COSC 4370 - Homework 1

Name: Morrison Fowlks PSID: 1855790

September 2021

1 Problem

The purpose behind this assignment is to implement an algorithm to rasterize a given eclipse. This eclipse is defined as (x/12)^2 + (y/6)^2 = 642 where y>=0. To accomplish the given task, one will need to focus on the principles of computer graphics while also keeping in mind how to scan converting circles.

2 Method

When generating an eclipse in the given main.cpp file, one should first create a function that generates the wanted eclipse. With this in mind, no other changes would need to be done to the code, other than eliminating the line of code outputting the original horizontal white line, being bmpNew.set_pixel(i, 100, 255, 255, 255, 0), and replacing it with that of the new generating eclipse function.

This function will be named genEclipse and will receive an address of a given BMP or bitmap as an input. Once this function has received a valid input, it will continue to render pixels on locations that would be seen on the given eclipse, although only those that have a y>=0. With this in mind however, when creating this function, one should keep in mind the importance cosine and sine will have on the rendering of said pixels. The function genEclipse will use cosine and sine in order to avoid large gaps within the rendering of pixels in terms of the given eclipse.

3 Implementation

When creating the function that generates the desired eclipse, it is important to keep in mind the specifications therein, as seen in the need to have the value of y greater than or equal to 0. This is in order to have the overall eclipse centered in the BMP, to make sure the eclipse fits in the given BMP, and to make sure the final eclipse is rendered smooth and accurately.

3.1 genEclipse

The function genEclipse was created by first simplifying the original equation into two variables, and then centering the overall graph on the x and y axis of the given BMP. This is done by making the center of x equal to bmp.bmp_info_header.width/2, and the center of y equal to bmp.bmp_info_header.height. After this step, one would then go on to graph the given ellipse through the use of a for loop, that fully covers the 1st and second quadrant of the graph, implementing the rule that the eclipse would only have a y greater than 0. This loop then renders pixels on the location of the desired eclipse using cosine and sine, the simplified original eclipse functions, and centered x and y coordinates. For these pixels to render however, there is one more "if" loop that renders the needed pixels as long as they indeed have a y value greater than or equal to 0. Once this is done, the pixels are rendered white, using the set pixel function, as seen in bmp.set_pixel(x,y,255,255,255,0). Overall, this function rasterizes the given eclipse in a simple yet very effective way.

4 Results

Given the created and filled bitmap in the main.cpp file and function, the function genEclipse is then looped until the entire bmp is filled accurately. Once this is complete, and the function has generated the desired eclipse, the output of the bitmap is then designated to "output.bmp" which showcases the final result.