6, February, 2024

TMC 401 Graphics and Visual Computing

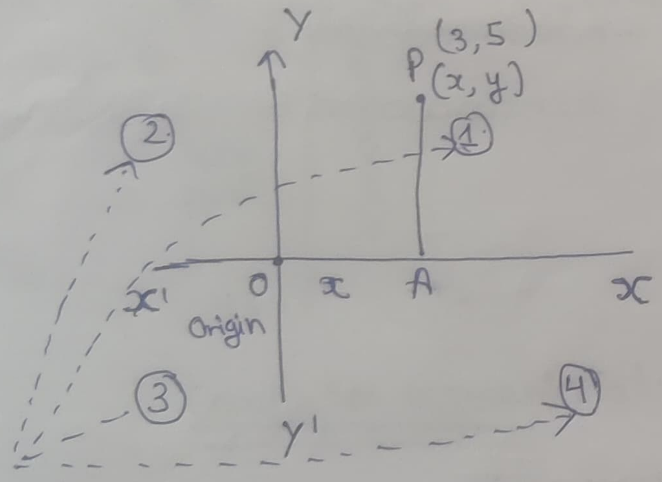
→ Computer graphics refers to a technology that generates images on a computer screen.

→ It is used in digital photography, film and television, video games, and on electronic devices and is responsible for displaying images effectively to users.

Screen / plain paper

Computer graphics is:-

Sim
→ stimulation of real life objects on ~~screen~~ computer screen.



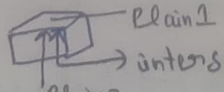
$x \in R$ {Real numbers}
$y \in R$ {Real numbers}
$(x, y) \in R \times R = R^2$

Quadrants

Notes to remember
→ point is dimension less
→ point can be infinite in a plain
→ We always need coordinate to identify point on the plane.

Equation of line

→ When two plain intersect with each other forms a line.



→ intersection of plain 1 & plain 2 forming a line in a square of rectangular wall made house.

Equation of line

→ $ax+by+c=0$

$y = mx+c$

gradient form

differentiate

$\frac{dy}{dx} = m$ ← this form is also called 1st order 1st degree

Order is recognized by power i.e 1

recognized by no differentiation

again differentiate

$\frac{d^2y}{dx^2} = 0$

$x \in R$

$y \in R$

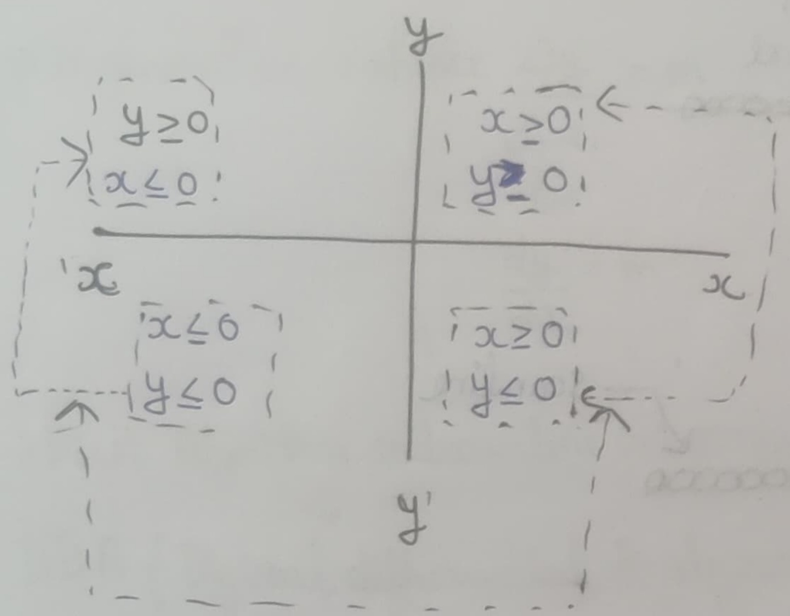
$(x,y) \in R \times R = R^2$

---

inequality $-1 \leq \sin x \leq 1 \ \forall x$

Left side | Right side

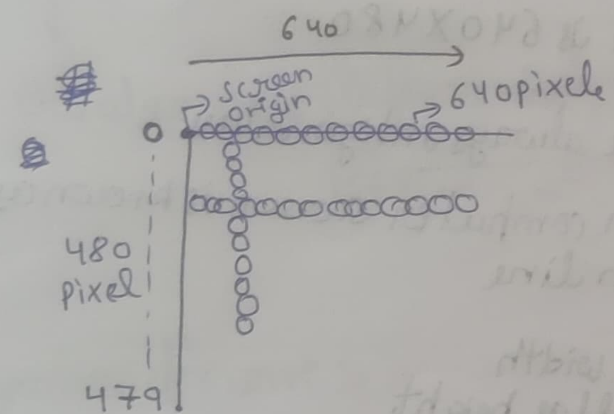
hence it is called 2 sided inequality



these are called inequalities

Note ⟶
∞ represents pixel here

Note



<u>Computer screen</u>'s matrix of pixel

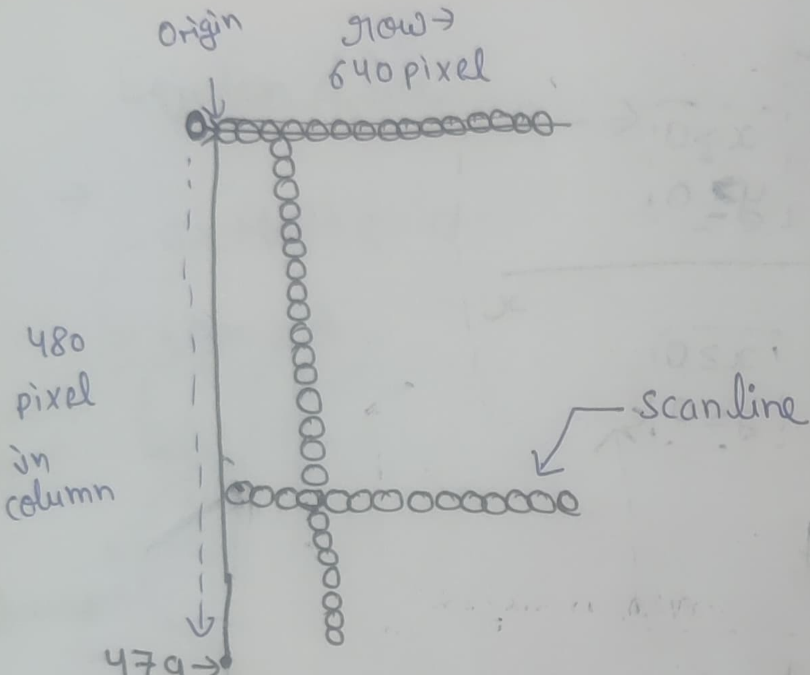
640X480

Computer screen ÷ is matrix {raster} of fixed surface

<u>Pixel</u> → Pixels are smallest geometrical entity which have width and height

Note:→ The no. of pixel horizontal in a line and no. of pixel in vertically column gives resolution of the screen.

→ Very common resolution of computer screen is 640X480

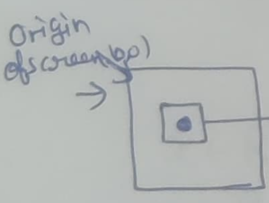


↓

so it's resolution is 640X480

→ pixel positions are always integer values.

→ A row of pixel in computer screen is known as:-
scan line



→ pixel has width as well as height

→ $\frac{dy}{dx}$ is limiting case of $\frac{\Delta y}{\Delta x}$

(i.e)

$$\frac{dy}{dx} = \lim_{\Delta x \to 0} \underbrace{\frac{\Delta y}{\Delta x}}_{\text{limitting case}} \quad \left\{ \begin{array}{l} \text{i.e} \\ y \text{ depends on} \\ \text{any change in } x \end{array} \right\}$$

→ There is no component line of –tive quadrant on computer screen.

→ in general we replace $\frac{\Delta y}{\Delta x} = m$

or

$\frac{dy}{dx} = m$

Imp → First Algorithm to draw line on computer screen.

↓

DDA {Digital differential Analyzer}

to draw line on computer screen

↓

this algo uses $\boxed{\text{Differential equation of line } \frac{\Delta y}{\Delta x} = m}$

equation of curve :→

→ line can be curve

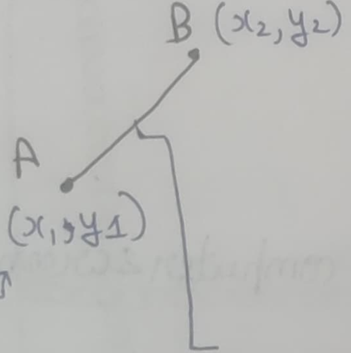
→

→ Circle → it is locus of varrying point that moves in such (सं) away that it distance from fixed point remain constant

Algorithm purposes :-

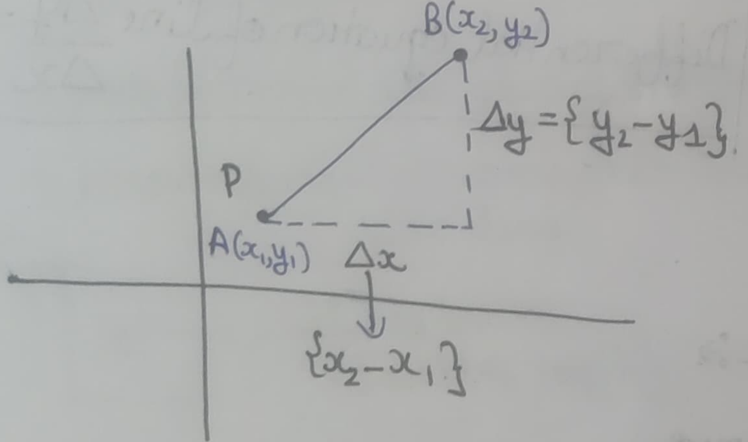
→ To identify pixel which are responsible for forming geometrical shape on computer screen.

→ To find a line uniquely we need to find two (2) distinct point on a line



→ there

— line segmentation —
{ if the line is between two points }

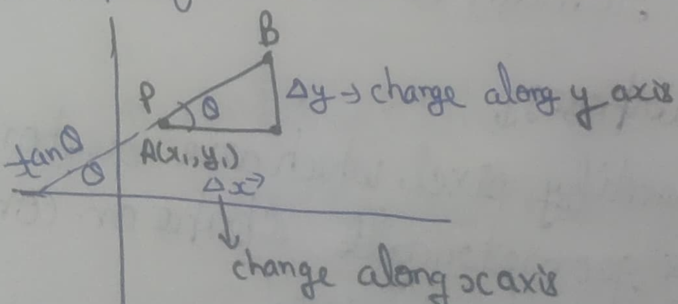


$\Delta x = (x_2 - x_1)$

$\Delta y = (y_2 - y_1)$

slope of the line

$\tan\theta$ of line which make angle from x axis is called slope of line

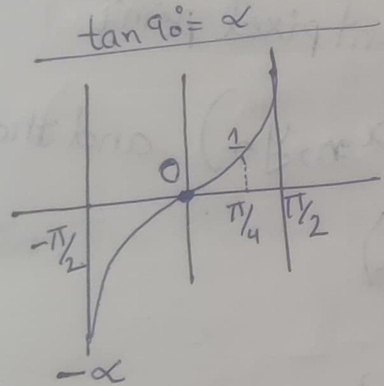


$$\tan\theta = \frac{\Delta y}{\Delta x} = m$$

$$\frac{y_2 - y_1}{x_2 - x_1} = m \{\text{slope}\}$$

→ remember pixel position are only integer on computer screen.

$\tan 90° = \infty$

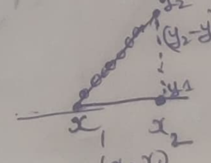


Note

$\boxed{\text{m stands for slope}}$

Case 1 : Now consider the case

↳ if $(0 < m < 1)$ and $(x_1 < x_2)$



Pixel → movement will be left to right

For a line segment

~~↳ pixel dot~~

if $(x_1 < x_2)$ starting point (x_1, y_1) (Left to right)

so

$$m = \frac{\Delta y}{\Delta x} < 1 = \Delta y < \Delta x$$

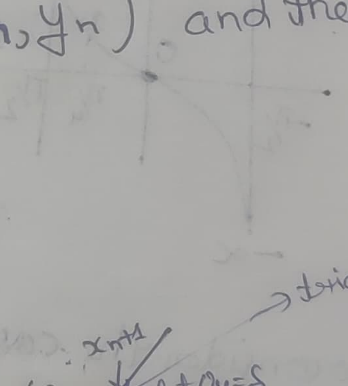
change in x will always be greater than y

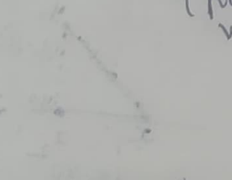
Since ~~Δx is $<\Delta y$~~ $\Delta y < \Delta x$, so at every step of pixel determination, the change in x coordinate, will be larger than the change in y coordinate. and the maximum possible increment which we can give to x or y coordinate will be 1

In this case we will give maximum possible increment of 1 to x coordinate

suppose we are at ~~current~~ pixel point (x_n, y_n) and the next position is (x_{n+1}, y_{n+1})

then, $x_{n+1} = x_n + 1$
i.e $\{$previous value $+ 1\}$





so slope is $\tan\theta = \frac{\Delta y}{\Delta x} = \frac{S}{1}$

and $m = \frac{S}{1}$ (according to figure)

$S = m$ (increment in y)

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

Thus we have $\begin{cases} x_{n+1} = x_n + 1 \\ y_{n+1} = y_n + m \end{cases}$

→ To get the pixel position coorresponding to a point we will use the round function.

ex:- Round(4.7) = 5

Round(4.29) = 4

Case 2: if $(m > 1)$

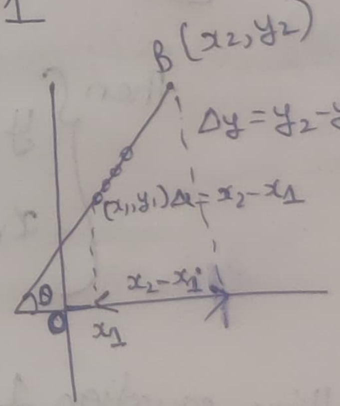
or slope is greater than 1 {Left to right} i.e $(x_1 < x_2)$

$$m = \frac{\Delta y}{\Delta x} > 1 \Rightarrow \Delta y > \Delta x$$

~~Then~~ Then the next position ~~is~~ in this case

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

Again from figure {acc. to figure} →



$$\tan\theta = m = \frac{1}{\sigma}$$

$m = \frac{1}{\sigma}$, where σ is the change in x coordinate at step of next position determination

then

$$x_{n+1} = x_n + \frac{1}{m}$$

$$\begin{cases} x_{n+1} = x_n + 1/m \\ y_{n+1} = y_n + 1 \end{cases}$$

Case(3) $(0 < m < 1)$ Right to left $(x_1 > x_2)$

$$m = \frac{\Delta y}{\Delta x} < 1$$

$$= \Delta y < \Delta x$$

then

Eqn $\begin{cases} x_{n+1} = x_n - 1 \\ y_{n+1} = y_n - m \end{cases}$

Case 4. $(m > 1)$ Right to left $(x_1 > x_2)$

$$\frac{\Delta y}{\Delta x} > 1 \Rightarrow \Delta y > \Delta x$$

then $\begin{cases} y_{n+1} = y_n - 1 \\ x_{n+1} = x_n - 1/m \end{cases}$

Using equation 1, 2, 3 and 4 as per case Algorithm determines the pixel position on computer screen and illuminate them, to generate the line on computer screen.

