

D Routing

2) Logical Addressing (IP Address)

ARP & RARP → Reverse address resolution

↓
Address resolution
protocol

MAC address

present for every host individually and uniquely.
both IP address are unique and are

→ Host ID different, network ID same like colony

same, house address different.

→ IP address consists of 2 parts → Host ID & network ID.

11/18

Transport Layer (Process to process delivery)

→ responsible for process to process delivery.

③ Port Addressing

④ Segmentation and reassembly

Sender → divides msg → sequence no
into packets is added to
packets.

10101101010101110

sequence
no.

Receiver end → reassembly

removes seq nos & retrieves the
data and joins accordingly.

(3) - Connection control

It decides whether it should be connection less or connection oriented.
(TCP)

(UDP)

(4) - Flow control

(end-to-end) flow control in transport layer is responsible for flow control in all the intermediary nodes as well.

(5) - Error control

end-to-end

Flow control in data link layer is internal & in transport layer is end-to-end.

Application layer (User support layer)

→ user interacts directly to this layer.

→ provides services to the user.

→ Mail service

→ file transfer service

→ remote login

→ accessing world Wide Web.

→ session + presentation layer job is done by Application layer

OSI- Model (7 layers) } + Session + Presentation
layer

→ It is a theoretical model. TCP/IP is the implemented model.

ISO - OSI Model (also called)

International standard organisation open system inter connect Model.

Session layer

- used to maintain session
- Network dialog control
- It is used to establish, maintain and synchronize interaction b/w communicating systems

Presentation layer

- syntax and semantics of information / data sent among systems.
- Encryption and Decryption
- used for security, original msg is converted to some form not readable by anyone else.
border end receiver end
- Compression
(some folders for eg can't be sent as it is and we compress it)

→ OSI - Model ? → explain 7 layers

→ TCP / IP Model ? → explain 5 layers

+ protocol & networking device used at each layer.

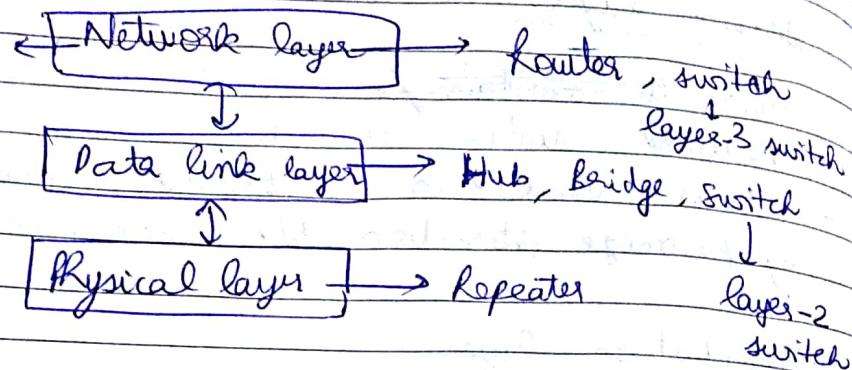
TCP / IP protocol suite

Protocols used

HTTP, SMTP, TELNET
FTP, TELNET

Uses Datagram Protocol
TCP, UDP

IP, ICMP
Internet control msg protocol



} Session + presentation layers generally doesn't use any device, protocols.

Assignment questions

Q1 Write short note on SNA, AppleTalk and Netware.

Q2 Differentiate between connection less and connection oriented.

Q3 Explain wired transmission media and wireless transmission media in detail.

Q4 Explain network topologies.

Submit by :- Next ~~Thursday~~ Friday (19/1/18)

X — (Unit - I completed) — X

Unit - 2

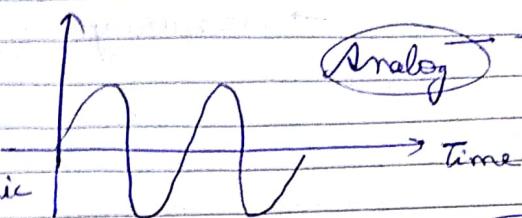
Physical Layer

classmate

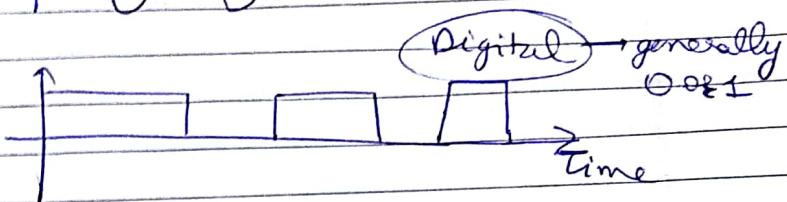
Date _____

Page _____

SIGNALS



Analog → infinite no. of values in a range



Digital → generally 0 or 1

Periodic — have a particular pattern in a particular span of time / measurable time frame.

where 1 pattern completes → cycle (completion of 1 full pattern)

Aperiodic — no pattern, changes without executing a pattern or cycle that changes over time

In Data communication, we use →

Periodic Analog signals & Aperiodic Digital Signals.

Analog signals → simple
→ composite

Simple

→ consists of a single sine wave

Composite

→ multiple sine waves.

Characteristics

- Amplitude
- Phase
- Frequency

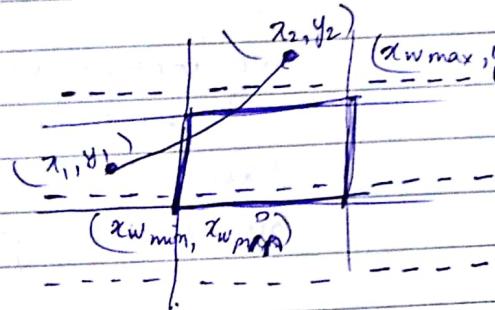
COMPUTER GRAPHICS

screen (2D)

Resolution

→ element
model
specification

- ① drawing → primitive → mathematical equation knowledge
- ② manipulation
 - clipping line circle
 - line ellipse parabola
 - polygon hyperbola
- ③ covering
 - a areas so 4 bit address



left 0001
Bottom 0010
Right 0100
top 1000

area for Cool ~~(x, y)~~ $\leftarrow (x, y, x_{w\min}, y_{w\min}, x_{w\max}, y_{w\max})$
int temp = 0;

if ($x > x_{w\max}$)

temp = temp | (1 << 2);

else if ($x < x_{w\min}$)

temp = temp | (1);

if ($y > y_{w\max}$)

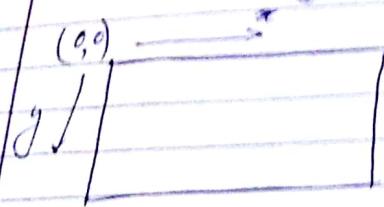
temp = temp | (1 << 3);

else if ($y < y_{w\min}$)

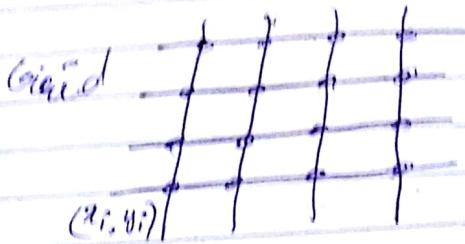
temp = temp | (1 << 1);
return temp

Line

$$y = mx + c$$



represented in ^{scanned} _{coordinates}



$$y_i = mx_i + c$$

x	10	.	.
y	10	.	.

$$y = mx + c$$

lets say $m < 1$

$$x_{i+1} = x_i + 1$$

$$y_{i+1} = y_i + m$$

we increment x

& not y as

$\tan \theta < 1 \Rightarrow \Delta x > \Delta y$

so more precise

$$x = x_1 \quad y = y_1$$

$$\Delta x = x_2 - x_1, \quad \Delta y = y_2 - y_1,$$

if $\left| \Delta y \right| < \left| \Delta x \right| \}$

putpixel(x, y, WHITE);

while ($x \leq x_2$)

{

$$y += m$$

$x++;$

putpixel(x, y, WHITE);

}

}

if $m > 1$

$$y = mx + c$$

$$x = \frac{1}{m}y - \frac{c}{m}$$

Page No.		
Date		

$$y_{i+1} = y_i + 1$$

$$x_{i+1} = x_i + \frac{1}{m}$$

$$\text{if } \text{abs}(dy) \geq \text{abs}(dx) \{$$

putpixel(x, y, WHITE)

while ($y <= y_2$) {

$$y += \frac{1}{m}$$

$$x += \frac{1}{m}$$

putpixel(x, y, WHITE)

// here floating part \rightarrow

3

Floating part calculation takes a lot of time

\Rightarrow instead \Rightarrow

while ($x <= x_2$) {

$x++$

$y += (\text{float}) \frac{dy}{dx}$

putpixel(x, round(y), WHITE);

3

when

round \hookrightarrow int round(float p) {

if ($p - \text{int}p < 0.5$)

return p

else return p+1

3

$$f(x, y) = ax + by + c = 0$$

< 0

> 0

$$f(m) = f(x_i + 1, y_i + \frac{1}{2}) = a(x_i + 1) + b(y_i + \frac{1}{2})$$

11.11

it is the next of $x_i + \Delta x$

step 1	step 2	step 3
x_i	$x_i + 1$	$x_i + 2$
y_i	$y_i + 1/2$	$y_i + 1$

$$\text{if } d < 0 \quad \text{selection (E)}$$

$$d_{\text{new}} = f(m_E) = f(x_i + 2, y_i + \frac{1}{2}) \\ = a(x_i + 2) + b(y_i + \frac{1}{2}) + c$$

- ② For a primitive drawing, note change

$$\Delta E = d_{\text{new}} - d \\ = a \\ = dy$$

$$y = mx + c$$

~~$$dy = dyx + c dx$$~~

~~$$dyx = dyx + c dx$$~~

~~$$dx + by + c$$~~

~~$$dyx - dyx + c dx = 0$$~~

$$\text{if } d \geq 0$$

selection (NE)

~~$$f(m_E) = f(x_i + 2, y_i + \frac{3}{2})$$~~

$$= a(x_i + 2) + b(y_i + \frac{3}{2}) + c$$

$$\Delta NE = a + b$$

$$= dy - dx$$

Q2 from
Q2 from
give
answer

$$d = f(M) = \frac{f(x_{i+1}, y_i + \frac{1}{2}) - f(x_i, y_i)}{2}$$

$$= a(x_{i+1}) + b(y_i + \frac{1}{2}) + c$$

$$= ax_i + by_i + c + a + b\frac{1}{2}$$

as $ax_i + by_i + c = 0$ since x_i, y_i are already given.

$\rightarrow b\frac{1}{2}$ is scaling line part

\rightarrow we can't multiply the whole by 2

$$\text{as } y = mx + c \equiv 2y = 2mx + 2c$$

$$d = 2a + b$$

$$f(x, y) = 2(ax + by + c)$$

$$\Delta E = 2dy$$

$$\Delta NE = 2(dy - dx)$$

$$x = x_1, y = y_1, dx = x_2 - x_1, dy = y_2 - y_1,$$

$$d = 2(dy) - dx$$

`putpixel(x, y, white)`

`while (x <= x_2) {`

`if (d < 0) { // selection E`

$$d += 2 * dy$$

`} else`

$$d += 2(dy - dx)$$

`} y++;`

`putpixel(x, y, white);`

`}`

day sun
m_i y_i
10 10

$$dx = 10$$

x₂ y₂
20 18

$$dy = 8$$

$$\begin{aligned}x_{\text{initial}} &= 10 &y_{\text{initial}} &= 10 \\d\text{initial} &= 2 &dy + dx &\end{aligned}$$

x _i	y _i	d _i
10	10	6
11	11	2
12	12	-2
13	12	14
14	13	10
15	14	6
16	15	2
17	16	-2
18	16	18
19	17	14
20	18	10

m_i y_i
10 10

x₂ y₂
18 20

$$dx = 8 \quad dy = 10$$

x⁰ y⁰ d
10 10 12

$$\text{dinitial} = 20 - 8 = 12$$

11 11 16
12 12 20
13 13 22
14 14 24
15 15 26
16 16 28
29

$$a(x_{i+1}) + b(y_{i+1/2}) + c$$

$$ax_i + by_i + c + 2a + b, \frac{1}{2}$$

$$= 4a + b$$

$$40 - 8 = 32$$

Scan conversion

primitive

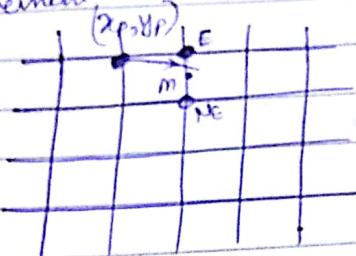
clade

void circleSymmetry(x, y)

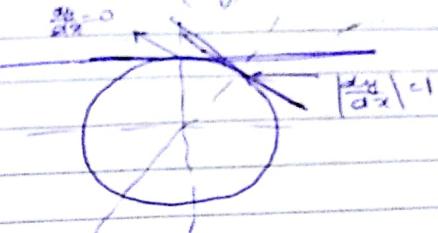
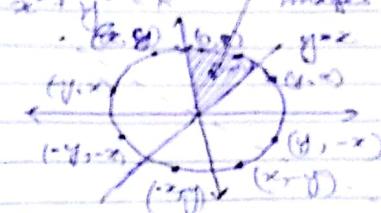
putPixel(x, y, WHITE)

putPixel(y, x, WHITE)

major movement in x dir



$x^2 + y^2 = R^2$ draw this
first quadrant image



$$\text{At } (x_p, y_p) \quad f(x_p, y_p) = x_p^2 + y_p^2 - R^2 \quad \text{--- (1)}$$

for decision consideration

$$d = f(m) = f(x_p+1, y_p-1/2)$$

$$= (x_p+1)^2 + (y_p-1/2)^2 - R^2 \quad \text{--- (2)}$$

If $d < 0$

selection $\rightarrow E$

$$d_{\text{new}} = f(m_E) = f(x_p+2, y_p-1/2)$$

$$= (x_p+2)^2 + (y_p-1/2)^2 - R^2 \quad \text{--- (3)}$$

$$\Delta E = d_{\text{new}} - d \quad \text{--- (3) --- (2)}$$

$$= \pm (2x_p + 3)$$

$$\Delta E = 2x_p + 3$$



else selection (SE)
 $d_{new} = f(m_{SE})$

$$= f(x_p + 2, y_p - \frac{3}{2}) \\ = (x_p + 2)^2 + (y_p - \frac{3}{2})^2 - R^2 \quad \textcircled{4}$$

$$\Delta SE = \textcircled{4} - \textcircled{2} = d_{new} - d$$

$$= (2x_p + 3) + (2y_p - 2)(-1) \\ = 2x_p - 2y_p + 5$$

$$\boxed{\Delta SE = 2(x_p - y_p) + 5}$$

Initialisation

$$d_{initial} \rightarrow d_{(0,R)} = \frac{5}{4} - R \approx 1 - R$$

$$x = 0; y = R$$

$$d = 1 - R$$

circle symmetry (x, y)

while ($x \leq y$) {

if $d < 0$ // selection E

$$d+ = 2*x + 3$$

else {

$$d+ = 2*(x - y) + 5$$

$y--;$

$x++;$

circle symmetry (x, y);

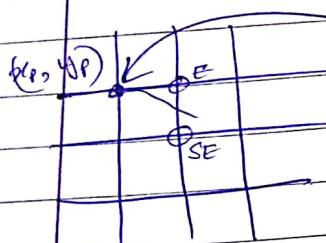
}

Eg

$$R = 8$$

x	y	d
0	8	-7
1	8	-1
2	8	1
3	7	-6
4	7	3
5	6	2
6	5	-5

Page No.	
Date	



$$\Delta E = 2x_p + 3$$

$$\Delta SE = 2(x_p - y_p) + 5$$

} → ①

if $d < 0$ then // selection(E)

At (E) → $[x_{p+1}, y_p]$ → ②

$$\Delta E_{\text{new}} = 2x_p + 5$$

$$\Delta SE_{\text{new}} = 2(x_p - y_p) + 7$$

② - ①

$$\Delta^2 E = 2$$

$$\Delta^2 SE = 2$$

, else

// selection(SE)

At (SE) → (x_{p+1}, y_{p-1})

$$\Delta E_{\text{new}} = 2x_p + 5$$

$$\Delta SE_{\text{new}} = 2(x_p - y_p) + 9$$

② - ①

$$\Delta^2 E = 2$$

$$\Delta^2 SE = 4$$

Initialisation

{ At $(0, R)$ }

$$d = 1 - R$$

$$\Delta E = 3$$

$$\Delta SE = 5 - 2R$$

$$2(0) + 3$$

$$2(0 - R) + 5$$

$$\begin{aligned}x &= 0 \quad y = R \\d &= 1 - R \\DE &= 3 \\DSE &= 5 - 2R\end{aligned}$$

circle symmetry (x, y)
 while ($x \leq y$) {
 { if ($d \leq 0$) { // selection E
 $d += DE$
 $DE += 2, DSE += 2$ }
 else {
 $d += DSE$
 $DE += 2, DSE += 4$
 $y --$
 $x ++$

