Shubhangi Bhalla
A15?0050

Computer Graphics
Assignment 1

Q1. Discuss the working of impact and non impact printers with example.

A1. Impact printers press formed character faces against an inked ribbon onto the paper. A line printer is an example, with the type faces mounted on bands, chains, drums etc. Characters and graphics are obtained by retracting certain ~~points~~ pins so that the remaining pins form the pattern to be printed.
Non impact printers and plotters, on the other hand, use laser technique, ink jet spray, xerographic processes, electrostatic methods and electrothermal methods to get images onto paper.

1) Laser: Laser beams create a charge distribution on a rotating to the drum coated with a photoelectric material eg: Selenium, toner is applied to the drum and then transformed to paper.

2) Ink jet: These methods produce output by squirting ink in horizontal rows across a roll of paper wrapped on a drum.

3) Electrostatic device places a negative charge on the paper, one complete row at a time. Then, paper is exposed to positively charged to -ve which is attracted to negatively charged areas.

Q2. How long would it take to load a 640 X 480 frame buffer 12 bits per pixel if 10^5 bits can be transferred per second?

Paper 1957 Q6(a)

Resolution = 640 X 480

= 307200 pixels

No. of bits transferred per second = 10^5

Given bits per pixel = 12

Total bits required = 307200 X 12

= 3686400 bits

Time required = $\dfrac{\text{Total no. of bits}}{\text{No. of bits transferred per second}}$

$= \dfrac{3686400}{10^5}$

= 35108.6 sec

Q3. If a screen has 513 scan lines and an aspect ratio of 3:4 and if each pixel contains 8 bits worth of intensity information, how many bits per second are displayed if 30 frames are displayed per second?

No. of scan lines = 513

Aspect ratio = 3:4.

No. of bits in each pixel = 8

No. of frames displayed per second = 30

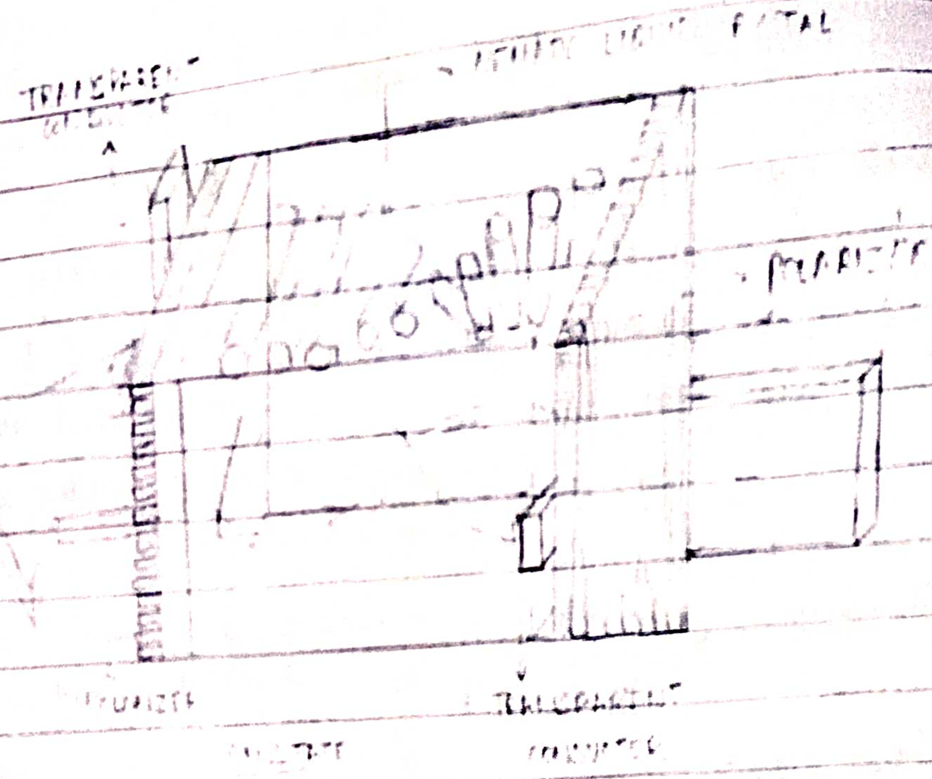
Asp

Q5. Differentiate between raster scan and random scan systems.