→ (means to show pixel on screen). Numerical

Q Digitize the line joining points $(\overset{x_1}{10}, \overset{y_1}{10})$ to $(\overset{x_2}{20}, \overset{y_2}{16})$ using DDA algorithm and plot the pixel on Cartesian grid?

Soln:- we have $(x_1, y_1) = (\overset{x_1}{10}, \overset{y_1}{10})$

$(x_2, y_2) = (\overset{x_2}{20}, \overset{y_2}{16})$

$(x_1 < x_2)$

$$\Delta x = x_2 - x_1 = 20 - 10 = 10$$

$$\Delta y = y_2 - y_1 = 16 - 10 = 6$$

(Slope) $m = \dfrac{\Delta y}{\Delta x} = \dfrac{6}{10} = 0.6$

Since

$(0 < m < 1)$ & $(x_1 < x_2)$ thus it is Case 1 of DDA Algorithm.

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

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

Now we will draw table

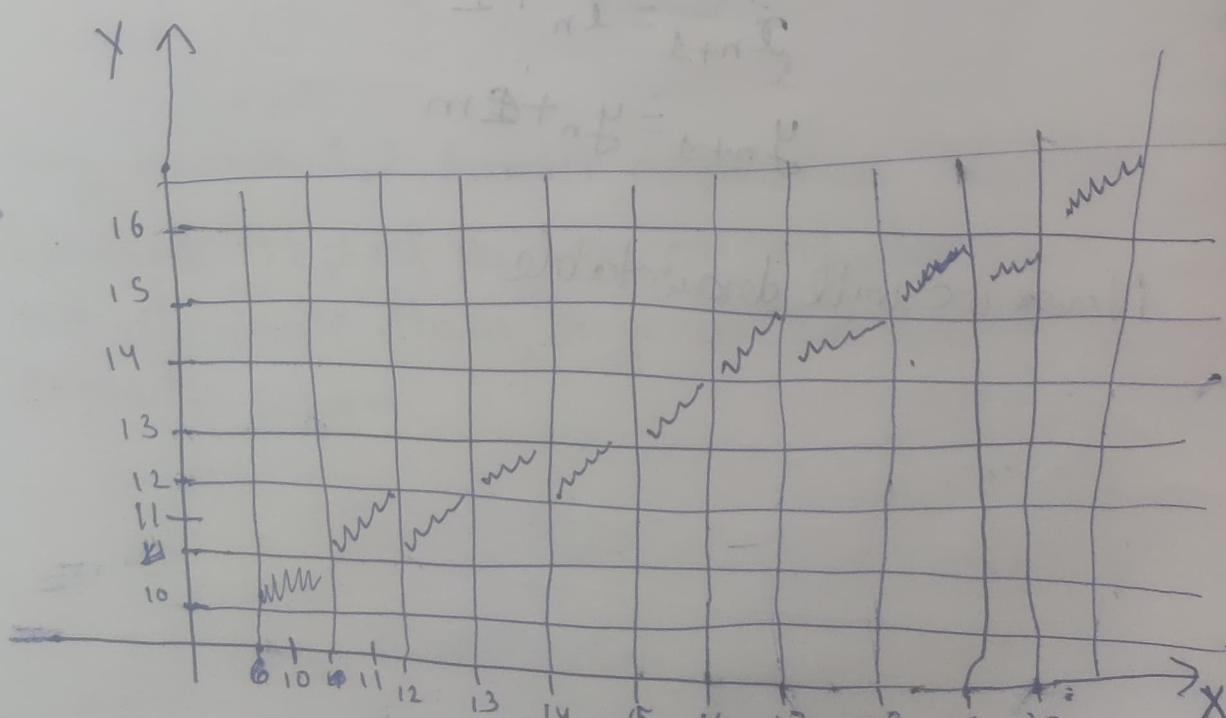
↙ current point, → current, → Next point

Bresenham line.

CP	C.pixel	N.P	N.Pixel
(10,10)	(10,10)	(11,10.6)	Round (11,11)
(11,10.6)	(11,11)	(12,11.2)	round (12,11)
(12,11.2)	(12,11)	(13,11.8)	round. (13,12)
(13,11.8)	(13,12)	(14,12.4)	(14,12)
(14,12.4)	(14,12)	(15,13.0)	(15,13)
(15,13.0)	(15,13)	(16,13.6)	(16,14)
(16,13.6)	(16,14)	(17,14.2)	(17,14)
(17,14.2)	(17,14)	(18,14.8)	(18,15)
(18,14.8)	(18,15)	(19,15.4)	(19,15)
(19,15.4)	(19,15)	(20,16)	(20,16)
(20,16)	(20,16)		

$$\begin{array}{r} 12.4 \\ 0.6 \\ \hline 13.0 \end{array}$$

$$\begin{array}{r} 13.6 \\ 0.6 \\ \hline 14.2 \end{array}$$

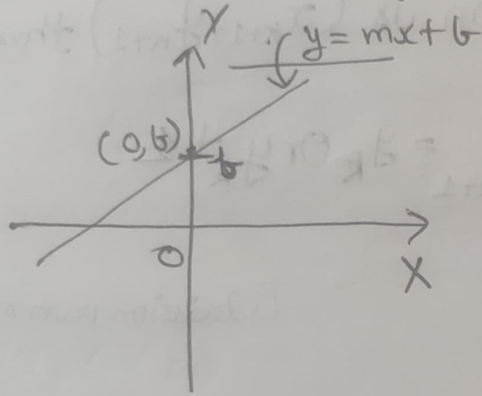


12/ Feb/2024

Breshamms line generating Algo.

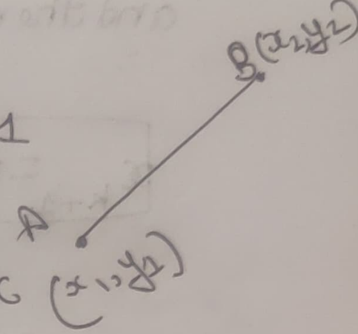
$(0<m<1)$

eqn of line $y = mx + b$



$m = \frac{\Delta y}{\Delta x} < 1$

$\Delta y < \Delta x$



→ In Bresenhem's algorithm the process of pixel determination is left to right i.e $\{x_1 < x_2\}$ x_1 is strictly less than x_2,

since m is less than one so $\Delta y < \Delta x$ at every step of pixel determination, to get the next pixel position, we will give the maximum possible increement of 1 to x coordinate, now we are only remaining with the problem for increement in y coordinate.

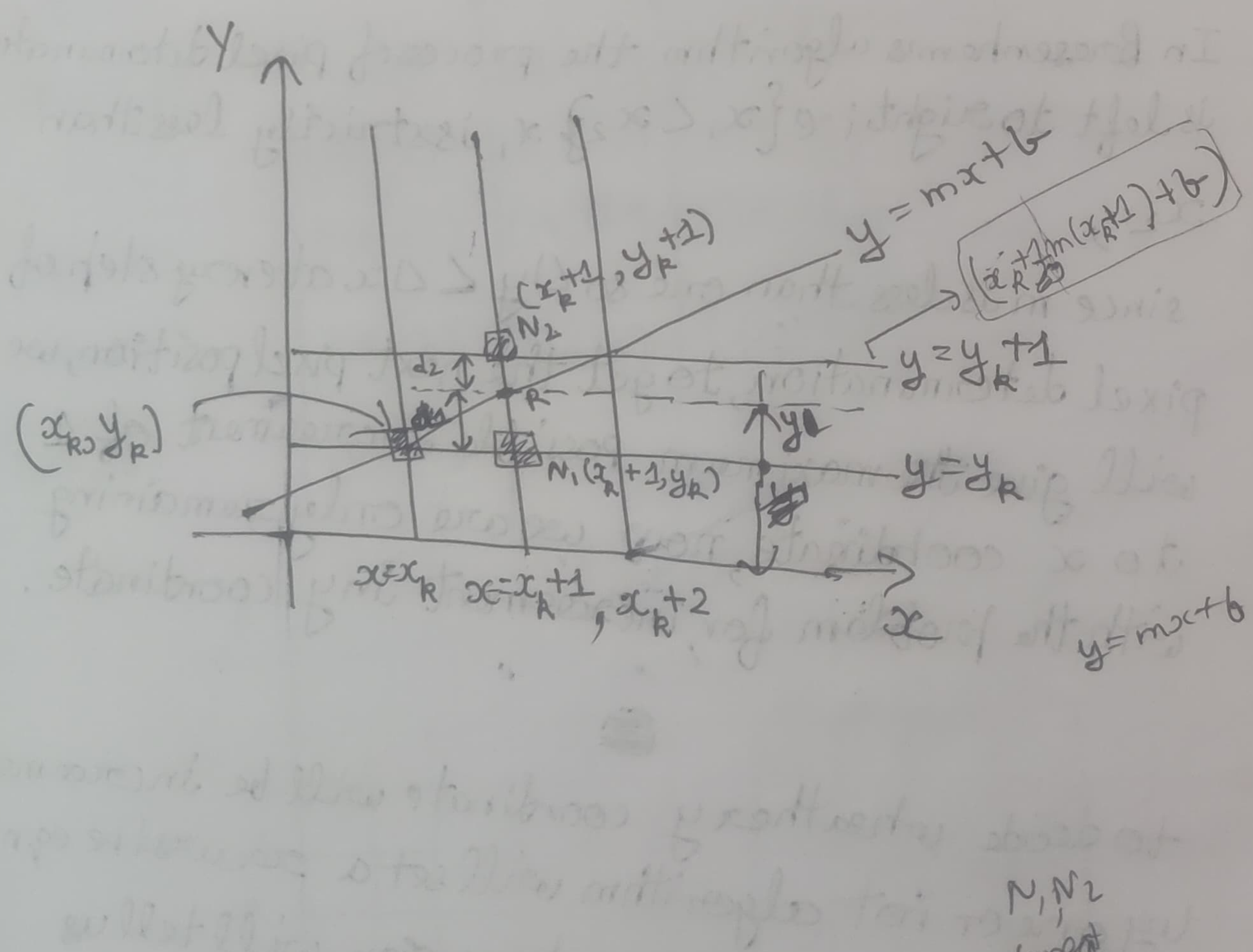
to decide wheather y coordinate will be increemented by one or not algorithm will set a recursive eqn. of decission parameter whose sign, will tell us about, the increement in y coordinate, Algo ÷

Let pixel (x_k, y_k) is to be displayed.

and the next pixel position is (x_{k+1}, y_{k+1}) then,

$\boxed{x_{k+1} = x_k + 1}$ and $y_{k+1} = y_k$ or $y_k + 1$

↳ decission parameter



N_1, N_2
↳ next possible pixels

then select that pixel which is near or closer to R i.e. N_1 or N_2 ⟵ R → near theoritical point {non-integer value}

From R is the rear theoritical point which is intersection of lines $y = mx + b$ and $x = x_k + 1$

→ cal the difference

$\boxed{\text{b/w } d_1 \,\&\, d_2 \text{ check +ve, -ve}}$

if $d_1 - d_2 = 0$,

$d_1 - d_2 = -ve$ d_1 is smaller than d_2

$d_1 - d_2 = +ve$ d_1 is ~~no~~ greater than d_2.

point of intersection (R)

has equality it satisfy equation of both eqn. of line.

$y = mx + b$ then

$y = m(x_k + 1) + b$

From figure ⇒

$$d_1 = RN_1 = y' - y_k = (m(x_k + 1) + b - y_k) \quad - ①$$

$$d_2 = RN_2 = y_k + 1 - y' = \{(y_k + 1) - m(x_k + 1) + b\} \quad ②$$

then, subtracting $\{d_1 - d_2\}$

$$d_1 - d_2 = \{m(x_k + 1) + b - y_k\} - \{(y_k + 1) - [m(x_k + 1) + b]\}$$

$$d_1 - d_2 = mx_k + m + b - y_k - y_k - 1 + mx_k + m + b$$

$$(d_1 - d_2) = 2mx_k - 2y_k + \underbrace{(2m + 2b - 1)}_{\text{its constant \{slope\}}}$$

$\Rightarrow d_1 - d_2 = 2mx_k - 2y_k + C$ $\{$ where
$C = 2m + 2b - 1$
i.e constant $\}$

Since (∵)

$m = \frac{\Delta y}{\Delta x}$

$\boxed{m = \frac{\Delta y}{\Delta x}}$

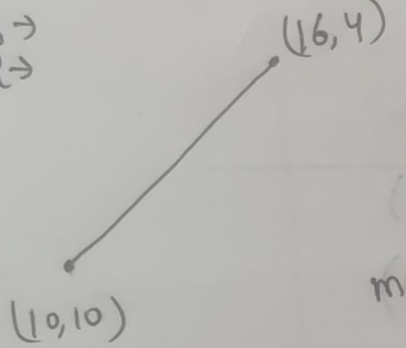
$\Rightarrow (d_1 - d_2) = \frac{2\Delta y \, x_k}{\Delta x} - 2y_k + C \rightarrow$ 3

Since $(\Delta x > 0)$
then, the sign of
$(d_1 - d_2)$ and $\Delta x (d_1 - d_2)$ will remain same.

Numerical Bresenham's line

Q Digitalize the line, Joining points (10,10) to (16,14) using Bresenham's line Algo rithym and plot points on cartesian grid?

Ans →
sol →



$\Delta x = 16 - 10 = 6$

$\Delta y = 14 - 10 = 4$

$m = \frac{\cancel{4}^{2}}{\cancel{6}_{3}} = \frac{2}{3} = 0.66$

$p_0 = \Delta x \times (d_1 - d_2) = \Delta x \{ m x_0 + m + b - y_0 - y_0 - 1 + m x_0 + m + b \}$

$p_0 = \Delta x \times \{ 2 m x_0 - 2 y_0 + 2m + 2b - 1 \}$

$p_0 = \Delta x . \{ 2 (m x_0 + b - y_0) + 2m - 1 \}$

Note: $y_0 = m x_0 + b$
↓
can be written as
$m x_0 + b - y_0 = 0$

$p_0 = \Delta x \{ 2 \frac{\Delta y}{\Delta x} - 1 \}$

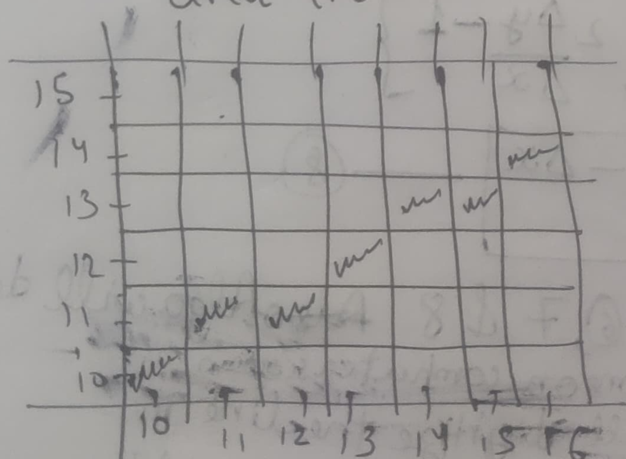
$\boxed{p_0 = 2 \Delta y - \Delta x}$ —— (8)

By using eqn (6), 7 & 8 ~~Algo~~ Algo will determine the pixel positions on computer screen and this will digitize the line (show on screen)

If the p_k value is +ve then value of both x and y will be incremented by 1, but if the p_k value is -tive then the value of x will be increased by 1 but not in y.