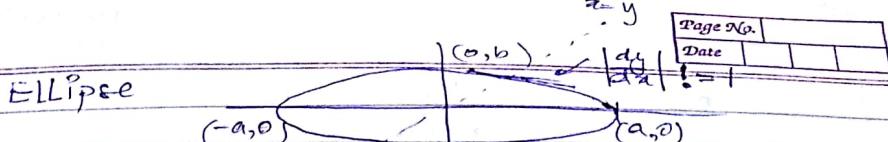
circle symmetry (x, y)

3

$R = 8$	x	y	d	DE	DSE
	0	8	-7	3	-11
	1	8	-4	5	-9
	2	8	1	7	-7
	3	7	-6	9	-3
	4	7	3	11	-1
	5	6	2	13	3
	6	5	5	15	7

Result is still the same as before

ELLIPSE



Page No. _____
Date _____

region II

$$(-a, y)$$

$$(-x, y)$$

$$(x, y)$$

$$(x, -y)$$

$$(-x, -y)$$

4-way symmetry
about x & y axis

Region II

$$|dy/dx| = 1$$

$$|dy/dx| > 1$$

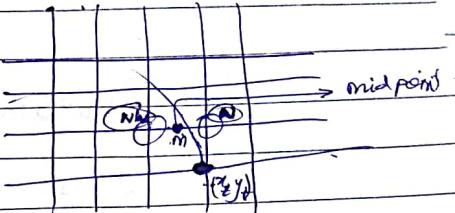
$$|dy/dx| < 1$$

$$|dy/dx| < 1$$

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

in Region I

$$|dy/dx| > 1$$



major movement in y as dy>dx

$$\text{At } (x_t, y_t) \quad d = f(m) = f\left(x_t - \frac{1}{2}, y_t + 1\right). \quad (1)$$

if $d < 0$

selection in N

$$b^2 \left(\frac{x_t - 1}{2}\right)^2 + a^2 (y_t + 1)^2 - a^2 b^2$$

$$b^2 \left(\frac{x_t^2 + 1}{4} - x_t\right) + a^2 (y_t^2 + 1 + 2y_t) - a^2 b^2$$

$$\frac{b^2 x_t^2 + b^2}{4} - b^2 x_t + a^2 y_t^2 + a^2 + 2a^2 y_t - a^2 b^2 < 0$$

$$d_N^\pm = f(m_N) = f\left(x_t - \frac{1}{2}, y_t + 2\right) \quad (2)$$

$$= b^2 \left(x_t - \frac{1}{2}\right)^2 + a^2 (y_t + 2)^2 - a^2 b^2$$

$$\Delta N = \frac{b^2 \left((x_t + 1)(-1/2)\right)}{4} + a^2 (2y_t + 1) = a^2 (2y_t + 1)$$

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

else if $d > 0$
selection is NW

$$d_{NW} = f(x_t - 3/2, y_t + 2) \quad \text{--- (3)}$$

$$d_{NW}^* = b^2 \left(x_t - \frac{3}{2} \right)^2 + a^2 (y_t + 2)^2 - b^2 a^2$$

$$\Delta_{NW} = b^2 (2x_t - 2)(-1) + a^2 (2y_t + 3)$$

$$= -2b^2(x_t - 1) + a^2(2y_t + 3)$$

initialisation \rightarrow at $(a, 0)$

$$d_{init} = f(a - 1/2, 1) = b^2 \left(a - \frac{1}{2} \right)^2 + a^2 - a^2 b^2$$

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

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

Region II

$$d_{II}^* = f(x_t - 1, y_t + 1) \quad d_{II} > d_{NW}$$

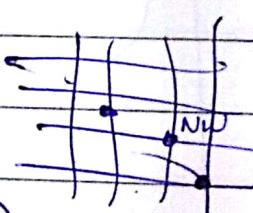
$$= f(x_t - 1, y_t + 1/2)$$



$$= b^2 (x_t - 1)^2 + a^2 \left(y_t + \frac{1}{2} \right)^2 - a^2 b^2 \quad \text{--- (4)}$$

if $d_{II}^* < 0$

selection is NW



$$d_{NW}^* = f(x_t - 2, y_t + 3/2)$$

$$= b^2 (x_t - 2)^2 + a^2 \left(y_t + \frac{3}{2} \right)^2 - b^2 a^2 \quad \text{--- (5)}$$

$$\Delta_{NW} = b^2 (2x_t - 3)(-1) + a^2 (2y_t + 2)(1)$$

$$\Delta NW = -b^2(2x_t - 3) + 2a^2(y_t + 1) = b^2(3 - 2x_t) + 2a^2(y_t + 1)$$

else if $d^{II} \geq 0$

selection is W

$$d_W^{II} = f(x_t - 2, y_t + 1/2)$$

$$= b^2(x_t - 2)^2 + a^2(y_t + 1/2)^2 - b^2a^2$$

$$\Delta W = b^2(2x_t - 3)(-1) = b^2(3 - 2x_t)$$



d initialisation

$$x = a$$

$$y = 0$$

$$d = a * a + (b * b) / 4 - a * b * b$$

$$\text{while } (b^2(x_t - 2) \geq a^2(y_t + 1/2))$$

{ /* REGION I */ }

* 3

$$d = b^2(x_t - 1)^2 + a^2(y_t + 1/2)^2 - a^2b^2$$

here x,y
from previous
will initialise

$$\text{while } (x_t > 0)$$

{ /* REGION II */ }

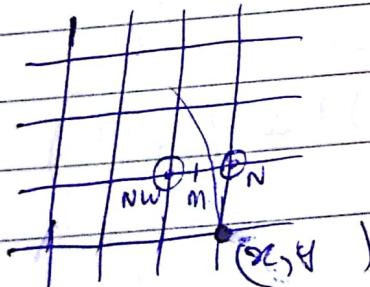
?

Ellipse \rightarrow 2nd differential eqn

$$\Delta N_t = a^2(2y_t + 3)$$

$$\Delta NW_t = a^2(2y_t + 3) - 2b^2(x_t - 1)$$

REGION-I



$$\begin{cases} \Delta N_{t+1} = a^2(2(y_{t+1}) + 3) \\ \Delta NW_{t+1} = a^2(2(y_{t+1}) + 3) - 2b^2(x_{t+1} - 1) \\ \Delta^2 NW = 2a^2 \end{cases}$$

Initialisation at $(a, 0)$

$$\Delta N = 3a^2 \quad \Delta NW = \frac{3a^2 - 2b^2a + 2b^2}{3a + 2b^2(1-a)}$$

$$d = f(x_t - 1/2, y_t + 1)$$

$$\Delta N = a^2(2y_t + 3) - ① \quad \text{if } d^+ < 0 \quad \text{selection } N \quad (x_t, y_t + 1)$$

$$\Delta NW = a^2(2y_t + 3) + 2b^2(1-x_t) - ②$$

$$\Delta d_{new} = f(x_t - 1/2, y_t + 2)$$

$$\Delta N_{new} = a^2(2y_t + 5) - ③$$

$$\Delta N^{2^I}_{\cancel{NW}} = 2a^2$$

$$\Delta NW_{new} = a^2(2y_t + 5) + 2b^2(1-x_t) - ④$$

$$\Delta N^{2^I}_{\cancel{NW}} = 2a^2$$

else if $d^+ > 0$

selection NW $(x_t - 1/2, y_t + 1)$

$$d_{new} = f(x_t - 3/2, y_t + 2)$$

$$\Delta N_{new} = a^2(2y_t + 5) - ⑤$$

$$\Delta NW_{new} = a^2(2y_t + 5) + 2b^2(2-x_t) - ⑥$$

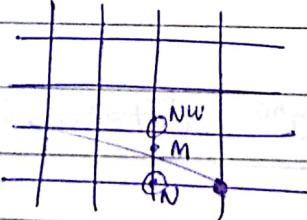
$$⑥ - ②$$

$$⑤ - ①$$

$$\Delta N^{2^I}_{\cancel{NW}} = 2a^2$$

$$\Delta NW^{2^I}_{\cancel{NW}} = 2(a^2 + b^2)$$

REGION - II



$$\Delta W = b^2(3 - 2x_t) - ⑦$$

$$\Delta NW = b^2(3 - 2x_t) + 2a^2(y_t + 1) - ⑧$$

Page No. _____
Date _____

$$d^{\text{II}} = f(x_t - 1, y_t + \frac{1}{2})$$

Let $d^{\text{II}} < 0$ selection is NW

$$(x_t - 1, y_t + 1)$$

~~$$\Delta W_{\text{NW}} = b^2 (5 - 2x_t) \quad \text{--- (9)}$$~~

$$\Delta NW_{\text{NW}} = b^2 (5 - 2x_t) + 2a^2 (2 + y_t) \quad \text{--- (10)}$$

$$(9) - (7)$$

$$(10) - (8)$$

$$\Delta^2 W_{\text{NW}}^{(II)} = 2b^2$$

$$\Delta^2 NW_{\text{NW}}^{(II)} = 2b^2 + 2a^2 = 2(b^2 + a^2)$$

else $d^{\text{II}} > 0$ selection is N

$$\Delta W_N = b^2 (5 - 2x_t) \quad \text{--- (11)}$$

$$\Delta NW_N = b^2 (5 - 2x_t) + 2a^2 (y_t + 1) \quad \text{--- (12)}$$

$$(11) - (7)$$

$$(12) - (8)$$

$$\Delta^2 W_{\text{N}}^{(II)} = 2b^2$$

$$\Delta^2 NW_{\text{N}}^{(II)} = 2b^2$$

Bresenham's Approach

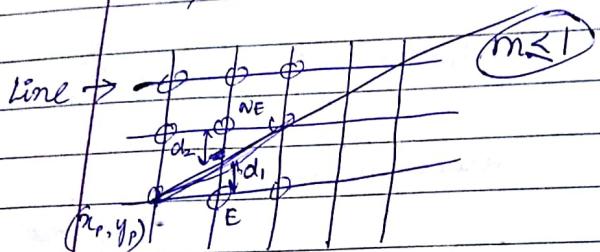
difficulties / issues

1. floating point computation
2. mid point approach (overcame floating point computation)
but $f(m) < 0$ } deciding this is
 > 0 all issue

purpose: 1) measurable quantity

↳ difference in measurable quantity will decide

↳ use the measurable quantity in finding the difference



$$d_1 = y - y_p = y - y_p - ①$$

$$\begin{aligned} y &= m(x_{p+1}) + c \\ &= m(x_p + 1) + c \end{aligned}$$

$$d_1 = m(x_p + 1) + c - y_p - \cancel{m} \cancel{x_p} - \cancel{c} - \cancel{y_p}$$

$$d_2 = (y_p + 1) - y - ②$$

$$D = d_1 - d_2 \quad \text{if } D < 0 \text{ we choose E.}$$

$D \geq 0$ we choose NE

$$d_1 - d_2 = y - y_p - y_p - 1 + y = 2(y - y_p) - 1$$

$$= 2(m(x_p + 1) + c - y_p) - 1$$

$$= 2\left(\frac{\Delta y}{\Delta x}(x_p + 1) + c\right) - y_p - (y_p + 1)$$

$$P = \Delta x * D = 2\left(\frac{\Delta y}{\Delta x}(x_p + 1) + c\right) - \frac{2(y_p + 1)}{\Delta x}$$

since $\Delta x > 0$ D will still determine the points

$$P = 2\Delta y x_p - 2\Delta x y_p + C$$

$$\text{where } C = 2(\Delta y + \Delta x c) - \Delta x$$

if $\Delta < 0$, selection is E

$$P_{\text{new}} = 2\Delta y(x_p + 1) - 2\Delta x(y_p) + C$$

$$\Delta E = P_{\text{new}} - P = 2\Delta y$$

else, selection is NE

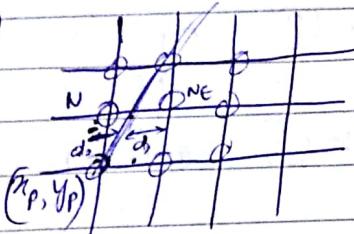
$$P_{\text{new}} = 2\Delta y(x_p + 1) - 2\Delta x(y_p + 1) + C$$

$$\Delta NE = 2\Delta y - 2\Delta x = 2(\Delta y - \Delta x)$$

$$P_{\text{initial}} = 2(\Delta y(x_p + 1) + \Delta x C) - \Delta x (2 \frac{\Delta y}{\Delta x} x_p + \frac{\Delta x}{\Delta y} C + 1)$$

$$\text{as } y_p = mx_p + c \\ = 2\Delta y x_p + 2\Delta y + 2\Delta x C - 2\Delta y x_p - 2\Delta x C - \Delta x \\ = 2\Delta y - \Delta x$$

$m > 1$



$$d_1 = x - (x_p + 1) \quad (x_p + 1) - x$$

$$d_2 = x - x_p \quad x - x_p$$

$$x = \frac{1}{m}([y_p + 1] - c)$$

$$P = d_1 - d_2 = (x_p + 1) - x - x + x_p = 2x_p - 2x + 1$$

$$P_{\text{initial}} = 2 \left[\frac{y_p - c}{m} \right] - \frac{2}{m} [y_p + 1] - c + 1$$

$$P_{\text{initial}} = 2\Delta x(y_p - c) - 2\Delta x(y_p + 1) - c + \Delta y \\ = \frac{\Delta y}{\Delta x} - 2\Delta x$$

$$D = 2x_p - 2\Delta x([y_p + 1] - c)$$

$$\Delta y \cdot D = 2\Delta y x_p - 2\Delta x((y_p + 1) - c)$$

$$P = 2\Delta y x_p - 2\Delta x y_p - 2\Delta x + 2\Delta x C$$

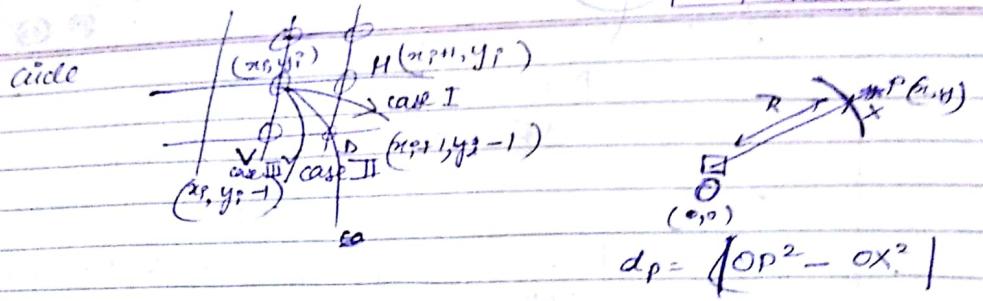
$$= 2\Delta y x_p - 2\Delta x y_p + C$$

If $P > 0$ we select N

$$P_{\text{new}} = 2\Delta y x_p - 2\Delta x(y_p + 1) + C$$

$$\Delta N = -2\Delta x$$

$$\text{Else } P_{\text{new}} = 2\Delta y(x_p + 1) - 2\Delta x(y_p + 1) + C \quad \Delta NE = 2(\Delta y - \Delta x)$$



$$d_P = \sqrt{x^2 + y^2 - R^2}$$

$$d_H = \sqrt{(x_i+1)^2 + y_i^2 - R^2}$$

case ①

$$OD^2 - R^2 < 0$$

$$d_D = \sqrt{(x_i+1)^2 + (y_i-1)^2 - R^2}$$

$$OD^2 < R^2$$

$$d_V = \sqrt{x_i^2 + (y_i-1)^2 - R^2}$$

case ② $OD^2 - R^2 = 0$

$$OD^2 = R^2$$

case ③ $OD^2 - R^2 > 0$

$$OD^2 > R^2$$

if one wants to select between H and D

$$\delta_{HD} = d_H - d_D$$

$$|(x_i+1)^2 + y_i^2 - R^2| - |(x_i+1)^2 + (y_i-1)^2 - R^2|$$

$$OD^2 - R^2 \geq 0$$

$$|OD^2 - R^2|$$

$$- |OD^2 - R^2|$$

$$OD^2 - R^2 \leq 0$$

$$(OD^2 - R^2)$$

$$- (R^2 - OD^2)$$

$$\text{open } | |$$

$$= (x_i+1)^2 + y_i^2 - R^2 - (R^2 - (x_i+1)^2 - (y_i-1)^2)$$

$$= 2(x_i+1)^2 + y_i^2 + (y_i-1)^2 - 2R^2$$

$$= 2((x_i+1)^2 + (y_i-1)^2 - R^2) + 2y_i - 1$$

$$\delta_{HD} = 2 \Delta D_i + 2y_i - 1$$

selection b/w v and D

$$S_{VD} = \Delta V - \Delta D = (DV^2 - R^2) - (DV^2 - R^2)$$

$$\text{case } DV^2 - R^2 > 0 \quad = \quad R^2 - DV^2 = DV^2 + R^2$$

$$DV^2 - R^2 < 0 \quad = \quad 2R^2 = (x_i^2 + (y_i - 1)^2)$$

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

$$= 2 [R^2 - (x_i + 1)^2 - (y_i - 1)^2]$$

$$+ 2x_i + 1$$

$$S_{VD} = 2x_i + 1 = 2\Delta D_i$$

$$\Delta D_i \rightarrow ((x_i + 1)^2 + (y_i - 1)^2 - R^2) \quad (x_i, y_i \text{ are changing after every choosing a point})$$

part I illuminating the pixel H
pt $\rightarrow (x_i + 1, y_i)$

$$\Delta D_{i, \text{new}} = ((x_i + 2)^2 + (y_i - 1)^2 - R^2)$$

$$\Delta H = \Delta D_{i, \text{new}} - \Delta D_i = (2x_i + 3)$$

part II illuminating pixel V
pt $\rightarrow (x_i, y_i - 1)$

$$\Delta D_{i, \text{new}} = (x_i + 1)^2 + (y_i - 2)^2 - R^2$$

$$\Delta V = \Delta D_{i, \text{new}} - \Delta D_i = (2y_i - 3)(-1) \\ = 3 - 2y_i$$

part III illuminating pixel D

$$\text{pt} \rightarrow (x_i + 1, y_i - 1)$$

$$\Delta D_{i, \text{new}} = ((x_i + 2)^2 + (y_i - 2)^2 - R^2)$$

$$\Delta D_D = \Delta D_{i, \text{new}} - \Delta D_i = (2x_i + 3) + (2y_i - 3)(-1) \\ = 2(x_i - y_i) + 6$$

initialisation if starting from $(0, R)$

$$\Delta_i = \frac{(1^2 + (R-1)^2 - R^2)}{2} \\ = 1 + R^2 + 1 - 2R - R^2 \\ = 2(1-R)$$