	Raster scan	Random scan
1)	Electron beam is swept from top to bottom, across the screen	Electron beam is directed only to the part of screen where a picture is to be drawn
2)	It results in poor resolution	It results in good resolution
3)	Picture definition is stored as a set of intensity values in a refresh buffer area.	Picture definition is stored as a set of line drawing instruction in a display file.
4)	Refresh rate ranges between 60 to 80 frames per second	Refresh rate ranges between 30 to 60 frames per second.

Q6. Briefly explain the working of a Liquid Crystal Display.
Paper 1957 Q7 (a)

Liquid crystal display have a crystalline arrangement of molecules yet they flow like liquid. Two glass plates, each containing a light polarizer at right angles to the other plate, sandwiching the liquid crystal material. Rows of horizontal conductor ~~are~~ on one plate & columns of vertical conductor on the other intersection is a pixel's position. Polarized light passes passing through the material is twisted so that it passes through the other opposite polarizer and is then reflected back to the user.



07: Describe the functioning of a plasma display panel which does not require any refreshing, gives very stable image at the output panel and allows erasing and writing selectively at some finite speed about 21.5 microsecond per unit cell.

Plasma panels are constructed by filling region between two glass plate with a mixture ~~two glass plates with~~ of gases that usually include 'Neon'. A series of vertical conducting ribbons are placed on the glass panel and a set of horizontal ribbons are built into the other glass panel. Firing voltage applied to a pair of horizontal and vertical conductors cause the gas at the intersection of the two conductors to break down into a glowing plasma of electrons and ions. Picture definition is stored in a refresh buffer and the firing voltages applied to refresh pixel position.

Q8. What are the disadvantages of DDA line drawing algorithm?

The accumulation of roundoff errors in recessive addition of the floating point increment can cause a pixel position to drift away from true line path for long times. Additionally, rounding operation and floating point arithmetic are time consuming

Q9. Indicate which raster locations would be chosen by Bresenham's algorithm when scan converting a line from:
a.) screen coordinate (1,1) to screen coordinate (8,5)
b) screen coordinate (0,0) to screen coordinate (4,-8)
Paper 1957 Q6 (b)

For $|m| < 1$,

plot (x_0, y_0)
calculate $p_0 = 2\Delta y - \Delta x$
if $p_k < 0$, plot (x_k+1, y_k)
and $p_{k+1} = p_k + 2\Delta y$
else plot (x_{k+1}, y_k+1)
and $p_{k+1} = p_k + 2\Delta y - 2\Delta x$

Given:
a) $\Delta y = 4$ $\qquad \Delta x = 7$
$\therefore 2\Delta y = 8$ $\qquad \therefore 2\Delta x = 14$

Calculating:
1) $m = \frac{\Delta y}{\Delta x} = \frac{4}{7} < 1$

2) $2\Delta y - 2\Delta x$
$= 8 - 14 = -6$

Sampling along x :

x	y	P_k
1	1	$1 > 0$
2	2	$1-6 = -5 < 0$
3	2	$-5+8 = 3 > 0$
4	3	$3+(-6) = -3 < 0$
5	3	$-3+8 = 5 > 0$
6	4	$5-6 = -1 < 0$
7	4	$-1+8 = 7 > 0$
8	5	

b) Calculating :

Given :

$(x_1, y_1) = (0,0)$

$(x_2, y_2) = (4,-8)$

1) $m = \frac{\Delta y}{\Delta x} = \frac{y_2-y_1}{x_2-x_1}$

$= \frac{-8-0}{4-0}$

$\boxed{m = -2}$

Here, $m < 1$ but $|m| > 1$

Now, since the slope of the line is negative, we draw a line on the ~~given~~ coordinates (0,0) and (4,8) ~~and~~ after ~~then~~ changing the value of the y coordinate

Decision parameter :

$P = 2\Delta x - \Delta y$

$= 2(4) - 8$

$= 8-8 = 0$

$\therefore \boxed{P_{initial} = 0}$

For :

