Date: ______ Page no:

and

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

Otherwise The next point to plot is (x_k+1, y_k+1) and

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

5) Repeat step 4 $(\Delta x - 1)$ times

The Bresenham Line Algorithm has the following advantages.

- An fast Incremental Algorithm
- use only Integer Calculation

Comparison:- DDA has following problem

1) Accumulation of Round off error can make the pixel line drift away from what was intended.

2) The Rounding operation and floating Point Arithmetic are time consuming.

MY ROUGH NOTE BOOK

$2x_{k+1} = 2x_k + 2$

$2y_{k+1} = 2y_k - 2$

$= 1 - r$
$= 1 - 10$
$= -9$

k	P_k	x_{k+1}	y_{k+1}
0	-9	(1, 10)	
1	-6	(2, 10)	
2	-1	(3, 10)	

$P_{k+1} = -9 + 2 + 1$
$= -7 + 1$
$= -6$

$2x+1$

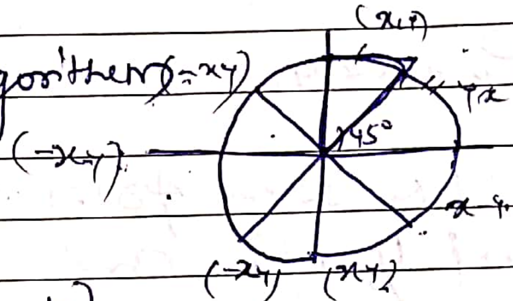
$-6 + 4 + 1$
$-2 + 1$
-1

Date:______ Page no:

Radius (from Bakes)

= x =

mid Point circle Algorithm



(first octent find out)

($x \geq y$) end point

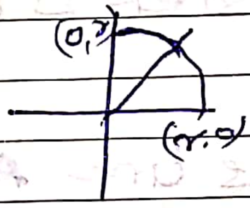
x unit interval

y ?

next Cordinate

(y_k y_{k-1})

(0, r)



Next Coordinate will be (x_k+1, y_k) (x_k+1, y_k-1)

mid point

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

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

= x =

DDA line drawing Algorithm

```
# include< graphics.h>
# include< stdio.h>
# include <math.h>
int Main ()
{
    float x, y, x1, y1, x2, y2, dx, dy, Pixel;
    int i, gd, gm;

Printf ("Enter the value of X1: ");
Scanf ("%f", &X1);
```

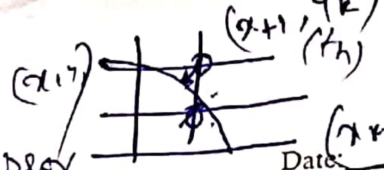
MY ROUGH NOTEBOOK

Scanned by CamScanner

```
Printf ("Enter the value of Y1 : ");
Scanf ("%f", &Y1);
printf ("Enter the value of X2 : ");
scanf ("%f", &X2);
printf ("Enter the value of Y2 : ");
scanf ("%f", &Y2);

detectgraph (&gd, &gm);
initgraph (&gd, &gm, "");

dx = abs (X2-X1);
dy = abs (Y2-Y1);
if (dx>=dy)
pixel = dx;
else
pixel = dy;
dx = dx/pixel;
dy = dy/pixel;
x = X1;
y = Y1;
i = 1;
while (i<=pixel)
{
putpixel (x, y, 1);
x = x+dx;
y = y+dy;
i = i+1;
delay (100);
}
getch ();
closegraph ();
```



$f(n) = \}$ which give less error selected
$f(s) =$

$\boxed{x^2 + y^2 - r^2}$

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

Bresenhems circle drawing Algorithem

$\boxed{d = 3 - 2r}$

$(x+1)(y_k)$ $\quad [d < 0]$ $\quad \boxed{d = d + 4x_{k+1} + 6}$

$(x_{k+1}, y_k - 1)$ $\quad d > 0$ $\quad d = d + 4(x_{k+1} - y_{k+1}) + 10$

If $R = 3$

$d = 3 - 2(3)$

$d = 3 - 6$

$= -3$

R	Dp	x_{k+1} y_{k+1}
0	-3	[1, 3]
1	7	[2, 2]

— x —

$f(N) = (x_i + 1, y_i)$

$f(S) = (x_i + 1, y_i - 1)$

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

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

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

decision parame : $d_i = f(N) + f(S)$

(E) pos SE

1) Inside the circle (-ve) distance
2) outside the circle (+ve).

