COSC 4370

Name: Larry Nguyen PSID: 2032098

1) Problem

The assignment involves making an algorithm to rasterize an ellipse for the equation $(x/16)^2 + (y/25)^2 = 30^2$ where x is <= 0.

2) Method

The method that definitely needs to be used to achieve rasterizing an ellipse is by using the midpoint ellipse algorithm. The main point of the algorithm is to help highlight the pixels that are part of the ellipse and decide the next pixel to go after and color using input variables such as xc (x center), yc(y center) which represent the coordinates for the ellipse center and a and b which are the horizontal and vertical radius. I will explain in the implementation section about how the inputs are used alongside other variables that are used to compute in the DrawEllipse function that uses the middle point ellipse algorithm I'm talking about here. And the function can also adjust for whether the current area is in a flat or steep slope and I will also explain in the implementation section, what the function does in reaction to it. And because of how the function has different sections for different areas of the ellipse, it's able to choose the next correct pixel to go for which helps in keeping adjacent pixels connected thus showing no gaps.

3) Implementation

First of all, the DrawEllipse function that uses the midpoint ellipse algorithm gets called and uses the b variable to start at the top most point of the ellipse to start with which would be (0, 25). The a and b variables represent the horizontal and vertical radius for the assignment which are 16 and 25 respectively. The input variables xc and yc are both 100 since the coordinate (100,100) is the center of the ellipse. And the variables d1(d1 = b^2 - a^2b + 0.25a^2) and d2(d2 = b^2(x + 0.5)^2 + a^2(y-1)^2 - a^2 b^2) inside the functions are decision parameters for regions 1 and 2 part of the function to help decide the best pixel to go to next. The variables dx and dy are used to incrementally change d1 and d2 when moving throughout the ellipse. While the x and y variables represent the current x and y coordinates that are being plotted.

Now we can get more into how the function works starting with how the function is under "Region 1" mode while the slope is steep which is represented by the while loop condition of dx < dy. The reason we start with this condition is due to how we start off of the ellipse steep. And inside the loop, we plot the pixels only in the first half of the ellipse due to x <= 0 thus why xc -x is used. After that, we check if d1 is negative or not. If d1 is negative then that means the next pixel is in the ellipse thus we move to the right (E). Then we increment some variables such as x by 1, dx by 2*b^2, and d1 by dx + b2. The reason for the incrementation is to update the d1 or d2 to then know whether to change directions or not. Also for the cases where d1 is positive or is zero meaning the pixel is outside, we move by the direction of right and down (SE) thus we adjust x by incrementing it by 1, decrease y by 1, increment dx by 2*b^2, decrease dy by 2*a^2 and increment d1 by dx-dy + b2. And we repeat the while loop until the slope becomes flattened (dx >= dy).

Once the function gets to the point where dx >= dy, that means the slope has flattened out and now goes under the "Region 2" phase which is similar to Region 1 but has some slight differences. These slight differences are that the while loop for this region depends on y being greater than or equal to 0. This part of the function now uses d2 instead of d1 as the decision parameter. And this different loop also checks whether or not d2 is positive or not. If d2 is positive, that means that the pixel is outside the ellipse thus we move down (S) thus we decrement y by 1, dy by 2* a^2 and increment d2 by a2 - dy. And if d2 is negative meaning the pixel is inside the ellipse, we move right and down (SE). We then increment x by 1 and decrement y by 1 and increment dx by 2*b^2, decrement dy by 2*a^2 and increment d2 by dx - dy + a^2. And this all repeats in a loop until y becomes 0 meaning the ellipse is now completely drawn. Also I want to mention how part of the code also plots the pixels part of the ellipse only in the first half just like with region 1 in the while loop.

4) Results

The output of the program is in the form of a bmp file and has canva width and height of (200,200). It shows the ellipse of the equation $(x/16)^2 + (y/25)^2 = 30^2$ in the 2nd and 3rd quadrant.