(0,0) : $m > 1$, $P \geq 0$: $x_{i+1} = x_i + 1$

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

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

$\therefore x_{i+1} = 0+1 = 1$

$y_{i+1} = 1$

$P_{k+1} = 0 + 2(4) - 2(8)$

$= -8$

Similarly for :

		x_{i+1}	y_{i+1}	P_{k+1}
(1,1)	:	x_i	$y_i + 1$	$P_k + 2\Delta x$
(0,2)	:	1	2	0
(1,2)	:	2	3	-8
(2,3)	:	2	4	0
(2,4)	:	3	5	-8
(3,5)	:	3	6	0
(3,6)	:	4	7	-8
(4,7)	:	4	8	0

Required table :

P	x_i	y_i	x_{i+1}	y_{i+1}
0	0	0	1	1
-8	1	1	1	2
0	1	2	2	3
-8	2	3	2	4
0	2	4	3	5
-8	3	5	3	6
0	3	6	4	7
-8	4	7	4	8

Final (x,y) coordinates:

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

Q10. A dashed line joining two points $P(x_1, y_1)$ and $Q(x_2, y_2)$ has to be plotted. Describe an algorithm to draw a dashed line for joining the points $P(x_1, y_1)$ and $Q(x_2, y_2)$. The length of dash is p pixels and length of gap b/w two dash is q pixels.

A10. DDA algorithm can be implemented here with the following modifications:

```
void lineDDA ( int x, int y, int x2, int y2, int p, int q)
{ int dx = x2-x1;
  int dy= y2-y1;
  float xinc, yinc;
  if (abs (dx) > abs(dy)) ~~steps= abs(dx)~~)
        steps = abs(dx);
  else
        steps = abs (dy) ;
  xinc = dx/(float) steps;
  yinc = dy/ (float) steps;
```

```
setPixel ( Round (x), Round (y));
for (int k=0; k < steps; k++) {
    for (int i=0 ; i<p ; i++) {
        x+ = xinc;
        y+= yinc;
    setPixel (Round (x), Round (y)) ;
    }
    for (int j=0; j<q ; j++) {
        x+ = xinc;
        y+ = yinc;
    }
}
```

Here, we have increased the value of x and y for p pixels and set the pixel position after q distance. After putting p pixels for next q pixels, we have first increment the value of x and y and left those pixel position blank. This process is continued till we reach the end of this line PQ.

~~Paper 1957 Q1. (b)~~

Q11. What is the condition that ellipse scan conversion algo. uses to divide the first quadrant of the ellipse in two regions?

A11. At the boundary between region 1 and 2,

$$\frac{dy}{dx} = -1 \quad \text{and} \quad 2 r_y^2 x = 2 r_x^2 y$$

whenever $2 r_y^2 x \geq 2 r_x^2 y$, we move out of region 1.

Q12. Enumerate the steps for scan line polygon fill algorithm. Also explain the data structures used.
Paper 1957 Q8 (b)

A12. Step 1: Find the intersection of the scan line with all edges of the polygon
Step 2: Sort the intersections by increasing x coordinate
Step 3: Fill in all pixels b/w pairs of intersection that lie interior to the polygon using odd-parity rule to determine that a point is inside a region
Parity is initially even and each encountered intersection inverts the parity bit
Draw when parity is odd, do not draw when it is even.

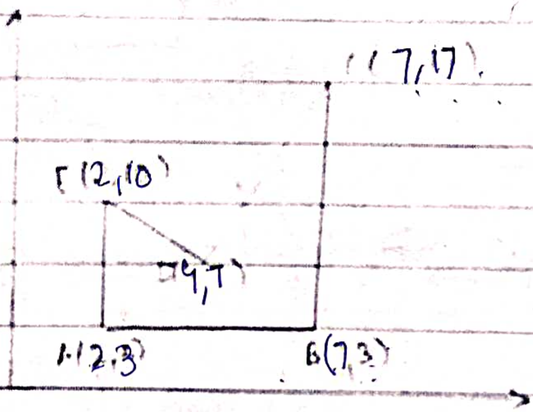
Now, once the edge table has been formed,
Step 4: Set y to the smallest y coordinate that has an entry in the ET i.e. y for the first nonempty bucket
Step 5: Initialise active ET (AET) to be empty.
Step 6: Repeat this (i) to (v) till AET and ET are empty.
(i) Move from ET bucket y to the AET those edges whose $y_{min} = y$.
(ii) Remove from AET those entries, for which $y = y_{max}$, then sort the AET on x
(iii) Fill in desired pixel values on scan line y by using pairs of x coordinates from AET.
(iv) Increment y by 1
(v) For each non vertical edge remaining in AET, update x for new y.