Steps	C.P	p_k	N.P
	(10,10)	$p_0 = 2 > 0$	(11,11)
	(11,11)	$p_1 = 2+8-12 = -2$	(12,11)
	(12,11)	$p_2 = 2+8 = 6$	(13,12)
	(13,12)	$p_3 = 6+8-12 = 2$	(14,13)
	(14,13)	$p_4 = 2+8-12 = -2$	(15,13)
	(15,13)	$p_5 = -2+8-12 = 6$	(16,14)

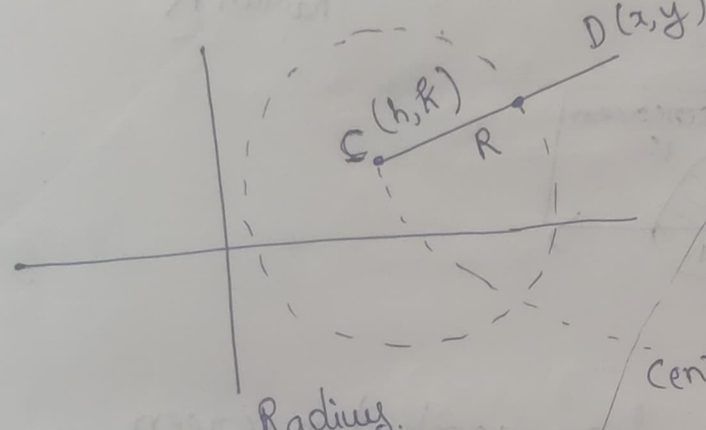
Now we have to plot Current point in this and the last next points.



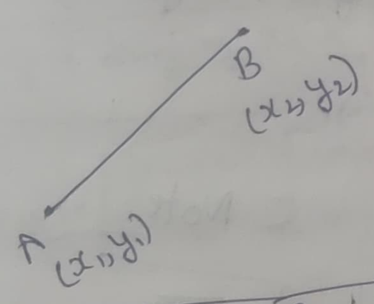
HW → (10,10) to 20,18)

Circle

it is the locus of point which moves in such a way that it's distance from fixed point always remain constant.



$R = \sqrt{(x-h)^2 + (y-k)^2}$



$d(A,B) = \sqrt{(x_2-x_1)^2 + (y_2-y_1)^2}$

$R^2 = (x-h)^2 + (y-k)^2$

$R^2 = x^2 - 2hx + h^2 + y^2 - 2ky + k^2$

$x^2 + y^2 - 2hx - 2ky + (h^2 + k^2 - R^2) = 0$

$-h \equiv g$

$-k \equiv f$

$\therefore$

$x^2 + y^2 + 2gx + 2fy + c = 0$

General eqn. of circle which lies of x, y plane

Circle with Centre

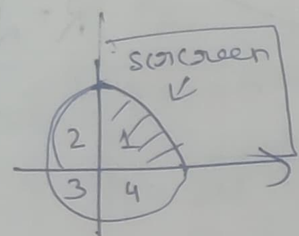
at O $(h=0, k=0)$

eqn $R^2 = x^2 + y^2$

$\hookrightarrow x^2 + y^2 = R^2$ $\{(0,0)\}$

$\downarrow$

Radius R

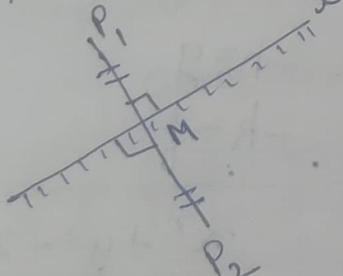


Generation of circle on computer screen.

Note ↓

To understand this we have understand few things about reflection

Symmetry.



Note

$\overline{P_1P_2}$

$\downarrow$

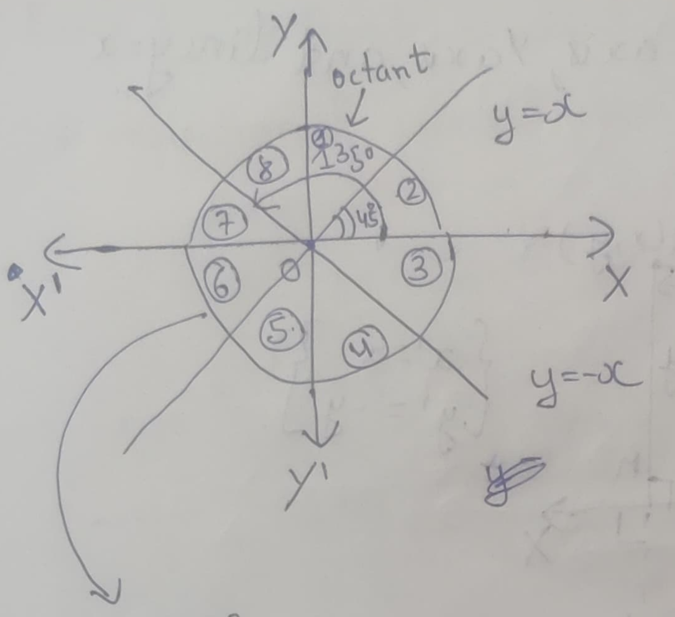
this means there is a line b/w P_1 & P_2

~~If $P_1M = P_2M$ then~~

P_1 & P_2 in figure are the symmetric point about line l

$\overline{P_1P_2} \perp l$

where $P_1M = P_2M$



$x^2+y^2=R^2$

Note
if $m_1 \& m_2 = -1$
then $\theta = 90°$
line will intersect at

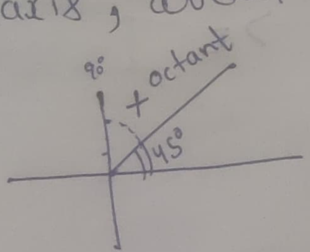
$y = mx$
$m = \tan\theta$
$m = 1, \theta = 45°$
$y = x$
$y = -x$

⇒ The circle $x^2+y^2=R^2$, is symmetrical about x axis, y axis about the line $y=x$ and line $y=-x$

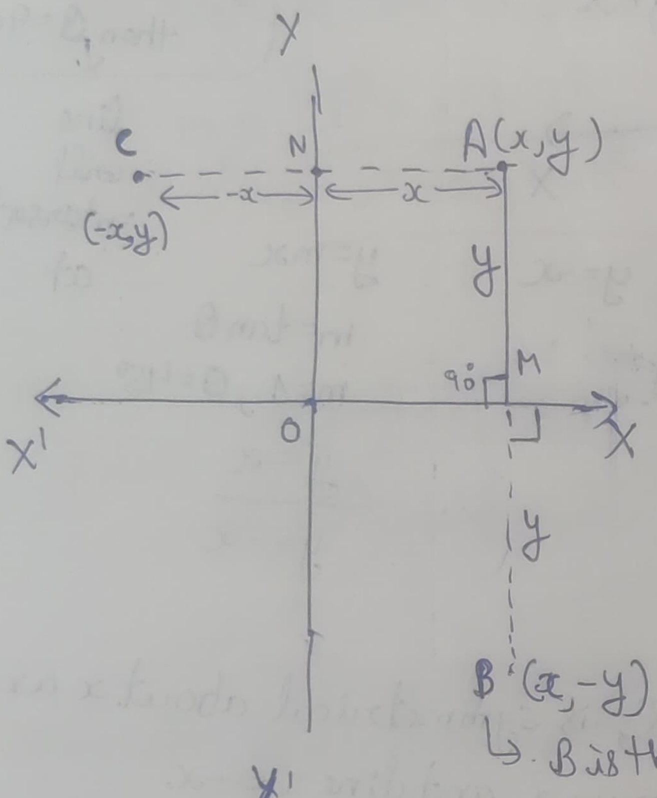
Note self → ~~dividing~~ The X line divide circle into 8 pieces.

Note self.
↳ we will compute the pixel by 1 octant
rest all we will form using symmetry.

⇒ While generating circle on computer screen the algorithm will determine the pixel only in one octant. $45° \leq \theta \leq 90°$, remaining 7 part of the circle will be generated by taking reflection of those pixel about x axis, y axis, about line $y=x$ and line $y=-x$



Reflection about x axis, Y axis, and line $y=x$

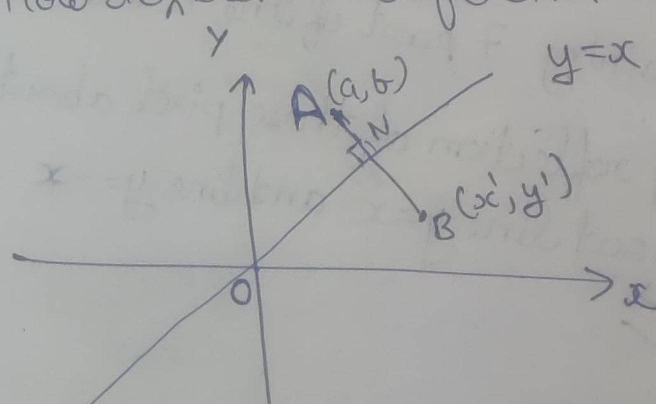


$$\begin{Bmatrix} x' = x \\ y' = -y \end{Bmatrix}$$

↳ B is the reflection of A about x axis.

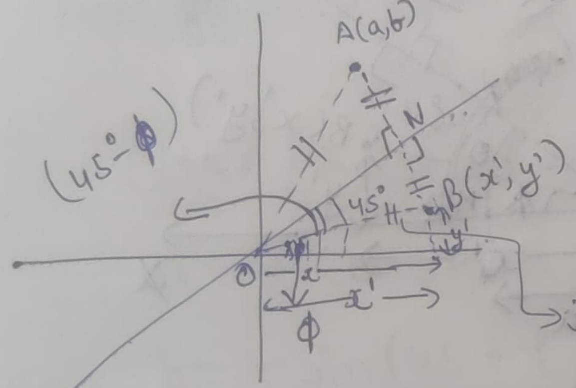
From figure point $A(x,y)$ it's reflection about x axis is given by $B(x,-y)$, also point C minus (x,y) is the refection of point A { about y axis }

How to find ~~get~~ the reflection about the line $y=x$



$AN = BN$

From figure A(a,b) is a point on x y plane & B(x', y') is reflection of point A about the line y = x



it is right angle triangle

so we can apply pythagorus theorem

AN = BN

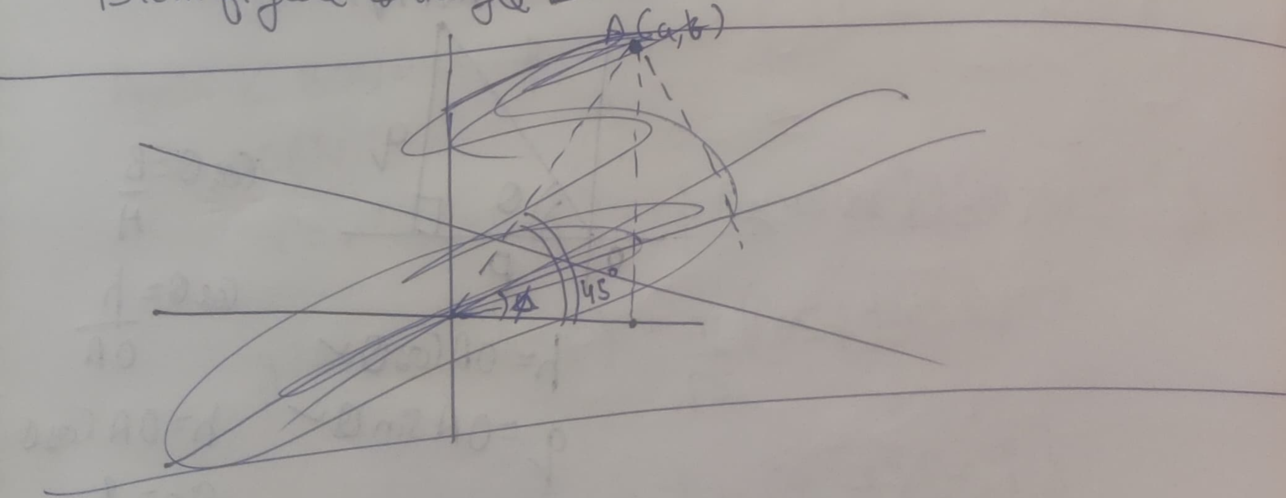
$OA^2 = ON^2 + AN^2$

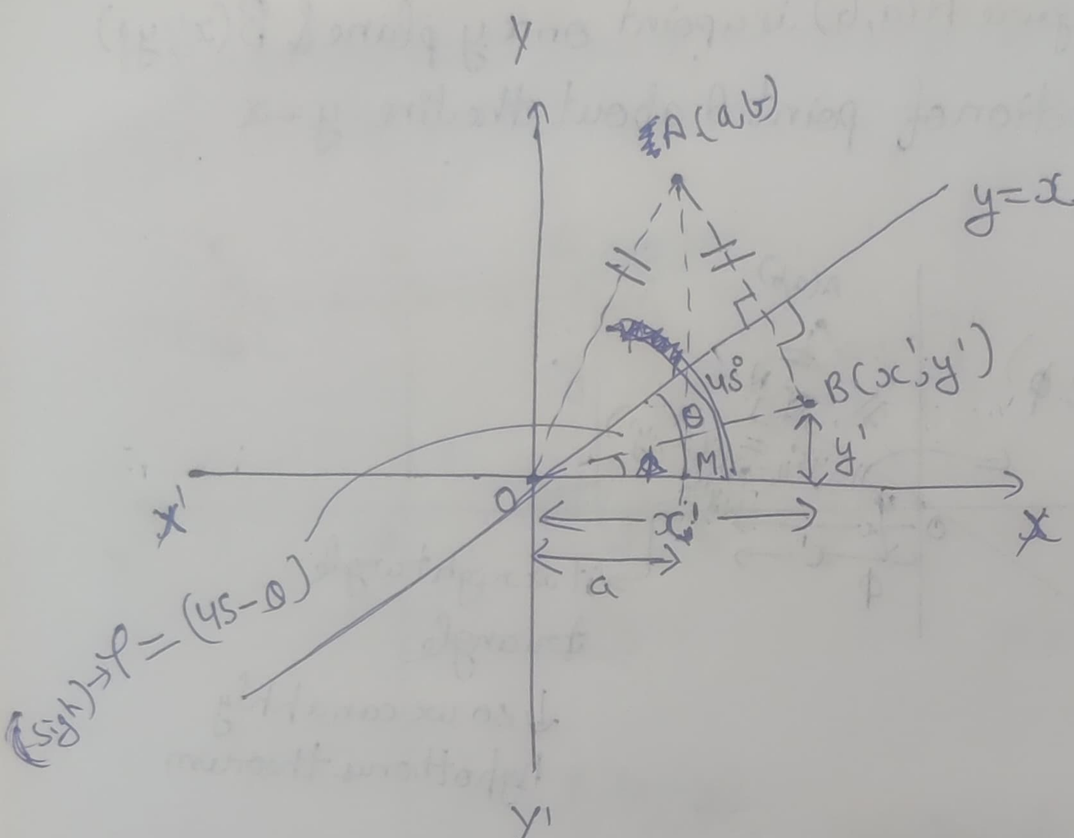
$= ON^2 + BN^2 = OB^2$

OA = OB = r (Say sights)

~~if tri~~ concurrent = { i.e two sides of triangle are same }

From figure triangle $\Delta OAN \cong \Delta OBN$





$A = BN$

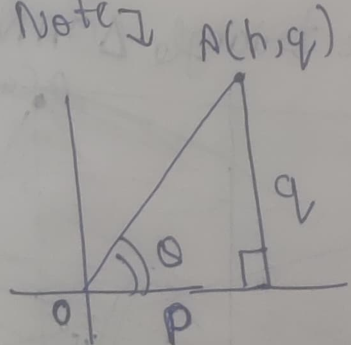
$OA^2 = ON^2 + AN^2$

$= ON^2 + BN^2 = OB^2$

$OA = OB = r$ (sigh)

(says)

Note]



$\cos\theta = \frac{B}{H}$

$\cos\theta = \frac{h}{OA}$

$h = OA\cos\theta$ ✓

$q = OA\sin\theta$ ✓

$h = OA\cos\theta$

$\sin = \frac{q}{OA}$

$q = OA\sin\theta$

$$\left\{\begin{array}{l}\cos(A\pm B)=\cos A\cos B\pm\sin A\sin B\\ \sin(A\pm B)=\sin A\cos B\pm\cos A\sin B\end{array}\right\}$$

H.W

Assignment
use plain geometry to find (x', y') i.e reflection

From figure: in ΔOAM

$$OM = r\cos(45+\varphi)$$

$$a = r\cos(45+\varphi)$$

$$b = r\sin(45+\varphi)$$

$$\Rightarrow a = r\{\cos 45\varphi - \sin 45\sin\varphi\}$$

$$a = \frac{r}{\sqrt{2}}(\cos\varphi - \sin\varphi) \quad \text{—} \quad (1)$$

$$b = r(\sin 45\cos\varphi + \cos 45\sin\varphi\}$$

$$b = \frac{r}{\sqrt{2}}\{\cos\varphi + \sin\varphi\} \quad \text{—} \quad (2)$$

Again $\angle BON = 45-\varphi$

in ΔOBN

$$x' = r\cos(45-\varphi) = r\{\cos 45\cos\varphi + \sin 45 \cdot \sin\varphi\}$$

$$y' = r\sin(45-\varphi) = \frac{r}{\sqrt{2}}(\cos\varphi + \sin\varphi) = b$$

$$y' = r\{\sin 45^\circ\cos\varphi - \cos 45^\circ\sin\varphi\}$$

$$y' = \frac{r}{\sqrt{2}}(\cos\varphi - \sin\varphi) = a$$

$$x' = b \quad y' = a$$

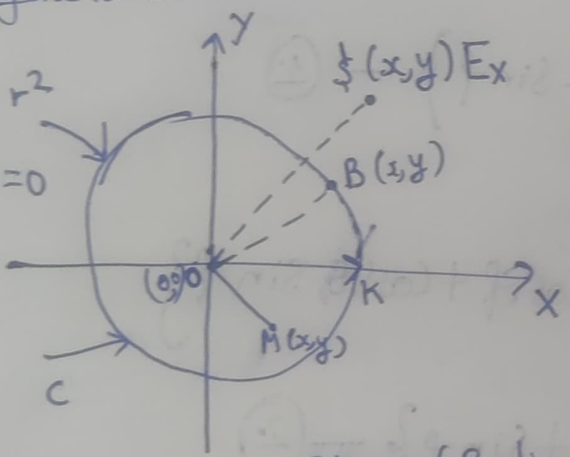
The reflection of poin $A(a,b)$ about line $y=x$ is $B(b,a)$

Date→ 19,2,2024

Circle generation:-

$x^2+y^2=r^2$

$x^2+y^2-r^2=0$



OK= r (Radius)

$$\boxed{f(x,y)=x^2+y^2-r^2}$$

for INT Point $M(x,y)$

$OM=\sqrt{x^2+y^2}$

$OM<r$

$OM^2<r^2$

$x^2+y^2<r^2$

$x^2+y^2-r^2<0$

$f(x,y)<0$ for interior points

~~$f(x$~~

for exterior point $S(x,y)$

$OS>r$

$OS^2>r^2$

$x^2+y^2>r^2$

$\Rightarrow x^2+y^2-r^2>0$

$f(x,y)>0$

Boundary point

$OB=r$

$OB^2=r^2$

$x^2+y^2-r^2=0$

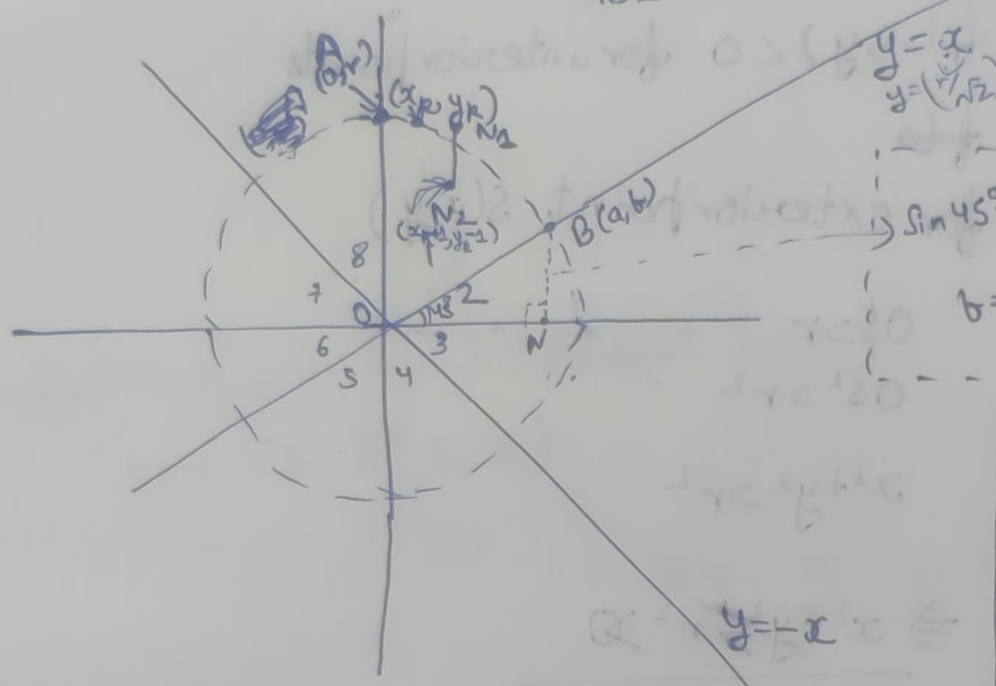
for boundary points

$f(x,y)=0$

The Bresenham's Circle generation Algorithm:-

$45° \leq \theta \leq 90°$

$45^\circ \le \theta \le 90^\circ$



$\sin 45^\circ = \frac{b}{r}$

$b = \frac{r}{\sqrt{2}}$

$B(a,b) = \frac{r}{\sqrt{2}}$

Note-> $\boxed{\sqrt{2} = 1.414}$

$\frac{1}{\sqrt{2}} = 0.707$

$\Delta x = \frac{r}{\sqrt{2}} - 0 = \frac{r}{\sqrt{2}} = 0.707 \times r$

$\Delta y = \frac{r}{\sqrt{2}} - r = -\left(1 - \frac{1}{\sqrt{2}}\right) r$

$= -(1 - 0.707) r = -0.293 \times r$

hence $\Delta x > \Delta y$ {Δx magnitude wise greater than Δy

i.e $|\Delta x| > |\Delta y|$

Since the total change in x value is larger than the change in y value while moving point ~~at~~ A to B along the arm of the circle so, Algorithm will give maximum possible increament of 1 to x value, while determining the next pixel position.

→ Now we are only remaining with the y coordinate, wheather y coordinate will be decreemented by one or not.

→ To decide this algorithm we will develop a recursive equation about decission parameter whose sign will tell us wheather y value will be decreemented by one or not.

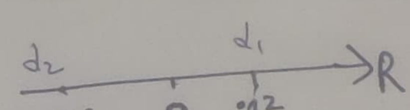
Let pixel (x_k, y_k) is to be displayed, the next pixel position is (x_{k+1}, y_{k+1}), where $x_{k+1} = x_k + 1$

$$\text{and } y_{k+1} = y_k \text{ or } y_k - 1$$

i.e The possible pixel positions

$$N_1 = (x_k + 1, y_k) \text{ or } N_2 = (x_k + 1, y_k - 1)$$

Out of these two pixel positions algorithm will select that pixel, which will be much close to the circumference of the circle



magnitude

$|.12| = 0.12$

$|-0.23| = 23$

Now

$d_1 = f(x,y)$ at N_1 (exterior)

$$d_1 = ((x_k+1)^2 + y_k^2 - r^2)$$ N_1 is exterior

$(d_1 > 0)$

Again, $d_2 = f(x,y)$ at N_2

$$d_2 = (x_k+1)^2 + (y_k-1)^2 - r^2$$, N_2 is interior

$(d_2 < 0)$

then setting

$p_k = d_1 + d_2$, where p_k is the decission parameter (d_h)

for k^{th} step.

$p_k = d_1 + d_2$

if $|d_1| \geq |d_2|$ ← positive value magnitude; ← negative value magnitude.

$p_k \geq 0$

or

if $|d_1| < |d_2|$

$p_k < 0$

if $p_k < 0$, then ~~the next~~ the next pixel positions is N_1, N_1 is close to circumference.

ie $y_{k+1} = y_k$

If $p_k \geq 0$, the next will be N_2

$y_{k+1} = y_k - 1$

20/2/2024

Bresenham's Circle Algorithm

$\Delta x > \Delta y$

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

$y_{k+1} = y_k$ or $y_k - 1$

$d_1 = f(x, y) > 0$ at N_1

$d_2 = f(x, y) < 0$ at N_2

Now $p_k = d_1 + d_2 = 2(x_k+1)^2 + y_k^2 + (y_k-1)^2 - 2r^2$ — ①

p_k is decission parameter for k^{th} steh so we canobtain, value of p_k+1 by substituting $k+1$, for k in eq ①

$$p_{k+1} = 2(x_{k+1}+1)^2 + y_{k+1}^2 + (y_{k+1}-1)^2 - 2r^2 \quad - ②$$

subtracting eqn 1 from eqn 2

$$p_{k+1} - p_k = 2\left\{ (x_{k+1}+1)^2 - (x_k+1)^2 \right\}$$
$$+ \left(y_{k+1}^2 - y_k^2 \right) + \left(y_{k+1} - 1 \right)^2$$
$$- (y_k - 1)^2$$

$$p_{k+1} - p_k = 2\left\{ (x_k+2)^2 - (x_k+1)^2 \right\}$$
$$+ \left\{ y_{k+1}^2 - y_k^2 \right\} + \left\{ y_{k+1}^2 - 2y_{k+1} + 1 \right\}$$
$$- \left\{ y_k^2 - 2y_k + 1 \right\}$$

$$p_{k+1} - p_k = 2\left\{ \cancel{x_k^2} + 4x_k + 4 - \cancel{x_k^2} - 2x_k - 1 \right\}$$
$$+ \left\{ y_{k+1}^2 - y_k^2 \right\} + \left\{ y_{k+1}^2 - y_k^2 \right\}$$
$$- 2\left\{ y_{k+1} - y_k \right\}$$

$$p_{k+1} - p_k = 4x_k + 6 + 2\left\{ y_{k+1}^2 - y_k^2 \right\} - 2\left\{ y_{k+1} - y_k \right\} \quad \text{--- eqn } 3$$

if p_k is negative $y_{k+1} = y_k$

i.e

if $p_k < 0$, $y_{k+1} = y_k$

then,

$$\boxed{p_{k+1} - p_k = 4x_k + 6}$$

eqn

$$\boxed{p_{k+1} = p_k + 4x_k + 6} \rightarrow 6$$

If $p_k \geq 0$, $y_{k+1} = y_k - 1$

$45° < \theta \leq 90°$

$$p_{k+1} = p_k + 4x_k + 6 + 2\{(y_k - 1)^2 - y_k^2\}$$
$$-2\{y_k - 1 - y_k\}$$

$$p_{k+1} = p_k + 4x_k + 6 + 2\{y_k^2 - 2y_k + 1 - y_k^2\}$$
$$+2$$

$$\boxed{p_{k+1} = p_k + 4x_k - 4y_k + 10} - 5$$

, when $p_k \geq 0$

eqn. 4 and 5 are helpful to determine the value of decission parameter for different steps, since, the degree of recursive eqn. is 1, so we need one initial value of decission parameter.

To get the initial value of decission parameter.

$$\therefore p_R = d_1 + d_2 = 2(x_R+1)^2 + (y_R-1)^2 + y_R^2 - 2r^2$$

The initial point is $A(0, r)$, so to get p_0 putting $x_R = 0$ and $y_R = r$ in eqn (1)

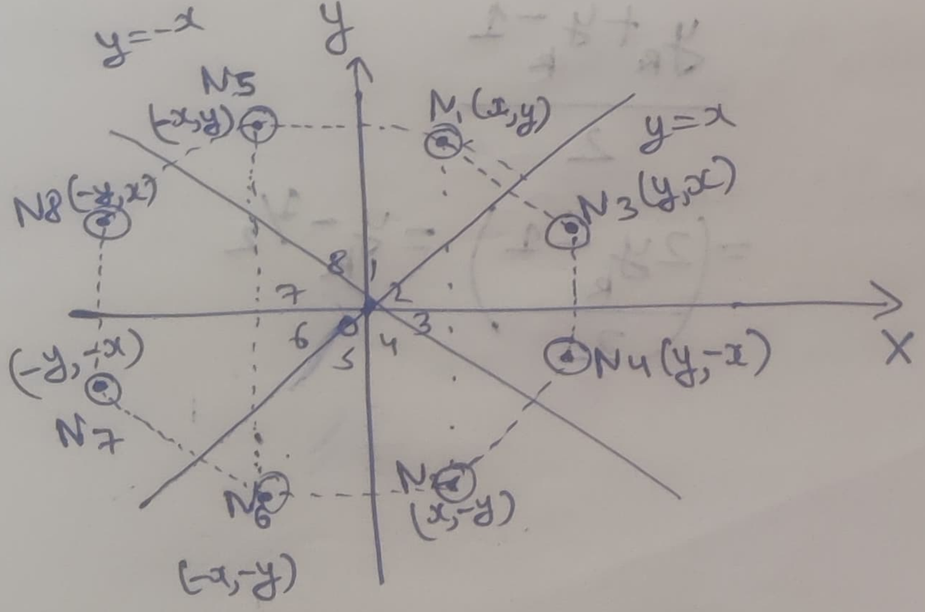
$$p_0 = 2 + (r-1)^2 + r^2 = 2r^2$$

$$p_0 = 2 + \cancel{r^2} - 2r + 1 + \cancel{r^2} - 2\cancel{r^2}$$

$$\boxed{p_0 = 3 - 2r} \quad (6)$$

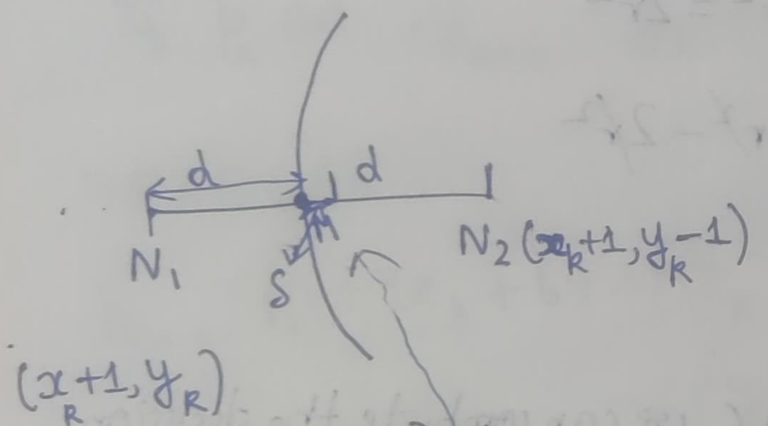
By using eqn 4, 5, and 6 we can compute the decission parameter for all the steps and thus, in position, to determinite all the pixel in, the octant ~~θ~~

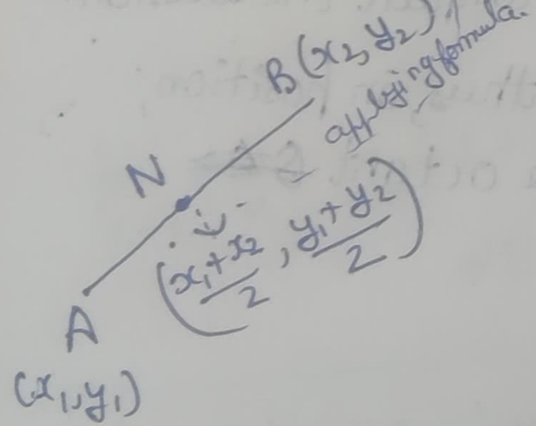
$$45^\circ \le \theta \le 90^\circ$$



Midpoint circle algorithm

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





$$M = \left(x_k + 1,\ y_k - \tfrac{1}{2}\right)$$

$$\frac{y_k + y_k - 1}{2}$$

$$= \left(\frac{2y_k - 1}{2}\right) = y_k - \tfrac{1}{2}$$

Also $x_{k+1} = x_k + 1$ $(\Delta x > \Delta y)$

Let pixel (x_k, y_k) is to displayed

then $N_1(x_k+1, y_k)$ or $N_2(x_k+1, y_k-1)$

To decide the next pixel position we need a decission parameter whose sign will tell us about the next pixel.

