

# A Window on the Beauty of Fractal Images: TI-92<sup>1</sup>

Kwon, Oh Nam

Department of Mathematics Education, Ewha Womans University,  
11-1 Daehyeon-dong, Seodaemun-gu, Seoul 120-750, Korea;

Email: onkwon@mm.ewha.ac.kr

(Received October 12, 2000)

Generating fractal images by graphing calculators such as TI-92 combines several important features, which convey the excitement of a living, changing mathematics appropriate to secondary or post-secondary students. The topic of fractal geometry can be illustrated using natural objects such as snowflakes, leaves and ferns. These complex and natural forms are often strikingly fantastic and beautiful. The examples highlight the fact that complex, natural behaviors can result from simple mathematical rules such as those embodied in *iterated function systems* (IFS). The visual splendor beauty of fractals, in concert with their ubiquity in nature, reveals the intellectual beauty of nonlinear mathematics in a compelling way. The window is now open for students to experience and explore some of the wonder of fractal geometry.

## I. INTRODUCTION

The field of mathematics has generated a handful of peculiar sets, complex beyond imagination, which have influenced mathematical thinking. The Cantor set, the Sierpinski triangle, and the Koch curve come to mind as examples of such sets. Although such objects were created as “mathematical monsters” during flights of fancy, no one imagined that they would be useful models for natural phenomena.

Mandelbrot (1982), noticing that these sets shared some common properties, called them fractals and produced computer drawings of various fractals, which drew much attention from mathematical research, even outside mathematics. Since generating fractals usually requires a process that is applied an infinite number of times, it is only with the advent of modern computer graphics that it has been possible to carry out the experiments necessary to explore them effectively.

---

<sup>1</sup> This research was supported by 1998 Ewha Womans University Research Fund.

Fractals are often beautiful geometric shapes that can be generated by simple rules. The study of fractals has great potential in the mathematics curriculum. Fractals provide instantly visible, almost concrete illustrations of many mathematical concepts including infinity, limits, creating and following algorithms, mappings, and matrix algebra. Course units that can introduce students to the fundamental ideas underlying fractals now exist.

In order to expose students to the intellectual beauty of the mathematics behind the fractals, a graphing calculator and computer graphing technology in mathematics classes can serve as a pedagogical tool to reveal the beauty of fractal images.

The purpose of this paper is to describe a number of fractals that can be easily generated on graphing calculators (TI-92) using simple programs.

## II. ITERATED FUNCTION SYSTEMS: A TOOL FOR CODING FRACTAL IMAGES

We turn to one of the most exciting and profound developments in the construction of fractal sets, the use of iterated function systems (IFS). The mathematics was developed by John Hutchinson, while the method was popularized by Michael Barnsley and others (Crowner, 1995). The iterated function system approach provides a good theoretical framework from which to pursue the mathematics of many classical fractals as well as more general types.

Before introducing the theory of IFS, let's look at a particular example. There are two ways to proceed, namely the deterministic algorithm and the random iteration algorithm. An example is the Sierpinski triangle, which is constructed by iteratively removing open middle triangular regions, starting with a single closed triangular region (see Figure 1).

The following three affine transformations are used in the construction.

$$\begin{aligned}
 T_1 \begin{bmatrix} x \\ y \end{bmatrix} &= \begin{bmatrix} \frac{1}{2} & 0 \\ 0 & \frac{1}{2} \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \\
 T_2 \begin{bmatrix} x \\ y \end{bmatrix} &= \begin{bmatrix} \frac{1}{2} & 0 \\ 0 & \frac{1}{2} \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} 3 \\ 0 \end{bmatrix} \\
 T_3 \begin{bmatrix} x \\ y \end{bmatrix} &= \begin{bmatrix} \frac{1}{2} & 0 \\ 0 & \frac{1}{2} \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} \frac{3}{2} \\ \frac{3\sqrt{3}}{2} \end{bmatrix}
 \end{aligned}$$