Data structures used:

1) ET : Edge table contains the list of all the edges by storing their endpoint coordinates

2) AET: Active Edge list contains all those edges of the polygon that are intersected by the current scan line. The edges are dropped into the table in a sorted manner i.e. Inc value of x.

Q13. Consider a polygon ABCDE : A (2,3), B(7,3), C(7,17), D(4,7), E (2,10). Use scan line fill algo to fill it scan line 8. Show the global and active edge table at each step.



increasing order
A < B < D < E < C

Global edge table:

C (7,17) → | 17 | 4 | 3 | 10 | (DC)

E (2,10) → | 10 | 2 | 0 | (AE) → | 10 | 4 | -2/3 | (DE)

D (4,7) → | 10 | 4 | -2/3 | (DE) → | 17 | 4 | 3/10 | (DC)

B (7,3) → | 17 | 7 | 10 | (BC)] vertical

A (2,3) → | 10 | 2 | 0 | (AE)

Active Edge Table :

(scan line with k)

i) k = 3

[ ] → [10 | 2 | 0] → [17 | 7 | 0]

∴ (2, 3) and (7, 3) are to be drawn

ii) k = 4

[ ] → [10 | 2 | 0] → [17 | 7 | 0]

∴ (2, 4) and (7, 4) are to be drawn

iii) k = 5

[ ] → [10 | 2 | 0] → [17 | 7 | 0]

∴ (2, 5) and (7, 5) to draw

iv) k = 6

[ ] → [10 | 2 | 0] → [17 | 7 | 0]

∴ (2, 6) and (~~B~~ 7, 6) to draw

v) k = 7

[ ] → [10 | 2 | 0] → [10 | 4 | -2/3] → [17 | 4 | 3/10] → [17 | 7 | 0]

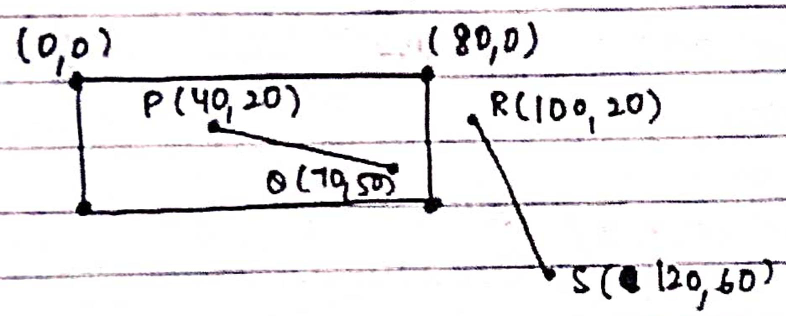
∴ (2, 7) to (4, 7) and (4, 7) to (7, 7) to draw

vi) k = 8

[ ] → [10 | 2 | 0] → [10 | 5 | −2/3] → [17 | 5 | 3/10] → [17 | 1 | 0]

∴ (2,8) to (3,8) and (5,8) to (7,8) to draw

Q14. A clipping window ~~is~~ has two vertices lying at (0,0) and (80,40). Use the line end point codes to determine whether the lines P(40,20), Q(70,50) and R(100,20), S(120,60) would be visible, partially or ~~totta~~ invisible.



I Region codes :

P → 0000 R → 0010

Q → 0000 S → 0110

AND operation on points :

P:	0000	R:	0010
Q:	0000	S:	0110
AND:	0000	AND	0010

∴ Line PQ will be ~~tollo~~ totally visible
and line RS will be totally invisible

Q15 Let R be a rectangular window whose lower left corner is at L(-3,1) and upper righthand corner is at R(2,6). If the line segment is defined with two end points with A(-4,2) and B(-1,7)

