plot (x_k+1, y_k+1) and

$$P_{k+1} = P_k + 2\Delta y - 2\Delta x$$

5. Repeat step 4 Δx times.

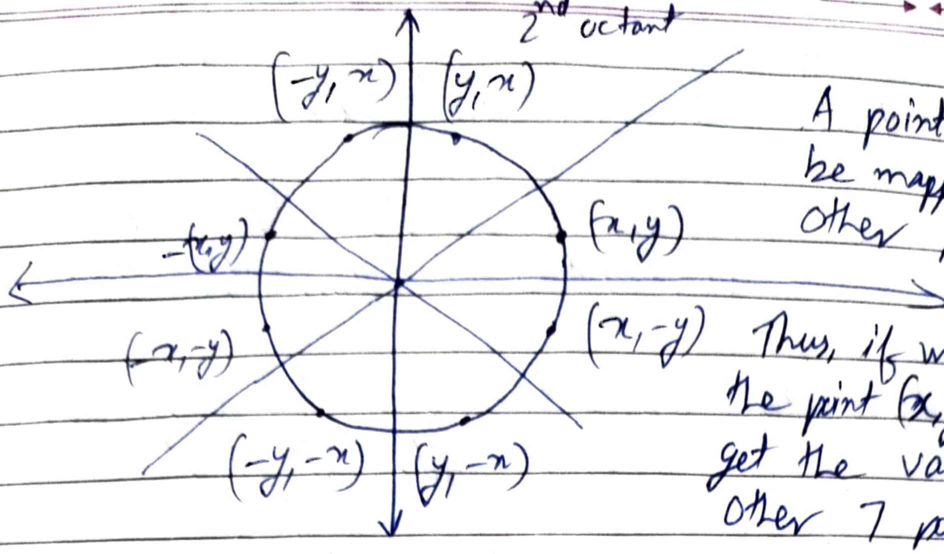
Drawing a circle

* Midpoint Algorithm

→ We sample at unit intervals and determine the closet pixel position to the specified circle path.

We use the equation of a circle centred at the origin ~~to do~~ as the decision parameter. We find the value of the equation at the midpoint of the two potential pixels and see if it lies inside the circle or outside the circle.

We also use the symmetry of circle in octants to reduce the number of computations needed to determine



A point (x, y) can be mapped to 7 other points.

Thus, if we determine the point (x, y) we can get the value for other 7 points.

Let's use the 2nd octant, starting from the point $(0, r)$

$$f_{circle}(x, y) = x^2 + y^2 - r^2$$

$$f_{circle}(x,y) \begin{cases} < 0, & \text{if } (x,y) \text{ is inside circle boundary} \\ = 0, & \text{if } (x,y) \text{ is on circle boundary} \\ > 0, & \text{if } (x,y) \text{ is outside the circle boundary} \end{cases}$$

If (x_k, y_k) is plotted correctly, next point is (x_{k+1}, y_k) or (x_{k+1}, y_{k-1})

$$P_k = f_{circle}\left(x_{k+1}, y_k - \tfrac{1}{2}\right)$$

$$= x_{k+1}^2 + \left(y_k - \tfrac{1}{2}\right)^2 - r^2$$

$$= x_{k+1}^2 + y_k^2 - y_k + \tfrac{1}{4} - r^2$$

$$= (x_k + 1)^2 + y_k^2 - y_k + \tfrac{1}{4} - r^2$$

$$\boxed{\therefore P_k = (x_k + 1)^2 + \left(y_k - \tfrac{1}{2}\right)^2 - r^2}$$

if $P_k < 0$, midpoint is inside circle, thus y_k is closer to path

$P_k > 0$, midpoint is outside circle, thus $y_k - 1$ is closer to path

Date :

