Name - Ameya Barapatre
Roll No - 06
SE- DS

Experiment No- 04

10

Aim → To implement midpoint Ellipse drawing algorithm for drawing ellipse.

Resource Required → Turbo C, Printer, Printout, Stationary

Theory →

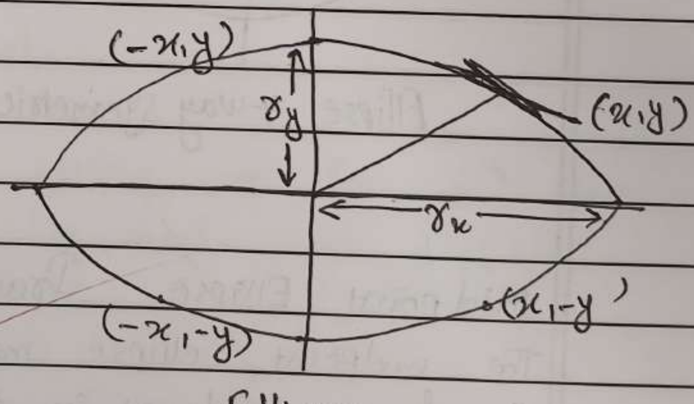
- Ellipse is a elongated circle, with two axes → major axis and minor axis.
Major axis = $2a = 2r_x$ and Minor axis $= 2b = 2r_y$
where r_x and r_y are the radius of major and minor axis respectively

Equation of Ellipse

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} = 1$$

$$\frac{b^2x^2 + a^2y^2}{a^2b^2} = 1$$

$$b^2x^2 + a^2y^2 - a^2 \cdot b^2 = 0$$



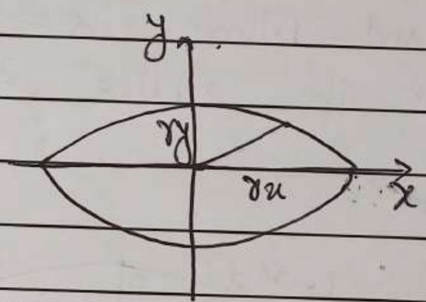
Ellipse

- If we put any coordinates into the equation and result is 0, then that coordinates are on ellipse boundary
- If it result in <0, then coordinates are inside ellipse. If the result >0, then the coordinates are outside ellipse.
- To find out the exact coordinates midpoint ellipse drawing algorithm is used.

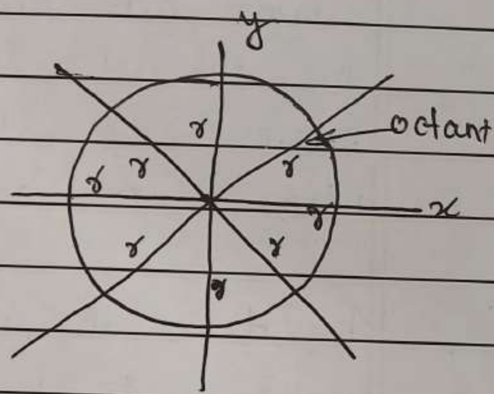
- Ellipse is symmetric between quadrant but not symmetric between the two octants of a quadrant.

Difference between circle and ellipse →

- Circle has 8-way symmetry and ellipse has 4-way symmetry.
- In circle we need to plot only one octant of any quadrant, but in ellipse we need to plot 2 octants i.e. I quadrant to plot entire ellipse needed to be calculated.



Ellipse 4-way symmetric



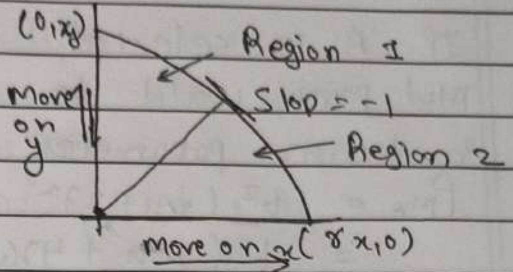
8-way Symmetric

Mid point Ellipse Drawing Algorithm →

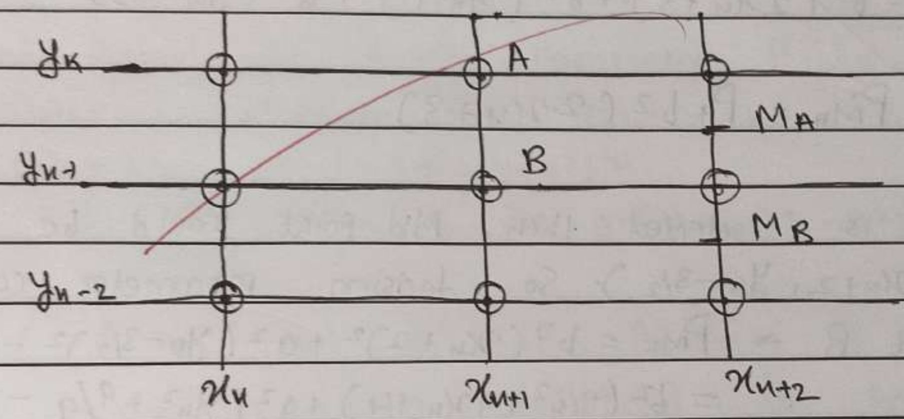
The midpoint ellipse method is applied throught the first quadrant in two parts i.e. region-1 and region-2. We are forming this regions by considering the slope of the curve. If the slope of the curve is less than (-1) then we are in region-1 and when the slope becomes greater than (-1) then in region-2. See figure below i.e. At the boundary between region 1 and 2. $\frac{dy}{dx} = -1$

The slope of the ellipse is calculated as

$$\frac{dy}{dx} = -\frac{2 r_y^2 x}{2 r_x^2 y}$$



For region-1 →



Region-1

Initial coordinates for ellipse are (x_k, y_k) then there are two choices for next pixel that is either point A or B.

To select any one from this the mid point should be calculated where coordinates of mid point are
$M = (x_k+1, y_k - 1/2)$

Let's derive the decision parameter P by evaluating ellipse equation for region-1

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

$$\therefore P = b^2(x_k+1)^2 + a^2(y_k - 1/2)^2 - a^2 \cdot b^2$$

If A is selected then
Mid point would be $M_A = (x_n+2 ; y_n - 1/2)$
So, decision parameter calculated for point A.

$P_{M_A} = b^2 (x_n+2)^2 + a^2 (y_n - 1/2)^2 - a^2 b^2$

$= b^2 (x_n^2 + 4x_n + 4) + a^2 (y_n - 1/2)^2 - a^2 b^2$

$= b^2 (x_n^2 + 2x_n + 1 \quad 2x_n + 3) + a^2 (y_n - 1/2)^2 - a^2 b^2$

$= b^2 (2x_n + 3) + b^2 (x_n+1)^2 + a^2 (y_n - 1/2)^2 - a^2 b^2$

$\therefore P_{M_A} = P + b^2 (2x_n + 3)$

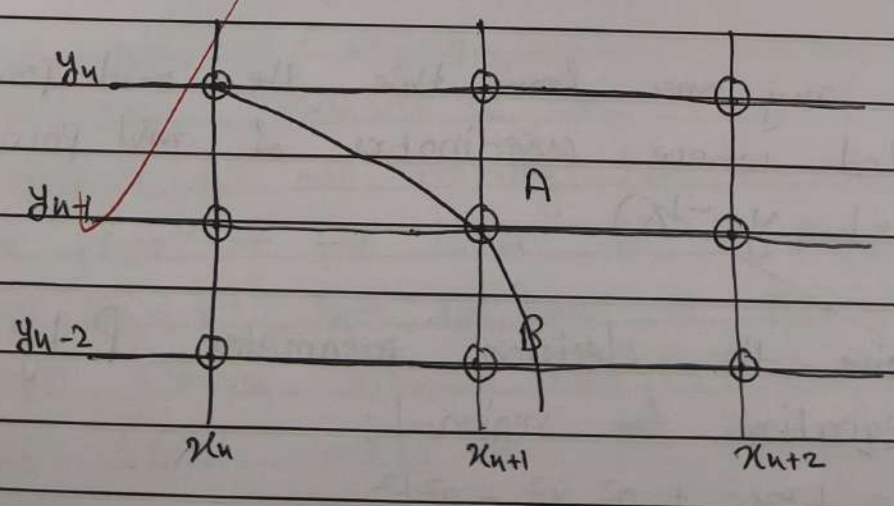
If B is selected then, Mid point would be
$M_B = (x_n+2, y_n - 3/2)$. So, decision parameter calculated for point B $\rightarrow$ $P_{M_b} = b^2 (x_n+2)^2 + a^2 (y_n - 3/2)^2 - a^2 b^2$

$= b^2 (x_n^2 + 4x_n + 4) + a^2 (y_n^2 + 9/4 - 3 y_n) - a^2 b^2$

$= b^2 [(x_n+1)^2 + (2x_n+3)] + a^2 [(y_n - 1/2)^2 + (2 - 2y_n)] - a^2 b^2$

$\therefore P_{M_B} = P + b^2 (2x_n + 3) + a^2 (2 - 2y_n)$

For Region 2 $\rightarrow$



Initial coordinates for ellipse are (x_u, y_u) then there are two choices for a next pixel that is neither point A or point B.
To select any one point from, we need to calculate mid point where midpoint lies in
$M = (x_u + \frac{1}{2}, y_u - 1)$

let derive the decision parameter P by evaluating ellipse equation for region - 2.
$f(x,y) = b^2x^2 + a^2y^2 - a^2b^2$
$\therefore P = b^2(x_u + 1/2)^2 + a^2(y_u - 1)^2 - a^2b^2$

If A is selected then. Midpoint is $M_A = (x_u + 1/2, y_u - 2)$
So decision parameter calculated for point A

$$P_{MA} = b^2(x_u + 1/2)^2 + a^2(y_u - 2)^2 - a^2b^2$$
$$= b^2(x_u + 1/2)^2 + a^2(y_u^2 - 4y_u + 4) - a^2b^2$$
$$= b^2(x_u + 1/2)^2 + a^2((y-1)^2 + 3 - 2y_u) - a^2b^2$$
$$\therefore P_{MA} = P + a^2(3 - 2y_u)$$

If B is selected then,
Midpoint is $M_B = (x_u + 3/2, y_u - 2)$
So, decision Parameter calculated for point B

$$P_{MB} = b^2(x_u + 3/2)^2 + a^2(y_u - 2)^2 - a^2b^2$$
$$= b^2((x_u + 1/2)^2 + 2x_u + 2) + a^2((y_u - 1)^2 + 3 - 2y_u) - a^2b^2$$
$$\therefore P_{MB} = P + b^2(2x_u + 2) + a^2(3 - 2y_u)$$

At Initially for region - 1

$x \leftarrow 0$ and $y \leftarrow b$

$P_{(initial)} = b^2 + a^2 (b - 1/2) - a^2 b^2$

$\therefore P_{initial} = b^2 - a^2 b + \frac{a^2}{4}$

Algorithm →

Step 1 - Read a (~~rx~~) and b(ry)

Step 2 - Initialise starting point of region I (R₁)

$x = 0$; $y = b$

Step 3 - Calculate $P = b^2 - a^2 b + a^4/4$

$dx = 2b^2 x$; $dy = 2a^2 y$

Step 4 - Repeat while $(dx < dy)$

Plot (x, y) if $(P_k < 0)$

$x_{k+1} = x_k + 1$

$dx = 2b^2 x_{k+1}$

$P_{k+1} = P_k + dx + b^2$

else, $x_{k+1} = x_k + 1$; $y_{k+1} = y_k - 1$

$dx = 2b^2 x$; $dy = 2a^2 y$

$P_{k+1} = P_k + dx - dy + b^2$

Step 5 - When $(dx \geq dy)$

Plot region 2 as, $P = b^2 (x + 1/2)^2 + a^2 (y-1)^2 - a^2 b^2$

Step 6 - While $(y>0)$

Plot (x, y) if $(P_n<0)$

$x_{n+1} = x_n + 1$

$y_{n+1} = y_n - 1$

$dx = 2b^2x$; $dy = 2a^2y$

$P_{n+1} = P_n + dx + a^2$

Else,

$y_{n+1} = y_n - 1$

$dy = 2a^2y$

$P_{n+1} = P_n - dy + a^2$

Conclusion - The midpoint ellipse ~~to~~ drawing ~~of~~ algorithm efficiently generates an ellipse by incrementally plotting points along the boundary, using decision parameters to determine the next point. It is computationally efficient because it only involves simple integer calculations rather than floating point operations, making it suitable for real time rendering in computer graphics. The algorithm's main advantage is its ability to maintain symmetry and precision ensuring a smooth and accurate ellipse.

2018