MATH 118 : COMPUTER PROJECT 3
DRAWING A FRACTAL

MATTHEW LEINGANG, CA

This is an exercise of how iterated function systems produce fractals as their attractors. Define four linear transformations of $\mathbb{R}^2$ as follows:

$$T_1 = \text{scale by } 1/3$$
$$T_2 = \begin{cases} \text{scale by } 1/3, \\ \text{rotate counterclockwise } 60^o, \\ \text{translate by } (1/3, 0). \end{cases}$$
$$T_3 = \begin{cases} \text{scale by } 1/3, \\ \text{rotate clockwise } 60^o, \\ \text{translate by } (1/2, \sqrt{3}/2). \end{cases}$$
$$T_4 = \begin{cases} \text{scale by } 1/3, \\ \text{translate by } (2/3, 0). \end{cases}$$

Exercise 1. Code these four linear transformations in *Mathematica*.

Note that a line is graphics primitive

```
Line[{{x1,y1},{x2,y2}, ..., {xn, yn}}]
```

that doesn't actually get drawn until you use the command `Show[Graphics[...]]` on it.

Exercise 2. (a) Extend your definitions of the T_i's so that they can take `Lines` as their arguments and output the Line connecting T_i of all the points. (You might find the functions `Part` and `Apply` useful).

(b) Extend further to a list, so that T_i of a list of lines gives the list of T_i of all the lines.

Now we can define

```
Kochize[x_] := {T1[x], T2[x], T3[x], T4[x]}
```

Define `myline = Line[{{0,0},{0,1}}]`.

Exercise 3. Output and use `Show[...]` to draw the iterations `Nest[Kochize, myline, n]` for $1 \leq n \leq 6$.

These are approximations to the *Koch curve*.

Exercise 4. Play with this technique to draw a Koch curve with "points" of different angles.

Date: April 29, 1999.