

drawing with mjoy – a supplement

(Ver 2, AE, 18/Mar/2017)

Fundamentals *init* creates a nested stack that is required for drawing. *draw* executes the stack.

The pen is started in default down position. *penup* and *pendown* change the pen status.

Black is the default color of the pen. It can be changed with the *pencolor* command. *Note – colors need to be defined – see Appendix 1.*

Coordinates are given in the cartesian convention (x,y), with +x left to right, and +y down to up. Default coordinates are (0,0), the upper left of the user screen (IV Quadrant).

The pen is moved using either *moveto* (dict x y -- dict) or *moverel* (dict x y -- dict). Each word takes x y coordinates and moves the pen either to an absolute position (*moveto*) or relatively from where the current pen position is (*moverel*). The numbers are pixel lengths.

The orientation of the imagined turtle is changed by *turnto* (dict r -- dict) or *turn* (dict r -- dict). Each word takes a radian measure and turns the orientation either to an absolute angle (*turnto*) or relatively from where the current angle is (*turn*). The numbers are assumed radian degrees. The default orientation is 0, which points directly East. Note: to provide an angle in degrees, use *rad* (d -- r) to convert from degrees to radians.

Drawing

To explore drawing in mjoy, start with the following methods:

`.s == stack reverse print` This allows inspecting the stack without changing it.

`cs == start ;` Clears the drawing screen.

`go == penup 100 -100 moveto pendown 0 turnto` Positions the pen ready to draw

There are two ways to draw in mjoy:

- 1) Specify coordinates (this is a connect-the-dots metaphor)
- 2) Control the rotation and forward movement of an imaginary turtle with pen attached (this is the follow-your-nose metaphor)

Example 1 – Triangle by specifying coordinates

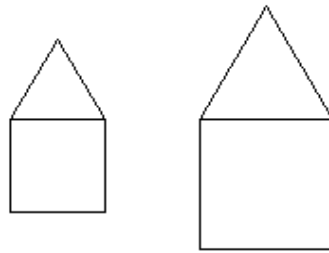
```
triangle == 25 20 moverel 25 -20 moverel -50 0 moverel  
init go triangle draw
```

Example 2 – Triangle by controlling Turtle

```
t-leg == 50 move 120 rad turn  
t2 == 3 [t-leg] times
```

Example 3 – Home & Church – an example program

```
go == penup 100 -100 moveto pendown 0 turnto  
go2 == penup 200 -100 moveto pendown 0 turnto  
  
t-leg == 50 move 120 rad turn  
triangle == 3 [t-leg] times  
square-leg == 50 move 90 rad turn  
square == 4 [square-leg] times  
home == triangle -90 rad turn square  
  
xtriangle == 3 [70 move 120 rad turn] times  
xsquare == 4 [70 move 90 rad turn] times  
church == xtriangle -90 rad turn xsquare  
  
init go home go2 church draw
```



```
go == penup 100 -100 moveto pendown 0 turnto
go2 == penup 200 -100 moveto pendown 0 turnto
```

```
t-leg == 50 move 120 rad turn
triangle == 3 [t-leg] times
square-leg == 50 move 90 rad turn
square == 4 [square-leg] times
home == triangle -90 rad turn square
```

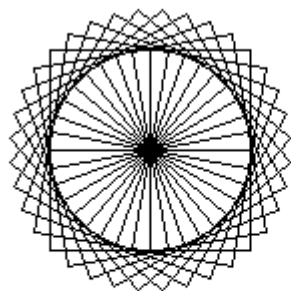
```
xtriangle == 3 [70 move 120 rad turn] times
xsquare == 4 [70 move 90 rad turn] times
church == xtriangle -90 rad turn xsquare
```

```
init go home go2 church draw
```

Example 4 – Mathematical Drawing - Squaring the Circle

This simple three line program draws a pattern that may be familiar:

```
square == 4 [50 move 90 rad turn] times
pivot == 10 rad turn
init go 36 [square pivot] times draw
```



```
square == 4 [50 move 90 rad turn] times
pivot == 10 rad turn
init go 36 [square pivot] times draw
```

Example 5 – An interactive turtle-logo in mjoy

```
red == 255
go == penup 100 -100 moveto pendown 90 rad turnto
square == 4 [50 move 90 rad turn] times
spin == -45 rad turn penup 50 move pendown

turtle == 1 pensize red pencolor 120 rad turn 12 move 210 rad turn 20
      move -120 rad turn 20 move 210 rad turn 11 move
; == dup dup ' pen dictget [turtle] [] if draw
cs == start ;

( ** Demo Program ** )

cs      (clearscreen)

go ;

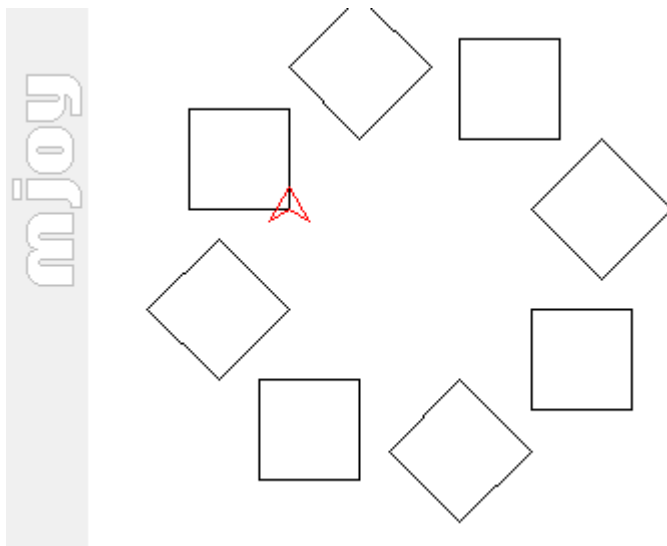
square ;

spin ;

square ;

spin ;

6 [square spin] times ;
```



Authors:

Assad Ebrahim <assad.ebrahim@mathscitech.org>

<http://www.mathscitech.org/articles/turtle-logo-forth>

+ some changes from fpstefan.de

Appendix 1. Color codes

It is recommended to put these definitions into your core.txt file.

```
black == 0
red == 255
white == 16777215
blue == 16711680
green == 32768
yellow == 65535
brown == 128
darkgray == 8421504
maroon == brown
darkgrey == darkgray
aqua == 16776960
fuchsia == 16711935
gray == 8421504
grey == gray
lime == 65280
lightgray == 12632256
navy == 8388608
olive == 32896
purple == 8388736
silver == 12632256
teal == 8421376
gold == 55295
orange == 42495
```

Reference: http://www.rapidtables.com/web/color/RGB_Color.htm