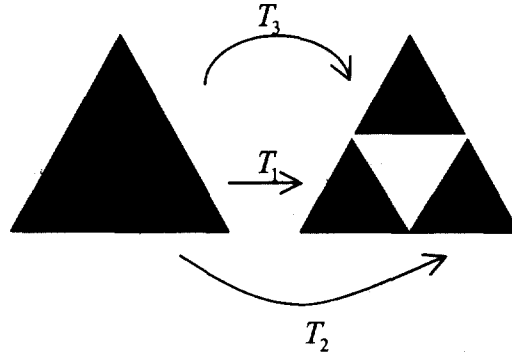


Figure 1. Sierpinski triangle generator

The deterministic algorithm is based on the idea of iteratively computing a sequence  $\{E_n\}$  of sets

$$E_n = \overbrace{T \circ T \circ \cdots \circ T}^{n \text{ times}}(E_0)$$

starting from an initial set  $E_0$ . In particular,  $E_0$  is the closed triangular region having vertices at  $(0, 0)$ ,  $(6, 0)$  and  $(3, 3\sqrt{3})$ . The images

$$T_1(E_0), T_2(E_0) \text{ and } T_3(E_0)$$

are the three smaller triangular regions. Then deterministic algorithm calls for an iterative construction of the following sets.

$$\begin{aligned} E_0 &= \text{a compact set (arbitrarily chosen)} \\ E_1 &= T_1(E_0) \cup T_2(E_0) \cup T_3(E_0) \\ &\vdots \\ E_n &= T_1(E_{n-1}) \cup T_2(E_{n-1}) \cup T_3(E_{n-1}) \end{aligned}$$

where  $n$  increases without bound.

When  $E_0$  is chosen as the closed triangular region, the sets  $\{E_n\}$  constructed in this way are precisely the sets produced by removing the open middle triangular regions. For the general iterated function system, let  $T = \{T_1, T_2, \dots, T_N\}$  denote a set of  $N$  continuous contraction maps on a compact subset  $K$  of  $\mathbb{R}^2$ . In other words  $T_i : K \rightarrow K$  and

$$|T_i(x) - T_i(y)| \leq s_i |x - y|, \quad \forall x, y \in K,$$

where  $0 \leq s_i < 1$ ,  $i = 1, 2, 3, \dots, N$ .

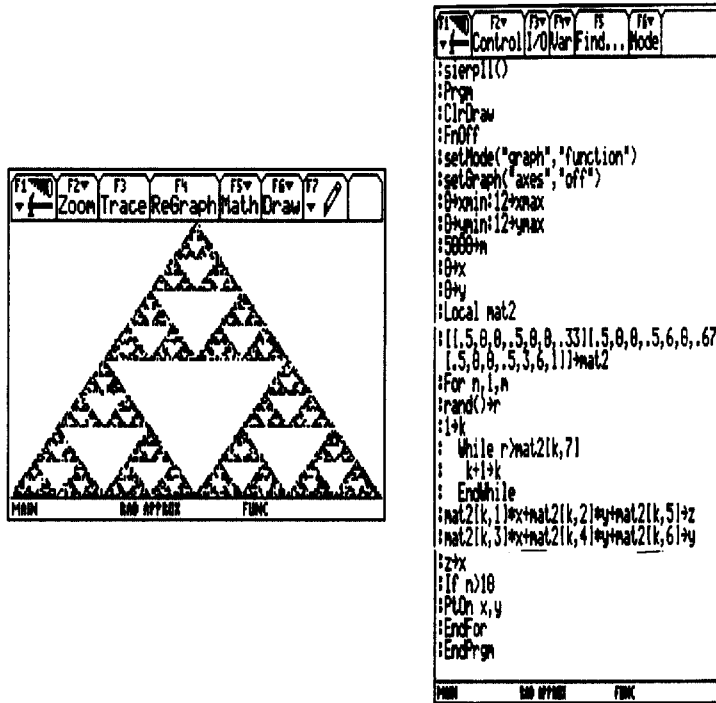


Figure 2. Sierpinski triangle and its program

Associated with these maps is a set of non-zero probabilities in the random iteration algorithm

$$P = \{p_1, p_2, p_3, \dots, p_N\}; \quad p_j > 0 \text{ for } j = 1, 2, \dots, N, \text{ and } \sum_{j=1}^N p_j = 1.$$

The system  $\{K, T, P\}$  will be referred to as a *contractive*<sup>1</sup> IFS. We refer to the data in Table 1 as an IFS code.

