HW1 Report

Created	@September 21, 2022 11:34 AM
Class	Graphics
Semester	Fall 2022

COSC 4370 - Homework 1

Name: Anthony Ciocco

PSID: 1600875

1 Problem

The assignment is to rasterize an ellipse defined by the function:

$$(\frac{x}{12})^2 + (\frac{y}{6})^2 = 64^2 : x \geq 0$$

Because the ellipse is so long, with it's major axis having a radius of 768 and its minor axis with a radius of 384, the image has to fit the ellipse where $x \geq 0$. This means the image will be about 800 x 800 pixels wide to fit the $x \geq 0$ side of the ellipse comfortably

2 Method

In the provided code, there is one main function belonging to the BMP struct. Upon initializing a new BMP struct, a size of 800 x 800 was determined to be the best fit for the half ellipse. Next a for loop is placed with the condition being equal to or less than the bounds of the ellipse. Since the x bounds is 768, that number was chosen.

Within the for-loop, function *set_pixel* takes an x and y value to plot on the grid of 800 x 800. The x value is the is the for-loop iterator starting at 0, and the y value is the above equation solved for y. This is done twice, once for the top half of the ellipse, once for the bottom half of the ellipse.

3 Implementation

As stated above, the initial condition for the for-loop starts at 0. As the for-loop continues, it is iterated by a thousandth and for every iteration, a new double is split out for the equation solve for y. That solve equation looks like this:

$$y = 384 \pm \sqrt{36 * (64^2 - \frac{x^2}{144})}$$

Where $i = x$ in the for-loop, the extra 384 is there to translate the computed points up in the y direction to fit into the image bounds since the origin is the bottom left of the image . The plus-or-minus is there to express two y values: the top and bottom halves of the ellipse respectively.

3.1 set_pixel

This function takes in an (x, y) coordinate along with an RGBA color value. Dependent on the size of the image bound from initializing the BMP struct, *set_pixel* will map out the (x, y) pixel given to it in the for-loop. Over the entire conditional, each mapped out (x, y) coordinate is approximated to its closest integer value and drawn onto the 800 x 800 array. Because *set_pixel*'s first two arguments are of the type *uint32_t,* then any double values given to it as parameters are cast into integers and rounded down. Because so many doubles are input into the function, the ellipse is more likely to hit every point in the 800 x 800 grid from the given ellipse formula. If any of the x or y values are outside of the image bounds, an error is thrown.

4 Result

The program outputs a .BMP file with the resolution of 800 x 800. White colors are used for those (x, y) integer values that the equation hits, and the remaining colors are in black.