If $d \le 0 \Rightarrow$ East Point Selected

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

$y_{i+1} = y_i$

$d > 0 \Rightarrow$ SE point selected

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

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

$-3+4\times1+6$

$-3+10$

7

$= 2+g\cdot$

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

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

$= 3-2R$

CDD :-

automobile assembly
circuit design
simulation of

Presantation of graphics

(i) 2 D
(ii) 3 D

Application :-

i) Computer art
ii) Entertainment
iii) Education and training
iv) Visualization
v) image processing
(vi) internet
(vii) medical application
(viii) simulation and virtual reality
(ix) computer aided learning (CAL)
(x) Geographical information system (GIS)

"श्री गणेशाय नमः"

Date:______ Page no:

1) Introduction Slide with Book [9:2 pm]

(3)

श्री गणेशाय नमः

"Computer graphics is a rapidly envolving field. Until the past two decades.

By graphics we will refer to any sketch, drawing special artwork or other material to pictorically depict an object or process or otherwise convey information. as a supplement to or instead of written discriptions the sketch may be cartoon or landscape building electorical network or of the human anatomy.

"It may be just a few regular lines or 2D or 3D drawing.

The graphics may also include text in various sizes and shapes."

* major use of comm is CAD

Pixel :- in computer graphics, pictures or graphics objects are representea as a collection of discrete picture element called pixels (Picture element).

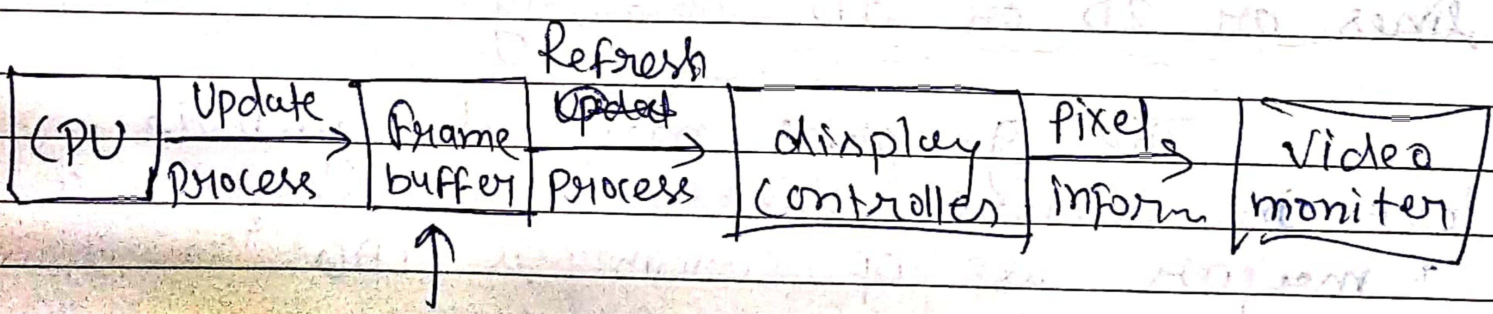
"the pixel is the smallest addressable screen element"

"it is a smallest piece of the display screen which we can control

the control is achived by setting the intensity and color of the pixel which compose the screen.

Components of Computer graphics

(i) Digital memory buffer
(ii) TV moniter
(iii) Display Controller



Picture defination is stored in a memory

(i) digital memory buffer :-

picture stored in array (matrix or 0 and 1)

frame buffer is the video RAM.

amount of memory Required

MB = X-resolution X Y-resolution X Bits Per Pixel

8 x 1024 x 1024

(iii) TV moniter :- helps to view the display and they make use of CRT (cathode Ray tube)

(iiii) Display controller :-

interface betn digital memory buffer and TV moniter.

the main function of this is to pass the content of frame buffer to the moniter. the display Controller reads each successive byte of data from fram buffer memory and converts 0's and 1's into corresponding video signal. produce black and white picture on the screen.

" ~~Now display Controller is recognised as display card~~ 640 X 480

image resolution :-

resolution of an image is the number of pixel per unit lenght in horizontal as well as vertical direction

image aspect ratio :-

can be defined as the ratio of it's width to it's height. unit length / number of Pixel.

Persistence :- is defined as the time it takes the emitted light from the ~~@~~ screen to decay to one-tenth of its orignal Intensity

Date:______ Page no:

~~Display Raster devices~~ –

image resolution – ~~height x width~~ (pixel per inch

image aspect ratio = $\frac{heigh}{width}$.

image size – $\frac{height}{\text{pixel per inch (resolution)}} \times \frac{width}{\text{pixel per inch}}$

Numerical :–

Compute the following

(i) "Size of 800 x 600 image" at 240 pixel per inch.

image size – $\frac{height}{resolution} \times \frac{width}{resolution}$

~~800 x 600~~ =

Size – $\frac{800}{240} \times \frac{600}{240}$

(ii) "Resolution" of 2X2 inch image size that has 512 X 512 pixels.

$$\text{image size} = \frac{\text{hight}}{\text{resolution}} \times \frac{\text{width}}{\text{resolution}}$$

$$2 \times 2 = \frac{512}{\text{resolution}} \times \frac{512}{\text{resoluti}}$$

$$\text{Resolution} = \frac{512 \times 512}{2 \times 2}$$

$$= \frac{512}{2}$$

(iii) image aspect Ratio = $\frac{\text{heigh}}{\text{width}}$

$$1.5 = \frac{5}{\text{wich}}$$

$$\text{widh} = 5 \times 1.5$$

$$= 7.5$$

(iv)

1.5

Display devices :-

The display system where the graphics are rendered in console screen of the computer.

it is responsible for graphic display.

1. Rastar scan display
2. Random scan display
3. Direct view storage tube
4. flat panel display

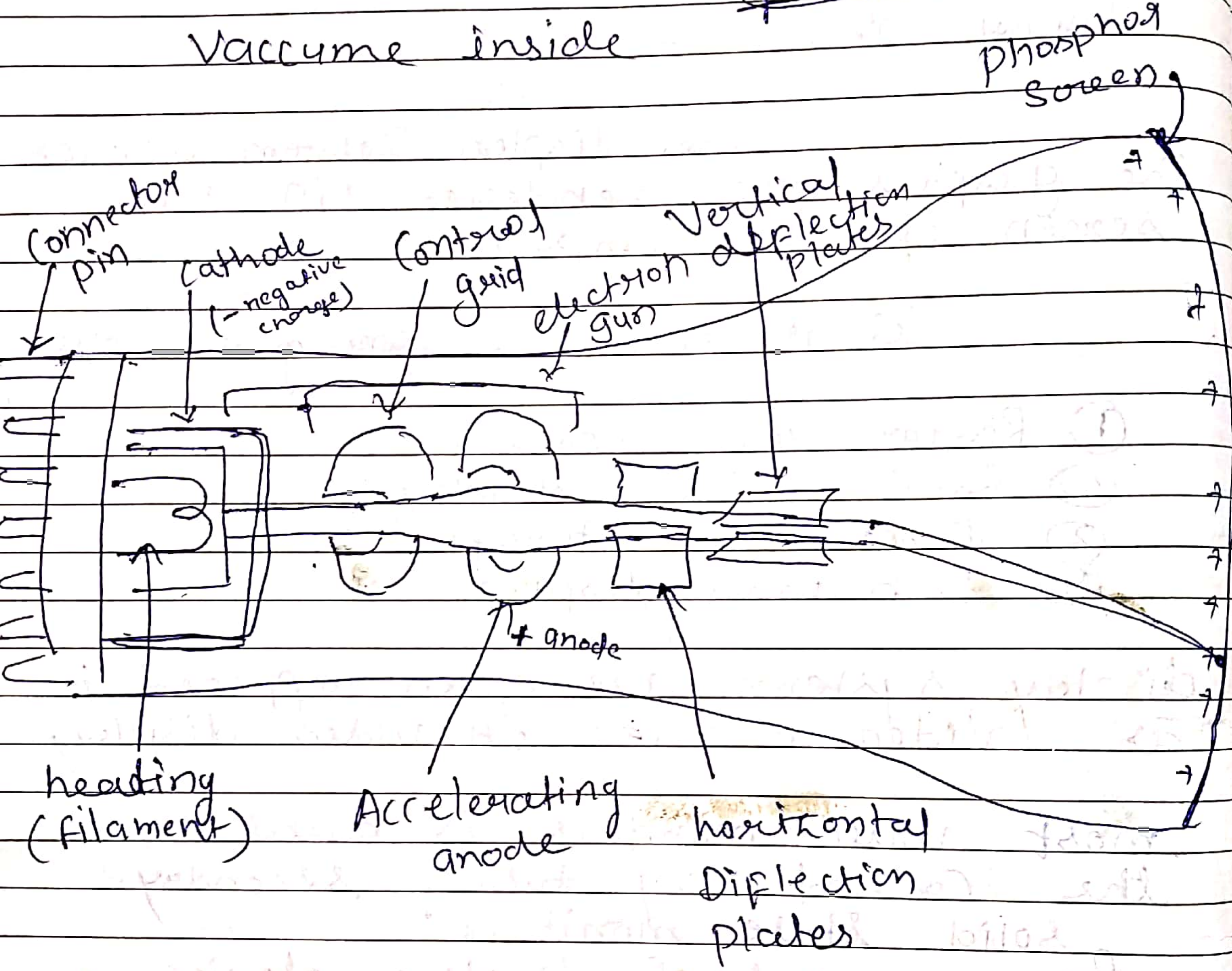
display systems are often reffered to as Video moniter or Video display

most video moniters is based on the Cathod ray tube. secondary solid State moniters

Primary output display device video moniter secondaly print.