Similarly,

$$P_{k+1} = f_{circle}\left(x_{k+1}+1,\ y_{k+1}-\tfrac{1}{2}\right)$$

$$= (x_{k+1}+1)^2 + \left(y_{k+1}-\tfrac{1}{2}\right)^2 - r^2$$

$$= [(x_k+1)+1]^2 + \left(y_{k+1}-\tfrac{1}{2}\right)^2 - r^2$$

$$P_{k+1} - P_k = \cancel{(x_k+1)^2} + 2(x_k+1) + 1 + \left(y_{k+1}-\tfrac{1}{2}\right)^2 - \cancel{r^2}$$
$$- \cancel{(x_k+1)^2} - \left(y_k-\tfrac{1}{2}\right)^2 + \cancel{r^2}$$

$$= 2x_{k+1} + 1 + y_{k+1}^2 - y_{k+1} + \cancel{1/4}$$
$$- y_k^2 + y_k - \cancel{1/4}$$

$$\boxed{\therefore P_{k+1} = 2x_{k+1} + 1 + (y_{k+1}^2 - y_k^2) - (y_{k+1} - y_k) + P_k}$$

If $P_k < 0$, then $y_{k+1} = y_k$ thus,

$$\boxed{P_{k+1} = 2x_{k+1} + 1 + P_k}$$

If $P_k > 0$, then $y_{k+1} = y_k - 1$ thus,

$$P_{k+1} = P_k + 2x_{k+1} + 1 + 1 - 2y_k + 1$$
$$= P_k + 2x_{k+1} - 2(y_k - 1) + 1$$

$$\boxed{\therefore P_{k+1} = P_k + 2x_{k+1} - 2y_{k+1} + 1}$$

For the start point $(0, r)$

$$P_0 = f_{circle}\left(x_1,\ y_0 - \tfrac{1}{2}\right)$$

$$= f\left(1,\ r - \tfrac{1}{2}\right)$$

$$= 1 + \cancel{r^2} - r + \tfrac{1}{4} - \cancel{r^2}$$

$$\boxed{P_0 = \frac{5}{4} - r}$$

Thus, Algorithm is:-

1. Input radius r and circle center (x_c, y_c) and obtain the 1^{st} point on the circumference of a circle centred on the origin as

$$(x_0, y_0) = (0, r)$$

2. Calculate initial value of the decision parameter as

$$P_0 = 5/4 - r$$

3. At each x_k position, starting at $k=0$, perform the following test:

If $P_k < 0$, ~~plot~~ next point is (x_{k+1}, y_k) and

$$P_{k+1} = P_k + 2x_{k+1} + 1$$

~~If $P_k \geq 0$~~ else, ~~plot~~ next point is $(x_{k+1}, y_k - 1)$ and

$$P_{k+1} = P_k + 2x_{k+1} + 1 - 2y_{k+1}$$

4. Determine symmetry points in the other 7 octants.
5. Move each calculated pixel position (x, y) onto the circular path centered on (x_c, y_c) and plot the co-ordinates

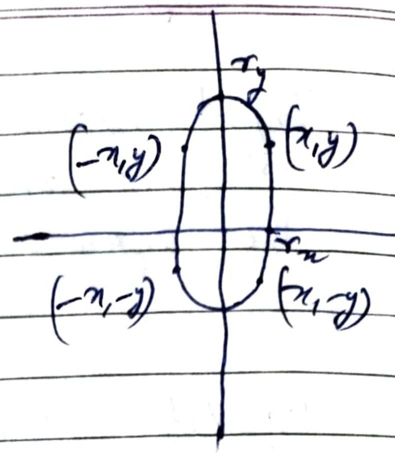
$$x = x + x_c, \quad y = y + y_c$$

6. Repeat steps 3 through 5 till $x < y$.

Page :
Date :

