*** (0-15)

date

date

category: **input** (shares line)

use: **change main variable** to string of the date: mm/dd/yyyy

arrcurl __

arrcurl *url*

category: **input** (shares line)

use: like **arropen**, except downloads ***url*** into the array

while

while

category: **loop -- own line**

use: mark the start of a loop (will keep going, without **break**)

sleep __

sleep *seconds*

category: **input** (shares line)

use: wait for number of ***seconds*** before continuing with program

command

command

category: **input** (shares line)

use: **change main variable** to array of command line parameters

for __ __ __ __

for *var start stop step*

category: **loop -- own line**

use: start a **for** loop, *changing* ***var*** from ***start*** to ***stop***, by ***step***

print

print

category: **output** (shares line)

use: output **main variable** to the screen (aka stdout.) command name dates back to teletypes

prints

prints

category: **output** (shares line)

use: output **main variable** to the screen; like **print** but (s)tays on the same line.

iftrue __

iftrue *ckv*

category: **conditional -- own line**

use: run lines between **iftrue** and **fig** if ***ckv*** is "non-zero."

plus __

plus *numstrarr*

category: **math** (shares line)

use: **change main variable** to itself **plus** *num* or *string* or *array*

minus __

minus *numeric*

category: **math** (shares line)

use: **change main variable** to itself **minus** *numeric*

sqr

sqr

category: **math** (shares line)

use: **change main variable** to the square root of itself

divby __

divby *numeric*

category: **math** (shares line)

use: **change main variable** to itself **divided by** *numeric*

times __

times *numeric*

category: **math** (shares line)

use: **change main variable** to itself **times** *numeric*

topwr __

topwr *n*

category: **output** (shares line)

use: raise numeric **main variable** to the ***n***-th power

oct

oct

category: **math** (shares line)

use: **change main variable** from numeric decimal to octal

hex

hex

category: **math** (shares line)

use: **change main variable** from numeric decimal to hexadecimal

tan

tan

category: **math** (shares line)

use: **change** numeric **main variable** to its tangent

cos

cos

category: **math** (shares line)

use: **change** numeric **main variable** to its cosine

sin

sin

category: **math** (shares line)

use: **change** numeric **main variable** to its sine

int

int

category: **math** (shares line)

use: **change** numeric **main variable** from decimal (aka "float") to integer

display

display

category: **output** (shares line)

use: **1st time:** stop automatic graphics update.
2nd, etc: update

ifmore __ __

ifmore *ckv1 ckv2*

category: **conditional -- own line**

use: run lines between **ifmore** and **fig** if ***ckv1*** is > ***ckv2***

ifless __ __

ifless *ckv1 ckv2*

category: **conditional -- own line**

use: run lines between **ifless** and **fig** if ***ckv1*** is < ***ckv2***

graphics

graphics

category: **output -- own line**

use: dont (or stop) suppress(ing) graphics. this is the default for programs

try

try

category: **conditional -- own line**

use: put code that might not work between **try** and **except** to catch or suppress the error

except

except

category: **conditional -- own line**

use: **if** code between **try**/**except** fails, run the code after **except**

colourtext __

colourtext *colourcode*

category: **output** (shares line)

use: change colour of upcoming text output to ***colourcode*** 0-15

colortext __

colortext *colorcode*

category: **output** (shares line)

use: change color of upcoming text output to ***colorcode*** 0-15

resume

resume

category: **conditional -- own line**

use: mark the end if **try** / **except** / **resume** command block

locate __ __

locate *row column*

category: **output** (shares line)

use: move to textmode position at ***row, column***

function __ _? _? …?

function *name p1 p2 …*

category: **function -- own line**

use: define function named ***name*** with optional params ***p1***, ***p2***, etc.

get __

get *parametername*

category: **function** (shares line)

use: (**no longer required**) copy ***parametername*** value to main variable

line __ __ __ __

line *x1 y1 x2 y2 c*

category: **output** (shares line)

use: draw line from (***x1***, ***y1***) to (***x2***, ***y2***) in colourcode ***c*** (0-15)

python

python

category: **function -- own line**

use: put inline python code between lines **python** and **fig**

fig/next/nextin/wend

fig (interchangeable)

category: **function -- own line**

use: finalize block started by **if**/**while**/**function**/**for**/**forin** etc

break

break

category: **loop -- own line**

use: put in the middle of a loop to exit (stop looping)

pass

pass

category: **function -- own line**

use: blocks (for/next, etc) require *something* inside lines; **pass** works / does nothing

