Exploring Fractals with Olafractal[1]

By Judith N Cederberg

The Olafractal Program:

Olafractal was developed by St. Olaf CS students. The program is modelled after a Macintosh program "Fractal Attraction" developed by Kevin Lee who gave St. Olaf permission to develop a web-based version of his program. This user-friendly program allows entry of transformation sets known as Iterated Function Systems (IFS) either by direct entry of numerical values in matrices or by cursor controlled changes on images of an original unit square.

Accessing the program:

Go to http://www.cs.stolaf.edu/projects/olafractal/ Open the link and scroll to the bottom of the page that appears. Right click on the 2.0 alpha 4 Standalone version and select *Save Link Target As ....* and save on the desktop.

When you open Olafractal, you will see the **Form** window that displays transformations, as well as a **Design** window and a **Fractal** window. Check out the *Help* menu to learn about the background of Olafractal, the current state of its development, and help contents.

Activities:

Getting Acquainted with Olafractal

When the *Form* window first opens it will contain three transformations each represented by a 2×2 matrix of a transformation that keeps the point $O(0, 0)$ invariant plus a translation vector. The *Design* window will contain 3 squares that result from applying the corresponding transformation to the unit square. The three default transformations all shrink the sidelength of the square by a factor of 2. One keeps the lower left-hand vertex of the original square fixed at O while the others translate this same vertex to 2 other points within the unit square.

1. In order to see the relevant windows, you may want to size the *Form* window so it fills the left-side of the screen while placing the *Fractal* and *Design* windows on top of each other to fill the right-side.

2. To see how the transformations work as entered, first place the *Fractal* window on top and then press the *Deterministic* button at the bottom of the *Form* window. (A "Warning" may appear noting a restriction in the number of iterations. If so, simply click on "O.K.") What appears to happen? What do the colors have to do with the transformations for this IFS? Which fractal has been generated?

3. Clear the current display in the *Fractal* window and now press *Random*. What happens this time? In what way does the generation of the fractal differ from the previous "deterministic" generation?

4. In the *Form* window, try changing the probabilities associated with each transformation making sure the sum of the probabilities equals 1. See how your new probability assignments change the fractal generated.

5. With the *Design* window displayed, select options from the *Resize* menu to reshape one or more of the squares while watching the matrix entries in the *Form* window. Display the *Fractal* window and generate the fractal resulting from your new IFS. How does its shape reflect your changes in the squares?

6. In the *Form* window, add additional transformations, again assigning probabilities so their sum equals 1, and generate the corresponding fractal.

[1]Designed to supplement Chapter 5 in *A Course in Modern Geometries, 2nd Ed.*; revised by JNC Feb. 5, 2007. Source File: My Documents\Text.dir\Web-exp.dir\olafractal07.tex

Tiling with Olafractal

In order to explore fractal dimension as well as experiment with entering matrices in Olafractal, the following activites ask you to tile well-known Euclidean point sets. Occasionally, the program may "hang-up." If this happens, select "Save" under the *File* menu to save your current transformations. Then close and reopen the program.

1. Remove all but 1 of the transformations. Change the remaining transformation so that it consists of a 2×2 dilation matrix with center $O(0, 0)$ and ratio $r = 1/3$ plus a zero translation vector. In the *Edit* menu, select *Clone* to produce a second transformation. Change this new transformation so that it will result in the same dilation plus a translation mapping the origin $O(0, 0)$ to $(2/3, 0)$.
2. Generate the fractal that results from your two transformations. Which well-known set appears?
3. Enter transformations that would make a complete tiling of the unit square with a 2-scale tiling. Note you'll need to use four transformations to do this. Since these transformations all make use of the same 2×2 matrix and differ only in their translation vectors, you may want to make use of the "Clone Transformation" feature. Then change the colors so each transformation is done in a different color.
4. Now generate a Sierpinski carpet as described in Exercise 26 on page 358.

Reactions and/or Suggestions:

If you have found any bugs in Olafractal or would like to offer reactions and suggestions for either the Olafractal program or this Exploration document, please e-mail them to cederj.