* Raster Scan - Ellipse
Mid point Algorithm

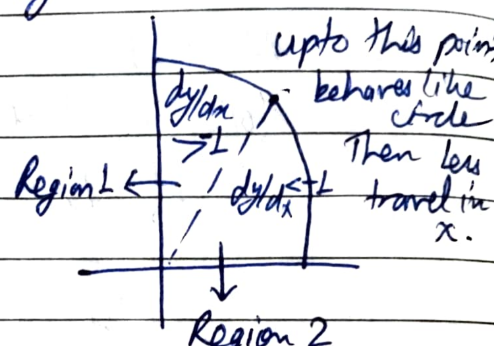
Equation: $\frac{x^2}{r_x^2} + \frac{y^2}{r_y^2} = 1$



It is symmetrical in the quadrants

$$f_{ellipse}(x,y) = r_y^2 x^2 + r_x^2 y^2 - r_x^2 r_y^2$$

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



In Region 1,
$(x_k, y_k) \longrightarrow (x_k+1, y_k)$ or (x_k+1, y_k-1)

Decision parameter in region 1

$$P1_k = f_{ellipse}(x_{k+1}, y_k - 1/2)$$

$$= r_y^2(x_k+1)^2 + r_x^2(y_k - 1/2)^2 - r_x^2 r_y^2$$

If $P1_k < 0$, plot point at y_k
else, plot point at $y_k - 1$

$$P1_{k+1} = r_y^2[(x_k+1)+1]^2 + r_x^2(y_{k+1} - 1/2)^2 - r_x^2 r_y^2$$

$$P1_{k+1} - P1_k = 2r_y^2(x_k+1) + r_y^2 + r_x^2\left[(y_{k+1} - 1/2)^2 - (y_k - 1/2)^2\right]$$

If $P1_k < 0$

$$P1_{k+1} - P1_k = 2r_y^2(x_k+1) + r_y^2$$

$$= 2r_y^2 \cdot x_{k+1} + r_y^2$$

Page :
Date :

If $P1_k \geq 0$

$$P1_{k+1} - P1_k = 2r_y^2(x_k+1) + r_y^2 + r_x^2\left[\left(y_{k+1} - \frac{1}{2} + y_k - \frac{1}{2}\right)\left(y_{k+1} - \frac{1}{2} - y_k + \frac{1}{2}\right)\right]$$

$$= 2r_y^2(x_k+1) + r_y^2 + r_x^2(2y_k - 2)(-1)$$

$$= 2r_y^2(x_k+1) + r_y^2 - 2r_x^2(y_k - 1)$$

$$= 2r_y^2 x_{k+1} + r_y^2 - 2r_x^2 y_{k+1}$$

At the starting point $(0, r_y)$

$$P1_0 = f_{ellipse}(1, r_y - 1/2)$$
$$= r_y^2 + r_x^2 r_y + \frac{1}{4} r_x^2$$

For region 2 (starting point is where $dy/dx = -1$)
We move in unit steps of y

$$P2_k = f_{ellipse}(x_k + 1/2, y_k - 1)$$
$$= r_y^2(x_k + 1/2)^2 + r_x^2(y_k - 1)^2 - r_x^2 r_y^2$$

if $P2_k > 0$ then select x_k
else ~~then~~ select $x_k + 1$

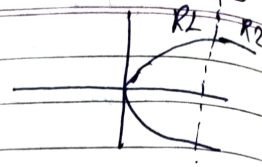
$$P2_{k+1} = r_y^2(x_{k+1} + 1/2)^2 + r_x^2[(y_k - 1) - 1]^2 - r_x^2 r_y^2$$

$$P2_{k+1} - P2_k = r_y^2\left[(x_{k+1} + 1/2) - (x_k + 1/2)^2\right] - 2r_x^2(y_k - 1) + r_x^2$$

If $P2_k > 0$, $x_{k+1} = x_k$
Then, $P2_{k+1} = P2_k - 2r_x^2(y_k - 1) + r_x^2$
else $P2_{k+1} = P2_k - 2r_x^2 y_{k+1} + r_x^2 + 2r_y^2 x_{k+1}$

