Lecture 3: Some Strange Properties of Fractal Curves

I have been a stranger in a strange land. Exodus 2:22

1. Fractal Strangeness

Fractals have a look and feel that is very different from ordinary curves. Unlike commonplace curves such as lines or circles, there are no simple formulas for representing fractals like the Sierpinski gasket or the Koch curve. Therefore it should not be surprising that fractal curves also have geometric features that are unlike the properties of any other curves you have previously encountered. In this lecture we are going to explore some of the strangest peculiarities of fractals, including their dimension, differentiability, and attraction.

2. Dimension

You may have been wondering about the origin of the term *fractal*. Many fractal curves have a non-integral dimension, a fractional dimension somewhere between one and two. Fractal refers to this fractional dimension.

Standard curves like the line and the circle are 1-dimensional. To say that a curve is 1-dimensional means that the curve has no thickness; if the curve is black and the background is white, then when we look at the curve we see white on either side of a thin black curve. But fractals are different. Look at the Sierpinski gasket or the C-curve. There seem to be regions that are neither black nor white, but instead are gray. Such curves typically have dimension greater than one, but less than two; these curves do not completely fill up any region of the plane, so they are not 2-dimensional, but neither are these gray curves as thin as 1-dimensional curves. To calculate the actual dimensions of fractal curves, we first need to formalize the notion of dimension for some standard geometric shapes.

The dimension of a line segment is one, the dimension of a square is two, and the dimension of a cube is three. There is a formal way to capture these dimensions. Suppose we split these objects by inserting new vertices at the centroids of their edges and faces. Then the line segment splits into 2 line segments, the square into $4 = 2^2$ squares, and the cube into $8 = 2^3$ cubes (see Figure 1). In each case the dimension appears in the exponent. There is nothing magical about splitting each edge into two equal parts. If we split each edge into N equal parts, then the line segment splits into N line segments, the square into N^2 squares, and the cube into N^3 cubes. Once again, the dimension appears in the exponent. Another name for an exponent is a logarithm, so we are going to formalize the notion of dimension in terms of logarithms.

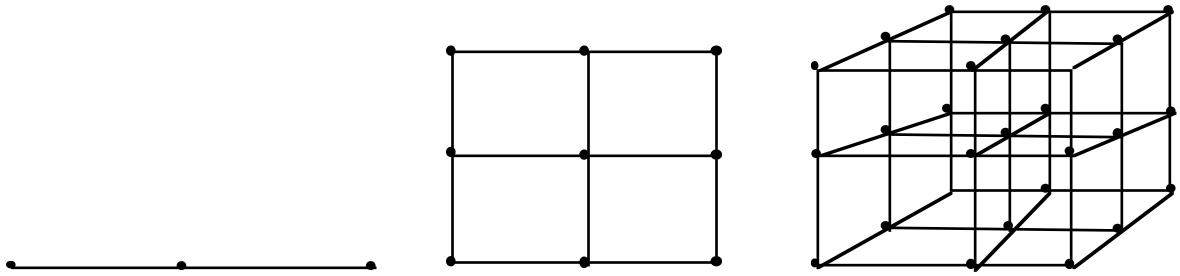

Figure 1: Inserting vertices at the centroids splits a line segment into 2 line segments, a square into $4 = 2^2$ squares, and a cube into $8 = 2^3$ cubes.

Another way of thinking about what we have just done is that we have split the line, the square, and the cube into identical parts, where each part is a scaled down version of the original. This decomposition should remind you of the fractals that you encountered in Lecture 2, where each fractal is composed of several identical scaled down copies of the original fractal. Evidently, in this construction for the line, the square, and the cube, if D denotes dimension, then

$$D = Log_N(E) = \frac{Log(E)}{Log(N)}, \tag{1}$$

where N is the number of line segments along each edge and E is the number of equal scaled down parts. But if N is the number of line segments along each edge, then $S = 1/N$ is the scaling along each edge. Since $Log(S) = -Log(N)$, we can rewrite Equation (1) by setting

$$D = -\frac{Log(E)}{Log(S)}, \tag{2}$$

where S is the scale factor and E is the number of identical scaled down parts.