Date:______ Page no:

Vaccume inside



heating (filament) :- heat is supplied to the cathod by directing a current through a coil of wire, called filament. magnetic field is generated.
This causes electrons to be "boiled off". fully negative charged elect are then accelerate.

Contro grid :- intensity of electron beam is controlled by setting voltage level. Control grid architecture.

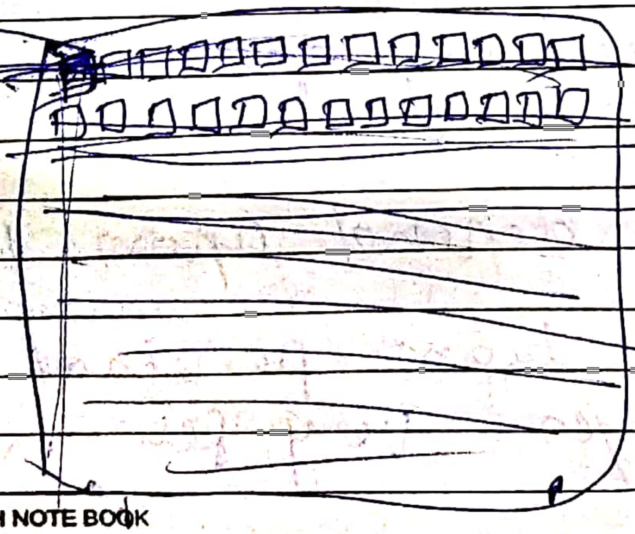
Accelerating anode :- +ve charge gives to focus and ~~real~~ give a way to go on.

Horizontal diflection plate :- to right to left

Vertical :- up to down.

Refresh CRT :- one way to keep the phosphor glowing is to redrawn the picture repeatedly by quickly directing the electron beam back over the same point. this type of display is called a Refresh CRT.