Metric Appraisal

measurable quantity is highest degree of equation

(x_i, y_i)

(\bar{x}_i+1, \bar{y}_i)

(\bar{x}_i, \bar{y}_i+1)

$$d_1 = y_i^2 - y^2$$

d_1 west E

$$d_2 = y^2 - (\bar{y}_i - 1)^2$$

d_2 west SE

$$\Delta_i = d_1 - d_2 = y_i^2 - y^2 + (\bar{y}_i - 1)^2$$

$$y^2 = R^2 - R^2$$

$$y^2 = R^2 - (\bar{x}_i + 1)^2 \Rightarrow$$

$$2y_i^2 + 1 - 2y_i - 2[R^2 - (\bar{x}_i + 1)^2]$$

$$\Delta_i^o = y_i^2 + (\bar{y}_i - 1)^2 - 2[R^2 - (\bar{x}_i + 1)^2] \quad \text{--- (1)}$$

if $\Delta_i^o < 0$ selection (\bar{x}_i+1, \bar{y}_i)

$$pt \rightarrow (\bar{x}_i+1, \bar{y}_i)$$

$$\Delta_{i+1}^o = \cancel{y_i^2} - y_i^2 + (\bar{y}_i - 1)^2 - 2[R^2 - (\bar{x}_i + 2)^2] \quad \text{--- (2)}$$

$$\Delta E = \Delta_{i+1}^o - \Delta_i^o = 2(2\bar{x}_i + 3)$$

Initialisation $\Delta_i^o(0, R) \rightarrow (R^2 + (R-1)^2 - 2(R^2 - 1))$

$$= R^2 + R^2 + 1 - 2R - 2R^2 + 2 \\ = 3 - 2R$$

else selection (SE) \rightarrow (x_{i+1}, y_{i+1})

$$\begin{aligned} D_{i+1} &= (y_i - 1)^2 + (y_i - 2)^2 \rightarrow 2[R^2 - (x_i + 2)^2] - 8 \\ \text{ASE} &= (2y_i - 2)(-2) - 2[R^2 - (x_i + 2)^2 + (y_{i+1})^2] \\ &= 4(1 - y_i) - 2(2x_i + 3)(-1) \\ &= 4(1 - y_i) + 2(2x_i + 3) \end{aligned}$$

Algo \rightarrow $x = 0, y = R, d = 3 - 2R$

while ($x \leq y$) {
if $d < 0$
 $d+ = 2 \times (x+3)$

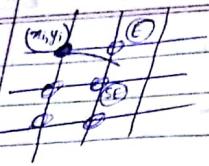
else {
 $d+ = 2 * (2x+3) + 4 * (1 - y_i)$

$y \leftarrow$

$x++$

3

$$\begin{aligned} (x_i, y_i) &\quad (x_{i+1}, y_i) \quad d_1 = (x_{i+1})^2 + y_i^2 - (x_{i+1})^2 - y_i^2 + R^2 \\ &\quad (x_{i+1}, y_{i+1}) \quad d_2 = y_i^2 - y_{i+1}^2 \\ (x_{i+1}, y_{i+1}) &\quad d_2 = (x_{i+1})^2 + y_{i+1}^2 - (x_{i+1})^2 - (y_{i+1})^2 \\ &\quad = y^2 - (y_{i+1})^2 \end{aligned}$$



$$\Delta E = 2 * (2x_i + 3)$$

$$\Delta SE = 4 * (1 - y_i) + 2 * (2x_i + 3)$$

(0, R) Initialisation

$$\Delta E = 6$$

$$\Delta SE = 4(1-R) + 6 = 10 - 4R$$

$$= 2(5 - 2R)$$

if $d < 0$ selection (E)

pt @E $(x_i + 1, y_i)$

$$\Delta E_{\text{new}} = 2(2(x_i + 1) + 3)$$

$$\boxed{\Delta^2 E^{(E)} = 4}$$

$$\Delta SE_{\text{new}} = 4(1 - y_i) + 2(2(x_i + 1) + 3)$$

$$\boxed{\Delta^2 SE^{(E)} = 4}$$

else selection (SE)

pt $(x_i + 1, y_i - 1)$

$$\Delta E_{\text{new}} = 2(2(x_i + 1) + 3)$$

$$\boxed{\Delta^2 E^{(SE)} = 4}$$

$$\Delta SE_{\text{new}} = 4(1 - (y_i - 1)) + 2(2(x_i + 1) + 3)$$

$$\boxed{\Delta^2 SE^{(SE)} = 8}$$

$$x=0, y=R, d=3-2R$$

$$\text{while } (x \leq y) \{ \quad \Delta E = 6 \quad \Delta SE = 2(5 - 2R)$$

if ($d < 0$) {

$$d+ = \Delta E$$

$$\Delta E+ = 4$$

$$\Delta SE+ = 4$$

else {

$$d+ = \Delta SE$$

$$\Delta E+ = 4$$

$$\Delta SE+ = 8$$

$$x++;$$

$$y--3$$

$$(y_i - \epsilon)(y_i + \epsilon) + (\epsilon - 1)(y_i + \epsilon - 1)$$

$$\epsilon(y_i + \epsilon) + (\epsilon - 1)(y_i + \epsilon - 1)$$

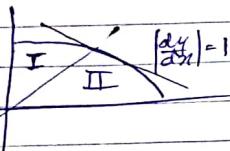
$$\frac{\epsilon}{\epsilon-1}(y_i + \epsilon) + (1-\epsilon)$$

Bresenham \rightarrow Ellipse

$$b^2 x^2 + a^2 y^2 = a^2 b^2$$

Page No. _____
Date _____

$$\frac{\epsilon}{\epsilon-1}(y_i + \epsilon) + (1-\epsilon)$$



Region I (x_i, y_i)

$$d_{(S)} = a^2 y_i^2 - a^2 y^2$$

$$d_{(SE)} = a^2 y^2 - a^2 (y_i - 1)^2$$

let $y_i - y = \epsilon$

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

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

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

$$+ (\epsilon - y - 1)(y_i + y - 1))$$

$$= a^2 (\epsilon(y_i + y) + (\epsilon - 1)(y_i + y - 1))$$

$$= a^2 [2\epsilon(y_i + y) + \epsilon - (y_i + y - 1)]$$

$$= a^2 ((y_i + y)(2\epsilon - 1) - (\epsilon + 1))$$

$$= a^2 ((2y_i - (y_i - y))(2\epsilon - 1) + (\epsilon + 1))$$

$$= a^2 (2y_i - \epsilon)(2\epsilon - 1) + (\epsilon - 1)$$

$$= a^2 (2y_i(2\epsilon - 1) - 2\epsilon^2 + \epsilon - 1)$$

$$= a^2 (b^2 + (b-1)^2) + 2b^2(2y_i + 3) - a^2 b^2$$

$$= a^2 (b^2 + b^2 + a^2 - 2a^2 b + 2b^2)$$

$$= a^2 (2y_i(2\epsilon - 1) + 2(\epsilon - \epsilon^2) - 1)$$

$$= a^2 (2y_i(2\epsilon - 1) + 2\epsilon(1 - \epsilon) - 1)$$

$$d_{(S)} = \cancel{b^2 x_i^2} - b^2 x_i^2 - b^2 y_i^2$$

$$d_{(SE)} = \cancel{b^2 x_i^2} - b^2 (x_i^2 + 1)^2 - b^2 (y_i + 1)^2$$

$$\Delta_i^I = d_{(S)} - d_{(SE)} = -(b^2 x_i^2 + b^2 (x_i^2 + 1)^2 - 2b^2 x_i^2)$$

$$= -b^2 (x_i^2 + (x_i + 1)^2) + 2(b^2 y^2 - a^2 b^2)$$

$$= 2(a^2 b^2 - a^2 y^2) - b^2 (x_i^2 + (x_i + 1)^2)$$

$$= 2(a^2 b^2 - a^2 (y_i - 1)^2) - b^2 (x_i^2 + (x_i + 1)^2)$$

if $D < 0$ select S

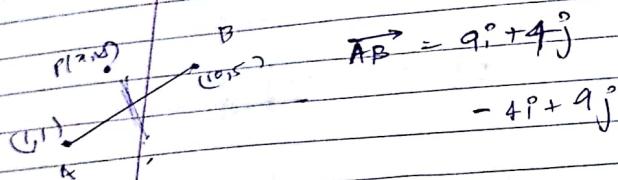
$$(x_i, y_i - 1) \quad \Delta_{i+1}^I = 2(a^2 b^2 - a^2 (y_i - 2)^2) \\ - b^2 (x_i^2 + (x_i + 1)^2)$$

$$\Delta S = 2 a^2 ((y_i - 1)^2 - (y_i - 2)^2) \\ = 2 a^2 (2y_i - 3)$$

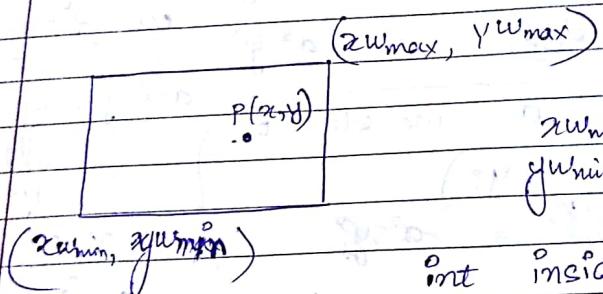
$$\text{else } 1/SE \quad x_{i+1}, y_{i-1} \quad \Delta_{i+1}^I = 2(a^2 b^2 - a^2 (y_i - 2)^2) - b^2 ((x_i + 1)^2 + (x_i + 2)^2)$$

$$\Delta SE = 2 a^2 (2y_i - 3) + b^2 (2x_i + 2)(2) = 2 a^2 (2y_i - 3) + 4b^2 (x_i + 1)$$

$$a^2 (y_i^2 + (y_r - y)^2 - 2y^2)$$



Clipping

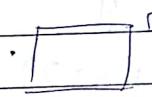


if P lies inside or on rectangle

$$x_{w\min} \leq x \leq x_{w\max}$$

$$y_{w\min} \leq y \leq y_{w\max}$$

Point inside (point P , A, B) {



return 0 when outside

return 1 when inside

Y

For a line with P_1 and P_2 end points we will get either

P_1, P_2 , P_1, P_2 , P_1, P_2 , P_1, P_2

OO , 01 , 10 , 11

line outside

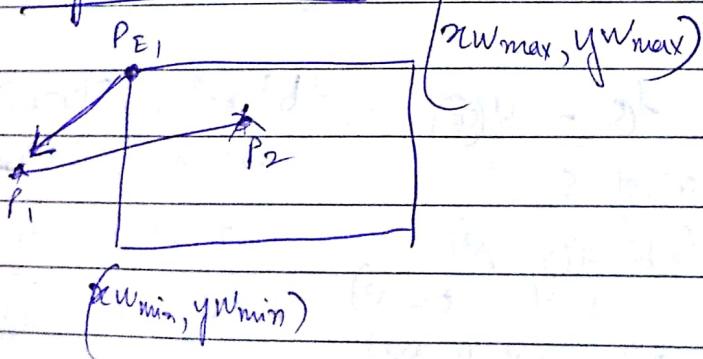
line inside

line or

may pass through

clipping should be performed

Cohen - Beck

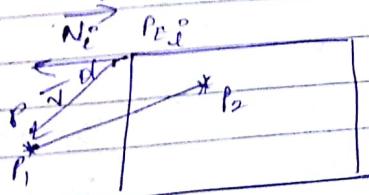


parametric eqn

$$P(t) = P_1 + (P_2 - P_1)t \quad 0 \leq t \leq 1$$

vector \vec{v} from P_E to line $P_1 P_2$ such that it swing from P_1 to P_2 and reverse

$$\vec{v} = P(t) - P_E$$



N_i is normal to any edge i

$$\vec{v} \cdot N_i = 0$$

$\cos \theta > 0$ that means $0 \leq 90^\circ$
or point outside edge

At intersection $\vec{v} \cdot N_i = 0 \quad (P(t) - P_{Ei}) \cdot N_i = 0$

$$(P_1 + (P_2 - P_1)t - P_{Ei}) \cdot N_i = 0$$

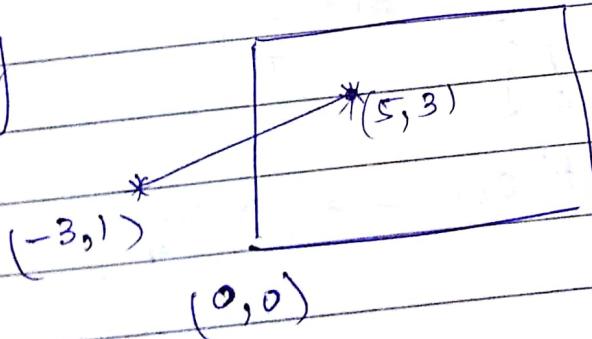
$$(P_1 - P_{Ei}) \cdot N_i + ((P_2 - P_1)t) \cdot N_i = 0$$

$$t = \frac{N_i \cdot (P_1 - P_{Ei})}{(P_2 - P_1) \cdot N_i}$$

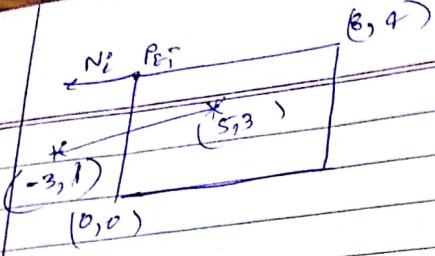
denominator
 $N_i \cdot ((P_2 - P_1) \cdot t)$

(8, 4)

