Lab 8: Recursion and Fractals

In this lab you'll get practice creating fractals with recursion. You will create a class that has will draw (at least) two types of fractals. Once completed, submit your .java file via Moodle. To make grading easier, please set up your class so that both fractals are drawn automatically when the constructor is executed.

Create a Sierpinski triangle

Step 1:
In your class’s constructor, ask the user how large a canvas s/he wants.

Step 2:
Write a method drawTriangle that draws a triangle on the screen. This method will take the x,y coordinates of three points as well as the color of the triangle. For now, start with

Step 3:
In a method createSierpinski, determine the largest triangle that can fit on the canvas (given the canvas's dimensions supplied by the user).

Step 4:
Create a method findMiddlePoints. This is the recursive method. It will take three sets of x,y coordinates for the outer triangle. (The first time the method is called, it will be called with the coordinates determined by the createSierpinski method.) The base case of the method will be determined by the minimum size triangle that can be displayed. The recursive case will be to calculate the three midpoints, defined by the three inputs. Then, by using the six coordinates (3 passed in and 3 calculated), the method will recur on the three interior triangles. Once these recursive calls have finished,
use drawTriangle to draw the triangle defined by the three original inputs to the method.

**Step 5:**
Instead of using black, make use of colors to make the triangle more interesting (e.g., the image above). Try adding a call to `sleep` to generate an animation rather than a still image. How do things change if you call `drawTriangle` in the recursive method before you recur?

More details, if you’re interested, can be found elsewhere (e.g., [http://en.wikipedia.org/wiki/Sierpinski_triangle](http://en.wikipedia.org/wiki/Sierpinski_triangle)), but we’ll just cover the basics.

**Create a second fractal**

**Step 6:**
Pick one of the three fractals below and create method(s) to draw it. My guess is that the third is the hardest, but I think it looks the coolest. As before, make it work for different sized canvases and use color. Remember to think about the base case and the recursive case before trying to write the recursive method.