0 □ 60 time



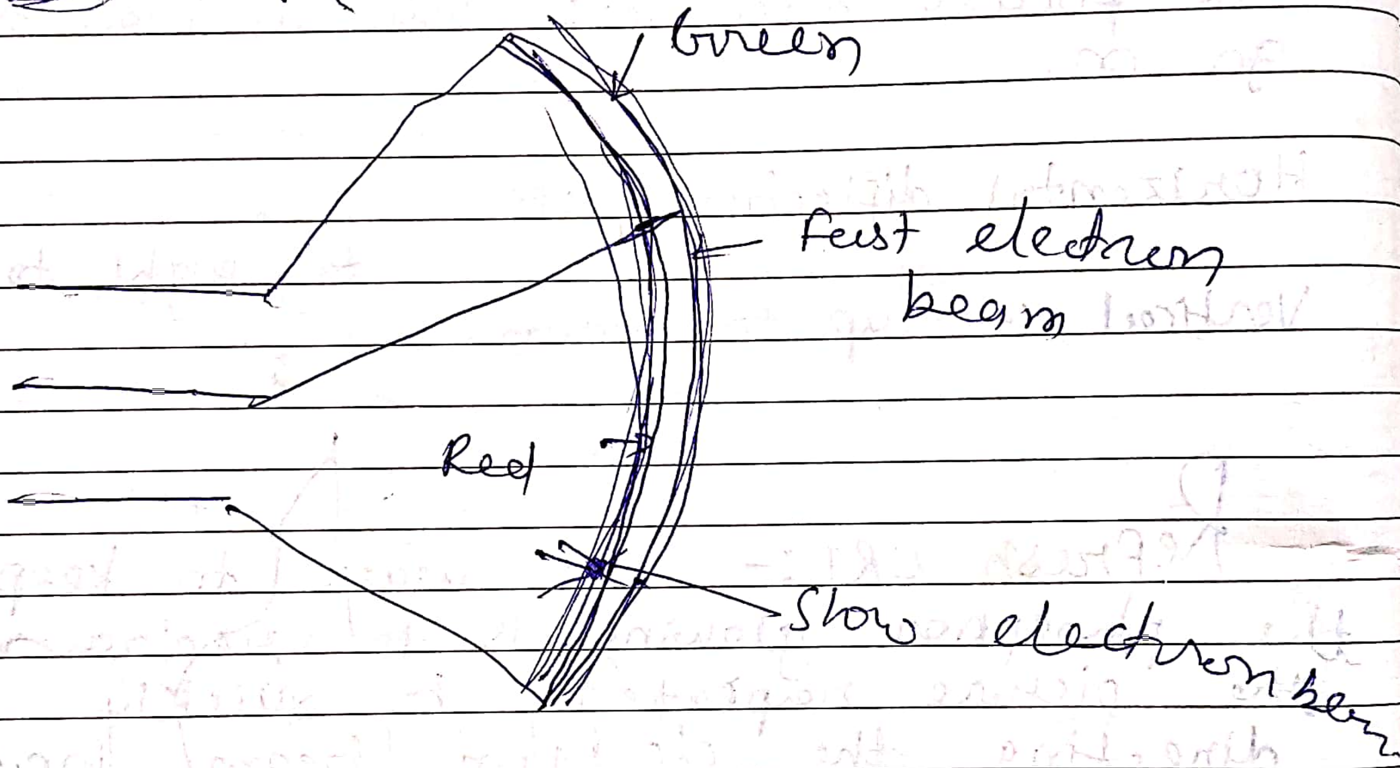
1/60 secon

Vertical retrace

Normal CRT is black n white

(b) Two type of color CRT

(1) Beam penetration



Used in Random Scan moniters

3 dots in a pixel.

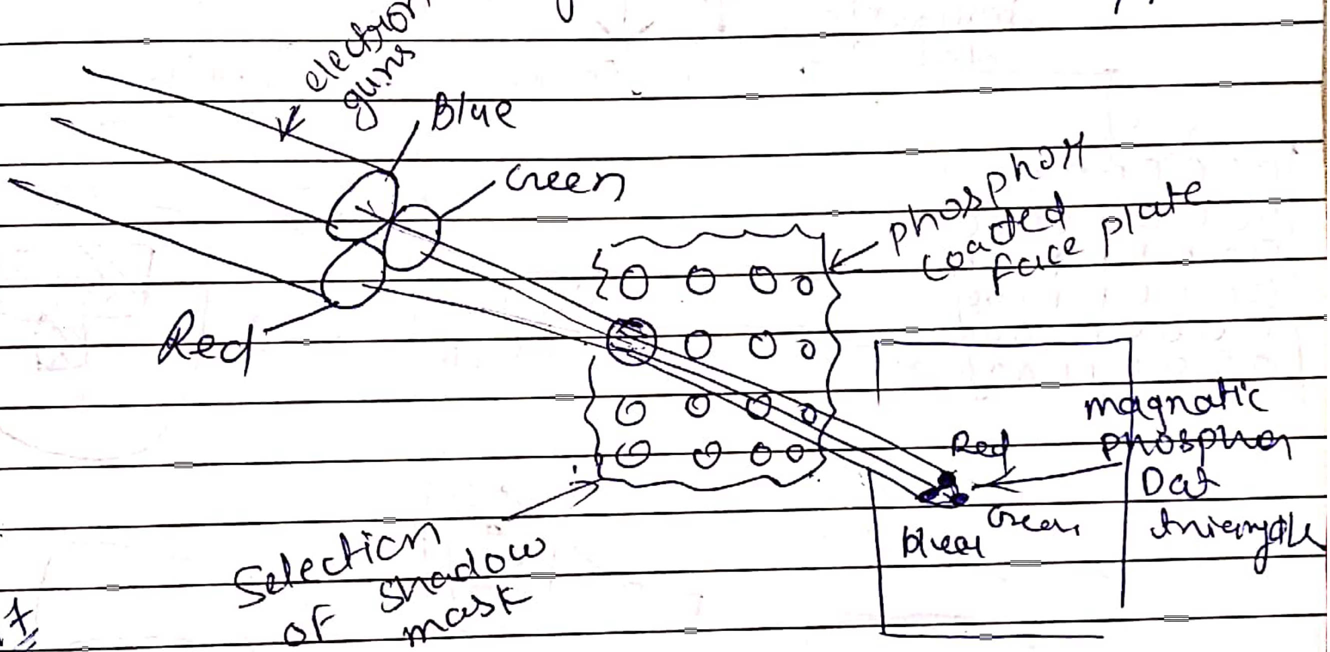
Two color Red and green coated to phosphor screen.

display color depend upon how deep electron beam penetrad to phosphor layer. very few colors advantage : the biggest advantage is that it is at half cost of shadow mask and its resolution is better.

