COSC 4370 - Homework 1

Name: Joseph Jophy UHID#: 2048762

February 2023

1 Problem

The assignment requires the rasterization of an ellipse. The ellipse is defined by the equation $(x/6)^2 + (y/12)^2 = 64^2$ where y>=0. The *radius* along the x-axis is 6*sqrt(64*64) and the radius along the y-axis is 12*sqrt(64*64).

2 Method

In order to draw our ellipse, we will be using the midpoint method. We utilize two functions **midPointEllipse** and **drawPoints.**

We receive the long values x_center, y_center, x_radius, y_radius. x_center and y_center represents the center position of the ellipse. x_radius represents the radius along the x-axis and y_radius represents the radius along the y-axis. The general idea behind the midpoint method is to find the distance from the center needed to create the correct coordinates of the ellipse. We will add or subtract from the x_center/y_center to generate all the coordinates to form an ellipse. We will not subtract from the y_center since we will only be generating the top half of the ellipse. We will iterate and use the midpoint method to generate all the points until we form an ellipse. We will go into more details of how we will iterate through all the necessary points to generate our ellipse in the implementation section.

3 Implementation

Given the x and y coordinates of the center, and the horizontal radius and vertical radius of the ellipse we are supposed to generate we will find all the coordinates need to generate the ellipse by calculating all the corresponding distances from the center.

3.1 midPointEllipse

The algorithm we will be using comes from https://www.geeksforgeeks.org/midpoint-ellipse-drawing-algorithm/.

Algorithm:

1. Obtain the initial decision parameter for region 1 as: $p1_0=r_y^2+1/4r_x^2-r_x\ ^2r_y$
2. For every x_k position in region 1 :
   If $p1_k<0$ then the next point along the is (x_{k+1}, y_k) and $p1_{k+1}=p1_k+2r_y^2x_{k+1}+r_y^2$
   Else, the next point is (x_{k+1}, y_{k-1})
   And $p1_{k+1}=p1_k+2r_y^2x_{k+1} – 2r_x^2y_{k+1}+r_y^2$
3. Obtain the initial value in region 2 using the last point (x_0, y_0) of region 1 as: $p2_0=r_y^2(x_0+1/2)^2+r_x^2\ (y_0-1)^2-r_x^2r_y^2$
4. At each y_k in region 2 starting at k =0 perform the following task.
   If $p2_k>0$ the next point is (x_k, y_{k-1}) and $p2_{k+1}=p2_k-2r_x^2y_{k+1}+r_x^2$
5. Else, the next point is (x_{k+1}, y_{k-1}) and $p2_{k+1}=p2_k+2r_y^2x_{k+1}$ - $2r_x^2y_{k+1}+r_x^2$
6. Now obtain the symmetric points in quadrant I and quadrant II and plot the coordinate value as: (x+xc, y+yc) & (x-xc, y+yc)
7. Repeat the steps for region 1 until $2r_y^2x>=2r_x^2y$.

Following this algorithm, will allow us to generate all the points in the array. On step 6, the drawPoints function will be called so we can add the point to output.bmp image file.

3.2 plotPoints

plotPoints is called inside the midPointEllipse function repeatedly and receives long values x_center, y_center, x_dist, y_dist. x_dist represents the distance we add to the x_center to get our coordinates. y_dist represents the distance we add to the y_center to get our coordinates. Adding x_dist and y_dist from the center will get us the right-hand portion of the ellipse. We also subtract x_dist from the x_center from the center to get the left half of the ellipse. Note: we will not subtract y_dist from the y_center since we will not generating the bottom half of the ellipse.

4 Results

The output of the program is a .bmp file. When viewing the output.bmp file, an image of half an ellipse can be seen.