else

else

category: **conditional - own line**

use: after **if-** line, before **fig**. run lines if condition **isnt** true

forin __ __

forin *var array*

category: **loop -- own line**

use: loop through each item in ***array***; for each, set ***var*** to item

lcase

lcase

category: **function** (shares line)

use: **change main variable** to all-lower-case copy of own value

ucase

ucase

category: **function** (shares line)

use: **change main variable** to all-upper-case copy of own value

str

str

category: **function** (shares line)

use: convert main variable from numeric to string

ifequal __ __

ifequal *ckv1 ckv2*

category: **conditional -- own line**

use: run lines between **ifequal** and **fig** if ***ckv1*** equals ***ckv2***

shell

shell

category: **function** (shares line)

use: run contents of main variable in a command shell (os specific)

asc

asc

category: **function** (shares line)

use: **change main variable** from string to ascii code of first character

mod __

mod *denominator*

category: **math** (shares line)

use: **change main variable** to main variable modulus ***denominator***

mid __ __

mid *position len*

category: **function** (shares line)

use: **change main variable** to range of ***len*** items or characters from ***position***

randint __ __

randint *smallst largst*

category: **input** (shares line)

use: **change main variable** to random number ranging from ***smallst*** to ***largst***

split __ __

split *string splitby*

category: **function** (shares line)

use: split ***string*** by separator ***splitby*** in to array, to main variable

join __ __

join *array usestring*

category: **function** (shares line)

use: **change main variable** to string by joining ***array*** using ***usestring***

atn

atn

category: **math** (shares line)

use: **change** numeric **main variable** to its arctangent

instr __ __

instr *lookin lookfor*

category: **function** (shares line)

use: **change main variable** to numeric position of ***lookfor*** in ***lookin***

chdir

chdir

category: **function** (shares line)

use: change current folder to path string from **main variable**

sgn

sgn

category: **math** (shares line)

use: **change main variable** to -1 if its value is < 0, or to 1 if > 0. keep at 0, if 0

system

system

category: **function** (shares line)

use: put on (usually at the end of) a line to stop the program

close

close

category: **function** (shares line)

use: close the open file designated by main variable

val

val

category: **function** (shares line)

use: **change main variable** from string to numeric (int if whole)

len

len

category: **function** (shares line)

use: **change main variable** to numeric length of main variable

arrshell

arrshell

category: **function** (shares line)

use: **change main variable** to array of shell output (from main variable)

not

not

category: **function** (shares line)

use: **change main variable** to **zero** if non-zero; or **-1** if zero

ltrim

ltrim

category: **function** (shares line)

use: strip whitespace from left side of main variable

reverse

reverse

category: **function** (shares line)

use: like **arreverse** (which might be faster for an array) but also works on strings

rtrim

rtrim

category: **function** (shares line)

use: strip whitespace from right side of main variable

chr

chr

category: **function** (shares line)

use: **change main variable** from numeric to ascii string

#

#

category: **comment** (can share)

use: place at beginning (or end) of line, prior to a comment

arreverse

arreverse

category: **function** (shares line)

use: **change main variable** from array to reverse order of array

left __

left *numofcharsoritems*

category: **function** (shares line)

use: **change main variable** to __ leftmost group of characters or items

right __

right *numofcharsoritems*

category: **function** (shares line)

use: **change main variable** to __ rightmost group of characters or items

arrsort

arrsort

category: **function** (shares line)

use: **change main variable** to sort the array stored in it

arrget __ __

arrget *array position*

category: **function** (shares line)

use: **change main variable** to ***position***-th item from ***array***

arrset __ __

arrset *position setto*

category: **function** (shares line)

use: change ***position***-th item in array in main variable to value of ***setto***

( ) : ; | = , .

(all optional)

category: **optional**

use: in a shared line (and some others) for aesthetics / notation or similarity to other languages

end

end

category: **function** (shares line)

use: interchangeable with **system** which ends the program

open __

open *mode*

category: **function** (shares line)

use: open file at filepath **main variable** in ***mode*** "r" or "w"

swap __ __

swap *var1 var2*

category: **function** (shares line)

use: change contents of ***var1*** to ***var2*** and vice-versa

return __

return *var*

category: **function** (shares line)

use: (optional) exit current **function**, returning value ***var*** (without return, defined **function** wont change main var)

string __ __

string len asciiorstr

category: **function** (shares line)

use: **change main variable** to ***len*** instances of ***ascii***or***str***

license: