Drawing an Ellipse

Tong Wu

September 20, 2019

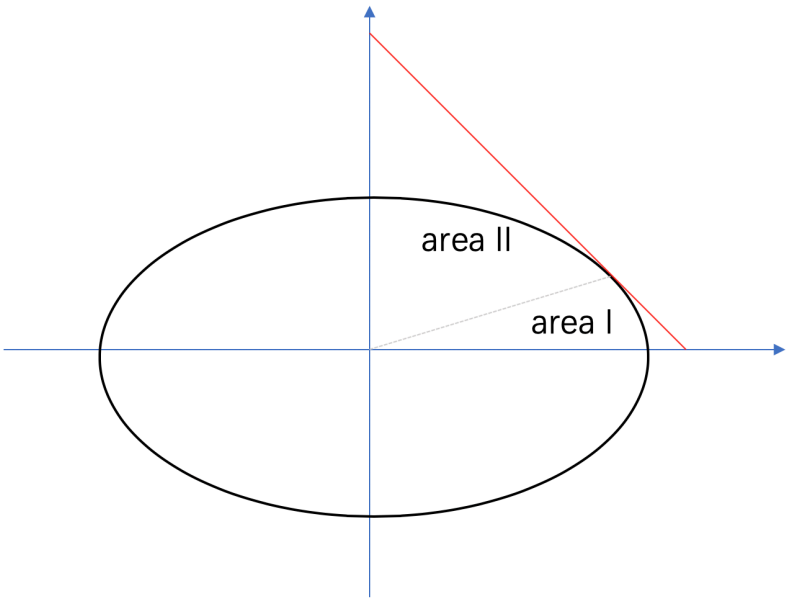
1 Introduction

This is an ellipse drawing program. The user specifies the coordinate of the center of the ellipse and the length of the semi-major axis and semi-short axis on a 1920*1080 canvas, and gives the angle of the ellipse around the center. The image of the ellipse is stored in an file in ppm format.

Running screenshot under macOS

2 Algorithm

The program uses the Bresenhem algorithm to draw the ellipse.



For points in the first quadrant, if a point is outside the ellipse,

$$b^2x^2 + a^2y^2 - a^2b^2 > 0.$$

If a point is inside the ellipse,

$$b^2x^2 + a^2y^2 - a^2b^2 - 1 < 0.$$

For the points in area I, we draw the ellipse point by point counterclockwise. If it is known that (x, y) is on the ellipse, then it can be inferred that the next point is either $(x, y+1)$ or $(x-1, y+1)$. The difference of the errors is

$$b^2x^2 + a^2(y+1)^2 - a^2b^2 + b^2(x-1)^2 + a^2(y+1)^2 - a^2b^2,$$

which is

$$2(b^2x^2 + a^2y^2 - a^2b^2 + a^2(2y+1)) + b^2(1-2x).$$

If the error > 0, the next point is $(x-1, y+1)$, otherwise it is $(x, y+1)$.

For the points in area II, we draw the ellipse point by point clockwise. The next point of (x, y) is either $(x+1, y)$ or $(x+1, y-1)$. The difference of the errors is

$$b^2(x+1)^2 + a^2y^2 - a^2b^2 + b^2(x+1)^2 + a^2(y-1)^2 - a^2b^2,$$

which is

$$2(b^2x^2 + a^2y^2 - a^2b^2 + b^2(2x+1)) + a^2(1-2y).$$

If the error > 0, the next point is $(x+1, y-1)$, otherwise it is $(x+1, y)$.

Now consider the slope to test which area a point belongs to. It can be used as condition of the iteraion. For points on the ellipse,

$$\frac{2x}{a^2} + \frac{2y}{b^2} = 0,$$

which means

$$y' = -\frac{b^2x}{a^2y}.$$

For the points in area I,

$$y' < -1,$$

which means,

$$b^2x > a^2y.$$

For the points in area II,

$$y' > -1,$$

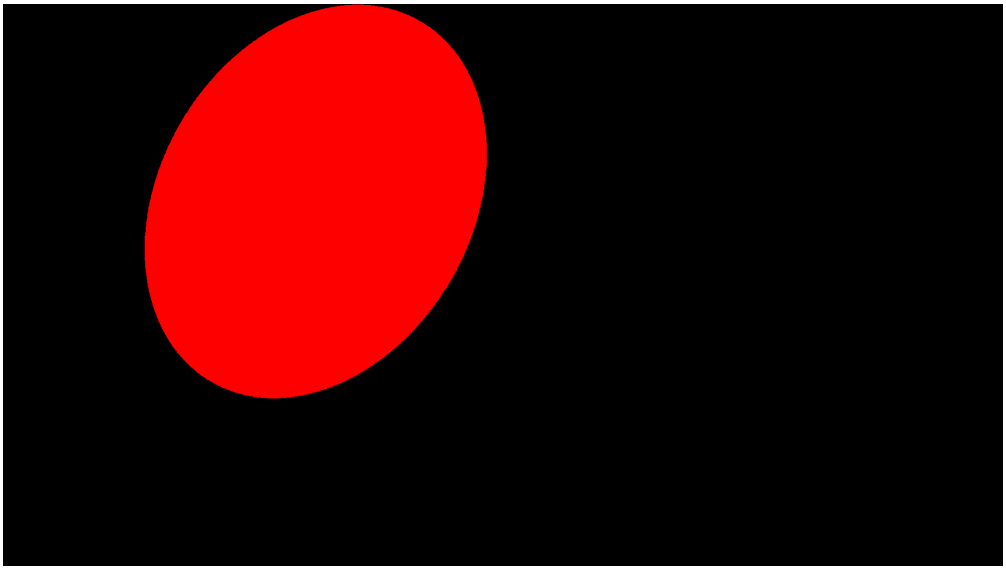
which means,

$$b^2x < a^2y.$$

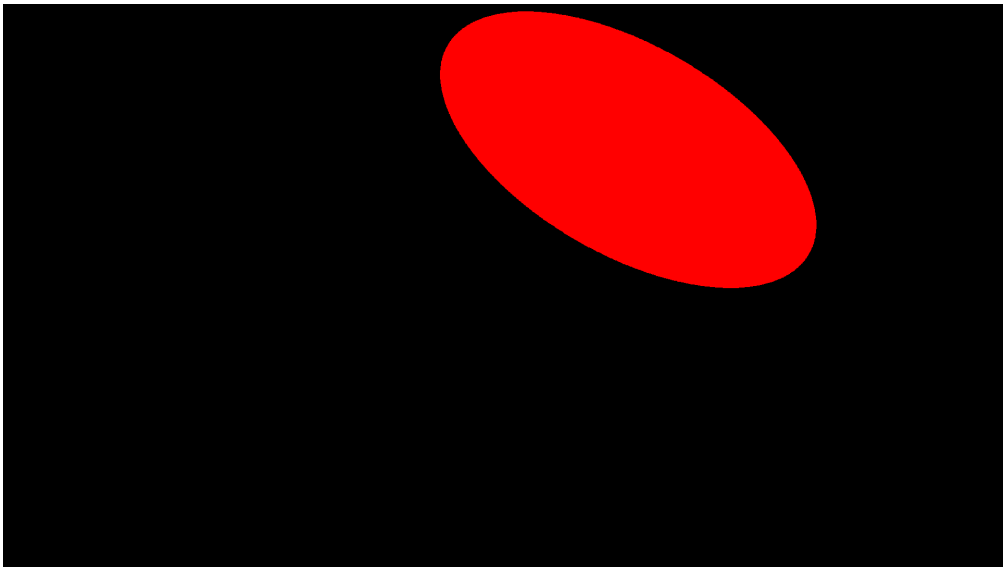
3 Implementation

The ellipse is designed as a class. When the constructor is called, an ellipse perpendicular to the axis is drawn first. The points in the first quadrant is drawn with Bresenhem algorithm and the points in other three quadrants are symmetric with points in the first quadrant. The rotated coordinates can be get by multiplying the initial coordinates. After the outline is drawn, the ellipse will be colored.