Date: Page no:

② Shadow masking :- ∆ delta shadow masking.

used in Raster-scan model 70
wider range of colour. 77

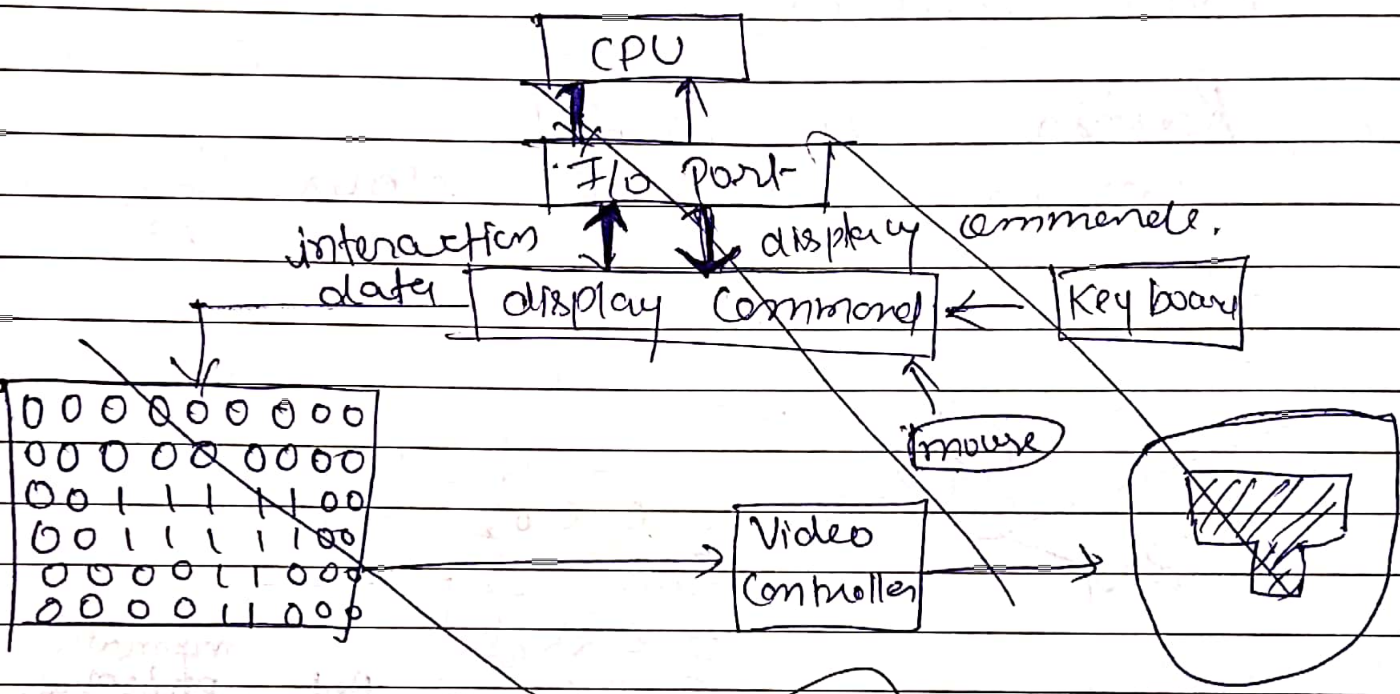


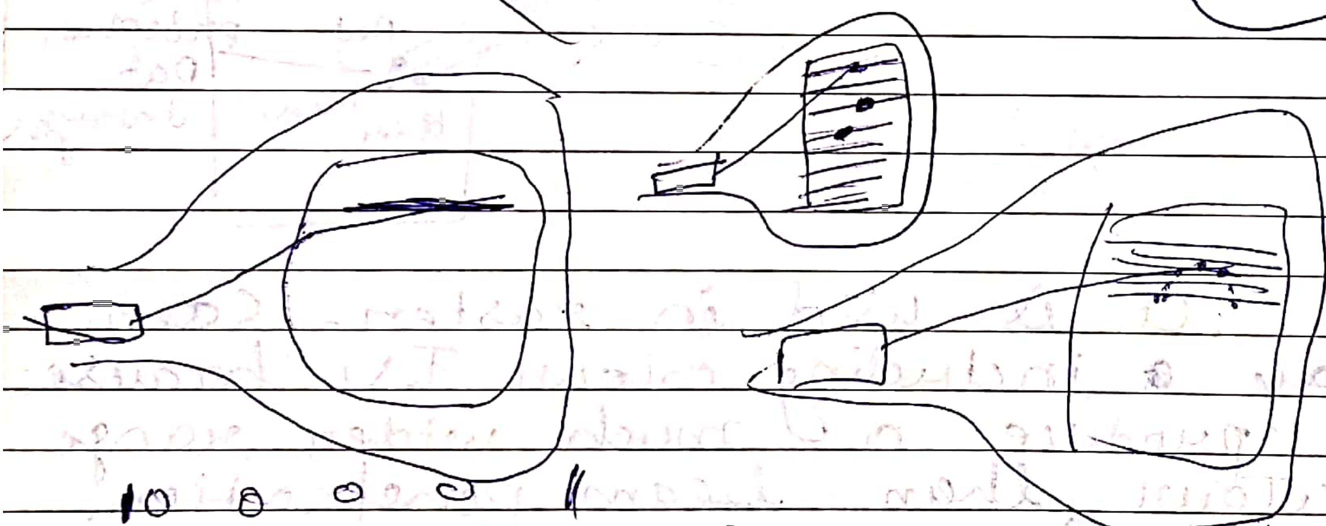
~~8bit~~
~~24bit~~

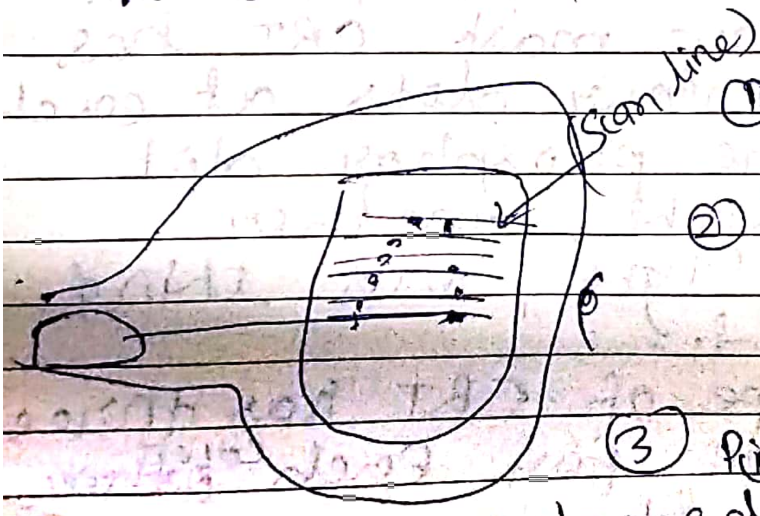
It is used in raster-scan display including colour T.V. because they produce a much wider range of colour than beam penetration method. A shadow mask CRT has three phosphor colour dots at each pixel position : one phosphor dot emits a red light, another emits a green light, and third emits blue light.

This type of CRT has three electron guns, one for each color ~~other~~ dots, and a shadow mask, ~~a sheet of metal~~ engraved with a mask grid just behind the phosphor coated screen. before the stream of electrons produced by the CRT's cathode reaches ~~the phosphor book~~

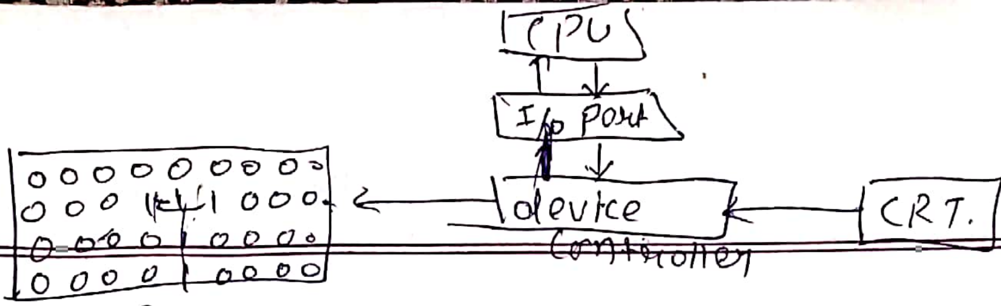
Raster - Scan moniters :-







① The most common type on TV
② electrons are move in row left - to right top to bottom
③ Picture beam intensity turned on and off An create pattern of illustr...



(4) Picture defination stored in a memory area called frame buffer or refresh buffer.

(5) stored intensity value will retrive from frame buffer to be painted

(6) for black and white only 1 bit is used 0 for white when the beam intencity is off vise versa

(7) additional bit are needed when coloured. up to 24 bits per pixel in high quality system depending upon resolution of system.

if 24 bits per pixel and resolution 1024 x 1024 will be have 3 megabyte of storage of frem buffer.

black and white 1 bit per pixel buffer → bitmap.

Coloured multiple pixel frame buffer — Pixmap.

each screen pixel on the screen can be specified by it row and colo

Date: Page:

Shadow masking (used in Raster Scan display)

:- ① In CRT colour monitors, we are using 3 e guns to produce the colour (single pixel is coated with Red, green, & blue dots of phosphorous)

② In shadow masking technique we can generate wider range of colours.

③ Three dots are arranged in triangular shape, so it is also called delta shadow mask method.