Table 1.

*IFS code for a Sierpinski triangle*

$T$	$a$	$b$	$c$	$d$	$e$	$f$	$p$
1	0.5	0	0	0.5	0	0	0.33
2	0.5	0	0	0.5	3	0	0.33
3	0.5	0	0	0.5	$\frac{3}{2}$	$\frac{3\sqrt{3}}{2}$	0.34

<sup>1</sup> "contractive" is synonymous with "hyperbolic" in Barnsley (1993).

The main business of IFS theory is to determine when the iterated function system produces a limit set  $E$ , that is, a set  $E$  such that

$$E = \lim_{n \rightarrow \infty} E_n$$

which converges according to the Hausdorff metric. When this happens, the set  $E$  is called the *attractor* for the iterated function system.

### III. HOW TO FIND AN IFS CODE?

Let's consider the Koch curve (see p. 7, Figure 4). From the initial segment  $\overline{AD}$ , we remove the middle one-third,  $\overline{BC}$ , and add two segments  $\overline{BE}$  and  $\overline{EC}$  having the same length (see Figure 3 below). We continue by repeating this process over and over, at each stage replacing the middle one-third by two new segments.

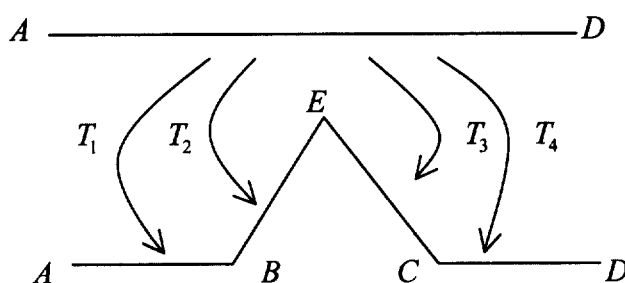


Figure 3. Koch curve (Steps 1-4)

The Koch curve can be encoded using the following four transformations, each with an associated probability of 0.25 (Bannon, 1991).

**Step 1.** The first transformation  $T_1$  involves shrinking one-third of the initial segment  $\overline{AD}$ . This procedure maps  $\overline{AD}$  to  $\overline{AB}$ , i.e.,

$$T_1 \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \frac{1}{3} & 0 \\ 0 & \frac{1}{3} \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \frac{1}{3}x \\ \frac{1}{3}y \end{bmatrix}.$$

**Step 2.** The second transformation  $T_2$  is a dilation by a factor of  $\frac{1}{3}$  followed by a rotation of  $\frac{\pi}{3}$  radians counterclockwise and a translation of  $(\frac{1}{3}, 0)$ . This procedure

maps  $\overline{AD}$  to  $\overline{BE}$ . Therefore  $T_2$  can be defined by

$$T_2 \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \cos \frac{\pi}{3} & -\sin \frac{\pi}{3} \\ \sin \frac{\pi}{3} & \cos \frac{\pi}{3} \end{bmatrix} \begin{bmatrix} \frac{1}{3} & 0 \\ 0 & \frac{1}{3} \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} \frac{1}{3} \\ 0 \end{bmatrix} = \begin{bmatrix} \frac{1}{6}x - \frac{\sqrt{3}}{6}y + \frac{1}{3} \\ \frac{\sqrt{3}}{6}x + \frac{1}{6}y \end{bmatrix}.$$