Equation (2) has several important properties. First, notice that $S < 1$, so $Log(S) < 0$. Therefore, the minus sign on the right hand side of Equation (2) insures that the dimension D is positive. Second, since D is defined as the ratio of two logarithms, the base of the logarithm does not matter; dimension is the same in all bases. Finally, Equation (2) gives the same result as Equation (1) for the line, the square, and the cube, since in these cases $E = N^D$ and $S = 1/N$, so

$$-\frac{Log(E)}{Log(S)} = -\frac{Log(N^D)}{Log(1/N)} = \frac{D\,Log(N)}{Log(N)} = D.$$

Let's see now what happens when we apply Equation (2) to fractal curves.

2.1 Fractal Dimension. To apply our dimension formula, we need to consider self-similar curves. Recall that a curve is *self-similar* if it can be decomposed into a collection of identical curves each of which is a scaled version of the original curve. Most of the fractal curves we encountered in Lecture 2 such as the Sierpinski gasket and the Koch curve are self-similar curves. In fact, self-similarity is what allows us to write simple recursive turtle programs to generate these curves. Let's look now at some examples.

Example 1: *Sierpinski Gasket*

The Sierpinski gasket consists of three smaller Sierpinski gaskets, where the length of each edge of the smaller gaskets is one-half the length of an edge of the original gasket (see Figure 2, left). Thus $E = 3$ and $S = 1/2$, so

$$D = -\frac{Log(E)}{Log(S)} = -\frac{Log(3)}{Log(1/2)} = \frac{Log(3)}{Log(2)} \approx 1.585 \Rightarrow 1 < D < 2\,.$$

Example 2: *Koch Curve*

The Koch curve consists of four smaller Koch curves, each one-third the size of the original curve (see Figure 2, right). Thus $E = 4$ and $S = 1/3$, so

$$D = -\frac{Log(E)}{Log(S)} = -\frac{Log(4)}{Log(1/3)} = \frac{Log(4)}{Log(3)} \approx 1.262 \Rightarrow 1 < D < 2\,.$$

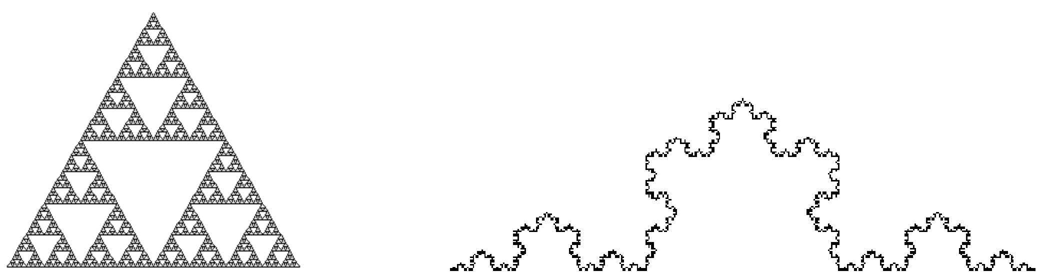

Figure 2: The Sierpinski gasket (left) and the Koch curve (right). The Sierpinski gasket is composed of three self-similar parts; the length of each edge of one of the smaller gaskets is one-half the length of an edge of the original gasket. The Koch curve is composed of four self-similar parts, each one-third the size of the original Koch curve.

2.2 Computing Fractal Dimension from Recursive Turtle Programs. The fractal dimension of a self similar fractal curve can often be computed directly from its recursive turtle program: the number of recursive calls corresponds to the number E of equal self similar parts, and the scale factor in the RESIZE command corresponds to the scale factor S. Thus

$$\textit{Fractal Dimension} = -\frac{Log(\#\,RecursiveCalls)}{Log(ScaleFactor)}\,. \tag{3}$$

This formula, like Equation (2), is valid provided that the self-similar parts do not overlap. To illustrate the validity of this formula, let's revisit the fractal dimension of the Sierpinski gasket and the Koch curve.

