COSC 4370 - Homework 1

Name: Cyrus Shekari **PSID:** 2042446

February 2023

1 Objective

The objective of this program is to implement an algorithm that rasterizes an ellipse using the BMP class. The BMP class is used to create an image with a dimension of 1600 x 2000 pixels and outline the ellipse with a color gradient of blue and green. The ellipse equation is defined as $\left(\frac{x}{6}\right)^2 + \left(\frac{y}{12}\right)^2 = 64^2$ where $y \geq 0$. The image data containing the ellipse is then written to a BMP file named "output.bmp".

2 Method

The method applied in this program is the midpoint algorithm to draw an ellipse. This approach iteratively calculates the decision parameter based on the distance from the midpoint to the actual ellipse curve, thus efficiently computing points along the ellipse boundary. The *Draw_Points* function is responsible for rendering individual points on the ellipse, while the *Draw_Ellipse* function is responsible for drawing the entire ellipse. However, since the ellipse equation is bounded by $y \geq 0$, the program only outlines the top half of the ellipse. *Draw_Ellipse* takes the BMP object and the ellipse parameters, such as the center and radii of the ellipse, as arguments to rasterize the image. The main function initializes a BMP object with specified dimensions and coordinates, invokes the *Draw_Ellipse* which then invokes the *Draw_Points* function to generate the ellipse, and saves the BMP object to a file.

3 Implementation

The implemented program encompasses 2 functions that work together to generate the desired ellipse using the midpoint algorithm. These functions include Draw_Ellipse and Draw_Points with each being responsible for where to draw and when to modify the color of the pixels in the desired area.

3.1 Draw_Points

The Draw_Points function in the program is used to draw individual points on the ellipse. It takes a BMP object, the x and y radii of the ellipse, as well as the x and y coordinates of the center of the ellipse as inputs. The function sets the thickness of the points and uses two nested loops to iterate over the x and y axes of the ellipse. For each (x, y) coordinate within the ellipse, the function calls the set_pixel method of the BMP object, which sets the pixel color at that location. The set_pixel method takes the x and y coordinates, as well as RGB values to specify the color of the point. In this program, the RGB values are set to (153, 255, 51, 0) and (255, 255, 51, 0) for the right and left halves of the ellipse, respectively. These colors are used to style the ellipse.

3.2 Draw_Ellipse

The function initializes the x and y coordinates to zero and the maximum y radius, respectively. It then calculates the parameters needed for the algorithm, such as the stopping points for each half of the ellipse and the quadratic equations used to determine the position of each point on the ellipse. On an important note, the y value is set to the maximum of *rad_y* and 0 since we only want the top half of the ellipse. The algorithm uses a while loop to draw the top half of the ellipse by calling the Draw_Points function for each calculated (x, y) position until the stopping condition is reached. If the calculated value of the quadratic equation is less than zero, x is incremented, the stopping point for x is updated, and the quadratic equation for the top half of the ellipse is also updated. If the calculated value of the quadratic equation is greater than or equal to zero, x

and y are incremented and decremented, respectively, and the stopping points for x and y are updated accordingly, as well as the quadratic equation for the top half of the ellipse.

4 Results

The result of the program is a 1600 x 2000 BMP image named "output.bmp" that is filled with a color gradient of color gradient of blue and green. The image contains an ellipse that is defined by the equation $\left(\frac{x}{6}\right)^2 + \left(\frac{y}{12}\right)^2 = 64^2$ and $y \geq 0$. When viewed through an image viewer, the top half of the ellipse can be seen.