**Step 3.** The third transformation  $T_3$  is a dilation by a factor of  $\frac{1}{3}$  followed by a rotation of  $\frac{\pi}{3}$  radians clockwise and a translation of  $(\frac{1}{2}, \frac{\sqrt{3}}{6})$ . This procedure maps  $\overline{AD}$  to  $\overline{EC}$ , i.e.,

$$T_3 \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \cos(-\frac{\pi}{3}) & -\sin(-\frac{\pi}{3}) \\ \sin(-\frac{\pi}{3}) & \cos(-\frac{\pi}{3}) \end{bmatrix} \begin{bmatrix} \frac{1}{3} & 0 \\ 0 & \frac{1}{3} \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} \frac{1}{2} \\ \frac{\sqrt{3}}{6} \end{bmatrix} = \begin{bmatrix} \frac{1}{6}x - \frac{\sqrt{3}}{6}y + \frac{1}{2} \\ -\frac{\sqrt{3}}{6}x + \frac{1}{6}y + \frac{\sqrt{3}}{6} \end{bmatrix}.$$

**Step 4.** The last transformation  $T_4$  is a dilation by a factor of  $\frac{1}{3}$  followed by a translation of  $(\frac{2}{3}, 0)$ . This procedure maps  $\overline{AD}$  to  $\overline{CD}$  and defines

$$T_4 \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \frac{1}{3} & 0 \\ 0 & \frac{1}{3} \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} \frac{2}{3} \\ 0 \end{bmatrix} = \begin{bmatrix} \frac{1}{3}x + \frac{2}{3} \\ \frac{1}{3}y \end{bmatrix}.$$

Now these four transformations define the IFS code for the Koch curve as indicated in Table 2.

**Table 2.**

*IFS code for the Koch curve.*

$T$	$a$	$b$	$c$	$d$	$e$	$f$	$p$
1	$\frac{1}{3}$	0	0	$\frac{1}{3}$	0	0	0.25
2	$\frac{1}{6}$	$-\frac{\sqrt{3}}{6}$	$\frac{\sqrt{3}}{6}$	$\frac{1}{6}$	$\frac{1}{3}$	0	0.25
3	$\frac{1}{6}$	$-\frac{\sqrt{3}}{6}$	$-\frac{\sqrt{3}}{6}$	$\frac{1}{6}$	$\frac{1}{2}$	$\frac{\sqrt{3}}{6}$	0.25
4	$\frac{1}{3}$	0	0	$\frac{1}{3}$	$\frac{2}{3}$	0	0.25

Sierpinski has added another object to the gallery of classical fractals, the Sierpinski carpet (Peitgen et al., 1992).

We begin with a rectangle in the plane. Then we subdivide the object into nine little congruent rectangles from which we drop the center one, and so on. If we carry out this process indefinitely, the resulting object can be viewed as a generalization of the Cantor set. The IFS code for the Sierpinski carpet is similar to that of Cantor set, as shown in Table 3.

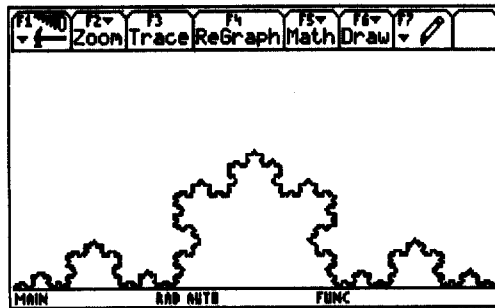


Figure 4. Koch curve

**Table 3.**

*IFS code for Sierpinski carpet.*

$T$	$a$	$b$	$c$	$d$	$e$	$f$	$p$
1	0.33	0	0	0.33	0	0	0.125
2	0.33	0	0	0.33	2	0	0.125
3	0.33	0	0	0.33	4	0	0.125
4	0.33	0	0	0.33	0	2	0.125
5	0.33	0	0	0.33	4	2	0.125
6	0.33	0	0	0.33	0	4	0.125
7	0.33	0	0	0.33	2	4	0.125
8	0.33	0	0	0.33	4	4	0.125

