CMPU 378: Computer Graphics
Assignment 1: Drawing the Mandelbrot Set

The Mandelbrot Set is probably the most well known fractal. It is an object of marvelous complexity and beauty. It is also extremely simple to define and to draw on a computer screen. The Mandelbrot set may be visualized in the following way. Consider the set C of all complex numbers s = x + y*i*. Each such number s may be visualized as a point with cartesian coordinates (x,y) in a plane. The Mandelbrot set M is a subset of the set C of complex numbers. The set M is connected, so that M is a single region of the plane. The boundary of the set M is extremely jagged. In contrast to the continuous smooth curves one studies in Calculus, the boundary of M remains jagged even if one examines an arbitrarily small portion of the boundary.

In this assignment, you will write a program that displays selected portions of the Mandelbrot set, and its boundary. Your program will be invoked from the shell in the following way:

mandelzoom x_1 x_2 y_1 y_2 w h

In this command, the parameters x_1, x_2, y_1 and y_2 are floating point numbers that define a rectangle in the complex plane. The parameters w and h are integers that indicate the number of pixels along the horizontal and vertical dimensions of the image your program will generate. A pixel at coordinates (u,v) in the image will represent the following complex number:

$$s = \left(x_1 + u\frac{x_2 - x_1}{w - 1} \right) + \left(y_1 + v\frac{y_2 - y_1}{h - 1} \right) i$$

The color of the pixel will indicate whether the number s is a member of the Mandelbrot set. For numbers that are not in the Mandelbrot set, the color will also indicate how close it comes to being a member of the Mandelbrot set.

Well you might ask, what is the Mandelbrot set? How do I tell if a given complex number s = x + y*i* is a member of the Mandelbrot set? How do I tell if a non-member is close to being a member? For each complex number s we define the following series:

$$z_0(s) = 0 + 0i$$
$$z_i(s) = (z_{i-1}(s))^2 + s$$

First recall how complex numbers are added and multiplied. Let z = x + y*i* be a complex number:

$$\begin{aligned} z_1 + z_2 &= (x_1 + x_2) + (y_1 + y_2)i \\ z_1 z_2 &= (x_1 x_2 - y_1 y_2) + (x_1 y_2 + y_1 x_2)i \\ z^2 &= (x^2 - y^2) + (2xy)i \end{aligned}$$

Also recall that the modulus (size) of a complex number z = x + y*i* is given by $|z| = \sqrt{x^2 + y^2}$. With these definitions in hand, we define the Mandelbrot set in the following way: A complex number s is a member of the Mandelbrot set if and only if $|z_i(s)| \le 2$ for all i = 0,1,2 … ∞. Of course you cannot write a program that will compute $|z_i(s)|$ for all i = 0,1,2 … ∞, so you will have to make some compromises. Here is what you should do: The pixel corresponding to the complex number s should be black if $|z_i(s)| \le 2$ for all i = 0,1,2 … 1000.The pixel corresponding to the complex number s should be some other color of your choice if $|z_i(s)| > 2$ for some i such that $0 \le i \le 1000$. In particular, you should find the smallest integer i such that $|z_i(s)| > 2$, and use the number i to choose a color. Since you have only a limited number of colors, you should partition the values of i from 0 to 1000 into intervals such that two values in the same interval get the same color. The simplest approach is to make all the intervals equal in size; however, you will probably generate more interesting images if you find a way to choose the intervals so that the image has roughly equal numbers of pixels of each color.

The program should have some features that help a user to identify and explore the most interesting regions of the Mandelbrot set:

1. The user should have the option to zoom in for a closer look at a particular region of the complex plane. When the user depresses the left mouse button, holds it down, and moves the mouse, a varying rectangular region should be indicated with rubber bands. When the user releases the left button, the program should clear the screen and draw an enlarged view of the selected region, using the entire window. The program should also print out the coordinates defining the selected region.
2. The middle mouse button should bring up a menu with three items: “Push”, “Pop” and “Exit”. The “Push” and “Pop” commands refer to a hierarchy of rectangular regions that should be maintained by the program. The “Pop” command reverts the image to a previously defined larger region of the complex plane. The “Push” command reverts the image to a previously defined smaller region of the complex plane. The “Exit” command causes the program to exit.

Once you have written your program, you can test it on the following parameters. The command “mandelzoom -2.0 0.5 -1.25 1.25 400 400” will give you an overview of the entire Mandelbrot set. In addition, the following commands display some interesting parts of the Mandelbrot set and its boundary:

- mandelzoom 0.262 0.263 0.002 0.003 400 400
- mandelzoom 0.26215 0.26225 0.00215 0.00225 400 400
- mandelzoom -1.4014 -1.4010 -0.0002 0.0002 400 400
- mandelzoom 0.281 0.291 -0.019 -0.009 400 400

Finally, for extra credit you may enter the Vassar CS378 Class Fractal Art Contest by including in the documentation of your program the parameters you used to generate your favorite picture. The winner of first prize will get 10% added to his/her grade. The winner of second prize will get 5% added to his/her grade.

For additional information, you may want to read the article “Computer Recreations: A computer microscope zooms in for a look at the most complex object in mathematics”, by A. K. Dewdney, from Scientific American, August 1985.