Laboratory: Randomized Drawing

Summary: We explore a technique for algorithmically generating interesting images that takes advantage of Scheme’s random procedure.

Contents:

- Preparation
- Exercises
  - Exercise 1: Random Art
  - Exercise 2: Splats
  - Exercise 3: New Kinds of Random Drawings
  - Exercise 4: Repeated Blobs

Preparation

a. Save a copy of random-art.scm in your directory structure (e.g., on the Desktop).

b. Start DrScheme and open your copy of random-art.scm.

c. In the GIMP, open a Script-Fu console and load random-art.scm.

Exercises

Exercise 1: Random Art

a. Read through random-art.scm to make sure that you know what all of the procedures do.

b. Create an image (say 200 wide by 100 high) and name it image.

c. Enter the following sequence of commands a few times to generate a few “random” lines.

(set-fgcolor (random-color))
(random-brush)
(random-line image 200 100)

Exercise 2: Splats

a. Read through the code to determine what splat does.

b. Use splat to generate a few additional lines.
c. As you may have noted, random-art.scm contains two different procedures for generating a random color in a limited range, random-blue and random-grey. Make sure that you understand the strategy used for each.

d. Update splat to that it only generates grey lines using a different shade of grey each time. Draw a few more lines.

e. Update splat so that it only generates blue lines using a different shade of blue each time. Draw a few more lines.

f. Update splat so that it only uses circular brushes.

g. Pick a few favorite brushes and update splat so that it selects between those brushes.

**Exercise 3: New Kinds of Random Drawings**

a. Write a procedure, `(random-ellipse img width height)`, that selects an unpredictable ellipse in img. (The parameters width and height represent the width and height of the image. You can use them or ignore them.)

b. Test your procedure with a sequence of commands like the following:

```
(define img (create-img 200 100))
(random-ellipse img 200 100)
(stroke img)
```

c. In addition to selecting a random ellipse, we might also want to select a random foreground color, a random background color, and even a random brush. Rather than retyping a sequence of commands, we might encapsulate them into a procedure, which we might call splat.

Write a procedure, `(blob img width height)` that

- chooses a random foreground color (perhaps from a certain group of colors),
- chooses a random background color (perhaps from a certain group of colors),
- chooses a random brush (perhaps from a certain group of brushes),
- selects a random ellipse (using `random-ellipse`),
- fills the ellipse with the background color, and
- strokes the ellipse.

d. Test your procedure by drawing a few blobs on the screen.

**Exercise 4: Repeated Blobs**

In the previous exercise, you wrote a procedure, `blob` and then called it a few times. In practice, most programmers don’t like to type the name of a procedure again and again and again. What’s the solution? A new procedure that repeatedly calls `blob`. 
Write and test a procedure, (blobs img width height times) that draws a blob on the image times times.