CptS 111 Project 2: 3D Fractals

Due

Look for turn-in instructions on the class website

Fractals are generated figures which can be highly complex, but are created using algorithms which are somewhat simplistic. For this project, first write a program that uses vpython to generate a Pyramid fractal. Then, pick a fractal of your choosing (or make up an algorithm which generates a 3d figure in a similar manner), and write a program to display that fractal (also using vpython).

Part 1: The Pyramid Fractal

Before beginning part 1, I recommend reviewing chapters 12 and 13 of the book. Chapter 12 will be discussed in some detail on Monday, December 2, but there is no need to wait for this lecture to begin the project.

The pyramid fractal is produced by subdividing a pyramid into 5 smaller pyramids, and then subdividing each smaller pyramid, continuing until a depth limit is reached. This is a recursive fractal. That is, a fractal generated using an algorithm which includes itself as a potential step.

In general, your program will look something like this:

```
Set up initial pyramid

def subdivide(big_pyramid, subdivisions_remaining)
    if subdivisions_remaining < 1:
        draw big_pyramid
    Create 5 small pyramids based on big_pyramid
    for each small pyramid:
        subdivide(the small pyramid, subdivisions_remaining - 1)

subdivide(initial pyramid, number of subdivisions)
```

Obviously the above is not correct Python code, and some of the steps require some elaboration. A pyramid consists of 5 points: A top, and four lower corners. To draw a pyramid, draw lines along each edge. Look at the `curve` object in vpython for line drawing. To find a point midway between two other points (for example, between the top point and one of the lower corners) average each of the three values that describes the point. You may find it convenient to create a point class. Alternatively, there is a vector class provided with vpython which supports vector operations. It is also possible to represent points as a simple list of three items - x, y, and z coordinate. Any of these three methods will work, and it is up to you to choose between them.

The number of subdivisions will be limited by your computer's capacity, but 3 or 4 should be within the capacity of most modern laptops. High-end gaming computers may handle many more than this.

Part 2: Choose Another Fractal

Wikipedia has a long list of fractals here:
http://en.wikipedia.org/wiki/List_of_fractals_by_Hausdorff_dimension
Choose one of these, and write a program to display it using the vpython. If you pick a 2d fractal, it can be displayed using x and y values only (leave z set to 0 for all points in the fractal). Choices are not limited to the list on Wikipedia. You are welcome to find one elsewhere, or make one up.