Page :
Date :

$$P2_0 = f_{ellipse}(x_0 + 1/2,\ y_0 - 1)$$



* Raster Scan Hyperbola

$$f_{hyperbola}(x,y) = b^2x^2 - a^2y^2 - a^2b^2 \quad ; \ a > b$$

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

The region will change when $dy/dx = 1$

R1: midpoint $(x_k + 1/2,\ y_k + 1)$

$$P1_k = b^2(x_k + 1/2)^2 - a^2y_{k+1}^2 - a^2b^2$$

$$P1_{k+1} = b^2(x_{k+1} + 1/2)^2 - a^2(y_{k+1} + 1)^2 - a^2b^2$$

$$P1_{k+1} - P_k = b^2\left((x_{k+1} + 1/2)^2 - (x_k + 1/2)^2\right) - a^2\left((y_k + 2)^2 - (y_k + 1)^2\right)$$

If $P1_k > 0$ plot (x_k, y_{k+1})

$$P1_{k+1} - P_k = -a^2(2y_k + 3)$$

else

$$P1_{k+1} - P_k = -a^2(2y_k + 3) + 2b^2(x_k + 1)$$

$$P1_0 = f_{hyperbola}(a,0) = b^2(a + 1/2)^2 - a^2(0+1)^2 - a^2b^2$$

$$= \frac{b^2}{4} - a^2 + ab^2$$

R2:

(x_{k+1}, y_k) or (x_{k+1}, y_{k+1})

$$P2_k = f_{hyper}(x_k + 1,\ y_k + 1/2)$$

$$= b^2(x_k + 1)^2 - a^2(y_k + 1/2)^2 - a^2b^2$$

$$P2_{k+1} - P2_k = b^2(2x_k + 3) - a^2\left[(y_{k+1} - y_k)(y_{k+1} + y_k + 1)\right]$$

If $P2_k < 0$ $y_{k+1} = y_k + 1$

$$\therefore P2_{k+1} - P2_k = b^2(2x_k + 3) - 2a^2(y_k + 1)$$

$P2_0 = b^2(x_0+1)^2 - a^2(y_0+1/2)^2 - a^2b^2$

* Scan Conversion of Parabola using Bresenham like Algo.

$y^2 = 2px$

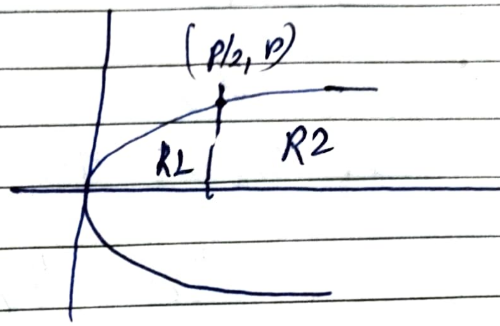
$\therefore$ The point $(P/2, P)$ where Region changes