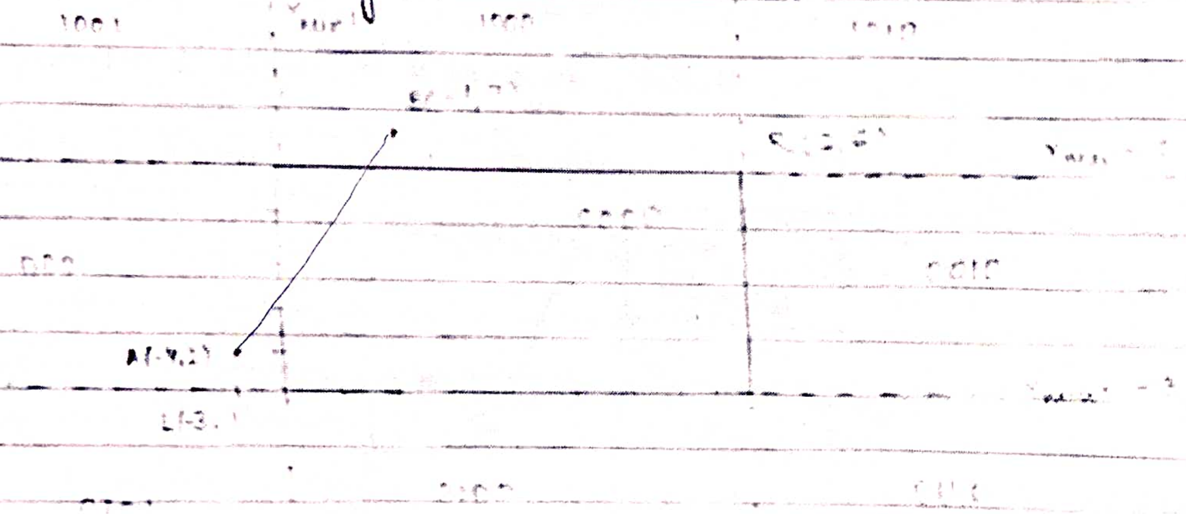
Paper 1957 Q8 (a)

i) The region codes of two end points

ii) The clipping category

iii) Stages in the clipping operation using Cohen-Sutherland Algorithm

A15. Window and line segment :



Given :

$W_{x_{min}} = -3$ $\quad$ $W_{y_{min}} = 1$

$W_{x_{max}} = 2$ $\quad$ $W_{y_{max}} = 6$

I Region code for (x,y) can be calculated by :

		A(-4,2)	B(-1,7)
Bit 4 =	$y - W_{y_{max}}$	$-6+2=-4$	$-6+7 = +1$
Bit 3 =	$W_{y_{min}} - y$	$-2+1=-1$	$-7+1=-6$
Bit 2 =	$x - W_{x_{max}}$	$-4+3=-1$ ~~2+4=6~~	~~2+1=3~~ $-1-2=-3$
Bit 1 =	$W_{x_{min}} - x$	~~-4-1=-5~~ $-2+4=6$	~~1+3=2~~ $-3+1=-2$

now, if sign (a) = +ve , val = 1

= -ve , val = 0

Now calculating region codes :

A : $\underset{(-4)}{0}$ $\underset{(-1)}{0}$ $\underset{(-1)}{0}$ $\underset{(5)}{1}$

(-4, 2)

B : $\underset{(+1)}{1}$ $\underset{(-6)}{0}$ $\underset{(-3)}{0}$ $\underset{(-2)}{0}$

(-1, 7)

II Clipping category :

Performing AND operation on both region codes for end points A and B.

$$
\begin{array}{lll}
A & : & 0001 \\
B & : & 1000 \\
\hline
 & & 0000
\end{array}
$$

Now, as the output of AND operation is zero (0000), none of the end points lie within the window
∴ It belongs to category 3 i.e. it is a candidate for clipping

III Stages in clipping operation :

1) Clipping ~~point~~ point A against boundary line $x_{min} = -3$ (to push 1 to 0) :

$(x_1, y_1) = (-4, 2)$, $(x_2, y_2) = (-1, 7)$

$$m = \frac{y_2 - y_1}{x_2 - x_1} = \frac{7-2}{-1+4} = \frac{5}{3}$$

Intersection point $I_1(x_1', y')$:

$$y' = y_1 + m(Wx_{min} - x_1)$$

$$= 2 + \frac{5}{3}(-3+4)$$

$$= \frac{11}{3}$$

$$\therefore \boxed{I_1(x', y') = (-3, 11/3)}$$

2) Clipping point B against boundary line $W_{ymax} = 6$

New point $I_1(x', y')$ and $B(x_2, y_2)$ form a line with point B lying outside the window.