Example 3: *Sierpinski Gasket*

Recall from Lecture 2 that in the recursive turtle program for the Sierpinski gasket $\#\,RecursiveCalls = 3$ and $ScaleFactor = 1/2$, so

$$Fractal\ Dimension = -\frac{Log(\#\,RecursiveCalls)}{Log(ScaleFactor)} = -\frac{Log(3)}{Log(1/2)} = \frac{Log(3)}{Log(2)}.$$

Example 4: *Koch Curve*
Again recall from Lecture 2 that in the recursive turtle program for the Koch curve $\#\,RecursiveCalls = 4$ and $ScaleFactor = 1/3$, so

$$Fractal\ Dimension = -\frac{Log(\#\,RecursiveCalls)}{Log(ScaleFactor)} = -\frac{Log(4)}{Log(1/3)} = \frac{Log(4)}{Log(3)}.$$

3. Differentiability

A curves is said to be *smooth* or *differentiable* if the curve has a well defined slope or equivalently a well defined tangent at every point. Lines and circles are smooth curves. Polygons and stars are piecewise smooth curves; polygons and stars have well defined slopes everywhere except at their vertices. The functions you studied in calculus -- polynomials, trigonometric functions, exponentials, and logarithms -- are all differentiable functions. What about fractals?

Differentiable functions are everywhere continuous, but continuous functions need not be everywhere differentiable. The function $y = |x|$ represents a continuous curve composed of two lines: the line $y = -x$ for $x \le 0$ and the line $y = x$ for $x \ge 0$. Thus the curve $y = |x|$ has slope -1 for $x < 0$ and slope +1 for $x > 0$, but the slope of $y = |x|$ is not well defined at the origin. A curve is said to be *piecewise linear* if like the function $y = |x|$ it is composed of a sequence of straight lines. A piecewise linear curve is smooth everywhere except where two lines join. Thus it is easy to generate curves that are smooth everywhere except at a finite number of points. The level *n* Koch curve is a piecewise linear curve that is smooth everywhere except at finitely many points.

Intuitively it seems evident that a continuous curve can have only a finite number of points where the slope is not well defined. But intuition can be misleading. *Many fractal curves are continuous everywhere, but differentiable nowhere*.

Consider the Koch curve (see Figure 3). Level zero is a straight line, which is smooth at every point. Level one, however, is a piecewise linear curve, and there are three points where the slope is not defined. Similarly, at level two there are 15 points where the slope is not defined, At level *n* there are $4^n - 1$ points where the slope is not defined. Thus in the limit as *n* approaches infinity, there are infinitely many points on the Koch curve where the slope is not defined.

In fact, in the limit, the slope of the Koch curve is undefined at every point. We can substantiate this assertion in the following manner. Suppose that the length of the line in level zero is 1. Then the length of the line segments in level one is $1/3$; the length of the line segments in level two is $1/9$; and, in general, the length of the line segments at level n is 3^{-n}. Thus the lengths of these line segments approach zero as the number of levels approaches infinity. Now the curve is smooth only at points on the interior of these line segments. But in the limit as the number of levels approaches infinity, there are no points in the interior of line segments because the lengths of the line segments approach zero. Thus in the limit the Koch curve is continuous everywhere, but differentiable nowhere. That is, the Koch curve has a wrinkle at every point!

The Koch curve is not an anomaly. In fact, much the same arguments can be used to show that virtually any bump fractal is continuous everywhere and differentiable nowhere. Fractals are indeed very strange curves.

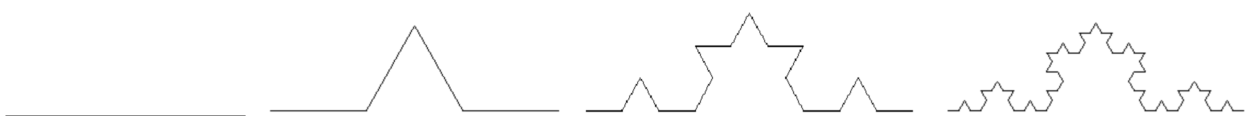

Figure 3: Levels 0, 1, 2, 3 of the Koch curve. Level zero is a straight line, which is smooth at every point. Level one has three points where the slope is not defined; level two has 15 points where the slope is not defined, and level 3 has 63 points where the slope is not defined. Notice too that the lengths of the line segments decrease from level to level by a factor of 3.