Let d.p (decission parameter) for k^{th} step is p_k

$p_k = f(x,y)_{at\,M}$, where M is midpoint of $N_1 N_2$

$M = (x_k+1, y_k - 1/2)$

$p_k = (x_k+1)^2 + (y_k - \cancel{1/2}\, 1/2)^2 - r^2$ — (1)

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

and if $p_k \geq 0$ then $y_{k+1} = y_k - 1$

To get the recursive eqn. for decission parameter we will substitute k+1 for k in eqn. 1.

$$p_{k+1} = (x_{k+1}+1)^2 + (y_{k+1} - 1/2)^2 - r^2 \quad \text{— eqn } 2$$

Now subtracting eqn. 1 from eqn. 2

$$p_{k+1} - p_k = (x_{k+1}+1)^2 - (x_k+1)^2 + (y_{k+1} - 1/2)^2 - (y_k - 1/2)^2$$

$$= (x_k+2)^2 - (x_k+1)^2 + \left(y_{k+1}^2 - y_{k+1} + \frac{1}{4}\right) - \left(y_k^2 - y_k + \frac{1}{4}\right)$$

Note applying $\boxed{(a-b)^2 = a^2+b^2-2ab}$

$$= x_k^2 + 4x_k + 4 - x_k^2 - 2x_k - 1 + (y_{k+1}^2 - y_k^2) - (y_{k+1} - y_k)$$

$$= p_{k+1} - p_k = 2x_k + 3 + (y_{k+1}^2 - y_k^2) - (y_{k+1} - y_k) \quad \text{— eqn. } 3$$

if p_k is negative

if $p_k < 0$, $p_{k+1} - p_k = 2x_k + 3 + 0 + 0$

$$\boxed{p_{k+1} = p_k + 2x_k + 3} \quad \text{if } p_k < 0 \quad \text{— eqn. } 4$$

if $p_k \geq 0$, $y_{k+1} = y_k - 1$

then

$$\Rightarrow p_{k+1} - p_k = 2x_k + 3 + \{(y_k - 1)^2 - y_k^2\}$$
$$- \{y_k - 1 - y_k\}$$

$$= 2x_k + 3 + \{y_k^2 - 2y_k + 1 - y_k^2\} + 1$$

$$p_{k+1} - p_k = 2x_k + 3 - 2y_k + 2$$

$$\boxed{p_{k+1} = p_k + 2x_k - 2y_k + 5}$$ ~~the~~ when $p_k \geq 0$ — eqn. 5

the degree of recursive eqn. 4 and 5 is 1, so we need one initial value of decission parameter.

To get p_0 A(0, r) innitial point
putting $x = 0, y = r$ in eqn. 1

$$p_0 = 1^2 + (r - 1/2)^2 - r^2$$

$$p_0 = 1 + r^2 - r + \frac{1}{4} - r^2$$

$$\boxed{p_0 = \frac{5}{4} - r} \quad (6)$$

Using eqn. 4, 5, and 6, we can determine the decissionparameter for eachsteh of next pixel determination and thus we can generate acircle on computer screen.

To generate the whole circle algorithm, will take, reflection of all the pixel in octant $45° \leq \theta \leq 90°$, about the line $y=x$, $y=-x$, about x axis & y axis.

Q Determine the pixel in an octant for circle $x^2+y^2=100$, using mid point circle algorithm and plot those pixels in cartesian grid graph.

Soln. eqn of Circle

$$x^2+y^2=10^2$$

$$r=10$$

Then d.p

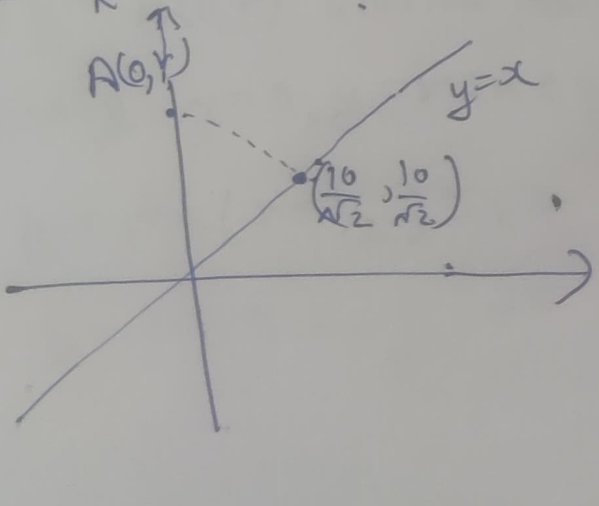
$$p_0=\frac{5}{4}-r$$

$$p_{k+1}=p_k+2x_k+3 \qquad p_k<0$$

i.e

$$p_{k+1}=p_k+2x_k+3,\ p_k<0$$

$$p_{k+1}=p_k+2(x_k-y_k)+5,\ p_k\geq 0$$



applying: $\boxed{p_0 = \frac{5}{4} - r}$ Table

in case of negative so $p_k = p_k + 2x_k + 3 \rightarrow y_{k+1} = y_k$.

increment in x only nothing in y.

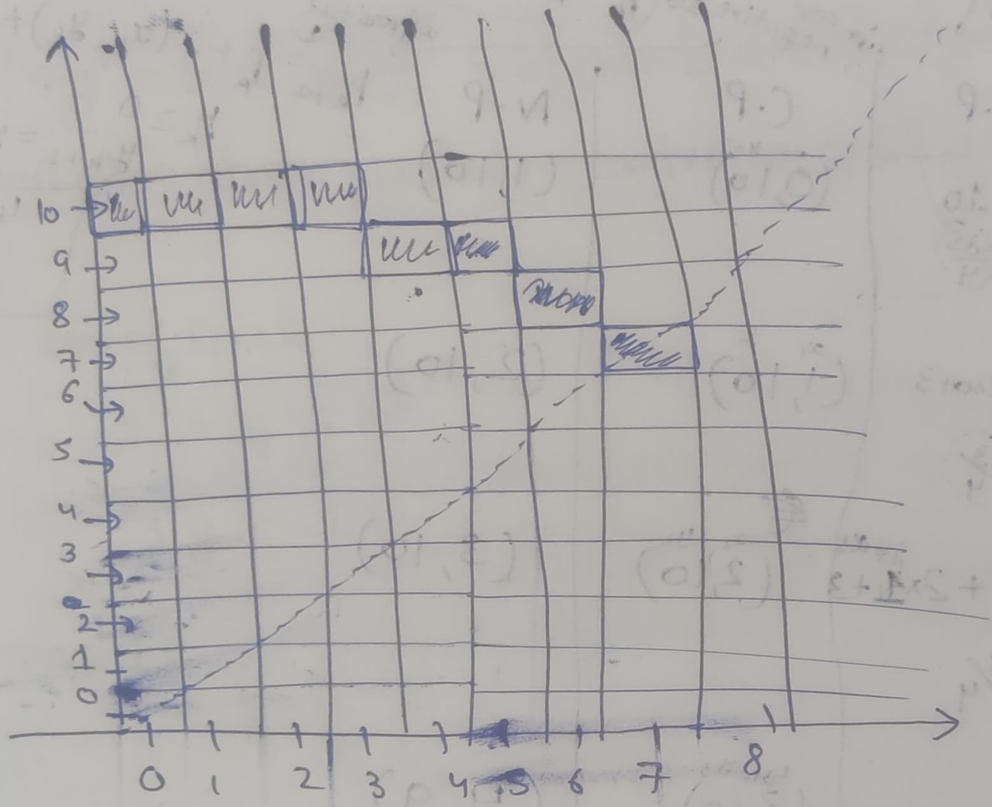
if positive $p_{k+1} = p_k + 2(x_k - y_k) + 5$

$p_k \geq 0$

$y_{k+1} = y_k - 1$

decrement in y. nothing in x

D.P	C.P	N.P
$p_0 = \frac{5}{4} - 10 = \frac{-35}{4}$	$\overset{x_0\ y_0}{(0,10)}$	$(1,10)$
$p_1 = \frac{-35}{4} + 2\times0+3$ $p_1 = \frac{-23}{4}$	$\overset{x_1\ y_1}{(1,10)}$	$(2,10)$
$p_2 = \frac{-23}{4} + 2\times\overset{\text{ie } x_1}{1} + 3 = -3/4$	$\overset{x_2\ y_2}{(2,10)}$	$(3,10)$
$p_3 = -\frac{3}{4} + 7 = \frac{25}{4}$	$\overset{x_3,\ y_3}{(3,10)}$	$(4,9)$
$p_4 = \frac{25}{4} + 2(-7) + 5 = -11/4$	~~3,10~~ $(4,9)$	$(5,9)$
$p_5 = -11/4 + 11$ $p_5 = 33/4$	$(5,9)$	$(6,8)$
$p_6 = \frac{33}{4} + 2(-4) + 5$ $p_6 = \frac{33}{4} - 3$ $p_6 = 21/4$	$(6,8)$	$(7,7)$



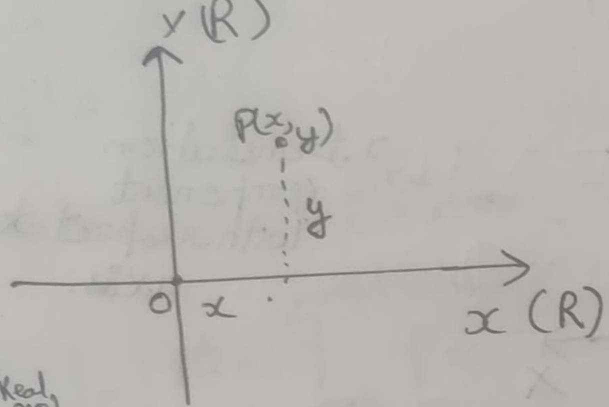
H.W

Q Plot the circle

$x^2+y^2=100$

using bresehams algorithm.

2D transformation



set
x R_r ← (Real no)
y R_r

$x \in X(R) \Rightarrow x \in R$
$y \in Y(R) \Rightarrow y \in R$

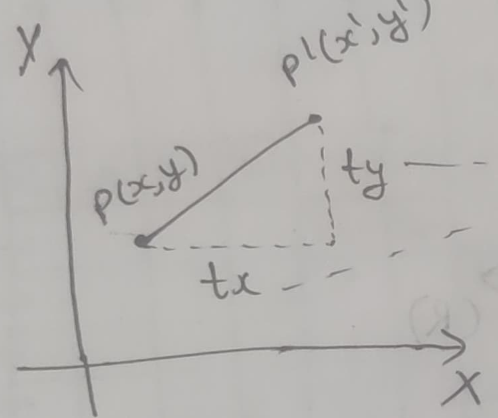
→ transformation is a general term which may include translation, rotation sccaling, reflection, ~~sheering~~ shearing, or few of them.

$A \times B = \{(a,b) : a \in A, b \in B\}$ {ordered pair}

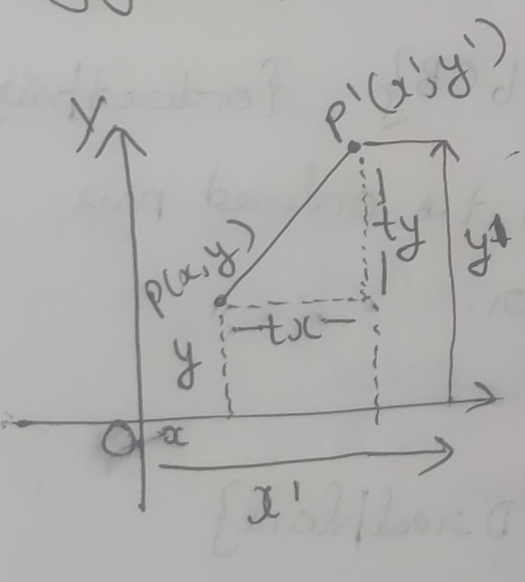
if $(a,b) \neq (b,a)$, then it is ordered pair.
else unordered pair.

$(x,y) \in R^2$ {2D real plane}

2 D translation



From figure P(x,y) is a point on 2D plain which is translated to point p'(x', y'), the ~~t(y)~~ translation component along x axis is t(x) and along y is t(y), then



$x' = x + t_x$

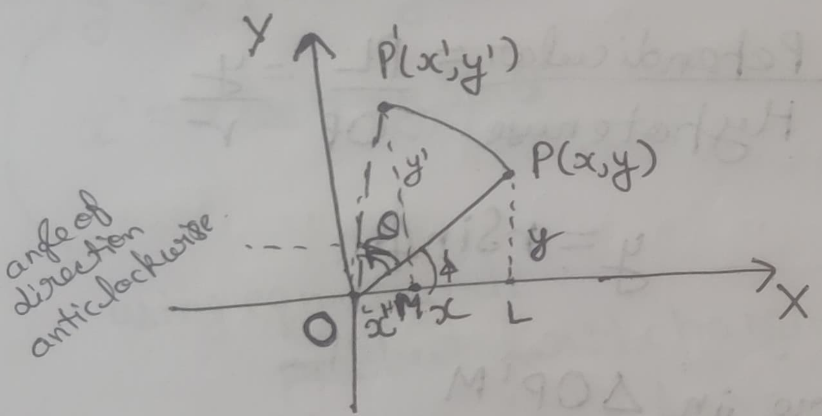
$y' = y + t_y$

in matrix form Addition

Addition

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} t_x \\ t_y \end{bmatrix} + \begin{bmatrix} x \\ y \end{bmatrix}$$

Rotation:⇒



→ anticlockwise +ve (θ)
→ clockwise -ve (θ)

↑

Rotation About origin

From figure P(x,y) is a point, which is rotated about origin through an angle θ in Anti-clockwise direction

If P'(x',y') is the new position of point P after rotation.

clearly $OP = OP' = r$ (says)

In right angle triangle OPL

$$OL = x, PL = y$$

$$\angle POL = \phi \text{ (says)}$$

$$\cos\phi = \frac{\text{Base}}{\text{Hypotenuse}} = \frac{OL}{OP} = \frac{x}{r}$$

$$x = r\cos\phi$$

$$\sin\phi = \frac{\text{Perpendicular}}{\text{Hypotenuse}} = \frac{PL}{OP} = \frac{y}{r}$$

$$y = r\sin\phi$$

Again From Figure in $\Delta OP'M$

$$\cos(\theta+\phi) = \frac{OM}{OP'} = \frac{x'}{r}$$

$$x' = r(\cos\theta+\phi)$$

$$x' = r\{\cos\theta\cos\phi - \sin\theta\sin\phi\}$$

Note]

$\cos(A \pm B) = \cos A\cos B \pm \sin A.\sin B$

$$x' = r\{r\cos\theta\cos\phi - r\sin\phi\sin\theta$$

$$x' = x\cos\theta - y\sin\theta \quad — \quad (1)$$

Again

$$\sin(\theta+\phi) = \frac{P'M}{OP'} = \frac{y'}{r}$$

$$y' = r\sin(\theta+\phi)$$

~~sin~~

$$y' = r(\sin\theta\cos\phi + \cos\theta\sin\phi)$$

imp.

$$\boxed{\begin{aligned}&\sin(A+B)\\&= \sin A\cos B + \\&\quad \cos A\sin B\end{aligned}}$$

$$y' = r\cos\phi\sin\theta + r\sin\phi\cos\theta$$

$$y' = x\sin\theta + y\cos\theta \quad \text{--- (2)}$$

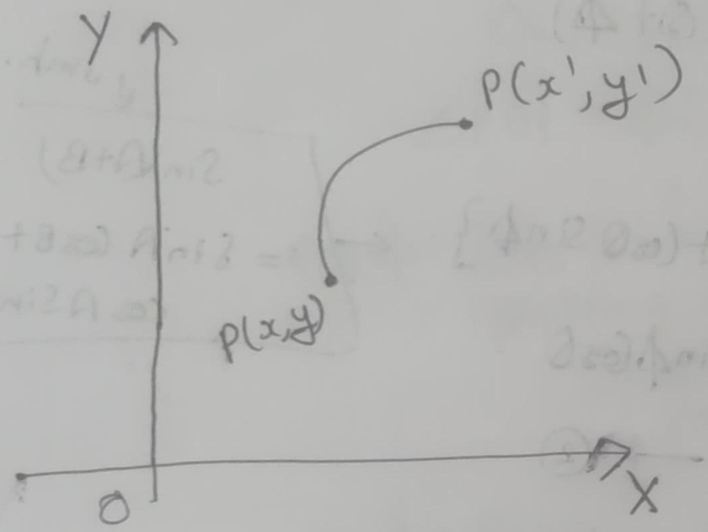
$$x' = x\cos\theta - y\sin\theta$$

using eqn: 1 and 2
~~put~~ but first changing
it in matrix form

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \times \begin{bmatrix} x \\ y \end{bmatrix}$$

Scaling

about origin :-



where p(x,y) is a point on 2D plane and after scaling p'(x',y') is the position of point P after scaling about origin, where the scaling factors are Sx and Sy along OX and OY respectively. then

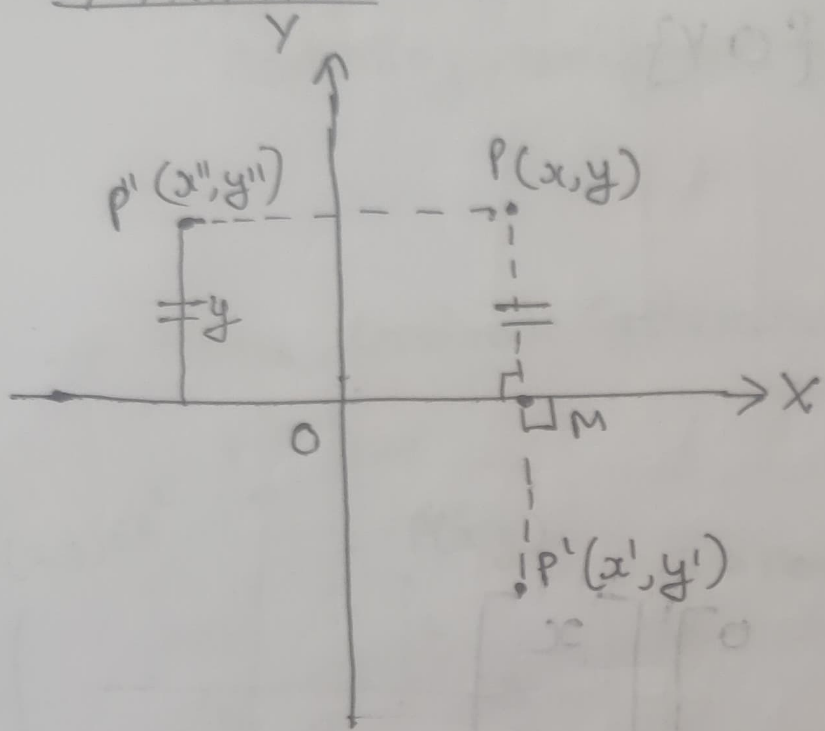
$x' = Sx \cdot x$

$y' = Sy \cdot y$

in matrix form

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} Sx & 0 \\ 0 & Sy \end{bmatrix} \times \begin{bmatrix} x \\ y \end{bmatrix}$$

Reflection



P(x,y) is point on 2D plane and it's reflection about x axis is P' (x',y')

with the help of simple geometric point ~~p&x~~ P' & P' are symmetric point

$$PM = P'M = y$$

Then $x' = x$ → reflection about x axis (OX)

$y' = -y$

in matrix form

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

Similarly reflection
about y axis {O Y}

$x'' = -x$
$y'' = y$

In matrix form

$$\begin{bmatrix} x'' \\ y'' \end{bmatrix} = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

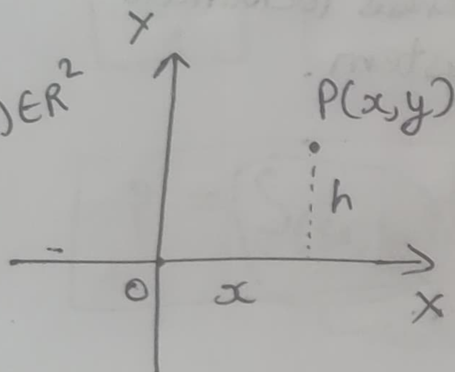
Transformation does n't consist only the translation, reflection, rotation, scaling, shearing it may be combination few or all of these operations, thus to determine, the resultant operator is a difficult task because the matrix operation is not uniform for all type of operation.

and so, we develope a new coordinate system known as homogeneous coordinate system.

homogenious coordinate system is helpful to bring uniformity in matrix operation.

Homes Coordinate System :->

$(x,y) \in R^2$



h -> Homogeneous factor.

(x_h, y_h, h)

$x_h = x.h \rightarrow x = \frac{x_h}{h}, \; y = \frac{y_h}{h}$

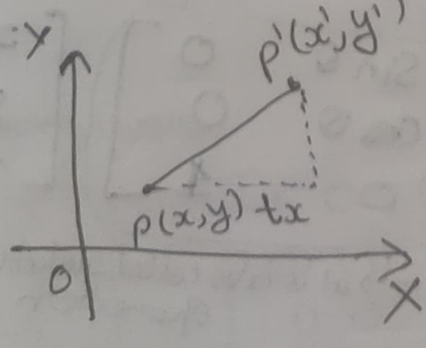
$y_h = y.h$

$h = 1$

$(x, y, 1)$

Translation :->

$$\begin{cases} x' = tx + x \\ y' = ty + y \end{cases}$$



$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} t_x \\ t_y \end{bmatrix} + \begin{bmatrix} x \\ y \end{bmatrix}$$

in matrix form using homogeneous coordinate system.

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix}$$

$$\boxed{P' = \underset{(t_x,t_y)}{T} \cdot P}$$

Rotation about origin

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

in homogeneous system

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} \cos & -\sin\theta & 0 \\ \sin & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

↳ it is also called rotation operator.

$P' = R(\theta) \cdot P$ → Rotation operator.

$-\theta$ = clockwise
θ = anticlockwise.

Scaling :->

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} Sx & 0 \\ 0 & Sy \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

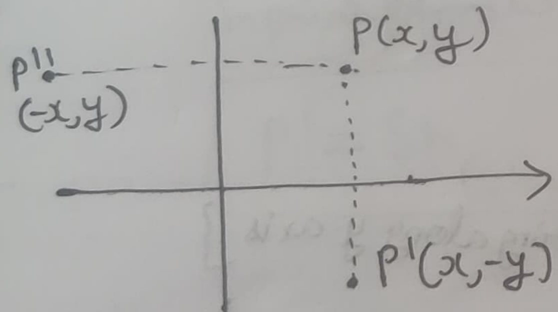
in homonious System

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} Sx & 0 & 0 \\ 0 & Sy & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

$$\boxed{p' = S_{(Sx, Sy)} P}$$

scaling operation.

Reflection



Reflection about OX

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

in homogenious

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

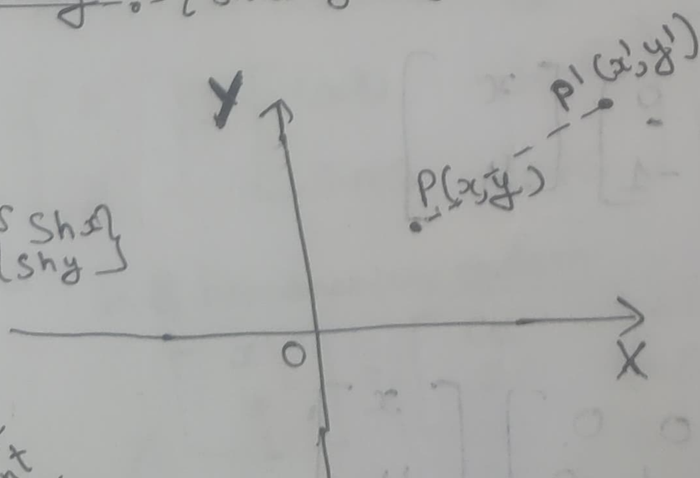
$$\boxed{P' = R_x . P}$$

Reflection about y axis ie (OY)

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} . \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

$$\boxed{P' = R_y . P}$$

Shearing :⇒ {shearing along x axis & shearing along y axis}



$\left\{ \begin{matrix} Sh_x \\ Sh_y \end{matrix} \right\}$

Shearing coefficient about OX and OY respectively

If $P(x,y)$ is a point on 2D space and $P'(x',y')$ is the position of point P after shearing

$$x' = x + y \cdot Sh_x$$
$$y' = y + x \cdot Sh_y$$

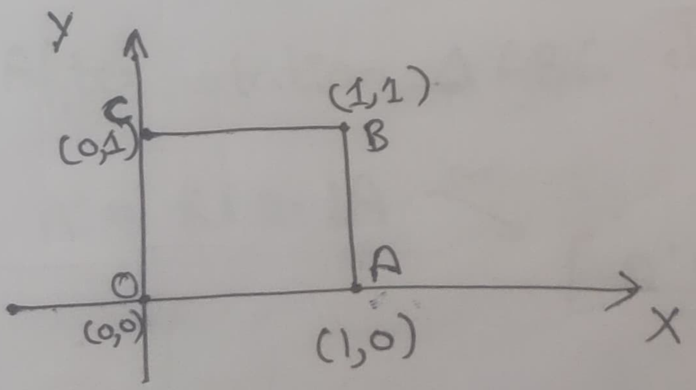
in matrix form :-

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & Sh_x & 0 \\ Sh_y & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

shearing operator

in point form we get.

$$\boxed{P' = Sh_{(Sh_x, Sh_y)} \cdot P}$$



~~0 with~~

taking shearing factor 1 we get :-

$$\left\{ \begin{array}{l} Sh_x = 1 \\ Sh_y = 1 \end{array} \right\}$$

Note ★ after shearing O is converted to O'

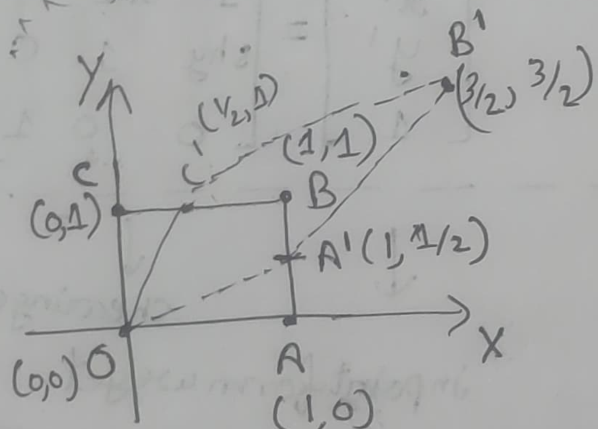
$$O' = \begin{bmatrix} x' = 0 \\ y' = 0 \end{bmatrix} = 0$$

After shearing of A becomes A'

$$A' = \begin{bmatrix} x' = 1 \\ y' = 1 \end{bmatrix}$$

taking after taking :

$$\left\{ \boxed{\begin{array}{l} Shx = 1/2 \\ \text{and } Shy = 1/2 \end{array}} \right.$$



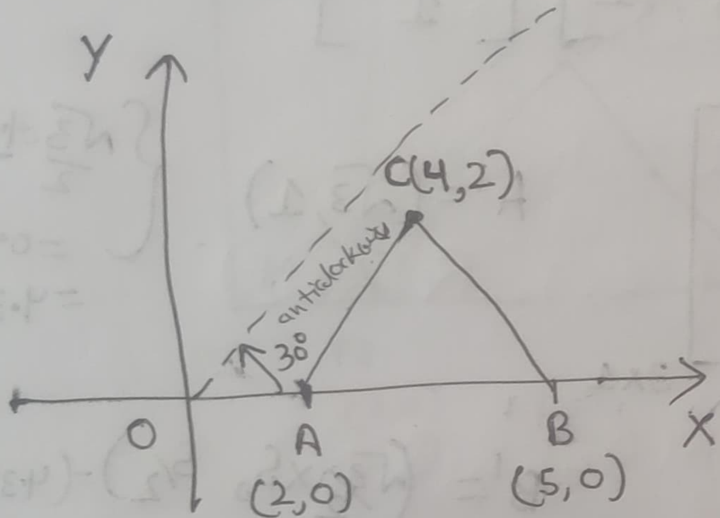
$$A' = \begin{bmatrix} x' = 1 \\ y' = 1/2 \end{bmatrix}$$

$$B' = \begin{bmatrix} x' = 3/2 \\ y' = 3/2 \end{bmatrix}$$

$$C' = \begin{bmatrix} x' = 1/2 \\ y' = 1 \end{bmatrix}$$

Numerical

Q Rotate the given ΔABC about origin through an angle 30° in anticlockwise direction