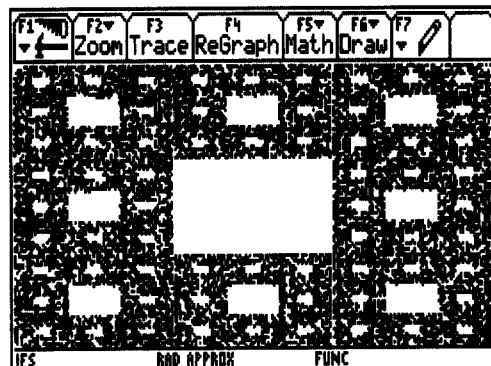


Figure 5. Sierpinski carpet

The Sierpinski carpet can be produced by using the same program as that for the Sierpinski triangle but with a modification of the IFS code. A display of the carpet on the TI-92 graphing calculator is presented in Figure 5.

#### IV. MORE NATURAL FRACTALS

Barnsley also used the IFS code to generate images of attractors that are found in nature. The attractor generated by the IFS code in Table 4, called Barnsley's fern is shown in Figure 6 (Barnsley, 1993).

**Table 4.**

*IFS code for Barnsley's fern.*

$T$	$a$	$b$	$c$	$d$	$e$	$f$	$g$
1	0.85	0.04	-0.04	0.85	0	1.6	0.85
2	0.2	-0.26	0.23	0.22	0	1.6	0.07
3	-0.15	0.28	0.26	0.24	0	0.44	0.07
4	0	0	0	0.16	0	0	0.01

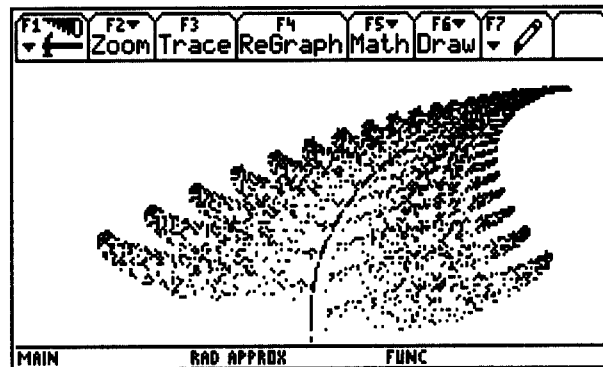


Figure 6. Barnsley's fern (window:  $-3 \leq x \leq 3$ ,  $0 \leq y \leq 11$ )

The importance of *Barnsley's fern* to the development of the subject is that his image looks like a fern, but lies in the same mathematical category of constructions as the Sierpinski triangle, the Sierpinski carpet, and the Koch curve. Another natural fractal, an oak leaf, can be generated on the TI-92 graphing calculator after



determining its IFS code (Nowark, 1995). These results suggest IFS as a tool for coding images.

**Table 5.**

*IFS code for an oak leaf.*

$T$	$a$	$b$	$c$	$d$	$e$	$f$	$p$
1	0.02	-0.07	-0.02	0.48	141	83	0.1
2	0.4	0	-0.04	0.65	88	10	0.4
3	-0.02	-0.45	-0.37	0.1	82	132	0.2
4	-0.11	-0.6	-0.34	0.22	237	125	0.3

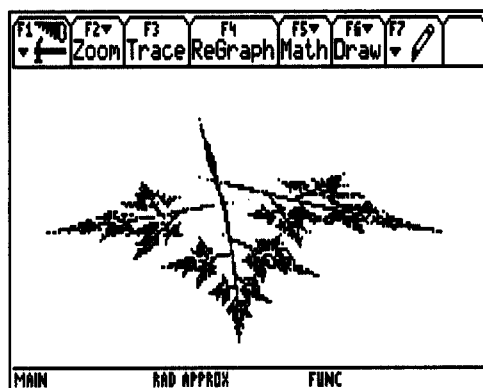


Figure 7. Oak leaf (windows:  $75 \leq x \leq 225$ ,  $0 \leq y \leq 175$ )