4. Attraction

How precisely does the fractal generated by a recursive turtle program depend on the choice of the base case? Intuitively you might think that changing the base case will alter the fractal in some fundamental way. But as we have just seen when we investigated differentiability, our intuition regarding fractals can often be misleading.

We shall begin then by considering two familiar examples: the Sierpinski gasket and the Koch curve. Based on our experience with these two fractals, we shall draw some highly counterintuitive and rather remarkable conclusions.

4.1 Base Cases for the Sierpinski Gasket. In the base case, the standard program for the Sierpinski gasket draws a triangle (see Lecture 2). The curves generated by levels 0,1,3,6 of this program are illustrated in Figure 4. In Figures 5-7, we have changed the base case to draw a square, a star, and a horizontal line, but we have not altered the recursive body of the program. Each of these programs generates curves that start out quite different at levels 0,1, but by level 6 they all seem to be converging to the same Sierpinski gasket!

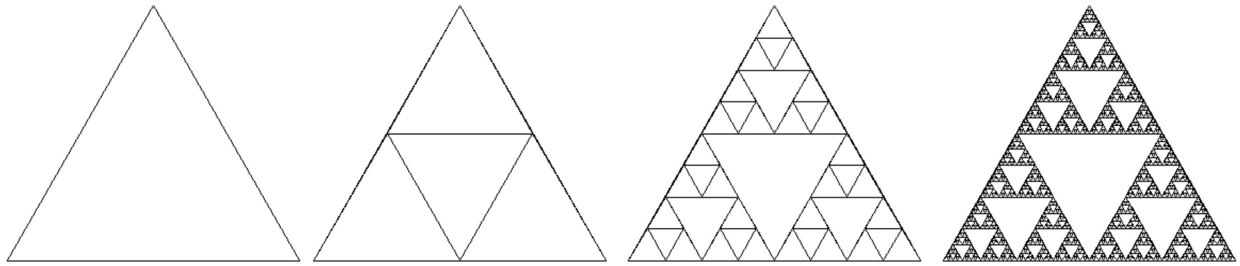

Figure 4: Levels 0,1, 3, 6 of the Sierpinski gasket. The base case is a triangle.

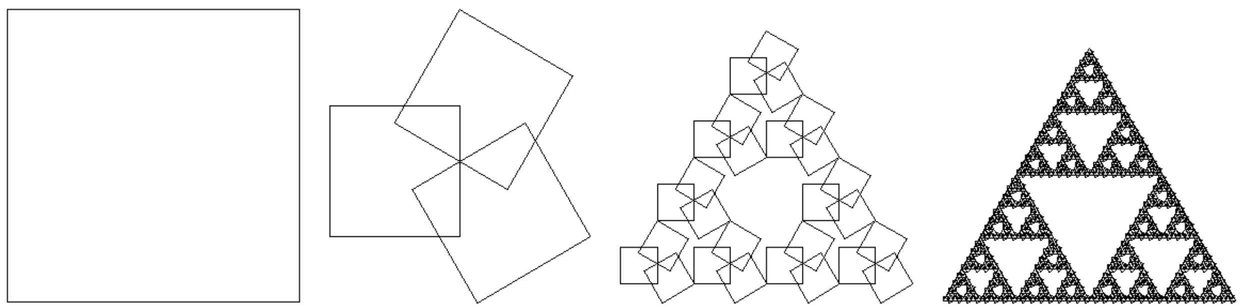

Figure 5: Levels 0, 1, 3, 6 of the Sierpinski gasket. The base case is a square.