Eg



We take this



left Edge

$$N_i = -i$$

$$P_1 = -3i + j$$

$$P_2 = 5i + 3j$$

$$P_{EP} = 4^\circ$$

$$\begin{aligned} t &= \frac{(-i)(-3i + j - 4j)}{(-i)(-3i + j - (-3i + j))} \\ &= \frac{(-i) \cdot (-3i - 3j)}{i \cdot (8i + 2j)} = \frac{3i(-i + j)}{2i(4i + j)} \end{aligned}$$

$$t = \frac{3}{8}$$

denominator $N_i \cdot (P_2 - P_1) \neq 0$

$\hookrightarrow > 0 \rightarrow$ potentially exiting rectangle

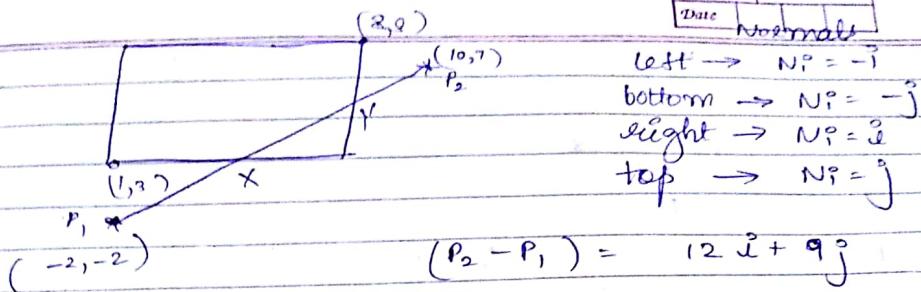
$< 0 \rightarrow$ potentially entering

$$\begin{aligned} P(t) &= P_1 + (P_2 - P_1)t \\ &= [-3i + j] + [5i + 3j + 3i - j] \frac{3}{8} \\ &= -3i + j + (8i + 2j) \frac{3}{8} \\ &= -3i + j + 3i + \frac{3}{4}j \\ &= \frac{7}{4}j \end{aligned}$$

thus intersection $\rightarrow (0, 7/4)$

can go from point to point
and enter

Eg 2



Page No. _____
Date _____

left $\rightarrow N_i = -\vec{i}$
bottom $\rightarrow N_i = -\vec{j}$
right $\rightarrow N_i = \vec{i}$
top $\rightarrow N_i = \vec{j}$

$$(P_2 - P_1) = 12\vec{i} + 9\vec{j}$$

Normal	$P_1 - P_{Ei}$	$N_i \cdot (P_1 - P_{Ei})$	$N_i \cdot (P_2 - P_1)$	t
left $N_i = -\vec{i}$	$P_{Ei} = (1, 8)\vec{i} + 8\vec{j}$ $-3\vec{i} - 10\vec{j}$	3	$-12 P_{exit}$	$3/12$
bottom $N_i = -\vec{j}$	$P_{Ei} = (1, 3)\vec{i} + 3\vec{j}$ $-3\vec{i} - 5\vec{j}$	-5	$-9 P_{exit}$	$5/9$
right $N_i = \vec{i}$	$P_{Ei} = (8, 3)\vec{i} + 3\vec{j}$ $-10\vec{i} - 5\vec{j}$	-10	$12 P_{exit}$	$10/12$
$\vec{N}_i + N_i = \vec{j}$	$P_{Ei} = (8, 8)\vec{i} + 8\vec{j}$ $-10\vec{i} - 10\vec{j}$	-10	$9 P_{exit}$	$10/9$

we will only take those t where $0 \leq t \leq 1$
we include 0

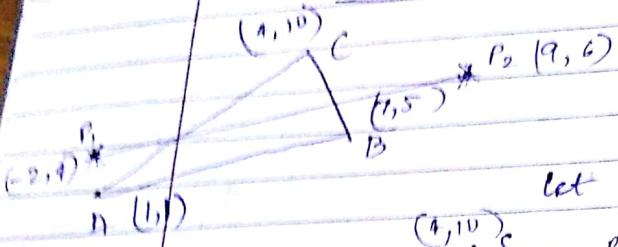
we make set of $S_1 \rightarrow (0, 3/12, 5/9)$ we find max = $5/9$
we include 1
 $S_2 \rightarrow (10/12, 1)$ we find min = $10/12$
 $= 5/6$

$$P(t) = P_1 + (P_2 - P_1)t$$

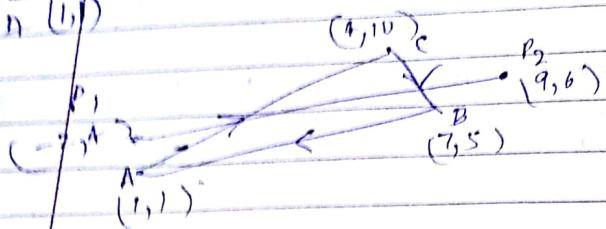
$$x = (-2, -2) + ((10, 7) - (-2, -2)) \frac{5}{9}$$

$$y = (-2, -2) + ((10, 7) - (-2, -2)) \frac{5}{6}$$

Non-Rectangular Window Cycles Back



let us do it clockwise



Edge

$$\vec{AC} = 3i + 9j$$

$$\vec{CB} = 3i - 5j$$

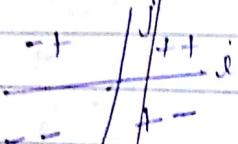
$$\vec{BA} = -6i - 4j$$

$$\vec{P_2 - P_1} = 11i + 2j$$

Edge	Normal	$P_E i$	$P_i - P_{E_i} i$	$N \cdot (P_i - P_{E_i})$	$N \cdot (P_2 - P_1)$
\vec{AC}	$3i + 9j$	$-9i + 3j$	$i + j$	$-3i + 3j$	$27 + 9 = 36$
\vec{CB}	$3i - 5j$	$5i + 3j$	$4i + 10j$	$-6i - 6j$	$-30 - 18 = -48$
\vec{BA}	$-6i - 4j$	$4i - 6j$	$7i + 5j$	$-9i - j$	$-36 + 6 = -30$
					$44 - 12 = 32$

~~Entel~~

See Normal



vector normal

+ -

+ +

- +

- -

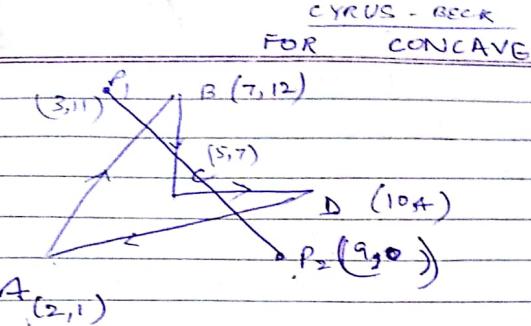
+ -

anticlockwise normal

t → 12/31 entel

48/61 exit

15/16 exit



Type No. _____
Date _____

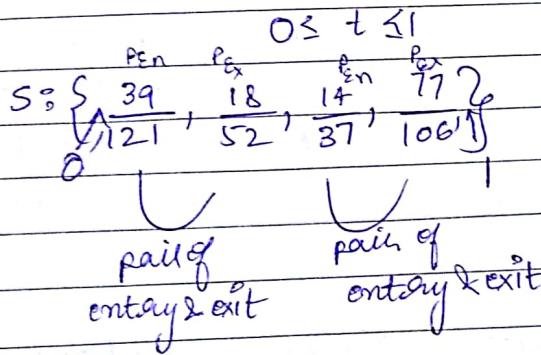
7420
0x 74-2

$$P_1 = 3i + 11j$$

$$P_2 - P_1 = \cancel{6i - 11j} \\ GP = 11j$$

Edge	Noemal	P_{E_i}	$P_i - P_{E_i}$	$N_i(P_i - P_{E_i})$	$N_i(P_2 - P_i)$
\overrightarrow{AB}	$5i + 11j$	$-11i + 5j$	$2i + j$	$i + 10j$	$-11 + 50 = 39$
\overrightarrow{BC}	$-2i - 5j$	$5i - 2j$	$7i + 12j$	$-4i + j$	$-20 + 2 = -18$
\overrightarrow{CD}	$5i - 3j$	$3i + 5j$	$5i + 7j$	$-2i + 4j$	$30 + 22 = 52$
\overrightarrow{DA}	$-8i - 9j$	$3i - 8j$	$10i + 4j$	$-7i + 7j$	$-6 + 20 = 14$
				$-21 - 56 = -77$	$(8 - 55) = -47$
					$18 + 88 = 106$

$$t =$$



\overrightarrow{AB}	$39/121$
\overrightarrow{BC}	$18/52$
\overrightarrow{CD}	$14/37$
\overrightarrow{DA}	$77/106$

(1)	(2)	COHEN - SUTHERLAND
1001	1000	$(x_{min}, y_{max}) \perp L \perp 0^{\circ}$
0001	0000	$0100^{(4)}$
		(x_{max}, y_{min})
0011	0010	$0110^{(6)}$

4 addresses

For a line P_1, P_2
 P_1 and P_2 would
in these 9 areas only

Input : P_1, P_2

Method : $\text{Code1} = \text{genCode}(\text{Point } P_1), \text{Code2}(\text{Point } P_2)$

$\Rightarrow \text{genCode}(\text{Point } P) \{$

$\text{if code} = 0;$
 $P_x < x_{min}$
 code 1 = LEFT

$P_x \geq x_{max}$
 code 1 = RIGHT

$P_y \geq y_{max}$
 code 1 = TOP

$P_y \leq y_{min}$
 code 1 = BOTTOM

done = 0

}

I $\{ (\text{code1} \& \text{code2} == 0) \&\& (\text{code1} | \text{code2}) == 0 \}$
 $\{ \text{if } (\text{code}(P_1) = 0000 \& \text{code}(P_2) = 0000) \}$

/* display line */

done = 1

// get out of recursion

II

if $(\text{code1} \& \text{code2} != 0)$

/* dont display line */

done = 1

// get out of recursion

III

if $((\text{code1} \& \text{code2} == 0) \&\& (\text{code1} | \text{code2} != 0))$

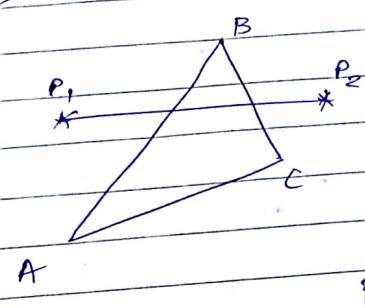
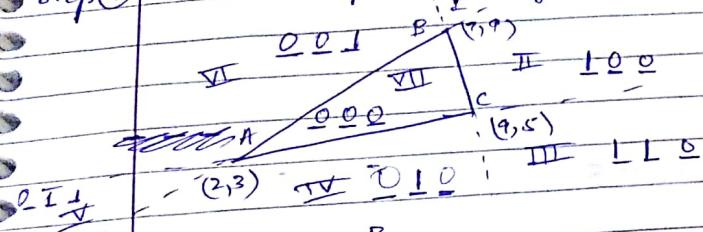
/* line clip() */

} while (done != 1)

Step(1)

Allocating the Region code for Address

Page No. _____
Date _____



code1 = genocode (P_1)

code2 = genocode (P_2)

genocode (Point A, Point B, Point C, P_1 , P_2)

code = 0

if

Edge AB $(2,3) \quad (7,9)$

$$f(x,y) \Rightarrow$$

$$\frac{6}{5} = \frac{y-3}{x-2}$$

$$6x - 12 - 5y + 15$$

$$6x - 5y + 3 = 0$$

~~sol took~~ ~~check~~

$$C(9,5) \times 0$$

$$\left. \begin{array}{l} -6(9) + 5(5) + 3 = 25 - 54 \\ = -29 \end{array} \right\}$$

$$AC \quad f(x,y) \quad \frac{2}{7} = \frac{y-3}{x-2}$$

$$2x - 4 = 7y - 21$$

$$2x - 7y + 17 = 0$$

$$B(7,9) \times 0$$

$$BC \quad f(x,y) = \frac{4}{-2} = \frac{y-5}{x-9}$$

$$4x - 36 = -2y + 10$$

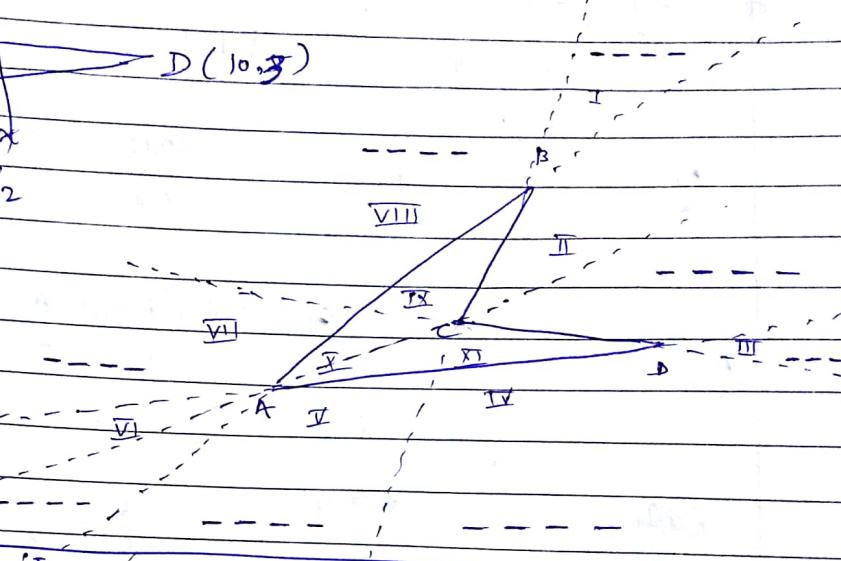
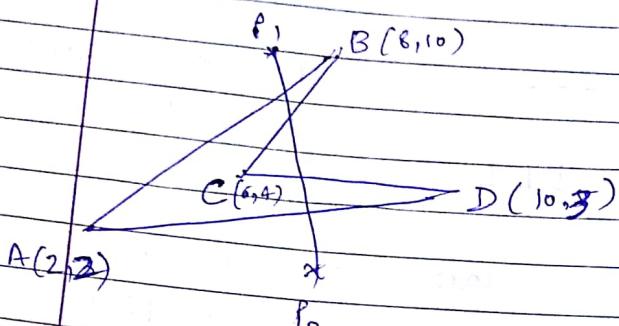
$$4x + 2y - 46 = 0$$

$$2x + y - 23 = 0$$

$$A(2,3) \times 0$$

gerade (point P)

wde \rightarrow 0×0



0101
 10111
 11

0101
 0110
 III
 VII
 $VIII$
 IX

0001
 0000
 XII
 $XIII$
 XIV

1001

1001

1000
 1010

time clipping

Page No. _____
Date _____

Violall - Lee - Violall

In rectangular window

9 regions \rightarrow 3 clusters

cluster-I

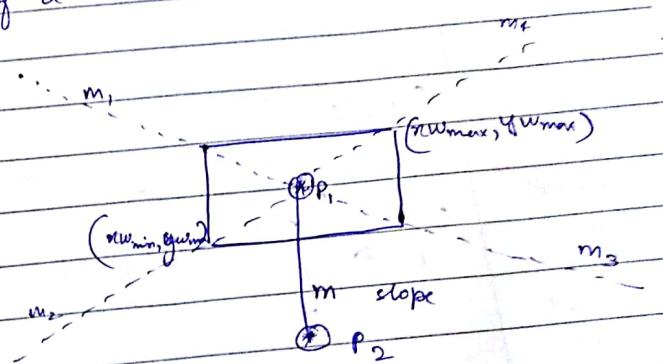
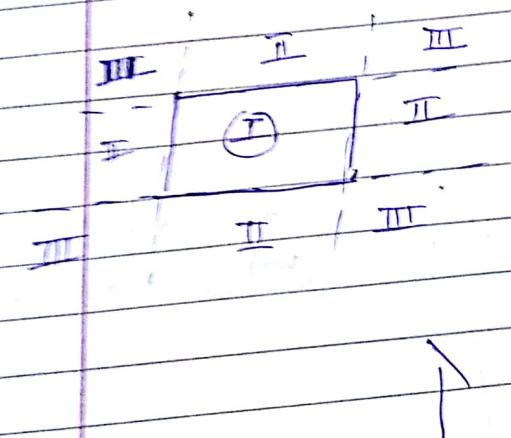
one of the end point of a line is at center/
or within the rectangular window is

cluster-II

one of the end point of a line is ~~at~~^{before} Edge
(left, Bottom, Right, Top)

cluster-III

one of the endpoint of a line lies at corner



$$m_1 \leq m \leq m_2 \quad m_2 \leq m \leq m_3 \quad m_3 \leq m \leq m_4 \quad m_4 \leq m \leq m_1$$

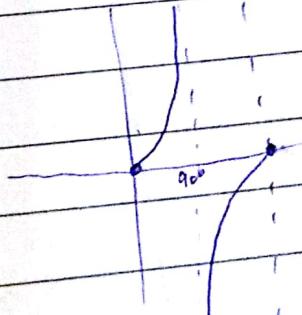
if

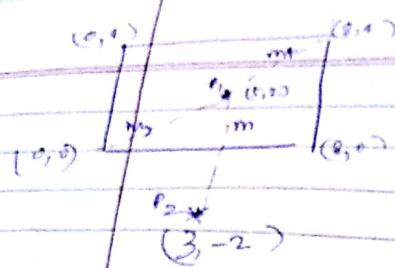
else

$P_2 - y_2 < y_{\min}$ no clipping

Clip

tan





$$m = \frac{-4}{-2} = 2$$

$$m_2 = \frac{-2}{-5} = 2/5$$

$$m_4 = \frac{2}{3}$$

$$m_3 = \frac{2}{-3}$$

~~0.1~~

~~5~~

~~2~~

~~3~~

if $(m < m_1) \wedge (m > m_2) \wedge (P_2 \cdot y_2 > P_1 \cdot y_1)$

if $P_2 \cdot y_2 < y_{\min}$

(* clipping is done *)