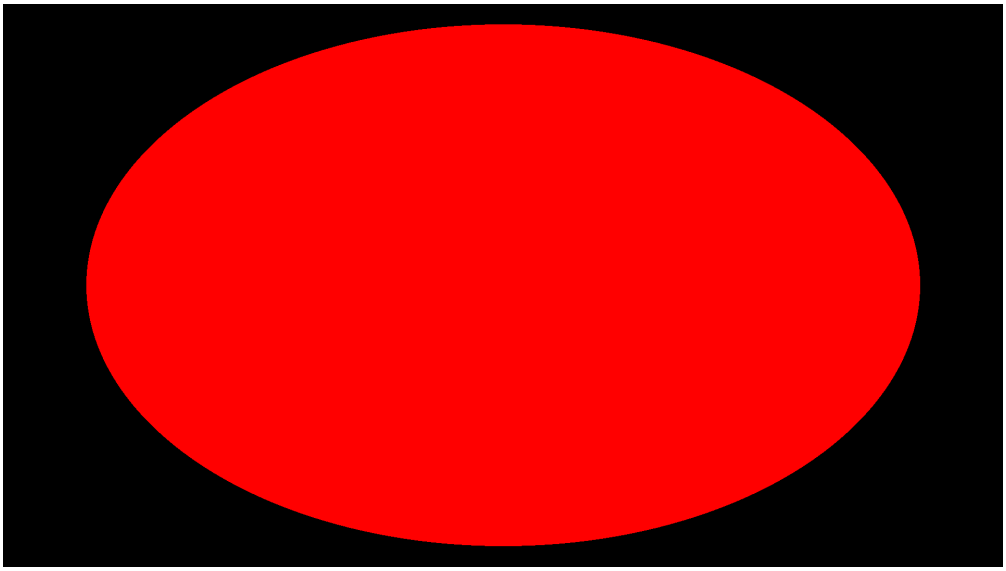
4 Result



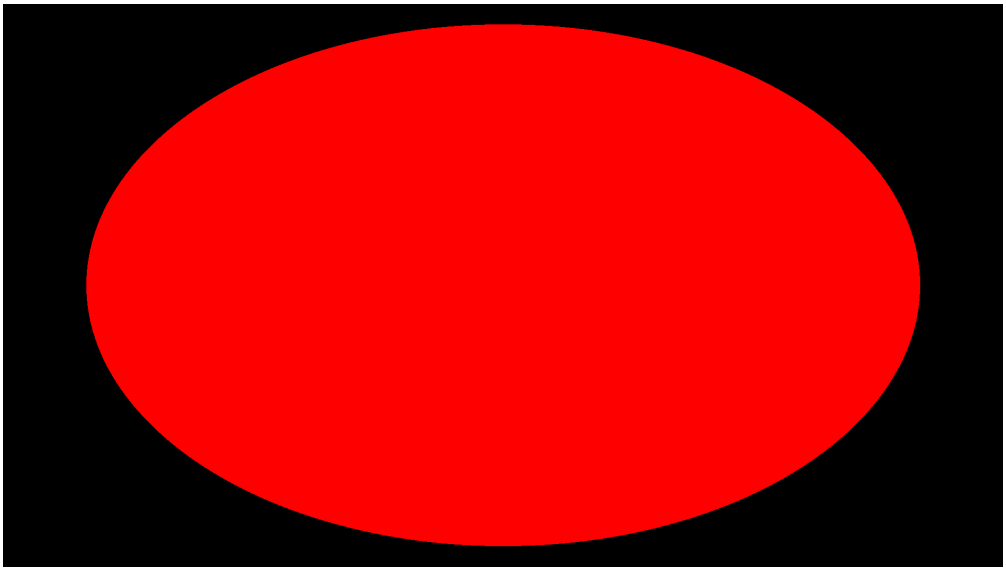
Sample 1: $center(600, 700)$, $a = 400$, $b = 300$, $\theta = 60^\circ$



Sample 2: $center(1200, 800)$, $a = 400$, $b = 200$, $\theta = 150^\circ$



Sample 3: $center(960, 540)$, $a = 800$, $b = 500$, $\theta = 0^\circ$



Sample 3: $center(960, 540)$, $a = 800$, $b = 500$, $\theta = 0^\circ$