Next, let's produce branching fractals. Kutzler & Stoutemyer (1997) created a program called "tg .:" to implement turtle graphics on a graphing calculator. The following program generates branching fractals.

```

:node(d,θ,n)
:Prgm
:If n≤0:Return
:tg\lft(θ)
:tg\rbranch(d,θ,n-1)
:rt(θ+θ)
:tg\rbranch(d,θ,n-1)
:tg\lft(θ)
:EndPrgm

```

Here  $d$ ,  $\theta$ , and  $n$  denote the length of the branches at each step, the angular increment, and the number of steps, respectively. A readable interpretation of the procedure above is as follows.

```

: rbranch(d, θ, n)
: Prgm
: tg\forward(d)
: tg\node(d, θ, n)
: tg\back(d)
: EndPrgm

```

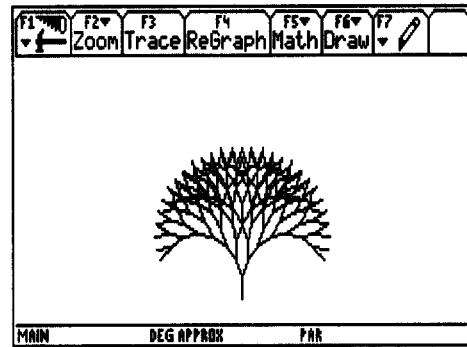


Figure 8. Branching fractal I

Repeat  $n$  times a process that moves the drawing stylus forward a distance  $d$  in the direction it is heading, and then turns the heading  $\theta$ , to the left to execute the previous step, and then turns the heading  $2\theta$  to the right to execute the previous step, and then turns the heading  $\theta$  to the left, and then moves backwards a distance  $d$ .

The result of this program is a tree which is symmetric with respect to the vertical axis (see Figure 8).

However, this tree looks somewhat unnatural. Suppose we replace the “rbranch” procedure by the following two lines:  $\text{rand}(d) \rightarrow d$ ,  $\text{rand}(\theta) \rightarrow \theta$ . Then a natural looking tree can be generated (see Figure 9).

```

: rbranch(d, θ, n)
: Prgm
: rand(d)→d
: rand(θ)→θ
: tg\forward(d)
: tg\node(d, θ, n)
: tg\back(d)
: EndPrgm

```

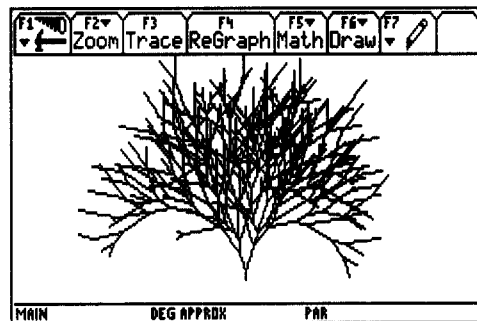


Figure 9. Branching fractal II

Now, how can we depict a tree on a windy day? A modification of two lines in the node program produces wind-swept trees. This resulting figures are amazingly natural-looking (see Figure 10).

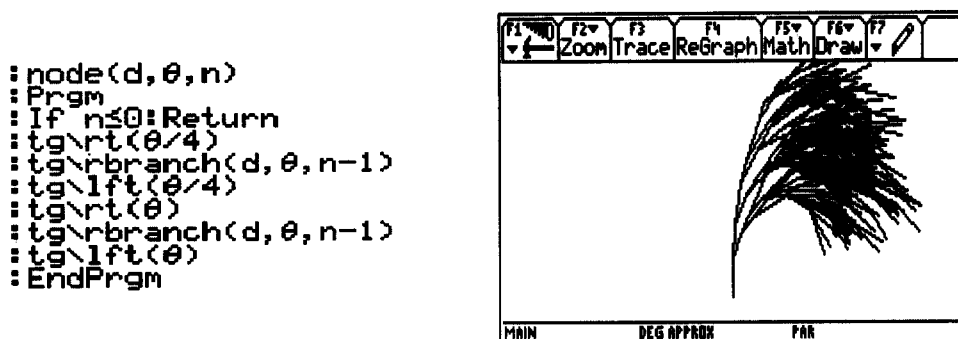


Figure 10. Wind-blown tree I

The following procedure renders a tree swaying in the opposite direction, as portrayed in Figure 11.

