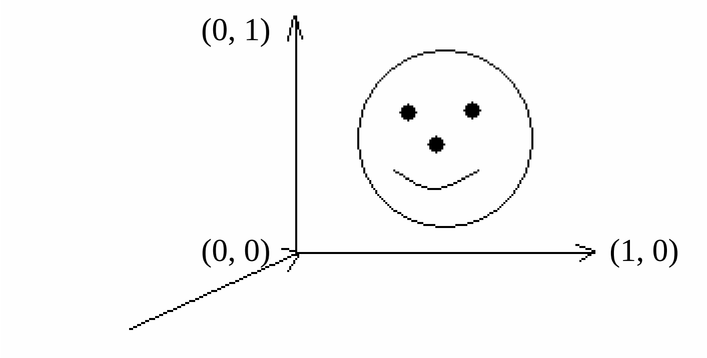
6.090, Building Programming Experience
Ben Vandiver
Lecture 9: Henderson Picture Language

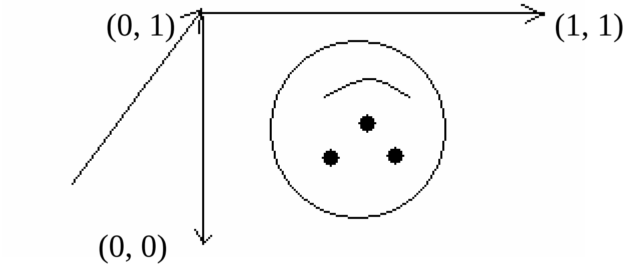
Lecture 9 involved defining frames (using vectors) and creating images that w could manipulate (by flipping over, altering coordinates, etc.). The file needed for this is called lec9.scm. It needs to be unzipped and the folders included in the file, one called “bpe” and one called “images”, should be in the directory C:\u6001. If they aren’t, place them there (copy/paste), or many of the load commands evaluated in scheme won’t work. The two files we will be working with are called hutils.scm and ps4go.scm. Procedures in hutils.scm must be completed before ps4go.scm can be used. Both must be opened in scheme (C-x, C-f → then type in hutils.scm or ps4go.scm, depending on which file you wish to open. M-o evaluates everything at once in an open file)

We can create a frame by specifying two vectors. The center point of the picture shown below is represented by (0.5, 0.5). We can draw pictures in the frame and specify the frame.

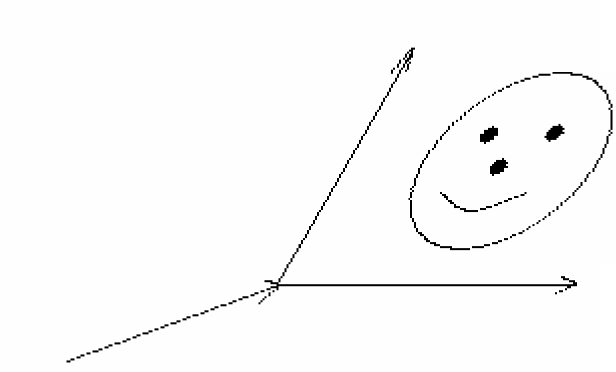


Manipulate the picture by specifying the coordinate frame. Listing (0, 0), (1, 0), and (0, 1) in that order gives the frame above. That is origin, edge 1, and edge 2.

We can flip the image



or distort it.



Picture not piece of data; it is a procedure. Give it one of the coordinate frames and it draws the image in that frame.

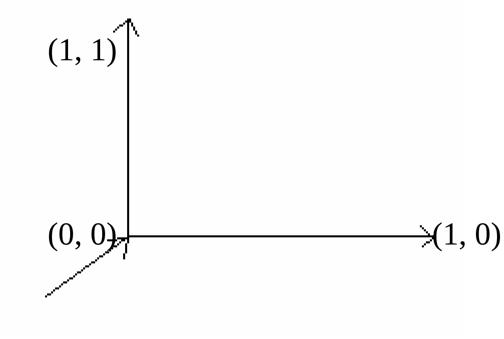
Painter is a procedure that takes in a frame and outputs an image. You can manipulate procedures.

```
(define make-vect list)
```

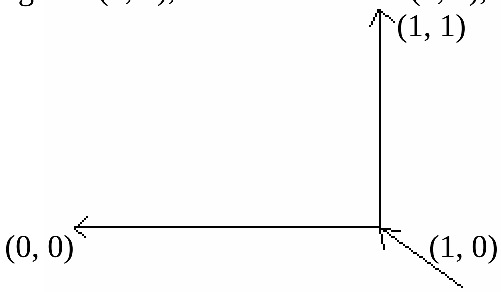
The above is equivalent to

```
(define (make-vect x y)
     (list x y))
```

Write a procedure (in lec9.scm, ps4go.scm) to flip an image horizontally.



Below has a new origin at (1, 0), a new vector 1 at (0, 0), and a new vector 2 at (1, 1)



The image above has been flipped horizontally and demonstrates the idea behind the procedure.