$2y\frac{dy}{dx} = 2p$

$\frac{dy}{dx} = \frac{P}{y}$

if $p=y$, $dy/dx = 1$

$\therefore y=p \Rightarrow x = P/2$



$d1 = px - px_i$

$d2 = px_{i+1} - \cancel{Px}$

$\therefore d1_i = d1 - d2$

$= (y_{i+1})^2 - px_i - p(x_{i+1}$

$d1_i$ at $(0,0) = (0+1)^2 - p\cdot 0 - p(0+1)$

$= 1-p$

$d1_i$ at $(P/2, P) = 1+p$

If $d1_i \geq 0$ then $x_{i+1} = x_i + 1$
else $x_{i+1} = x_i$

$d1_i = (y_i+1)^2 - px_i - px_{i+1}$

$d1_{i+1} = (y_{i+1}+1)^2 - px_{i+1} - p(x_{i+1}+1)$

$d1_{i+1} - d1_i = 2(y_i+1) + 1 - 2p$

if $d1_i \leq 0$ then

$d1_{i+1} - d1_i = 2(y_i+1) + 1$

Region 2

$d1 = (y_i+1)^2 - y^2$

$d2 = y^2 - y_i^2$

$d2_i = d1 - d2$

$= (y_i+1)^2 - 2y^2 + y_i^2$

$= (y_i+1)^2 \cancel{-2y^2} + y_i^2 - 4p(x_i+1)$

If $d2_i < 0$ then $y_{i+1} = y_i + 1$

$d2_{i+1} = d2_i + 4(y_i+1) - 4p$

If $d2_i > 0$ then $y_{i+1} = y_i$

$d2_{i+1} = d2_i - 4p$

$d2_i(P/2, p) = 1 - 2p$

$d2_i - d1_i = y_i^2 - p(2x_i + 3)$

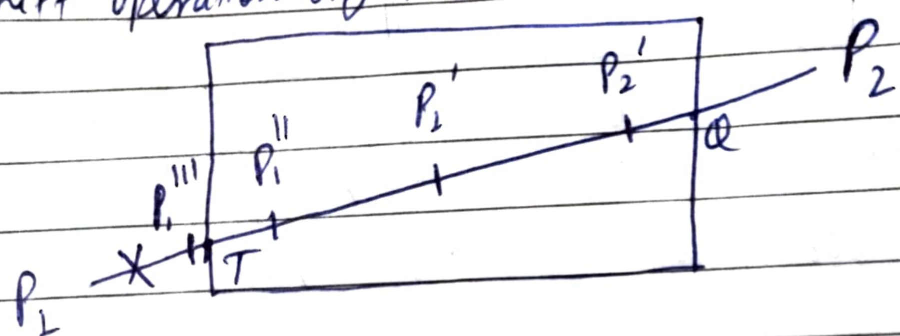
If $d1_i = 1$ then $d2_j = 1 - 4p$

If $d1_i = 1 + p$ then $d2_j = 1 - 2p$

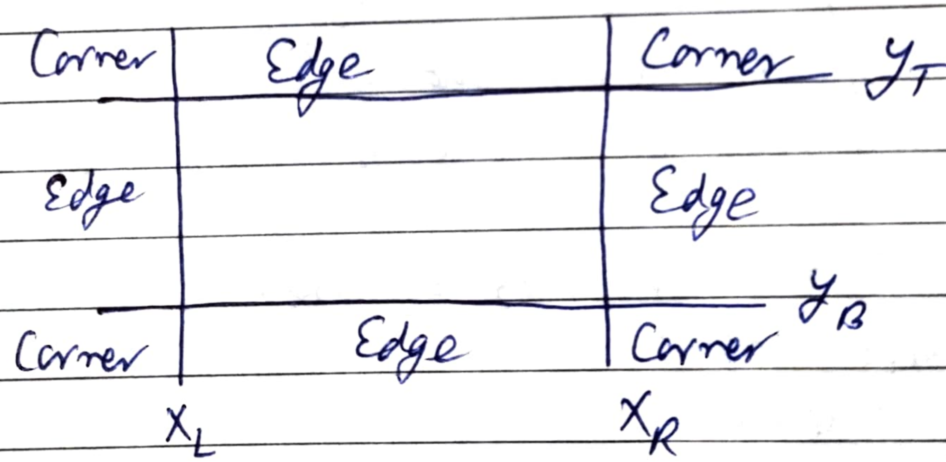
* Cohn - Sutherland Subdivision line clipping algorith

Advantage:-
H/w implementation is simple. Logical AND & right shift operation only needed.