$\cos 30^\circ = \sqrt{3}/2$

$\sin 30^\circ = 1/2$

Angle of Rotation $+30^\circ$

The Rotation operator :>

$$R(\theta) = R(30^\circ) = \begin{bmatrix} \cos 30^\circ & -\sin 30 & 0 \\ \sin 30^\circ & \cos 30^\circ & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \sqrt{3}/2 & -1/2 & 0 \\ 1/2 & \sqrt{3}/2 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

After rotation ΔABC changed to $\Delta A'B'C'$

$A' = R(30^\circ)A$ or

$$[A'B'C'] = R(30^\circ)[ABC]$$

or

$$= \begin{bmatrix} \sqrt{3}/2 & -1/2 & 0 \\ 1/2 & \sqrt{3}/2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{array}{c} \begin{array}{ccc} A & B & C \end{array} \\ \begin{bmatrix} 2 & 5 & 4 \\ 0 & 0 & 2 \\ 1 & 1 & 1 \end{bmatrix} \end{array}$$

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} \sqrt{3}/2 & -1/2 & 0 \\ 1/2 & \sqrt{3}/2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 2 \\ 0 \\ 1 \end{bmatrix}$$

$$= \begin{bmatrix} \sqrt{3} \\ 1 \\ 1 \end{bmatrix}_{3\times 1}$$

$A' = (\sqrt{3}, 1)$

$\left\{ \frac{\sqrt{3}}{2} = \frac{1.732}{2} \right.$

$= 0.866 \times 5$

$= 4.330$

$B' = R(30).B$

$C' = R(30).C$

$B' = \left(\frac{\sqrt{3}}{2} \times 5,\ 5/2\right) = (4.33, 2.5)$

$C' = \left(2\sqrt{3} - 1,\ \cancel{2+\sqrt{3}}\ 2+\sqrt{3}\right)$

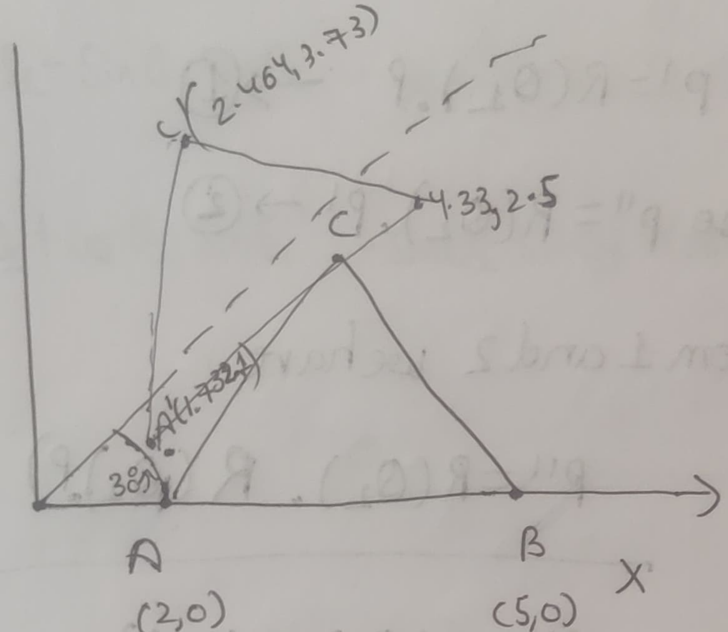
$= 1.732$

$\times 0.2$

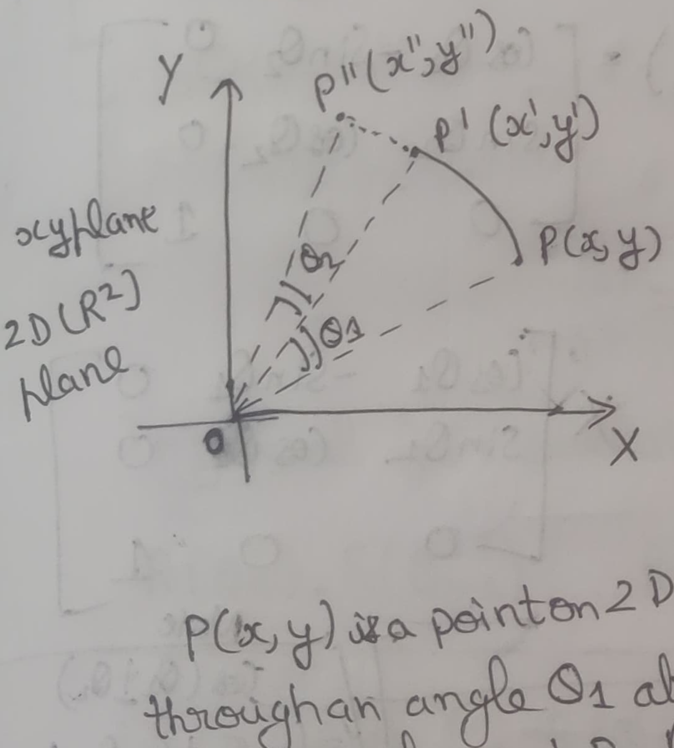
$3.464 \quad -1$

$C' = (2.464, 3.732)$

$$= \begin{bmatrix} \sqrt{3}/2 & -1/2 & 0 \\ 1/2 & \sqrt{3}/2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 4 \\ 2 \\ 1 \end{bmatrix}$$



Successive Rotation and scaling



$P(x, y)$ is a point on 2D plane, which is rotated through an angle θ_1 about origin $P'(x', y')$ is the new position of point P after rotation.

P' further rotated through an an angle θ_2 and $P''(x'', y'')$ is the new position of the point.

Then we have.

$$p' = R(\theta_1).P \rightarrow ①$$

Also $p'' = R(\theta_2).P' \rightarrow ②$

from 1 and 2 we have

$$p'' = R(\theta_2).\ R(\theta_1).P)$$

$$p'' = R(\theta_2).R(\theta_1)).P$$

$$A.(BC) = (AB).C$$

$$R(\theta_2).R(\theta_1) = \begin{bmatrix} \cos\theta_2 & -\sin\theta_2 & 0 \\ \sin\theta_2 & \cos\theta_2 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\times \begin{bmatrix} \cos\theta_1 & -\sin\theta_1 & 0 \\ \sin\theta_1 & \cos\theta_1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} \cos\theta_2 \cos\theta_1 & & \\ & & \\ & & \end{bmatrix}_{3\times3}$$

Note

$\cos(\theta_1 \pm \theta_2)$

$= \cos\theta_1 + \cos\theta_2 \mp \sin\theta_1 \sin\theta_2$

$\sin(\theta + \theta_2) = \sin\theta_1 \cos\theta_2 \pm \cos\theta_1 \sin\theta_2$

$$= \begin{bmatrix} \cos\theta_2 \cos\theta_1 - \sin\theta_2 \sin\theta_1 & -\cos\theta_2 \sin\theta_1 - \sin\theta_2 \cos\theta_1 & 0 \\ \sin\theta_2 \cos\theta_1 + \cos\theta_2 \sin\theta_1 & -\sin\theta_2 \sin\theta_1 + \cos\theta_2 \cos\theta_1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$R(\theta_2).R(\theta_1)$

$$= \begin{bmatrix} \cos(\theta_2+\theta_1) & -\sin(\theta_2+\theta_1) & 0 \\ \sin(\theta_2+\theta_1 & \cos(\theta_2+\theta_1) & 0 \\ 0 & 0 & 1 \end{bmatrix} = R(\theta_2+\theta_1)$$

$$\boxed{\begin{array}{l} -1 \leq \sin\theta \leq 1 \\ -1 \leq \cos\theta \leq 1 \\ \forall \theta \end{array}}$$

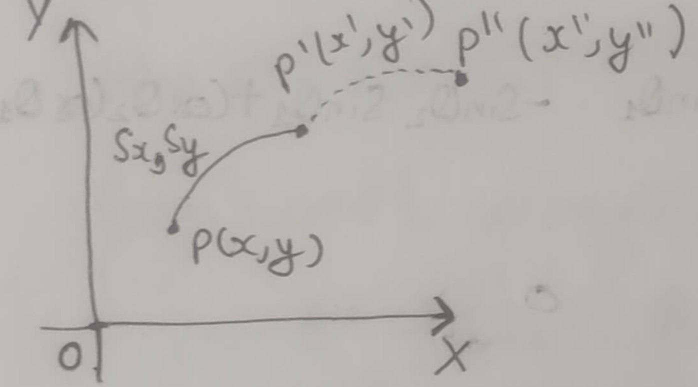
$$= R(\theta_1+\theta_2) = R(\theta_1).R(\theta_2)$$

In general for n successive rotation

$$\underline{R(\theta_n).R(\theta_{n-1}) \ldots\ldots R(\theta_2).R(\theta_1) = R(\theta_1+\theta_2+\cdots+\theta_n)}$$

$$= R(\theta_1).R(\theta_2) \ldots R(\theta_n)$$

Successive Scaling



Let $P(x,y)$ is a point on ~~to the~~ 2D plane

After scaling with scaling factor S_x, S_y, $P'(x',y')$ is a position of point P

After doing further scaling, with scaling factor S_{x_2}, S_{y_2} $P''(x'',y'')$ be the position of point P, then

Point equation.

$$P' = S_{(S_{x_1}, S_{y_1})} \cdot P \quad \text{—— (1)}$$

$$P'' = S_{(S_{x_2}, S_{y_2})} \cdot P \quad \text{—— (2)}$$

from 1 and 2 we have

$$P'' = S_{(S_{x_2}, S_{y_2})} \cdot \{ S_{(S_{x_1}, S_{y_1})} \cdot P \}$$

$$P'' = \{ S_{(S_{x_2}, S_{y_2})} \cdot S_{(S_{x_1}, S_{y_1})} \} \cdot P \quad \text{—— (3)}$$

$$S_{(Sx_2, Sy_2)} \cdot S_{(Sx_1, Sy_1)} = \begin{bmatrix} Sx_2 & 0 & 0 \\ 0 & Sy_2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} Sx_1 & 0 & 0 \\ 0 & Sy_1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} Sx_2 Sx_1 & 0 & 0 \\ 0 & Sy_2 Sy_1 & 0 \\ 0 & 0 & 1 \end{bmatrix}_{3\times 3}$$

$$= S_{(Sx_2 Sx_1,\ Sy_2 \cdot Sy_1)} = S_{(Sx_1 \cdot Sx_2,\ Sy_1 \cdot Sy_2)}$$

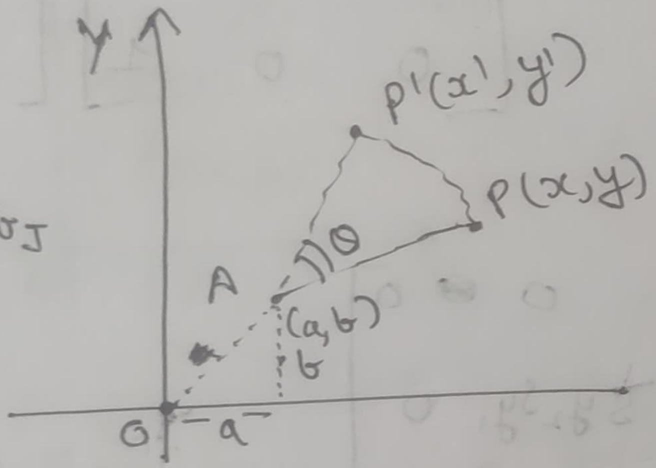
In general, for n successive scaling

$$S_{(Sx_1, Sy_1)} \cdot S_{(Sx_2, Sy_2)} \cdots S_{(Sx_n Sy_n)}$$

$$= S_{(Sx_1 \cdot Sx_2 \cdots Sx_n,\ Sy_1 \cdot Sy_2 \cdots S_n)}$$

Rotation of 2D object (Point) about a point other than origin.

$\overline{OA} = ai + bj$



From figure Point P(x, y) is rotated about point A(a, b), through an angle θ in anticlockwise direction and P'(x', y') is the position of point P after rotation.

Step 1 Shifting A(a, b) to origin O

represent $T_{-\bar{v}}$ → position vector
↳ point is coming towards origin

$$T_{-\bar{v}} = \begin{bmatrix} 1 & 0 & -a \\ 0 & 1 & -b \\ 0 & 0 & 1 \end{bmatrix}$$

S_2 :- Rotation about origin $R(\theta)$

$$R(\theta) = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Step 3

$S_3 \rightarrow$ Shifting point A back to its initial position

$$T_{\bar{V}} = \begin{bmatrix} 1 & 0 & a \\ 0 & 1 & b \\ 0 & 0 & 1 \end{bmatrix}$$

so the resultant operator is,

$$\cancel{R_A(\theta) = T_{\bar{V}} R(\theta) T_{-V}}$$

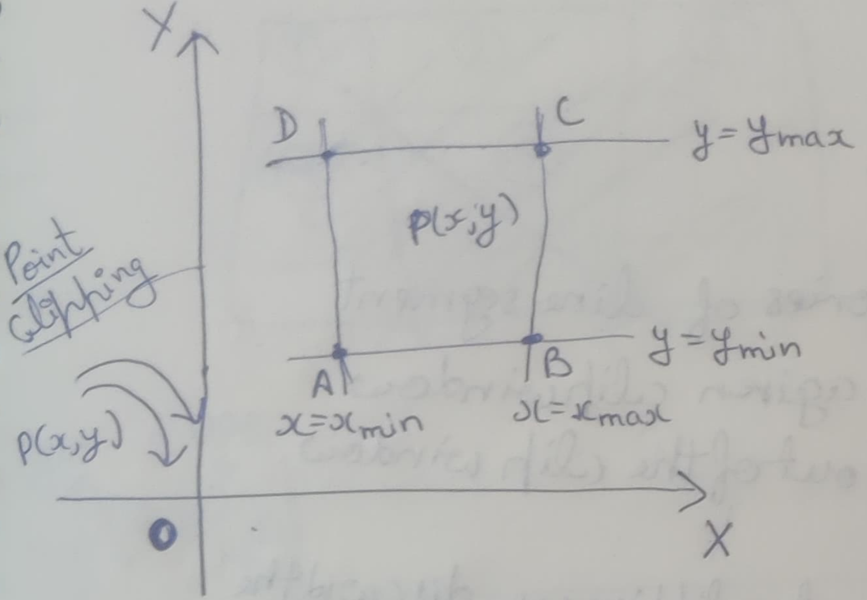
$$\boxed{R_A(\theta) = T_{\bar{V}} \cdot R(\theta) \, T_{-V}}$$

left out monday 4 march

5 March, 2024

Clipping in 2D: {point clipping}

Point clipping



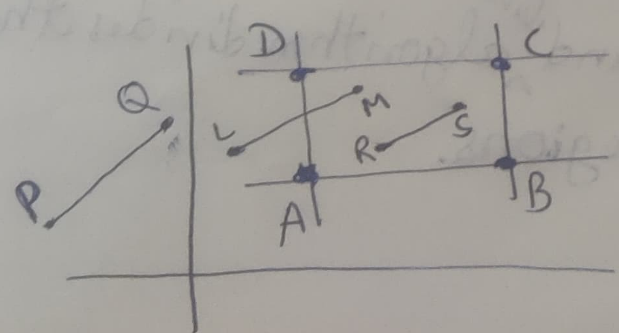
$$\left.\begin{array}{l} x_{min} \le x \le x_{max} \\ (x,y) \in R^2 \quad y_{min} \le y \le y_{max} \end{array}\right\} \quad (1)$$

Point $P(x,y)$ lies on 2D plane and ABCD is a clip window if point P ~~st~~ satisfies inequality set ~~one~~ (1) then point P will be inside the clip window or on it.