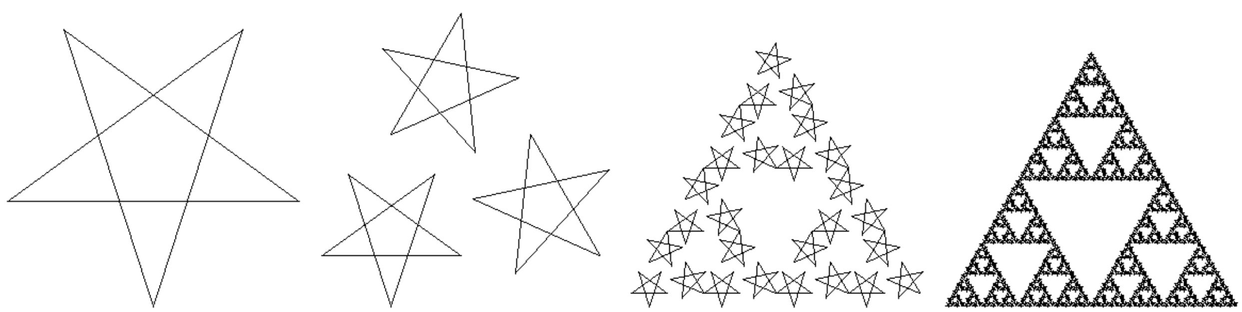

Figure 6: Levels 0, 1, 3, 6 of the Sierpinski gasket. The base case is a star.

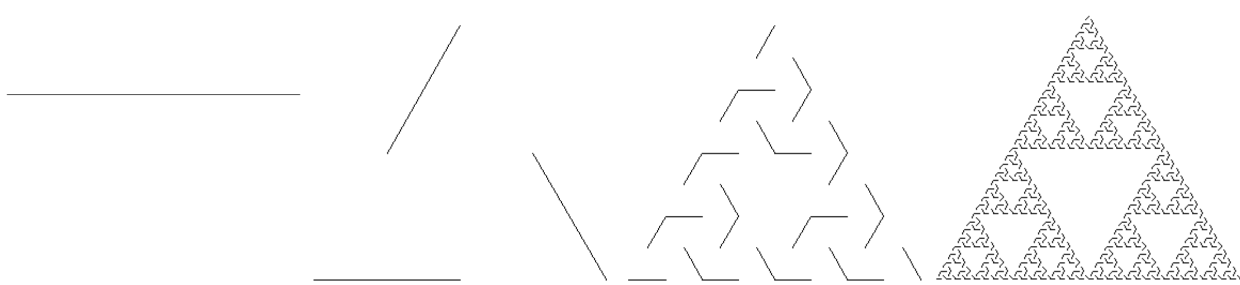

Figure 7: Levels 0, 1, 3, 6 of the Sierpinski gasket. The base case is a horizontal line.

4.2 Base Cases for the Koch Curve. Let's try another example: the Koch curve. In the base case, the standard program draws a straight line. The curves generated by levels 0,1,2,5 of this program are illustrated in Figure 8. In Figures 9-11, we have changed the base case to draw a square bump, a square, and a square where the turtle repeats the first side after completing its path around the square, but we have not altered the recursive body of the program. Again each of the curves generated by these programs starts out quite different at levels 0,1, but, except for the program where the base case is a square (see Figure 10 -- we will return to a discussion of this anomalous case shortly below), by level 5 they all seem to be converging to the same Koch curve!

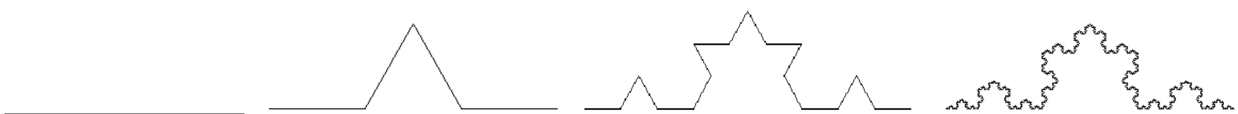

Figure 8: Levels 0, 1, 2, 5 of the Koch curve. The base case is a horizontal line.