Region code for B : 1000

Pushing 1 to 0, resulting in $I_2(x'', y'')$:

$m = \frac{5}{3}$ (from (1))

$$x'' = x_2 + \left(\frac{W_{ymax} - y_2}{m} \right)$$

$$= -1 + \frac{(6-7)}{5/3}$$

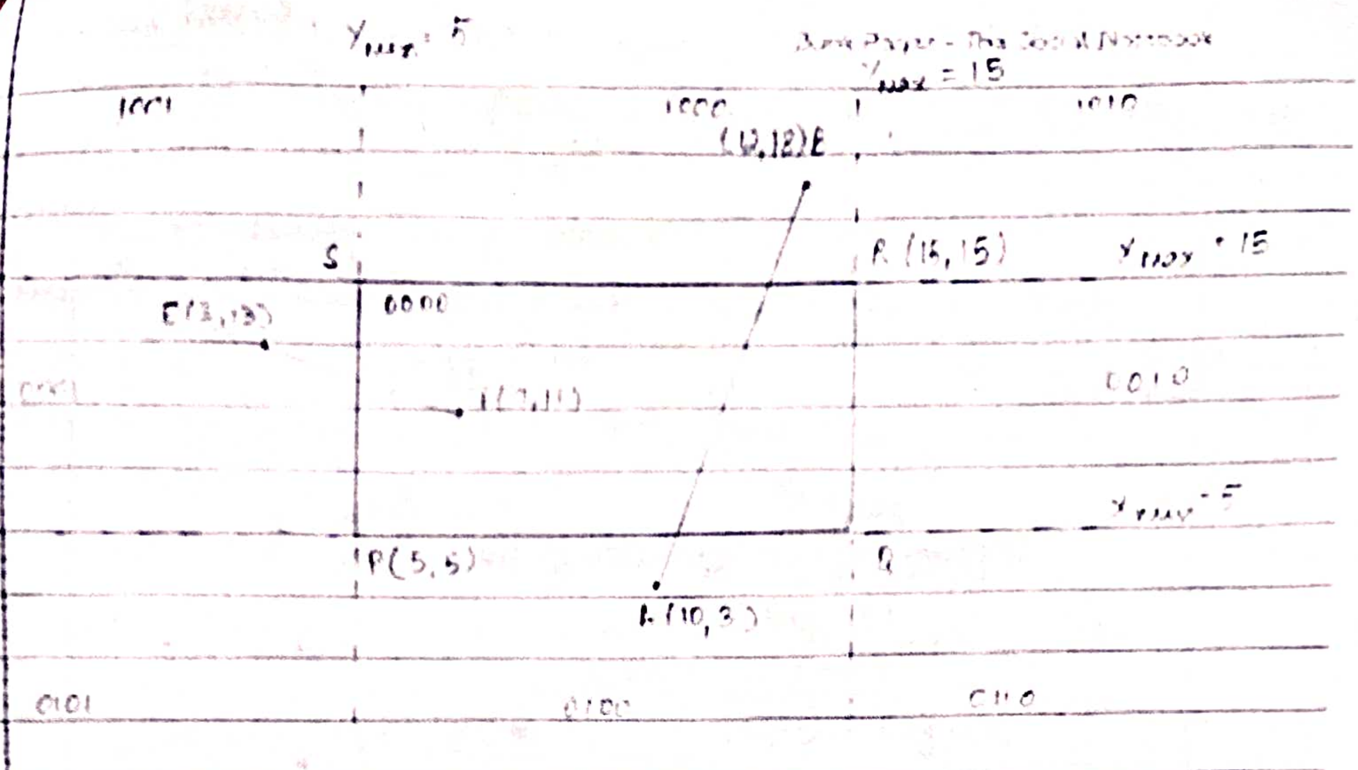
$$= \frac{-8}{5}$$

$$\therefore \boxed{I_2(x'', y'') = (-8/5, 6)}$$

3) Finally, clipped line $I_1(-3, 11/3)$ to $I_2(-8/5, 6)$ has region code 0000 and lies within the window.

Q16. Consider the following clip Rectangle PQRS and lines AB and EF. Assume P(5,5) and R(15,15), E(3,13), F(7,11), A(10,3) and B(12,18).

Using Cohen Sutherland clipping Algorithm, compute the region codes for line AB and EF and show the stages in the clipping operations.



Given :

$X_{min} = 5$ $\quad$ $Y_{min} = 5$

$X_{max} = 15$ $\quad$ $Y_{max} = 15$

I Region codes for a point can be calculated by

			A (10,3)	B (12,18)	E (3,13)	F (7,11)
Bit 4	=	$Y - Y_{max}$	$3-15=-12$	$18-15=3$	$13-15=-2$	$11-15=-4$
Bit 3	=	$Y_{min} - Y$	$5-3=2$	$5-18=-13$	$5-13=-8$	$5-11=-6$
Bit 2	=	$X - X_{max}$	$10-15=-5$	$12-15=-3$	$3-15=-12$	$7-15=-8$
Bit 1	=	$X_{min} - X$	$5-10=-5$	$5-12=-7$	$5-3=2$	$5-7=-2$
∴			-ve +ve -ve -ve	+ve -ve -ve -ve	-ve -ve -ve +ve	-ve -ve -ve -ve

Now, if sign(o) = +ve, val = 1

= -ve, val = 0

Calculating region codes :

A : 0100 $\quad$ E : 0001

B : 1000 $\quad$ F : 0000

Case 1 :

line : AB

A: 0100

B: 1000

(AND operation) 0000

$\therefore$ None of the points lie within the window and it is a candidate for clipping

Clipping point A against boundary line $y_{min} = 5$ (to push 1 to 0) :

$(x_1, y_1) = (10, 3)$, $B(x_2, y_2) = (12, 18)$

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

$$= \frac{18 - 3}{12 - 10} = \frac{15}{2}$$

Intersection point $I_1 (x', y')$:

$$\cancel{y' = y_1 + m(x_{min} - x_1)}$$

$$\cancel{= 3 + \frac{15}{2}(5 - 10)}$$

$$x' = x_2 + \frac{(y_{min} - y_2)}{m}$$

$$= 12 + \left(\frac{5 - 18}{15/2}\right)$$

$$= 12 + \frac{2}{15}(-13)$$

$$= \frac{180}{15} - 12 = \frac{154}{15} = 10 \cdot 26$$

$\therefore \boxed{I_1 (x', y') = (10 \cdot 26, 5)}$

~~Case 2~~:

~~the~~ Clipping point B against boundary line $Y_{max} = 15$
(to push 1 to 0):

$\boxed{m = \frac{15}{2}}$ & $(x_1, y_1) = 10,3$ $(x_2, y_2) = 12,18$

$$x'' = x_2 + \frac{Y_{max} - Y_2}{m}$$

$$= 12 + \frac{2}{15}(15 - 18)$$

$$= 12 + \frac{2}{15}(-3)$$

$$= \frac{174}{15} = 11.6$$

$$\therefore \boxed{I_2(x'', y'') = (11.6, 15)}$$

Now, clipped line $I_1(10.26, 5)$ to $I_2(11.6, 15)$ lies within the window

Case 2 :

line : EF

	E :	0001
	F :	0000
(AND)		0000

∴ It is a candidate for clipping point E. Point F already lies within the window as it's region code is zero. (0000).

Clipping point E against boundary line $X_{min} = 5$
(to push 1 to 0) :

$(x_1, y_1) = 3,13$ $\qquad (x_2, y_2) = (7, 11)$

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

$$= \frac{11 - 13}{7 - 3} = \frac{-2}{4} = \frac{-1}{2}$$

Intersection point $I_1(x', y')$:

$$y' = y_1 + m(X_{min} - X_1)$$

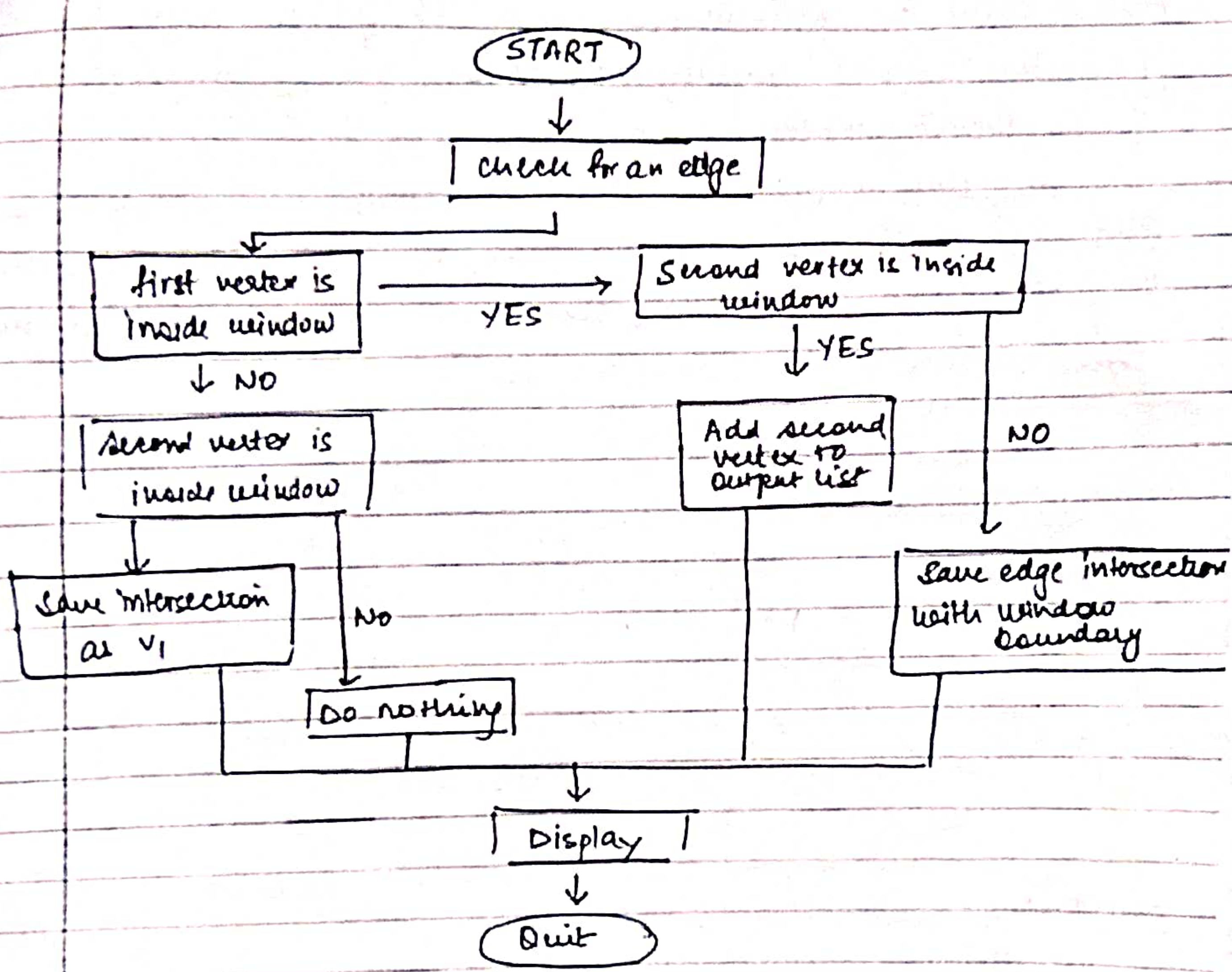
$$= 13 + \left(\frac{-1}{2}\right)(5 - 3)$$

$$= 13 - \frac{2}{2} = 12$$

$$\therefore \boxed{I_1 (x', y') = (5, 12)}$$

Now line $I_1 (5, 12)$ to $F (7, 11)$ lie within the window.

Q17. Draw a flowchart for the logic of Sutherland-Hodgeman polygo clipping algorithm.



Q18. What is aliasing? Discuss any one anti aliasing method. How can the effects of aliasing be minimized?

A18. Primitives drawn so have jaggies or staircasing. This phenomenon is called aliasing. It can be minimized by increasing resolution. Even unweighted Area Sampling can be used to produce noticeably better results.

- Intensity of a pixel intersected by a line edge decreases as the distance b/w pixel centre and edge increases.
- Farther away a primitive is, less the pixel intensity
- Primitive cannot influence the intensity at a pixel at all if it doesn't intersect the pixel.
- When the line covers the pixel completely, the overlap area and ∴ the intensity are at the max. When primitive is just tangent to the boundary, the area and thus intensity is zero.