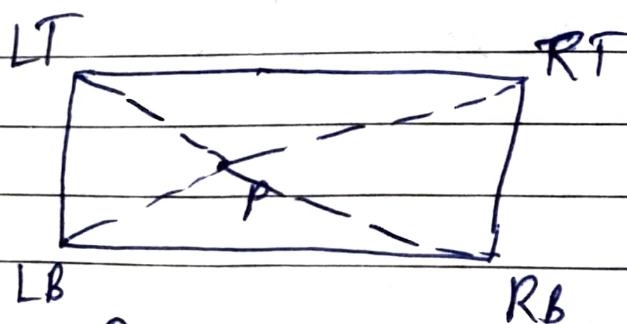
Continue halving until get T & Q



* Nicholi - lee - Nicholi Algo
→ Based on slope of line

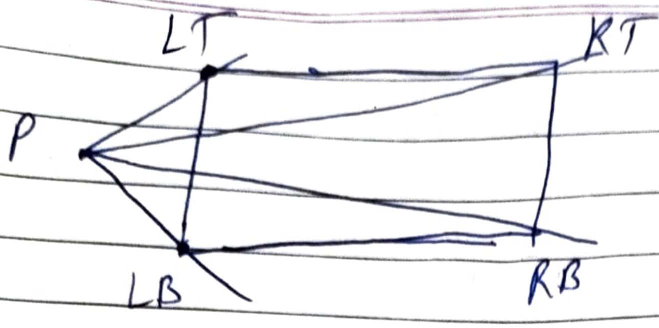


There can be 3 cases:-

① 

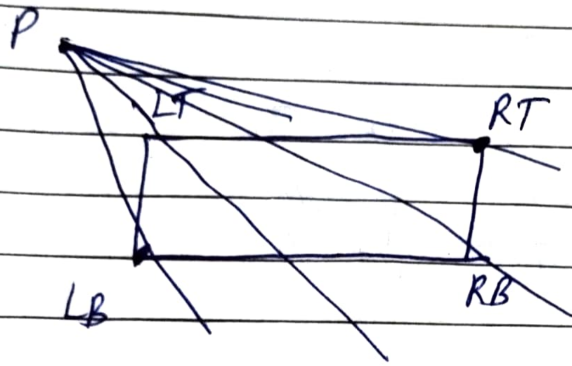
P is inside. Visible in any case (doesn't matter wh[ere] other endpoint is)

(2)

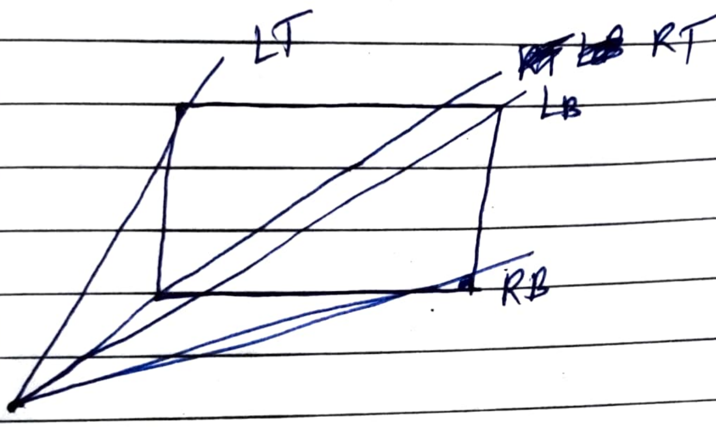


The line is only visible if other endpoint is b/w LT & LB.

(3)



The line is only visible if other endpoint is b/w LB & RT



* Liang-Barsky Line clipping

$P_1(x_1, y_1)$, $P_2(x_2, y_2)$

$x = x_1 + (x_2 - x_1)t \quad 0 \le t \le 1$

$y = y_1 + (y_2 - y_1)t$

For point (x, y) to be displayed,

$x_L \le x \le x_R$

$y_B \le y \le y_T$

$x_L \le x_1 + t\Delta x \le x_R$

$y_B \le y_1 + t\Delta y \le y_T$

$x_L - x_1 \leq t\Delta x$

$t\Delta x \leq x_R - x_1$

$y_B - y_1 \leq t\Delta y$

$t\Delta y \leq y_T - y_1$

$t P_k \leq q_k \qquad k = 1, 2, 3, 4$

$P_1 = -\Delta x$	$q_1 = x_1 - x_L$
$P_2 = \Delta x$	$q_2 = x_R - x_1$
$P_3 = -\Delta y$	$q_3 = y_1 - y_B$
$P_4 = \Delta y$	$q_4 = y_T - y_1$