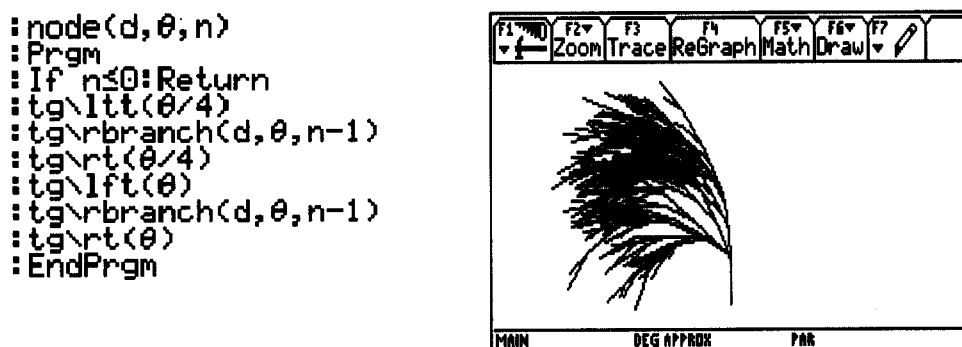


Figure 11. Wind-blown tree II

## V. CONCLUSION

Generating fractal images by graphing calculators or computers combines several important features, which convey the excitement of a living, changing mathematics appropriate to secondary or post-secondary students.

The topic of fractal geometry can be illustrated using natural objects such as snowflakes, leaves, and ferns. These complex and natural forms are often strikingly fantastic and beautiful. The examples highlight the fact that complex and natural behaviors can result from simple mathematical rules such as those embodied in IFS.

The visual splendor beauty of fractals, in concert with their ubiquity in nature, reveals the intellectual beauty of nonlinear mathematics in a compelling way. The

window is now open for students to experience and explore some of the wonder of fractal geometry.

## REFERENCES

- Barnsley, Michael F. (1993): *Fractals Everywhere*, 2nd ed. Boston, MA: Academic Press, Inc. MR 90e:58080
- Bannon, Thomas J. (1991): Fractals and Transformation. *Mathematics Teacher* 84(3), 178–85. MATHDI 1991g.37095
- Crownover, Richard M. (1995): *Introduction to Fractals and Chaos*. Jones and Bartlett Publishers.
- Kutzler, Bernard and Stoutemyer, David R. (1997): *Great TI-92 Programs*, Vol. 1. Translated from German. With diskette, version 2.0. Ser. Title: bk teachware Series “Support in Learning”, no. SL-01. Hagenberg, Austria: bk teachware. MATHDI 1998a.00117
- Kwon, Oh Nam; Kwon, Kisun & Choi, Yunyoung (1998): Fractal geometry using graphing calculators (In Korean). In: *Proceedings of the Meeting of the Korea Society of Educational Studies in Mathematics held at Seoul National Univ., June 26, 1998* (pp. 569–594). Seoul: Korea Society of Educational Studies in Mathematics.
- Mandelbrot, Benoit B. (1982): *The Fractal Geometry of Nature*. San Francisco, CA: W. H. Freeman and Co. MR 84h:00021
- Nowark, Ray (1995): *Mathematical Peepshows for the TI-82*. Brendern Kelly Publishing, Inc.
- Peitgen, Heinz-Otto; Jürgens, Hartmut & Saupe, Dietmar (1992): *Fractals for the Classroom*, Part 1. Introduction to fractals and chaos. With a foreword by Benoit B. Mandelbrot. New York: Springer-Verlag; Reston, VA: National Council of Teachers of Mathematics. MR 92k:00023