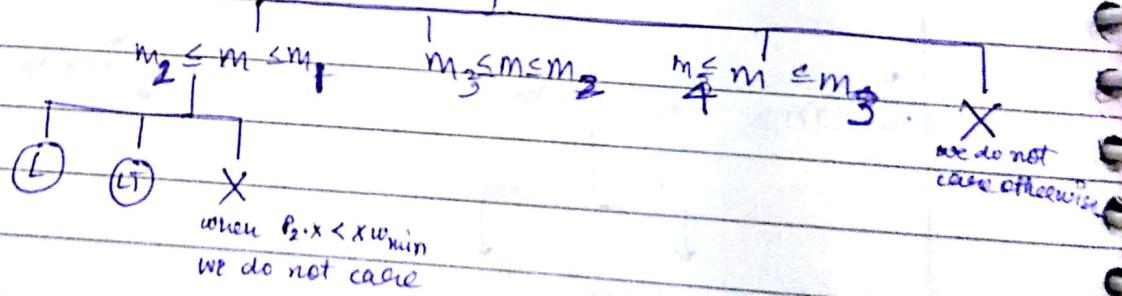
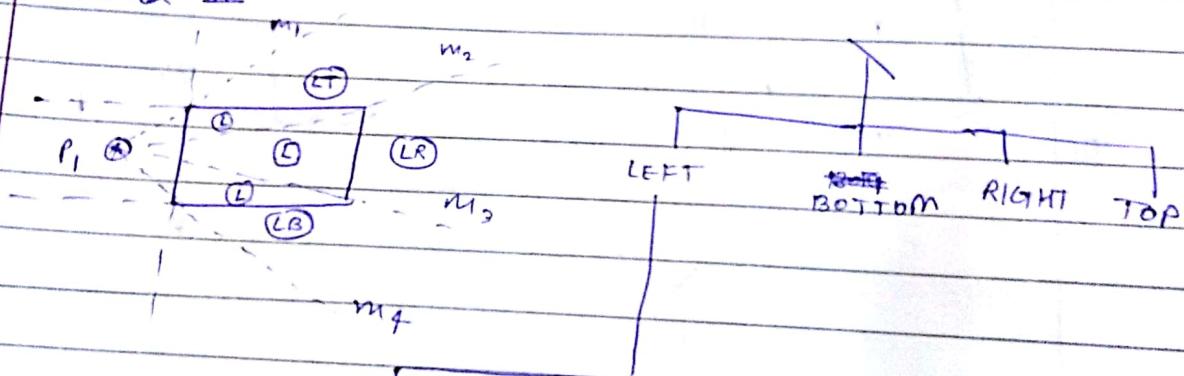
}

Cluster II

L means left

clipping

LT means left & top both

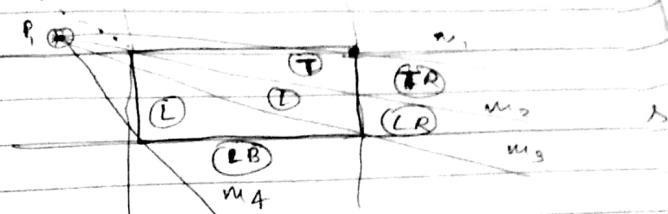
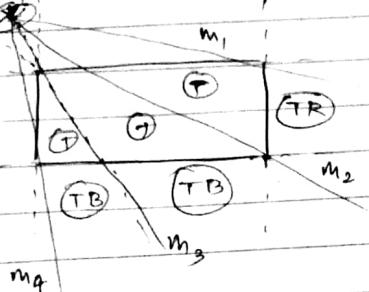


X
we do not care otherwise

cluster III

@ up

@ bottom



DMM

LTC

RTC LBC RBC

@ up

@ bottom

$$m_1 \leq m \leq m_2 \quad m_2 \leq m \leq m_3 \quad m_3 \leq m \leq m_4$$

T TR X

Non - Rectangular Window

C (14, 10)

A (2, 3)

B (10, 5)

m₃

$$m_1 \leq m \leq m_2 \quad m_2 \leq m \leq m_3 \quad m_3 \leq m \leq m_4$$

10⁸

Recursive

a) code1 & code2 = 0
line visible

b) code1 & code2 ≠ 0
line not visible

c) code1 & code2 = 0
recursive

mid point

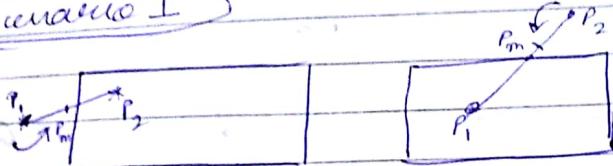
line
clipping

$$P_m = (P_1 + P_2)/2$$

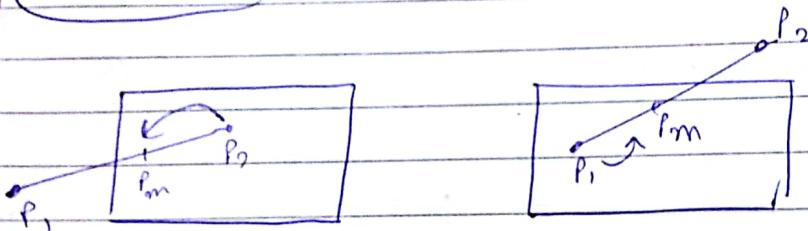
$$\Rightarrow P_1 P_m$$

$$\Rightarrow P_m P_2$$

Scenario I



Scenario II



Algorithm

input: $P_1, P_m, x_{w\min}, x_{w\max}, y_{w\min}, y_{w\max}$

method: $\text{code1} = \text{gencode}(P_1)$ $\text{code2} = \text{gencode}(P_2)$

step① if $\text{code1}, \text{code2}$ both = 0

draw $P_1 P_2$

step② if $\text{code1} \neq \text{code2}$!= 0

/* ignore line */

step③ $P_m = (P_1 + P_2)/2$

$\text{code } m = \text{gencode } (P_m)$

(P_1, P_2)

(P_1, P_m)
 (P_m, P_2)

Step(4) if code m != 0
then

if code 1 & code m != 0

then $P_1 \leftarrow P_m$; goto step(1)

else if code 2 & code m != 0

then $P_2 \leftarrow P_m$; go to step(1)

Step(5) if code m == 0

if code 1, code m both == 0

then consider $P_m P_2$

else if code 2, code m both == 0

then consider $P_1 P_m$

Step(6) consider $P_1 P_m$
do {

$$P_{m1} = (P_1 + P_m)/2;$$

code m1 = genode(P_{m1})

if ($\text{code } m1 \neq 0$) then $P_1 \leftarrow P_{m1}$,
else $P_m \leftarrow P_{m1}$,

3 while ($P_1 \cdot x \neq x_{\min}$ & $P_1 \cdot x \neq x_{\max}$ & $P_1 \cdot y \neq y_{\min}$
& $P_1 \cdot y \neq y_{\max}$)

$P_1 \leftarrow P_{m1}$

Step(7) consider $P_m P_2$

do { $P_{m2} = (P_2 + P_m)/2$; code m2 = genode(P_{m2})

if ($\text{code } m2 \neq 0$) then $P_2 \leftarrow P_{m2}$

else $P_m \leftarrow P_{m2}$

3 while ($P_2 \cdot n - - - - -$)

$P_2 \leftarrow P_m$



Page No. _____
Date _____ Juy _____

$E_j \rightarrow P_1, P_2$ $P_1(120, 5)$ $P_2(180, 30)$
 $\text{solution} = 100, \text{minmax} = 160, y_{\min} = 10, y_{\max} = 90$

ITER

P_1 P_2 P_m codem Remarks
 $(120, 5)$ code 1 $(180, 30)$ code 2 $(150, 18)$ 0000 consider P_1, P_m
 0010 0100 $(150, 18)$ 0000 consider P_m, P_2

$(120, 5)$ 0010 $(150, 18)$ 0000 $(135, 12)$ 0000

$(120, 5)$ 0010 $(135, 12)$ 0000 $(128, 9)$ 0010

$(128, 9)$ 0010 $(135, 12)$ 0000 $(132, 11)$ 0000

$(128, 9)$ 0010 $(132, 11)$ 0000 $(130, 10)$ 0000 $P_1 \in (130, 10)$

~~$(128, 9)$~~ ~~0010~~ $(130, 10)$ 0000

↑
boundary condition reached

consider P_m, P_2

$(150, 18)$ 0000 $(180, 30)$ 0100 $(165, 24)$ 0100

$(150, 18)$ 0000 $(165, 24)$ 0100 $(158, 21)$ 0000

$(158, 21)$ 0000 $(165, 24)$ 0100 $(162, 23)$ 0100

$(158, 21)$ 0000 $(162, 23)$ 0100 $(160, 22)$ 0000 $P_2 \in (160, 22)$

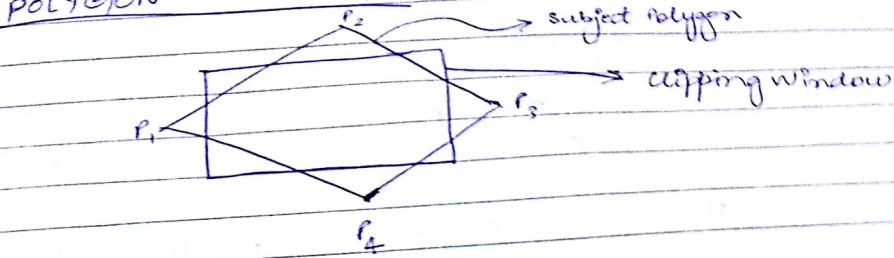
$(160, 22)$
↑

boundary condition reached

Liang-Barsky (Diy)

Date _____
Page No. _____

POLYGON CLIPPING

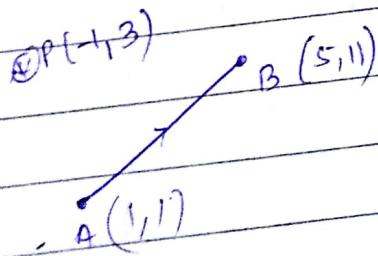
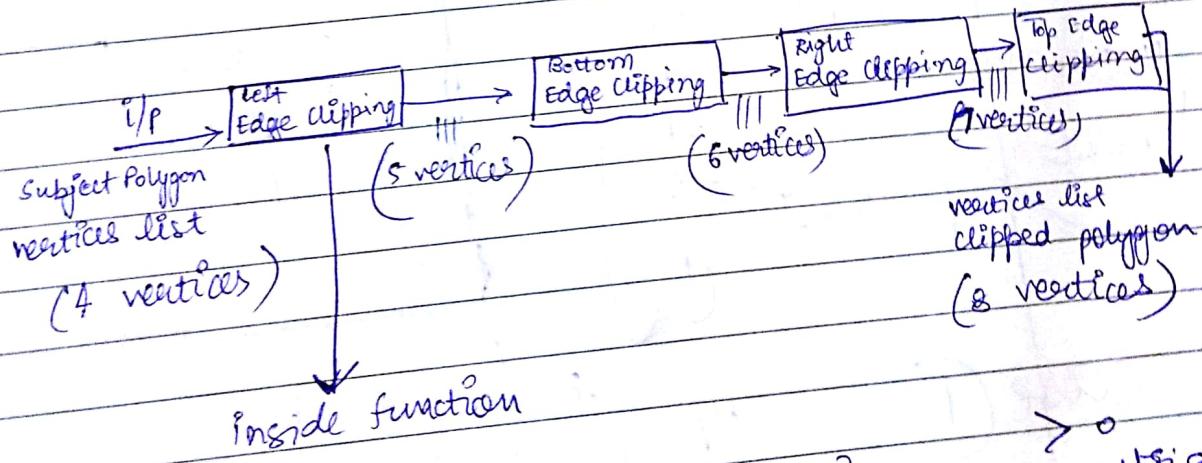
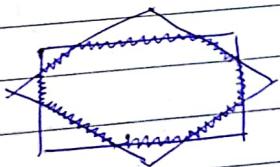


Input:

- ① subject Polygon (vertices list)
- ② clipping window (vertices list)

Method:

Output: clipped Polygon (list of vertices)



$$\vec{AB} = 4i + 10j$$

$$\vec{N_1}(A) = -10i + 4j$$

$$\vec{AP} = \vec{AP}$$

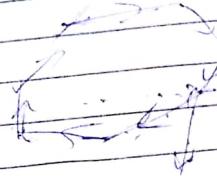
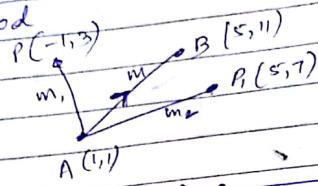
$$\vec{P} - \vec{P}_e(A) = -2i + 2j$$

$$N_1 \cdot (\vec{P} - \vec{P}_e(A)) = 20 + 8 = 28 > 0$$

outside
polygon

dot product > 0 then outside polygon else inside polygon

Analytic method



$\Rightarrow (m_1 \cdot m_2 > 0)$ if $(m_1 \cdot m_2 < 0)$ then outside else inside

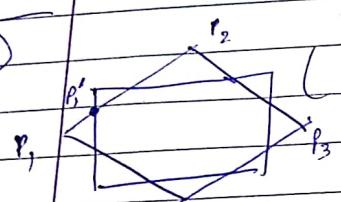
subject polygon list

vertex list 2

P₃

P₄

(P₁)



we are seeing
here only want
left edge

(not as whole
clipping window)

$$x_I = \min$$

$$y_I = y_1 + \frac{(y_2 - y_1)}{(x_2 - x_1)} (x_I - x_1)$$

repeat
the first vertex

P₁

P₂

P₃

P₄

(P₁)

i → o

left Edge

o → i

i → i

i → i

i → o

o → i

i → i

i → o

o → i

i → i

i → o

o → i

i → i

i → o

o → i

i → i

i → o

o → i

i → i

i → o

o → i

i → i

i → o

o → i

i → i

i → o

o → i

i → i

i → o

o → i

i → i

i → o

o → i

i → i

i → o

o → i

i → i

SUTHERLAND

HODGMAN

RULES

P₁ → P₂

P₂ → P₃

P₃ → P₄

P₄ → P₁

(P₁) → P₁

Rule

i → o

o → i

i → i

o → o

Intersection(I)

① Intersection point(I)

② second vertex (P₁)

second vertex (P₁)

- Nil -

case (1)

i/p

left Edge output

P₁ → P₂

P₁

P₂ → P₃

P₂

P₃ → P₄

P₃

P₄ → P₁

P₄

(P₁) → P₁

(P₁)

Now this

output

becomes

input

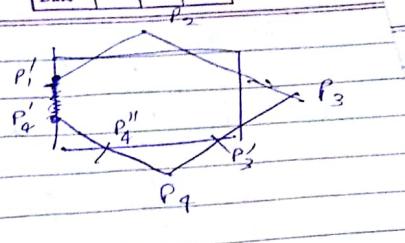
for next

edge

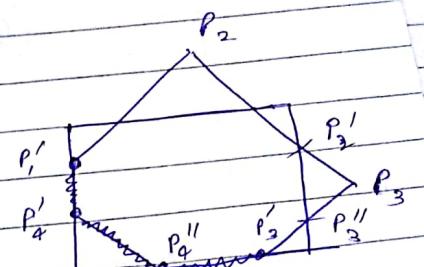
Front view
el

Page No.	
Date	

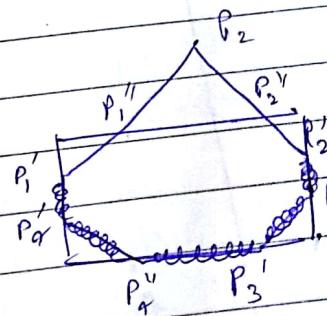
i/P	bottom edge
P_1'	$i \rightarrow i$
P_2'	$i \rightarrow i$
P_3'	$i \rightarrow o$
P_4'	$o \rightarrow ?$
P_4'	$i \rightarrow i$
(P_1')	$i \rightarrow i$
	$\underline{P_2}$