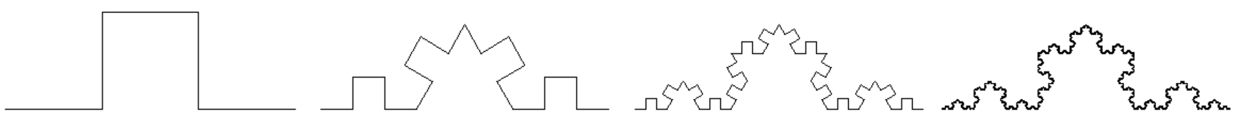

Figure 9: Levels 0, 1, 2, 5 of the Koch curve. The base case is a square bump.

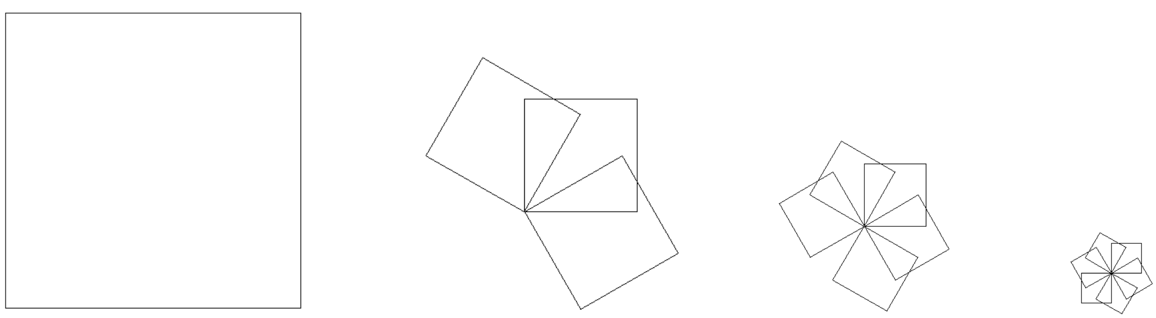

Figure 10: Levels 0, 1, 2, 5 of the Koch program. The base case is a square.

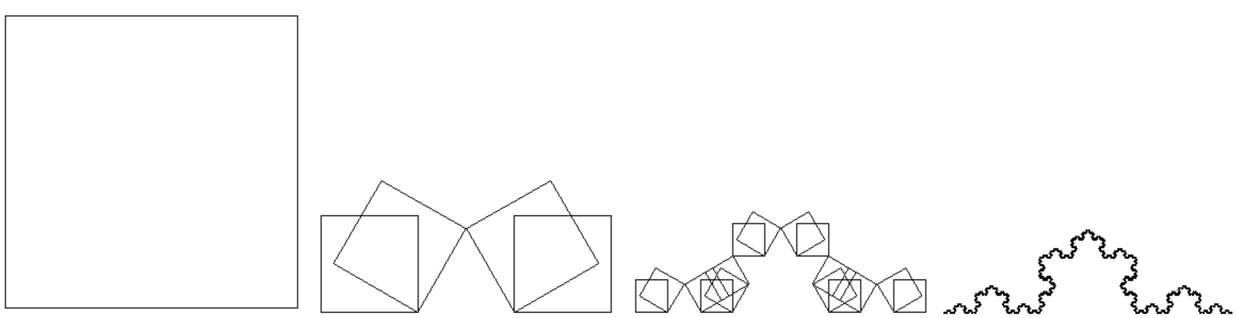

Figure 11: Levels 0, 1, 2, 5 of the Koch curve. The base case is a square, where the turtle repeats the first side after completing its path around the square.

4.3 Attractors. When we first encountered recursive turtle programs, our intuition lead us to believe that the precise form of the base case is critical for generating the desired fractal curve. The natural base case for the Sierpinski gasket is a triangle; the natural base case for the Koch curve either a straight line or a triangular bump. But the examples we have just witnessed both for the Sierpinski gasket and for the Koch curve indicate that in the limit, (and as a practical matter after only about five or six levels of recursion) the base case is all but irrelevant. Apparently, a recursive turtle program will always converge to the same fractal curve independent of the turtle program in the base case. When a process converges to the same limit value independent of the choice of the initial value, the limit value is called an *attractor*. Fractals are attractors.