Otherwise, the point will be out of the clip.

Algorithm for line clipping.

→ In line clipping we check the position of line with respect to a given clip window.



Position of line ~~whi~~ with respect to given clip window

Cohen Sutherland

There are 3 categories of line segment with respect to a given clip window

1. line completely out of the clip window (PQ)

During the process of clipping we discard the line

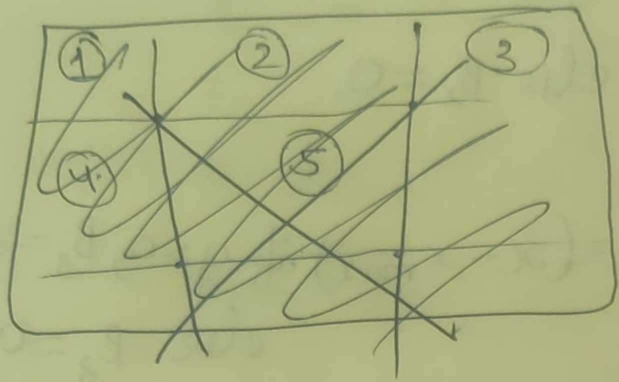
2. line completely inside clip window
→ there will be no change in line when we apply clipping.

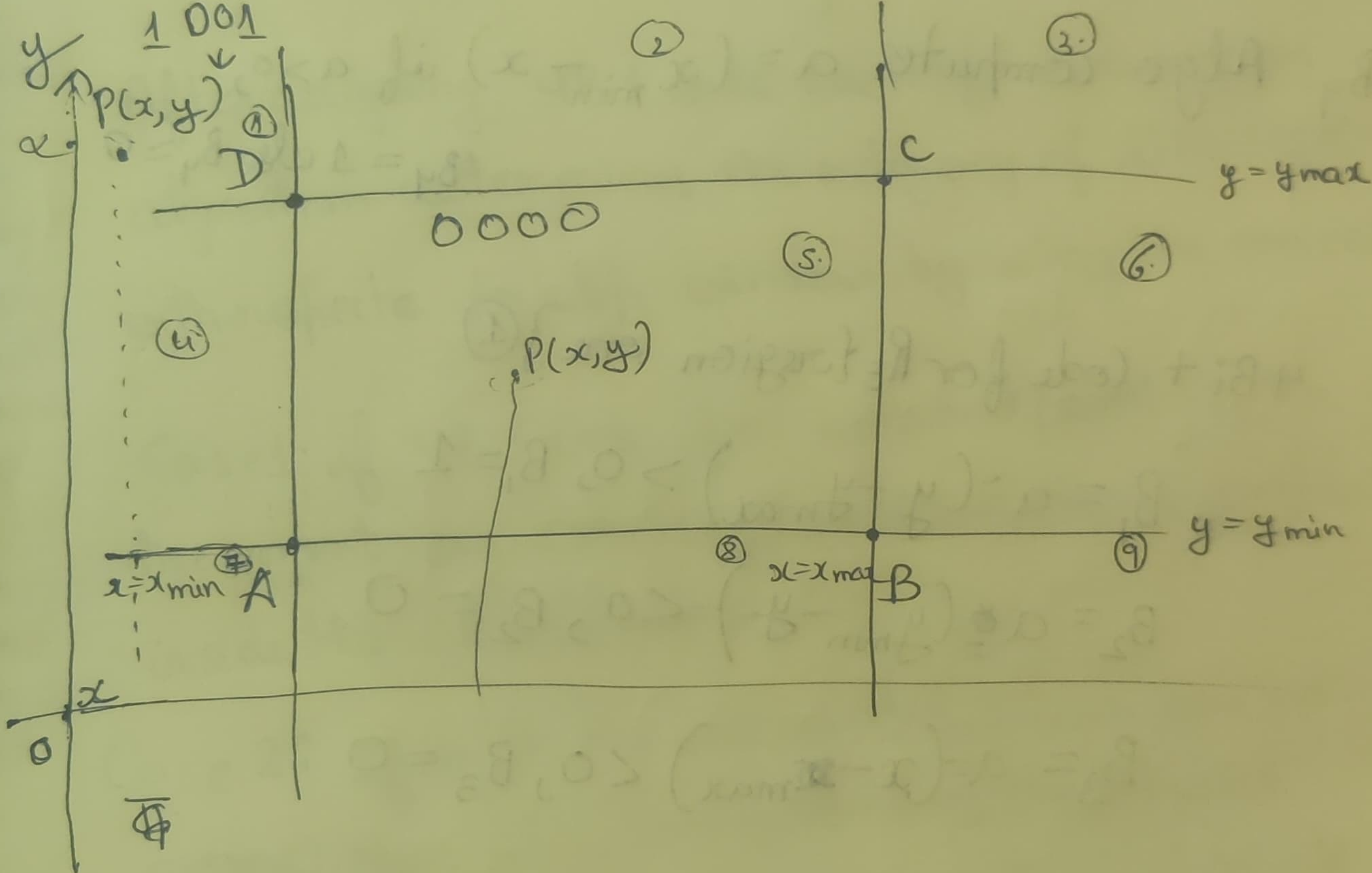
3. Partial in Partial out {clipping candidate line}

If some portion of line is inside the clip window and some outside such lines are known as clipping candidate line.

Cohen Sutherland line algorithm divides the 2D plane in 9 regions.

and assign a 4 bit code to each of these region.





→ The Bit assignment scheme :

B_1 {means Bit 1} : Algorithm Compute $a = (y - y_{max})$, if $a > 0$

$B_1 = 1$ else $B_1 = 0$

↳ less a is than equal to zero

B_2 : Algo Compute $a = (y_{min} - y)$ if $a > 0$;

$B_2 = 1$ else $B_2 = 0$

B_3 : Algo Compute $a = (x - x_{max})$ if $a > 0$, $B_3 = 1$
else $B_3 = 0$

B_4 : Algo compute, $a = (x_{min} - x)$ if $a > 0$,
$B_4 = 1$ else $B_4 = 0$

4 Bit Code for R_E {region one} 1

$B_1 = a = (y - y_{max}) > 0$, $B_1 = 1$

$B_2 = a = (y_{min} - y) < 0$, $B_2 = 0$

$B_3 = a = (x - x_{max}) < 0$, $B_3 = 0$

$B_4 = a = (x_{min} - x) > 0$, $B_4 = 1$

4 Bit Code for RE ⑤ (clip window)

$B_1 = 0$

$B_2 = 0$

$B_3 = 0$

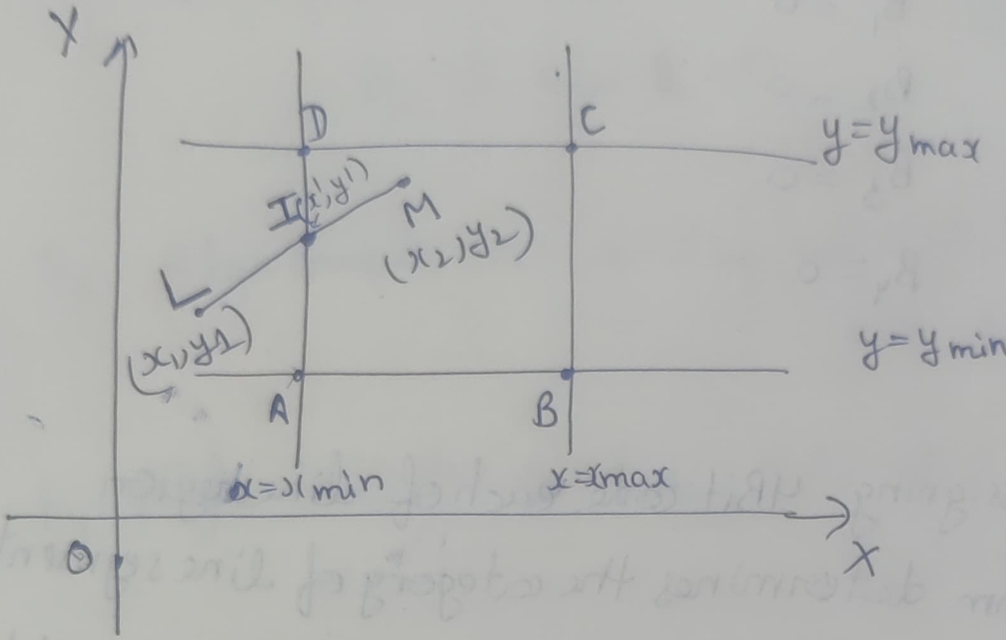
$B_4 = 0$

After assigning 4Bit code each of line region algorithm determines the category of line segment with respecte to clip window by using the concept.

Case1: if 4Bit code for both end point of the line segment are 0000, then the line is completely inside the clip window

Case 2: If both the end point Bit code are not 0000, then algorithm determines the Bitwise logical AND of 4 Bit codes of the end points of line segment, if it is 0000 then the line is clipping candidate.

→ The 4Bit code of the end points of line will help us to determine with which boundary of clip window the line is intersection.



eqn. of line LM

$$y - y_1 = m(x - x_1)$$

$$\boxed{m = \frac{(y_2 - y_1)}{(x_2 - x_1)}}$$

$$y - y_1 = \frac{(y_2 - y_1)}{(x_2 - x_1)}(x - x_1)$$

If I is point of intersection of line LM with left boundary of clip window then,

clearly $x' = x_{min}$

$$y' - y_1 = \frac{(y_2 - y_1)}{(x_2 - x_1)}(x' - x_1)$$

$$\boxed{y' = y_1 + \left(\frac{y_2 - y_1}{x_2 - x_1}\right)\left(x_{min} - x_1\right)}$$

with the help of these eqn. we can determine the point of intersection I protecting the line from I to M and deleting the rest will give us a clip line.

6, March, 2024

Liang Barsky line Clipping algorithm.

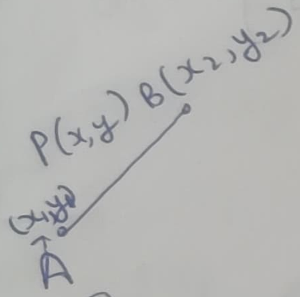
eqn of line.

$$ax + by + c = 0$$

$$y = mx + c$$

Parametric eqn. of Line

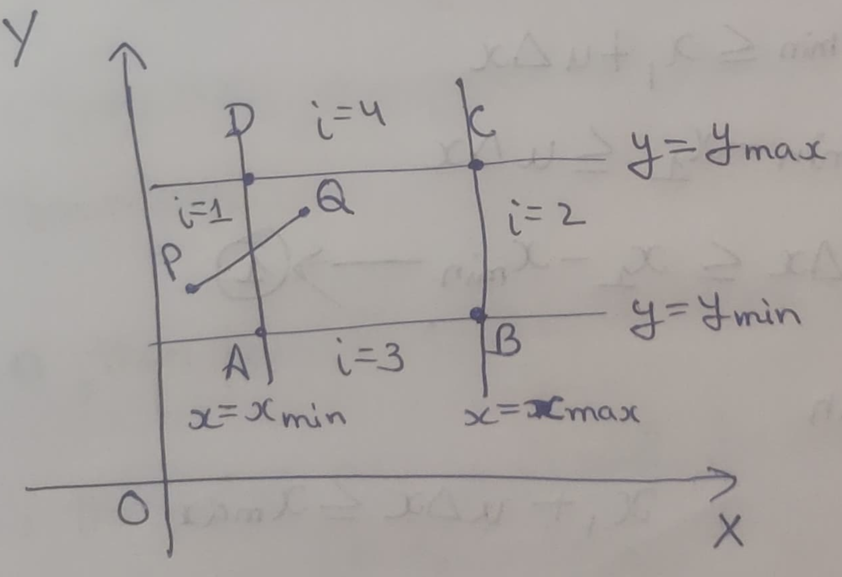
$$\left.\begin{aligned} x &= x_1 + u(x_2 - x_1) \\ y &= y_1 + u(y_2 - y_1) \end{aligned}\right\} \quad (1)$$



where $0 \le u \le 1$

$(x^2 + y^2 = r^2)$

P(x, y)



Liang Barsky Line Clipping Algo uses the concept of point clipping to clip a given line against the clip window

The point clipping is :-

$$\left.\begin{array}{l} x_{min} \le x \le x_{max} \\ y_{min} \le y \le y_{max} \end{array}\right\} (2)$$

from eqn (1) and eqn (2) we have.

$\cancel{x_{min} \le x_1 + u\Delta x}$

$$\left\{\begin{array}{l} x_2 - x_1 = \Delta x \\ y_2 - y_1 = \Delta y \end{array}\right\}$$

$x_{min} \le x_1 + u\Delta x \le x_{max}$

$y_{min} \le y_1 + u\Delta y \le y_{max}$

$x_{min} \le x_1 + u\Delta x$

$x_{min} - x_1 \le u\Delta x$

$-u\Delta x \le x_1 - x_{min} \longrightarrow (1)$

Again

$y = y_{max}$

$x_1 + u\Delta x \le x_{max}$

$y = y_{min}$

$u\Delta x \le x_{max} - x_1 \longrightarrow (2)$

and

$-u\Delta y \le y_1 - y_{min}$ —— (3)

$u\Delta y \le y_{max} - y_1$ —— (4)

in equality ①, ②, ③ & ④ in general form,

$u \times d_i \le q_i$ for $i = 1, 2, 3, 4$

$d_1 = -\Delta x$, $q_1 = x_1 - x_{min}$

$d_2 = \Delta x$ $q_2 = x_{max} - x_1$

$d_3 = -\Delta y$ $q_3 = y_1 - y_{min}$

$d_4 = \Delta y$ $q_4 = y_{max} - y_1$

if $d_i = 0$, then the line will be parallel to i^{th} boundary

Again If, $q_i < 0$ {negative}, then point $P(x_1, y_1)$ is out of i^{th} boundary.

if $q_i = 0$, then point $P(x, y)$ will lie on i^{th} boundary.

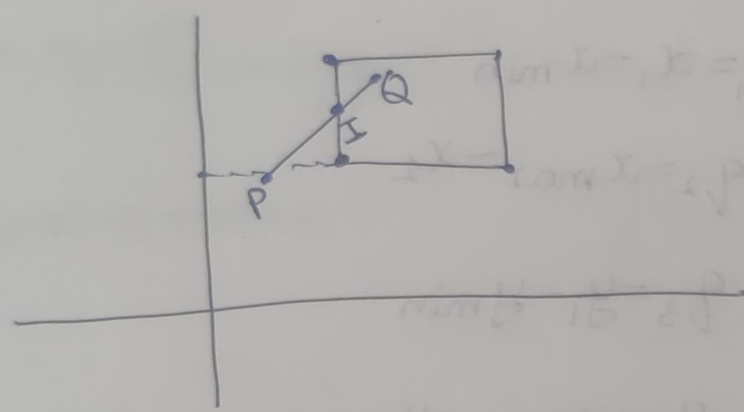
if $q_i > 0$, the point $P(x_1, y_1)$ will lie inside the i^{th} boundary.

→ To find the point of intersection of the line segment with the boundary of clip window we need the corresponding parametric values.

we use this eqn.

$u = (q_i / d_i)$ where $d_i \neq 0$
for $i = 1, 2, 3, 4$

this equation will give us the values of u, those values of u which are out of interval [0, 1]



~~ViewGroup~~