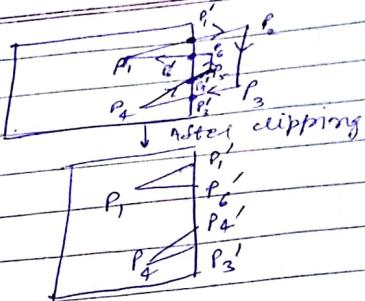
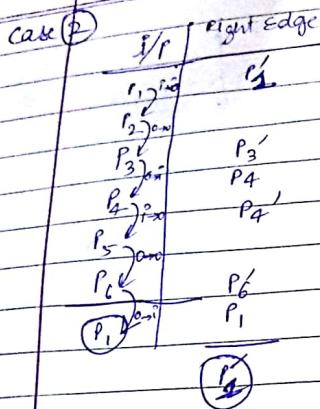


i/P	right edge
P_2'	$i \rightarrow o$
P_3'	$o \rightarrow i$
P_3'	$i \rightarrow i$
P_4''	$i \rightarrow i$
P_4'	$i \rightarrow i$
P_4'	$i \rightarrow i$
(P_2')	$i \rightarrow i$
	$\underline{P_2}$



i/P	top edge O/P
P_2'	$i \rightarrow i$
P_3''	$i \rightarrow i$
P_3'	$i \rightarrow i$
P_4''	$i \rightarrow i$
P_4'	$i \rightarrow i$
P_4'	$i \rightarrow i$
P_1'	$i \rightarrow o$
P_2	$o \rightarrow i$
(P_2')	$i \rightarrow i$
	$\underline{P_3''}$





output expected

Received hex

Thus, we need to modify

Bit Traversal

we have
set $BPt=1$
whenever
there was
 $P \rightarrow 0$

P_1'	1	0
P_3'	0	0
P_4'	0	0
P_1'	1	0
P_6'	0	0
P_1'	0	0
P_1'	0	0

intersection points occur
only with $P \rightarrow 0$
or $0 \rightarrow i$

step (2) (Filtered)

intersection
points

P_1'
P_3'
P_4'
P_6'

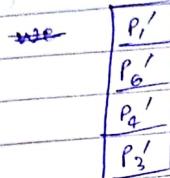
sort acc to decreasing
y coordinate

P_1'
P_6'
P_4'
P_3'

Bit Traversal		
P_1'	1	0
P_3'	0	0
P_5'	0	0
P_7'	1	0
P_9'	0	0
P_1'	0	0
P_1'	1	0

Traversal = 0 means not yet traversed

Page No.	90
Date	17/09/2023



we start from here

since $\text{bit}^t = 1$ here, we go to the filtered array

Bit Traversal		
P_1'	1	01
P_3'	0	01
P_5'	0	01
P_7'	1	0
P_9'	0	01
P_1'	0	01
P_1'	1	01

we do
line(P_1' , P_6')

& set traversal $P_1' = 1$
and check for P_6'
now

For P_6' traversal = 0 & $\text{bit}^t = 0$, thus we make
 P_6' and P_1'

line(P_6' , P_1')

line(P_1' , P_1')

when we reach P_1' , we have already traversed
 P_1' in line(P_1' , P_6'), thus we traverse till
we reach a traversal = 0 i.e. P_3'

line(P_3' , P_4')

line(P_4' , P_4')

At P_4' traversal = 0 but $\text{bit}^t = 1$, hence
move to filtered array

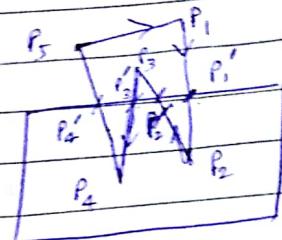
line(P_4' , P_3')

Now when we go to P_3' , it has already
traversal = 1.
We again go in loop to check if traversal = 0
anywhere

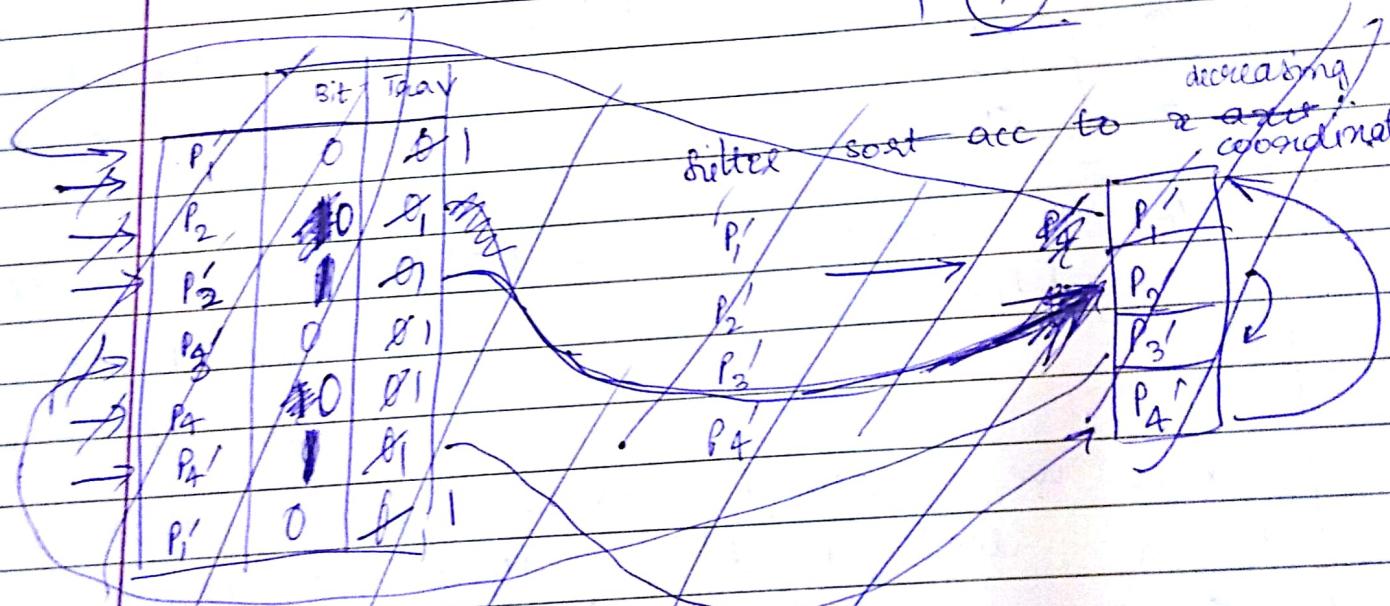
since its not we end here

Finally → line (P_1', P_6')
 line (P_6', P_1)
 line (P_1, P_1')
 line (P_3', P_4)
 line (P_4, P_4')
 line (P_2', P_3')

CASE (3)



Top edge	
P_1'	
P_2	
P_3'	
P_2'	
P_3	
P_4	
P_4'	
P_5	
P_5'	
(P_1)	
$\underline{(P_1')}$	

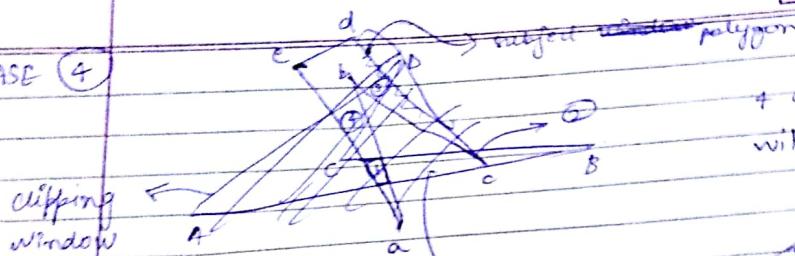


- 1) line (P_1', P_2')
- 2) line (P_2', P_2')
- 3) line (P_2', P_3')
- 4) line (P_3', P_4')
- 5) line (P_4', P_4')
- 6) line (P_4', P_1')

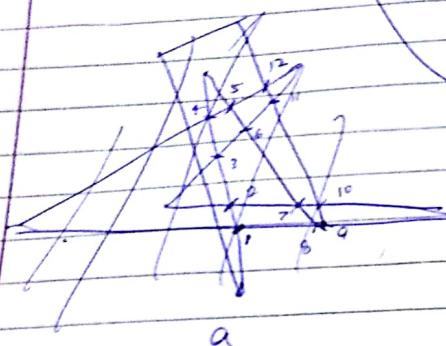
Weiler Atherton

Page No. _____
Date 10/10/18

CASE 4



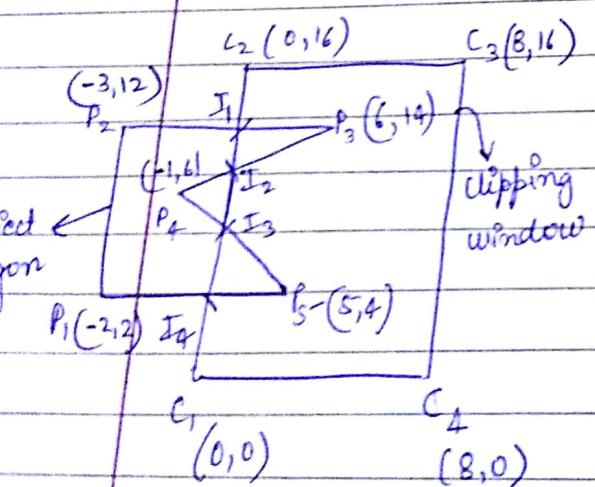
+ areas that will be visible



subject Polygon (SP) \rightarrow a b c d e
CW \rightarrow A B C D

SP \rightarrow a 1 2 3 4 b 5 6 7 8 c 9 10 d e 11 12 13 14

CW \rightarrow A 1



Weiler Atherton

Edge

$$\begin{aligned} &C_1 C_2 (16j) \\ &P_1 P_2 (-1i + 10j) \\ &P_2 P_3 (9i + 2j) \\ &P_3 P_4 (-7i + 8j) \\ &P_4 P_5 (6i - 2j) \\ &P_5 P_1 (-7i - 2j) \end{aligned}$$

Normal

$$\begin{aligned} &-10i - j \\ &-2i + 9j \\ &8i - 7j \\ &2i + 6j \\ &2i - 7j \end{aligned}$$

$$t = \frac{N_j - P_i - B}{-(B - R)t + N_i}$$

P_i^+

$-2i + 2j$

$-3i + 12j$

$6i + 14j$

$-i + 6j$

$5i + 4j$

from
or from
going
measures

Page No.	
Date	

$$C_2 - C_1 = 16^{\circ}$$

	$N_i \cdot (P_i - P_{E_i})$	$N_i(C_2 - C_1)$	t
$2P - 2J^{\circ}$	$-20 + 2 = -18$ Enter	-16	$18/16 = 9/8$
$3P - 12J^{\circ}$	$-6 - 108 = -114$ Exit	144	$114/144 = 13/12$
$-6P - 14J^{\circ}$	$-48 + 98 = 50$ Enter	-112	$50/112 = 25/56$
$P - 6J^{\circ}$	$2 + 36 = -34$ Enter	96	$178/96 = 48$
$-5P - 4J^{\circ}$	$-10 + 28 = 18$ Enter	-112	$18/112 = 9/56$

$$t \left(\frac{19}{24} + \frac{30}{112} + \frac{1}{12} + \frac{9}{56} \right)$$

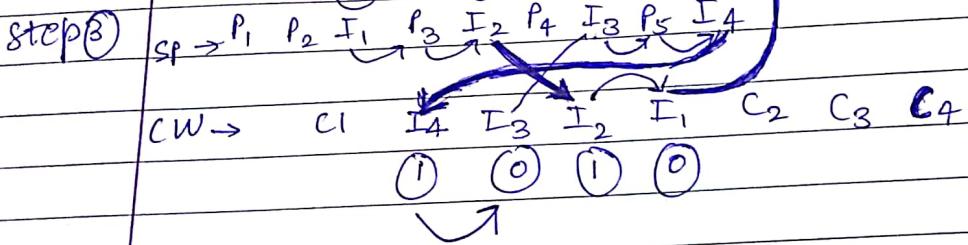
↓ ↓
Entry Exit

step ① SP → $P_1 P_2 P_3 P_4 P_5$
 CW → $C_1 C_2 C_3 C_4$

- step ②
- a) Intersection points evolve
 - b) Entry / Exit note

using t values we
 arrange the points
 and mark the entry &
 exit

Entry points are
 flagged with ①
 & exit with ②



Here we
 write off of
 SP

{ We sorted using t values as $C_1 C_2$ is the ~~clip~~
 line to be clipped for our subject polygon as clipping
 window }

Entry list { I_4, I_3, I_2 }

we start from a point

- step ④
- line (I_1, P_3) (traversing on subject polygon)
 - line (P_3, I_2) (I_2 is exit point so move to CW)
 - line (I_2, I_1) (traverse on CW) -
 - line (I_3, P_5) (traversing on SP)

GP from I₁ from I₂ give intersection

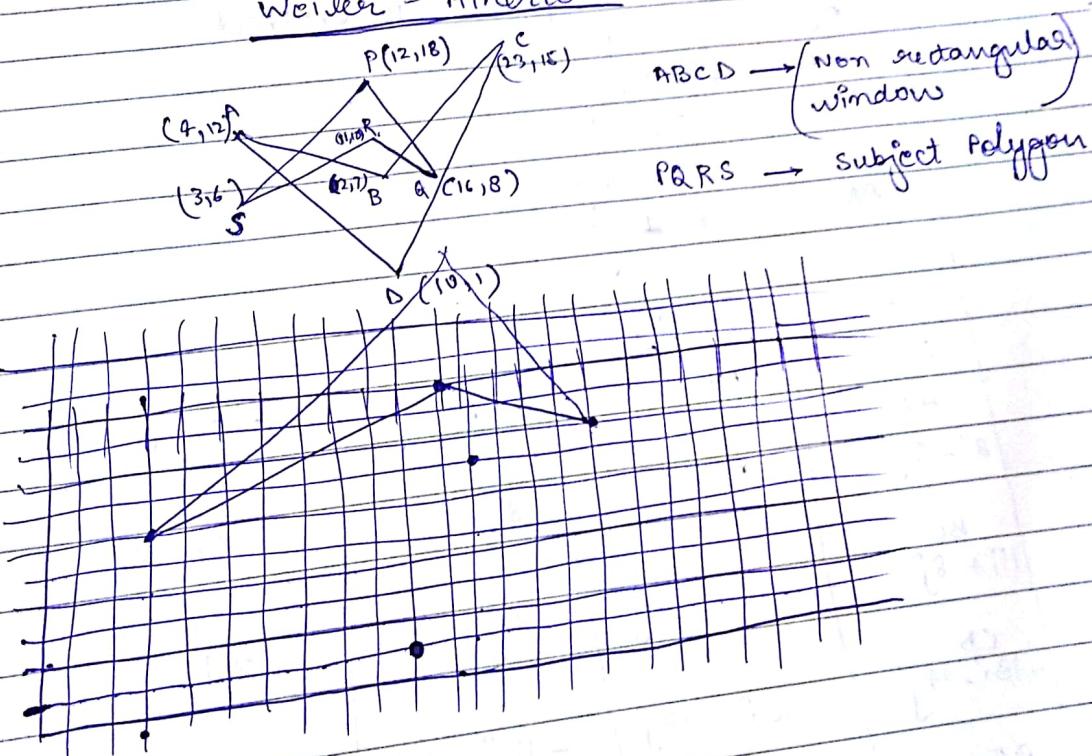
Page No.	
Date	

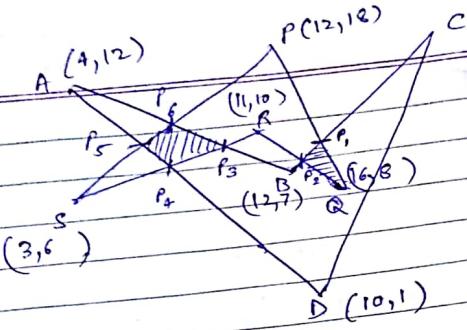
line (P₅, I₁) (thus I₁ is exit point connection)
line (I₄, I₂) (traverse on cw)

→ Here we saw that I₁ is already visited
thus we move to the next intersection point in our Entry Set.

→ the same case is repeated here but since no more intersection points in our entry set, we conclude our process here.

Weiler - Atherton





Page No. _____
Date _____

Step ① input : { CW AB CD
EP PQ RS }

Method :

Step ② intersection point is evolved

$$PQ \quad 4i - 10j$$

$$AB \quad 8i - 5j$$

$$BC \quad 11i + 8j$$

$$CD \quad -13i - 14j$$

$$DA \quad -6i + 11j$$

PQ	$4i - 10j$	Edges	Normal	$C_1 - P_{ti}$	$N_t(C_1 - P_{ti})$	$N_t(C_2 - C_1)$
AB						
$8i - 5j$	$8i + 8j$			$8i + 6j$	$40 + 48 = 88$	$20 - 80 = -60$
BC						
$11i + 8j$	$-8i + 11j$			$11j$	121	$-32 - 110$ $= -142$
CD						
$-13i - 14j$	$14i - 13j$			$-11i + 3j$	$-154 - 39 = -193$	$56 + 130$ $= 186$
DA						
$-6i + 11j$	$-11i - 6j$			$2i + 17j$	$-22 - 102 = -124$	$-44 + 60$ $= 16$

From
or
going
inwards.

$\frac{208}{253}$ $\frac{11}{12}$ $\frac{314}{12}$
Page No. Date

[S P : P P₁ Q]

$$RS = -8i - 4j$$

	Edge Normal	$C_1 - P_{t1}$	$N_t(C_1 - P_{t1})$	$N_t(C_2 - C_1)$	T	BITes
AB	$5i + 8j$	$7i + 2j$	$35 - 16 = 19$	$-40 - 32 = -72$	Entering $1\frac{1}{12} \times 0$	
BC	$-8i + 11j$	$-i + 3j$	$8 + 33 = 41$	$64 - 44 = 20$	Exit $-\frac{1}{2} \times 0$	
CD	$14i - 13j$	$-12i - 5j$	$-168 + 65 = -103$	$-112 + 52 = 60$	Entering $\frac{103}{60} \times 0$	
AA	$-6i + 11j$	$i + 9j$	$-11 - 54 = -65$	$88 + 24 = 112$	Exit $\frac{65}{112} \times 1$	

P	P ₁	Q	R	P ₃	P ₄	S
-1	0	-1	-1	0	1	-1

Bit	t	
Entering	X	> 1
Entering	0	($12\frac{1}{142}$) \rightarrow as entering
Exit	X	> 1
Exit	X	> 1

From here I am changing
notation of entering/exiting by multiplying
denominator by -1

$$SP = 9i + 12j$$

Edge	Normal	$C_1 - P_t$	
AB	$-5i + 8j$	$-i - 6j$	
BC	$-8i + 11j$	$-9i - j$	
CD	$14i - 13j$		
DA	$-13i - 19j$		
	$-6i + 11j$		

$$\frac{t}{53/11} \quad 1$$

$$-\frac{5}{6}$$

$$\frac{163}{30}$$

$$\frac{47}{171}$$

O

$$\frac{19}{72} \quad \frac{65}{112} \quad \frac{43}{112} \quad \frac{53}{11}$$

SP:

P	P ₁	Q	P ₂	R	P ₃	P ₄	S	P ₅	P ₆
-1	0	-1	1	-1	0	1	-1	0	1
0	0	0	0	0	0	0	0	0	0

for P → Q → R → S

for A → B → C → D → A

SP → updated list

for A → B → C → D

~~for~~ for P → Q → R → S → P

CW → updated list

CW:

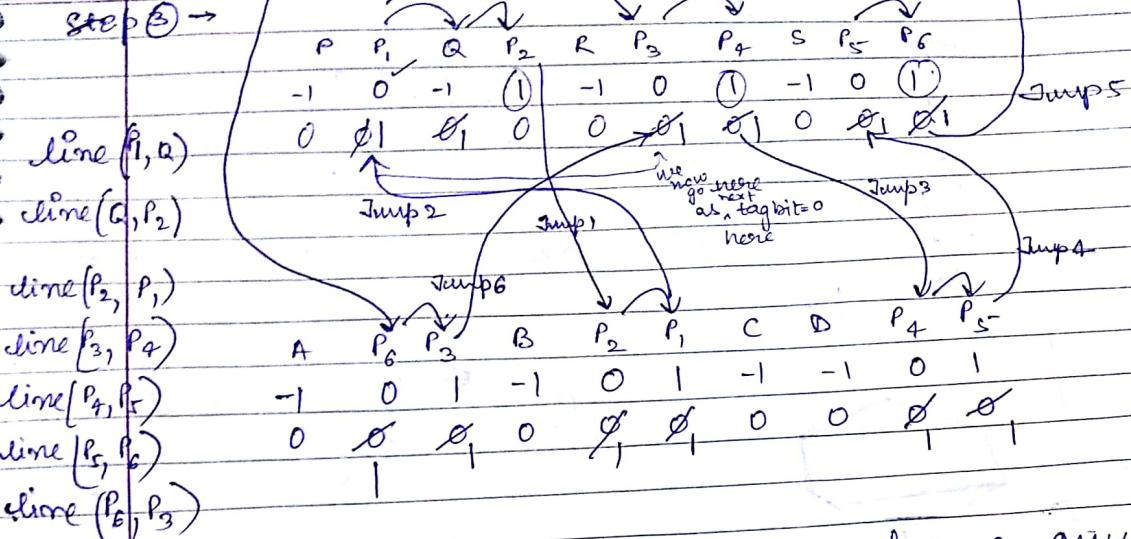
A	P ₆	P ₃	B	P ₂	P ₁	C	D	P ₄	P ₅
-1	0	1	-1	0	1	-1	-1	0	1
0	0	0	0	0	0	0	0	0	0

Traversal
of bit

From outer towards inner

Page No. _____
Date _____

Step 3 →



since tag bit = 0 has no longer any point where traversal = 0, thus we end

§ No more untraversed point in the entire set of subject Polygon

Q1
Q2
Q3
Q4
Q5
Q6
Q7
Q8
Q9
Q10

(100, 100)
(100, 400)

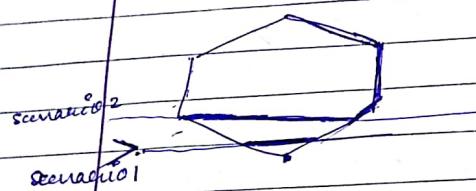
POLYGON FILLING

Approach - I

scanning = more vertical
or in any angle

cases: non-Horizontal Edges

$V_1(8, 13)$
 $V_2(2, 9)$
 $V_3(10, 6)$
 $V_4(6, 11)$
 $V_5(1, 1)$
 $V_6(4, 1)$



GLOBAL EDGE TABLE

structure → Edge

$y_{\max} | x_{y_{\min}} | Y_m | \rightarrow$

Example for $V_5 - V_6$ here

$| 4 | 7 | -2 | \rightarrow$

BOTTOM TO TOP

$y_{\max} = 13$

$\begin{array}{|c|c|c|c|} \hline 11 & \rightarrow & | 3 | 10 | -1 | \rightarrow \lambda \\ \hline 9 & \rightarrow & | 3 | 1 | 7 | 4 | \rightarrow \lambda \\ \hline 6 & \rightarrow & | 1 | 10 | 0 | \rightarrow \lambda \\ \hline 4 & \rightarrow & | 9 | 1 | 0 | \rightarrow \lambda \\ \hline 3 & \rightarrow & \lambda \\ \hline 2 & \rightarrow & \lambda \\ \hline \end{array}$

$y_{\min} = 1$

$\begin{array}{|c|c|c|c|} \hline 4 & 7 & -2 & \rightarrow \\ \hline V_5 & V_6 & & \\ \hline \end{array}$

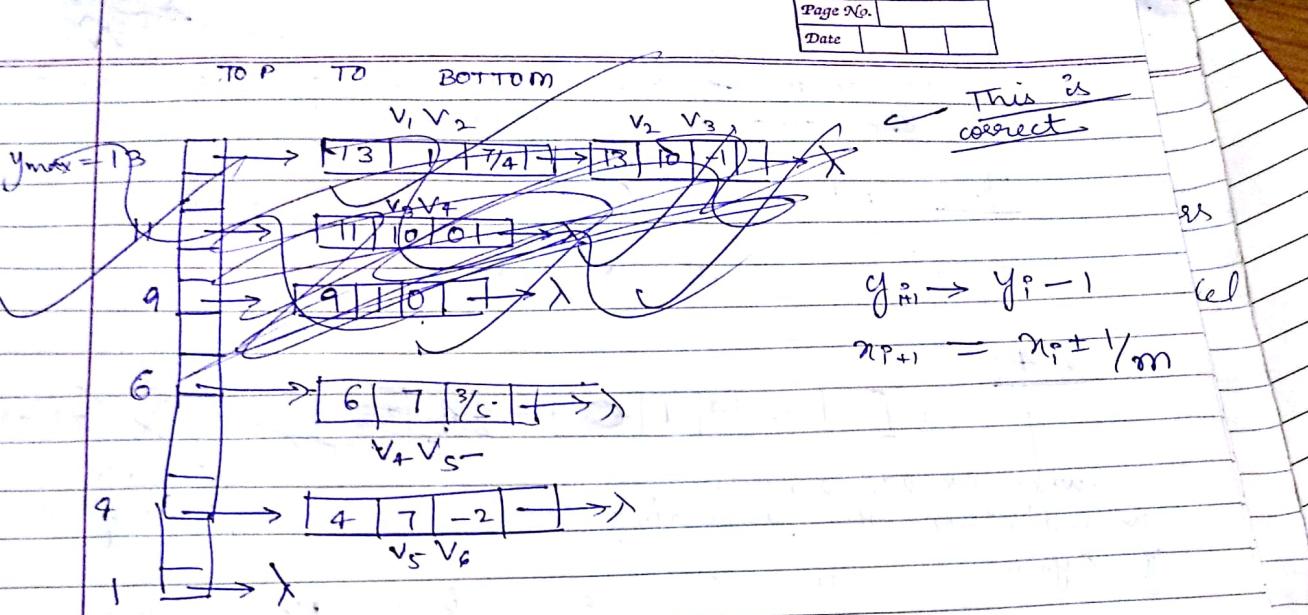
$y_{i+1}^o \rightarrow y_i^o + 1$

$x_{i+1}^o \rightarrow x_i^o + 1 / m$ all others are set to λ

(we only store unique edges & do not repeat any edge)

\uparrow
NULL

We kept $V_5 - V_6$ before $V_5 - V_4$ as we are going from left to right

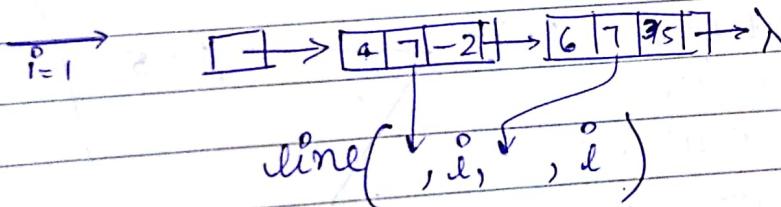


Active Edge Table (AET) [For filling the polygon]

Using Bottom to Top
It starts with an empty node & ends with an empty node

NULL

for $P \rightarrow y_{\min}$ to y_{\max}



Rules

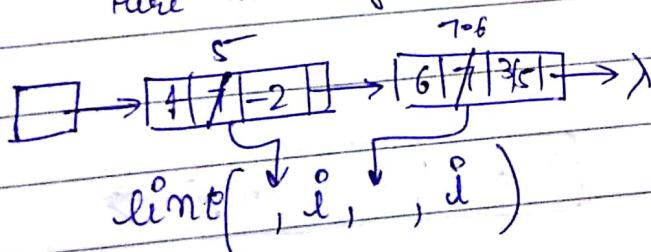
1. merge
2. If nodes are old, update them

$x_{i+1} \rightarrow x_i \pm 1/m$

draw line

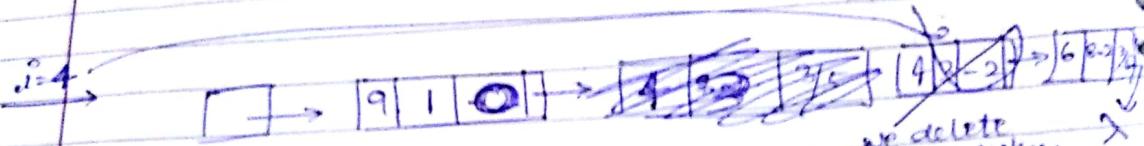
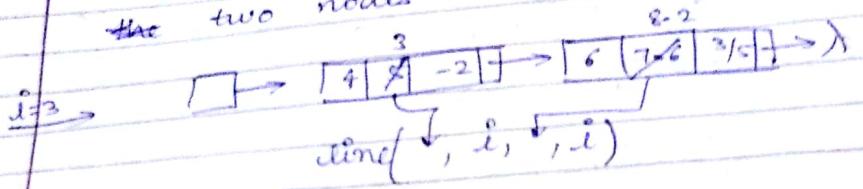
$i=2$

Here no merge req as no edge

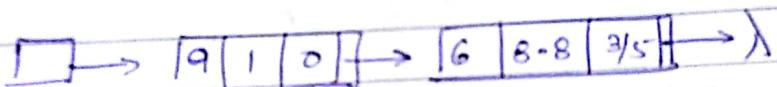


draw line

we delete ~~the~~ any node when $i = y_{\max}$ of any of
the two nodes taken



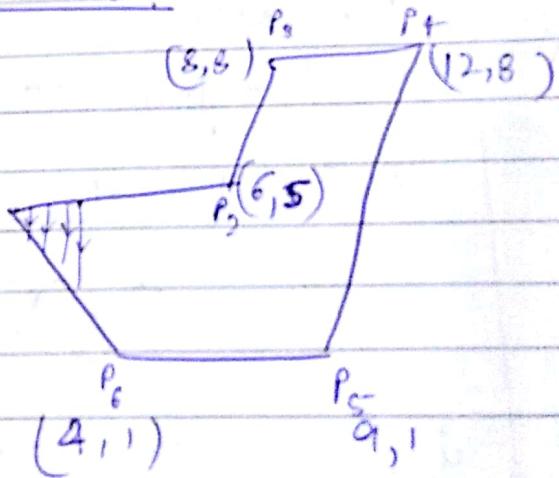
when we add a node
we maintain the ascending order
of x_{\min} only $\boxed{6 \mid 8 \cdot 2 \mid 3 \cdot 5 \mid}$ is old hence, we only
update this node.



Scaline Polygon Filling

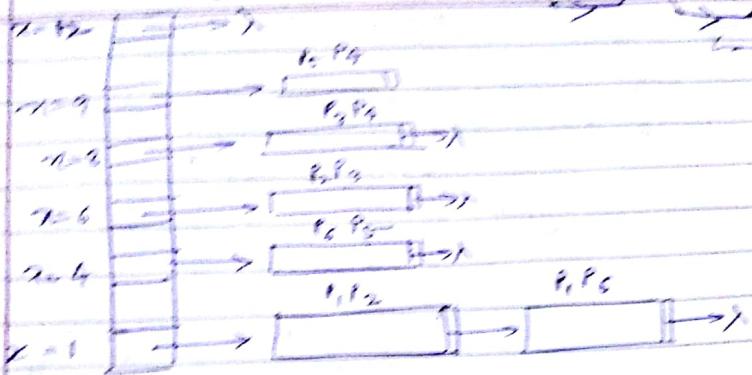
Vertical Scanning

scanning top to bottom $(1, 5) P_1$
vertical scanning
(starting from the
left side to right)
of polygon



Edge \rightarrow $\boxed{x_{\max} \mid 4x_{\min} \mid m \mid} \rightarrow$

Line on edges



Intersection

Active Edge Table rules/steps

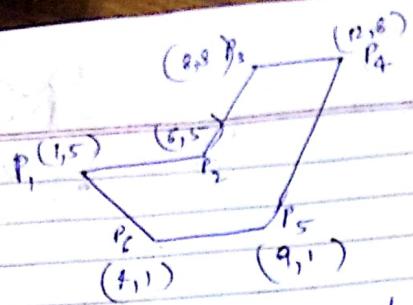
1. merge
2. update (if nodes are the previous one)
3. delete (if node (first number element)=i)
4. Draw line

we start with an empty node and must end with an empty node.

start with empty node bcs if no polygon then edge.