Figure 10 seems to challenge this conclusion; here the expected convergence to the Koch curve fails to occur. However, the reason for this failure turns out to be that the recursion in the body of the turtle program for a bump fractal assumes that the turtle advances to the end of the line on which the bump occurs, rather than return to her initial position. This assumption is violated by the base case in Figure 10, but is satisfied by the base case in Figure 11, where the turtle repeats the first side of the square after completing her path around the square.

In future lectures we shall prove that in the limit as the number of levels approaches infinity the fractals generated by recursive turtle programs are indeed independent of the turtle program in the base case, provided that the final state of the turtle in the base case is the same as the final state of the turtle in the recursion. Usually, as in the Sierpinski gasket, the initial state and the final state of the turtle are identical, but, as illustrated by the Koch curve, one must always be careful to check this assumption.

Thus, with some mild assumptions, the fractal generated in the limit by any recursive turtle program is indeed independent of the turtle program in base case. Fractals are attractors. Indeed, attraction is the signature property by which we recognize fractals.

But why does such a strange property hold? To understand the reason behind this curious phenomenon, we shall have to forgo for now our study of Turtle Graphics and investigate instead a very different approach to fractals called *iterated functions systems*. To understand iterated functions systems, we shall first need to take up the study of affine transformations and affine graphics. We shall commence with these topics in our next lecture.

5. Summary

From the turtle's point of view, fractals are simply *recursion made visible*. But from our perspective, fractals generated from recursive turtle programs can have many strange properties.

1. Fractals are self similar curves; fractals can be decomposed into a collection of identical

curves each of which is a scaled version of the original fractal curve.

2. Fractals can be continuous everywhere, yet differentiable nowhere.
3. Fractals can have fractional dimensions, non-integer dimensions between one and two.
4. Fractals are attractors, independent of the base case in the corresponding recursive turtle program.

These four properties are characteristics of fractal curves that we do not find in common everyday curves; together these features are what set fractals apart from our ordinary experience of geometry.

Exercises:

1. Let P_n denote the perimeter of the *nth* level of the Koch snowflake.
   a. Show that
      i. $P_n = \frac{4}{3} P_{n-1}$
      ii. $P_n = \left(\frac{4}{3}\right)^n P_0$
   b. Conclude that the perimeter of the Koch snowflake is infinite.

2. Let A_n denote the area enclosed by the *nth* level of the Koch snowflake.
   a. Show that
      i. $A_n = A_0 + 3A_0(1/9 + 4/9^2 + 4^2/9^3 + \cdots 4^{n-1}/9^n)$
      ii. $Lim_{n \to \infty} A_n = (8/5)A_0$
   b. Conclude from Exercise 1 and part *a* that the Koch snowflake has an infinite perimeter but encloses a finite area.

3. Show that the bump fractals corresponding to the bump curves in Lecture 2, Figure 13 are continuous everywhere but differentiable nowhere.

4. Compute the fractal dimension of the *C*-curve.

5. Compute the fractal dimension of the gaskets in Lecture 2, Figure 8.

6. Compute the fractal dimension for each of the curves in Lecture 2, Figure 11.

7. Compute the fractal dimension for each of the bump fractals corresponding to the bump curves in Lecture 2, Figure 13.

8. Consider the following recursive turtle program:

*SQUARE (*Level)

IF Level = 0, POLY (1, $\pi/2$)

OTHERWISE

REPEAT 4 TIMES

RESIZE 1/2

*SQUARE (*Level - 1)

RESIZE 2

MOVE 1

TURN $\pi/2$

a. Show that as the level approaches infinity, this program generates a filled in square.

b. Compute the dimension of the square from this recursive turtle program.

9. Generate a fractal pentagonal gasket using the same recursive turtle program with three different turtle programs in the base case.

10. Generate the fractal *C*-curve using the same recursive turtle program with three different turtle programs in the base case.

11. Generate the fractal tree in Lecture 2, Figure 11, using the same recursive turtle program with three different turtle programs in the base case.