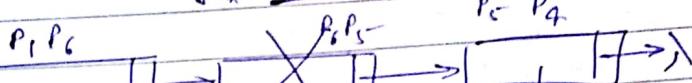
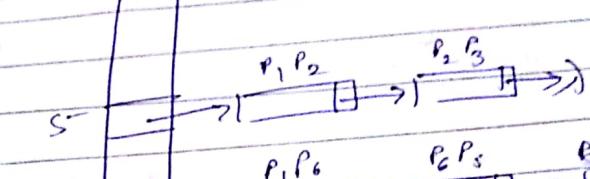
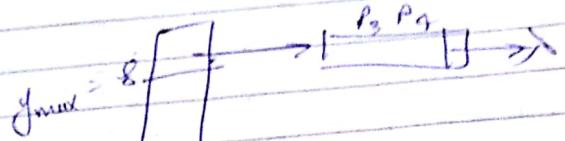
In the end, we have traversed all edges & hence no edge left to traverse

Page No. 15
Date 01/01/2023



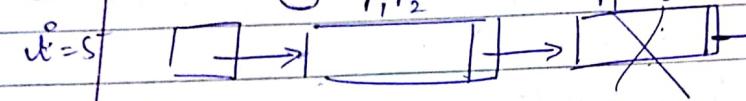
horizontal scanning

Global Edge Table



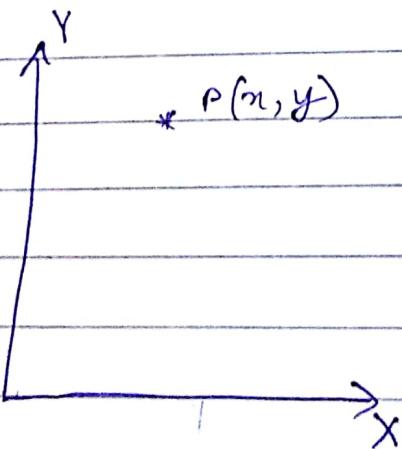
Quick horizontal edge list
its position in list
is even, delete the edge

$i = 5$



Since position
is odd
for horizontal
edge

2 D Transformation



- Translation
 - Scaling
 - Rotation
- Animation
- wait a point

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

Translation

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

$$x' = x + \Delta x$$

$$y' = y + \Delta y$$

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

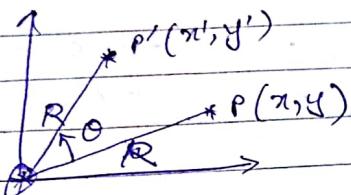
Scaling

$$y' = s_y \cdot y$$

multiplication operator

Rotation

anticlockwise rotation as +ve



$$R \sin \alpha = y$$

$$R \cos \alpha = x$$

$$R \sin(\theta + \alpha) = y'$$

$$R (\sin \theta \cos \alpha + \sin \alpha \cos \theta) = y'$$

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

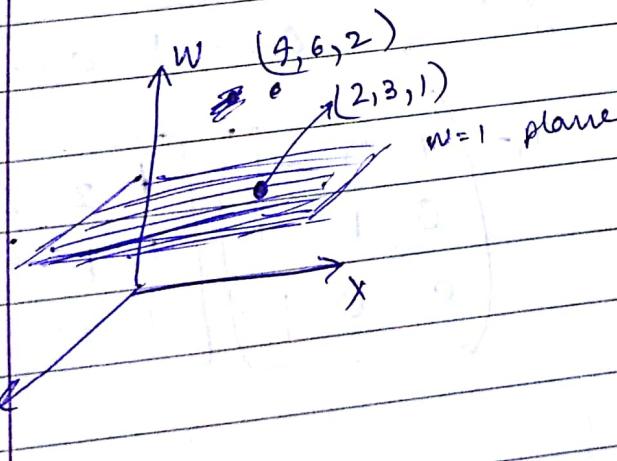
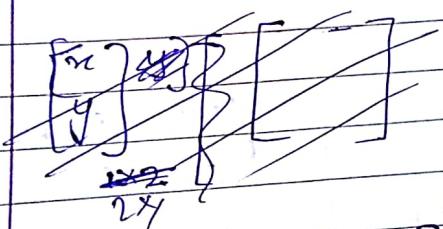
$$R \cos(\theta + \alpha) = x'$$

$$R (\cos \theta \cos \alpha - \sin \theta \sin \alpha) = x'$$

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

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

$$\{P' = R_\theta \cdot P\}$$



2D plane \rightarrow 3 vector
to homogenise

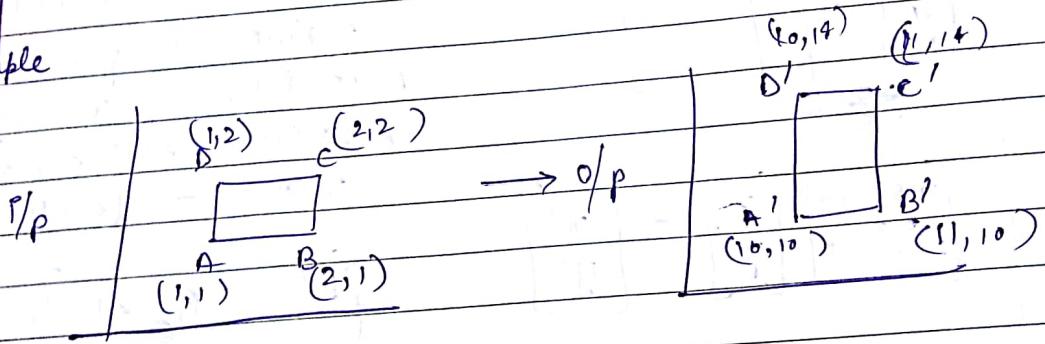
Homogeneous
Coordinate System

Translation $\rightarrow \begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & \Delta x \\ 0 & 1 & \Delta y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$

Scaling $\rightarrow \begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$

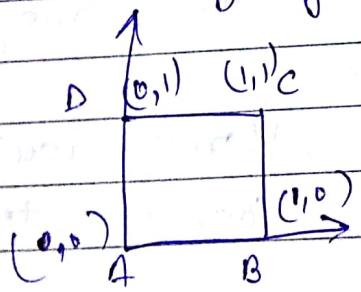
Rotation $\rightarrow \begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$

Example



Everything is done w.r.t origin

Step 1 Bringing to origin $\Delta x = -1 \quad \Delta y = -1$



$$T_1 = \begin{bmatrix} 1 & 0 & -1 \\ 0 & 1 & -1 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 2 & 2 & 1 \\ 1 & 1 & 2 & 2 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

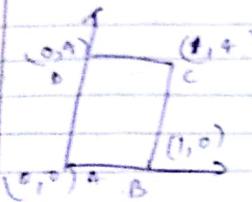
{ Check result should give }

$x + by + c = 0$

Page No.	
Date	

check ② scale

$$sx = 1, sy = 4$$

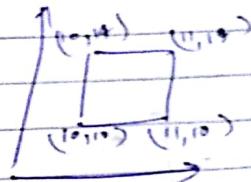


$$S_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

Step ③ Bring A for all vert. $(1, 10)$

$$\Delta x = \Delta y = 10$$

$$T_2 = \begin{bmatrix} 1 & 0 & 10 \\ 0 & 1 & 10 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 1 & 1 & 0 \\ 0 & 0 & 4 & 4 \\ 1 & 1 & 1 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 11 & 11 & 10 \\ 1 & 10 & 14 & 14 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$



$$C = T_2 \cdot S \cdot T_1$$

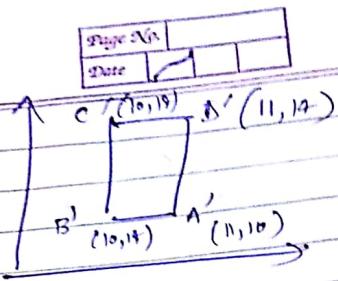
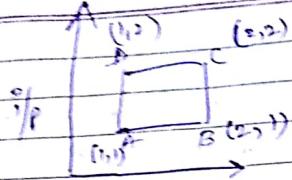
$$\text{as } (T_2(S(T_1, \text{right})))$$

$$C = \begin{bmatrix} 1 & 0 & 10 \\ 0 & 1 & 10 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & -1 \\ 0 & 1 & -1 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 10 \\ 0 & 1 & 10 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & -1 \\ 0 & 4 & -4 \\ 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 0 & 9 \\ 0 & 4 & 6 \\ 0 & 0 & 1 \end{bmatrix}$$

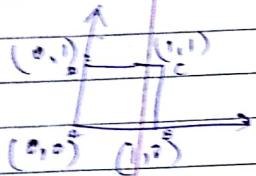
$$\text{Final result} = \begin{bmatrix} 1 & 0 & 9 \\ 0 & 4 & 6 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 2 & 2 & 1 \\ 1 & 1 & 2 & 2 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

Example ②



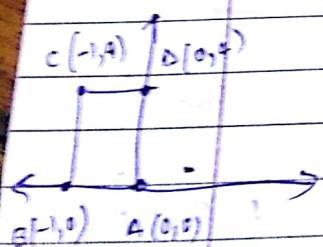
Step ① Translate to origin $\Delta x = -1 \Delta y = -1$

$$T_1 \rightarrow \begin{bmatrix} 1 & 0 & -1 \\ 0 & 1 & -1 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 2 & 2 & 1 \\ 1 & 1 & 2 & 2 \\ 1 & 1 & 1 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$



② scale by $\delta_x = 1 \Delta y = +4$

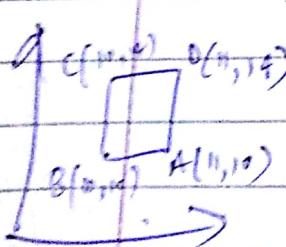
$$S = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix} = \begin{bmatrix} 0 & -1 & -1 & 0 \\ 0 & 0 & 4 & 4 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$



③ Translate B & all to (10,10)

$$\Delta x = 11 \quad \Delta y = 10$$

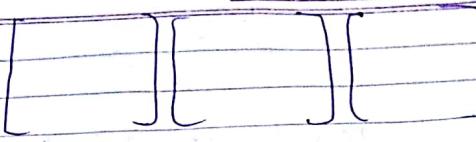
$$T_2 \rightarrow \begin{bmatrix} 1 & 0 & 11 \\ 0 & 1 & 10 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & -1 & -1 & 0 \\ 0 & 0 & 4 & 4 \\ 1 & 1 & 1 & 1 \end{bmatrix} = \begin{bmatrix} 11 & 10 & 10 & 11 \\ 10 & 10 & 14 & 14 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$



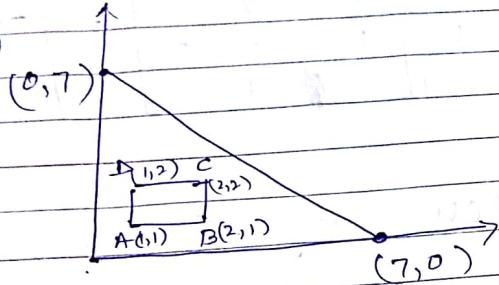
can go from center
or from your
giving 1 side
inwards

Page No.	
Date	

$$C = T_2 S T_1 =$$

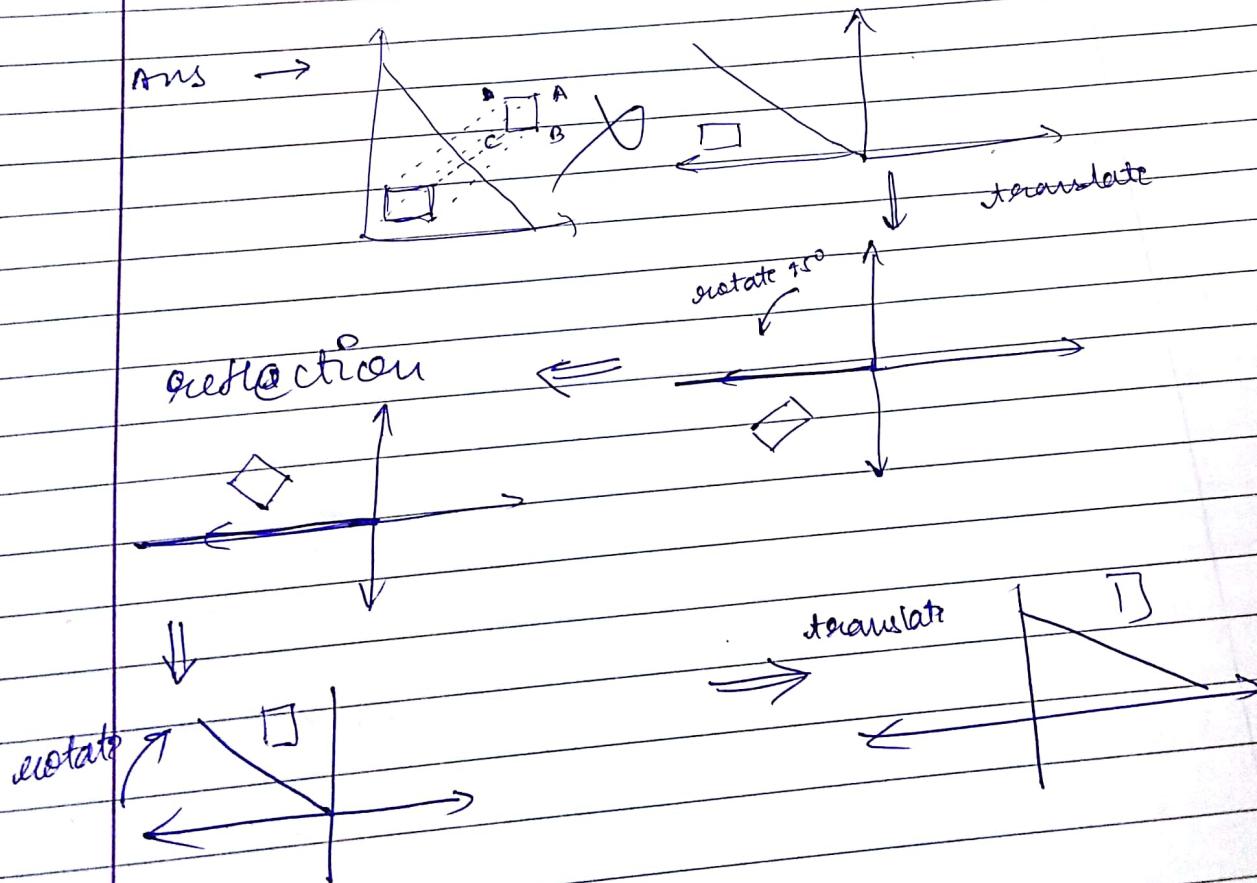


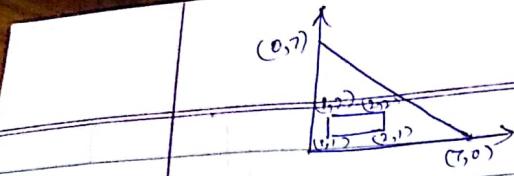
Example ③



Find mirror image
of ABCD about
the given line

Ans →





$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} \Delta x \\ \Delta y \end{pmatrix}$$

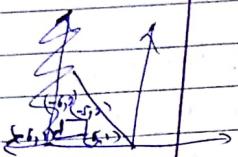
Page No.		
Date		

$$\begin{aligned} Ax + By &= C \\ A(x_0 \cos \theta - y_0 \sin \theta) + B(x_0 \sin \theta + y_0 \cos \theta) &= C \\ Ax_0 \cos \theta - Ay_0 \sin \theta + Bx_0 \sin \theta + By_0 \cos \theta &= C \\ Ax_0 \cos \theta + By_0 \cos \theta - Ay_0 \sin \theta + Bx_0 \sin \theta &= C \\ A(x_0 \cos \theta + Bx_0 \sin \theta) + B(y_0 \cos \theta - A y_0 \sin \theta) &= C \end{aligned}$$

step (1)

translate all by $\Delta x = -7$ $\Delta y = 0$

$$T_1 = \left[\begin{array}{ccc|ccccc} 1 & 0 & -7 & 1 & 2 & 2 & 1 \\ 0 & 1 & 0 & 1 & 1 & 2 & 2 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 \end{array} \right] \rightarrow \left[\begin{array}{ccc|ccccc} -6 & -5 & -5 & -6 & 1 & 1 & 2 & 2 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{array} \right]$$



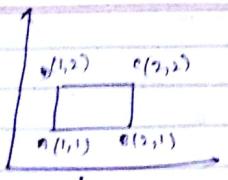
step (2)

rotate ACW by $+5^\circ$

R_1

Example T

Example 74

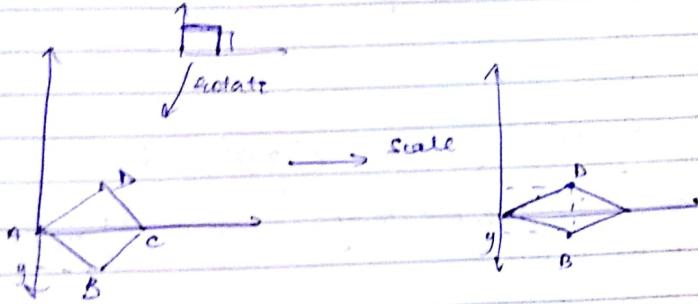
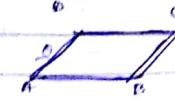


Given object

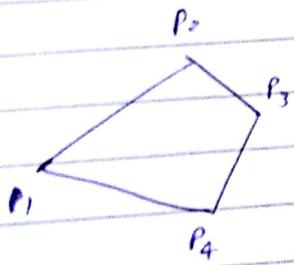
Scanned

Page No. _____
Date _____

00



SEED FILL (POLYGON FILLING)

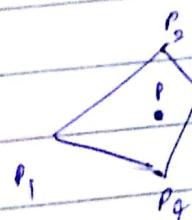


If concave polygon,
break it into convex
polygons.

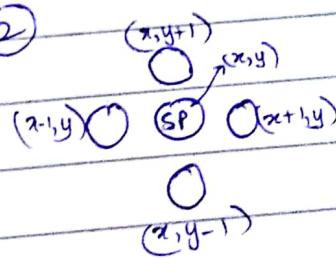
seed point has to be inside

point inside-poly-point { }

Step ① Finding a seed point



Step ②



0 0 0
0 1 0
0 0 0

Type No.	
Date	

void seedfill (color, background color, boundary color, ^{SP}
 boundary color, bg color, boundary color, ⁽ⁿ⁼⁴⁾ SP)

we can remove bg color as it is constant

seedfill (

void seedfill (color, bkg color, boundary color, x, y)
 condition } is { seed(x,y) == color || seed(x+1,y) == boundary
 existing } color)

exit

recursive function } seedfill (c, bkg, boundary color, x+1, y)
 seedfill ()
 ,
 ,
 ,
 ,
 }

loop