Structure of Shadow masking technique

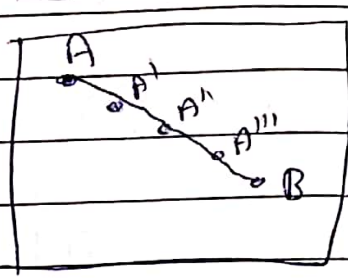
A metal plate series of aligned holes with phosphorous dots are placed in between electron guns and display Screen

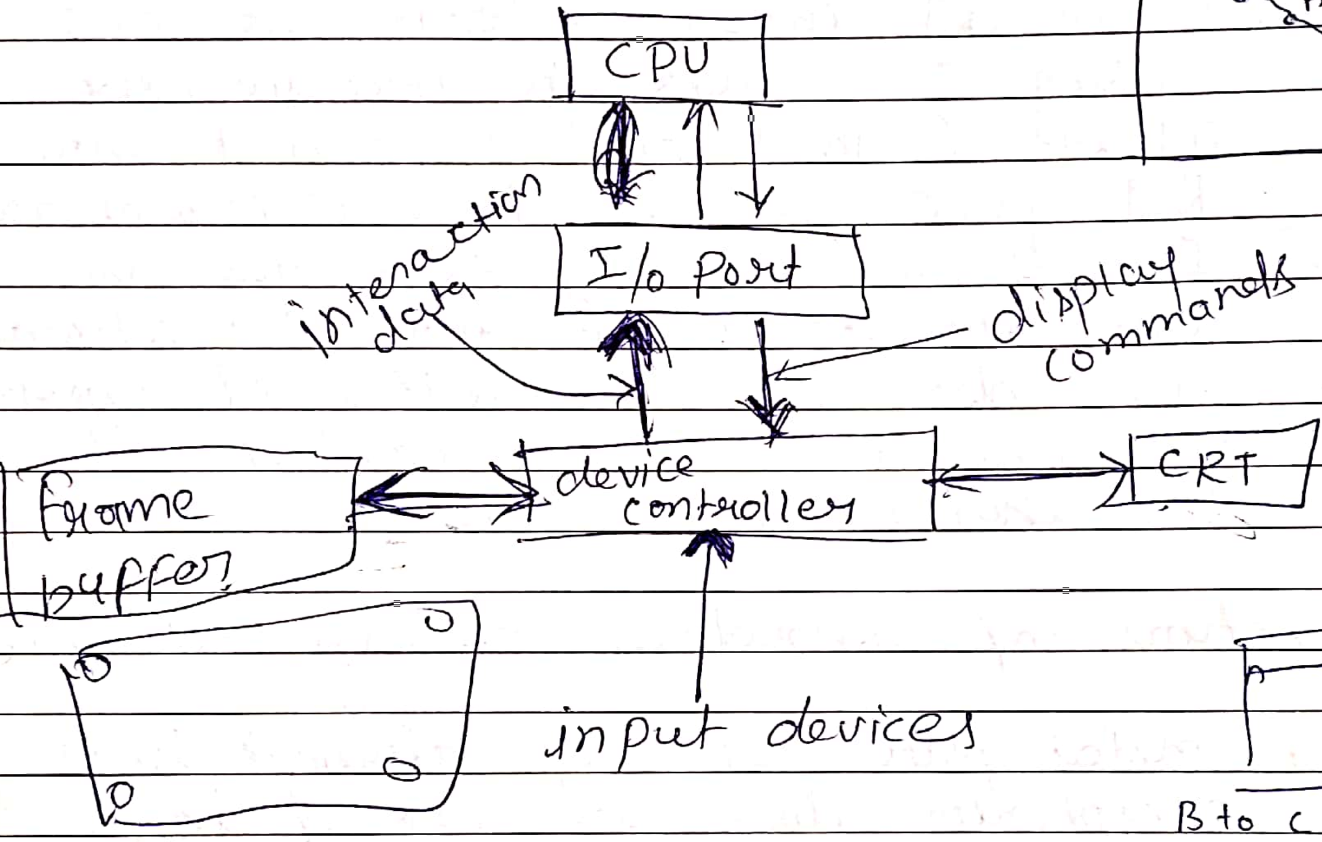
The objective is that when e ~~gun~~ beam penetrate the particular pixel, it's near by pixel should not illuminate.

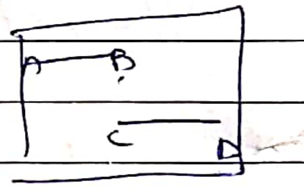
The e beam should activate the corresponding colour dot only.

(video caligraphic, Vector scan, Stroke displ

Random Scan display :-

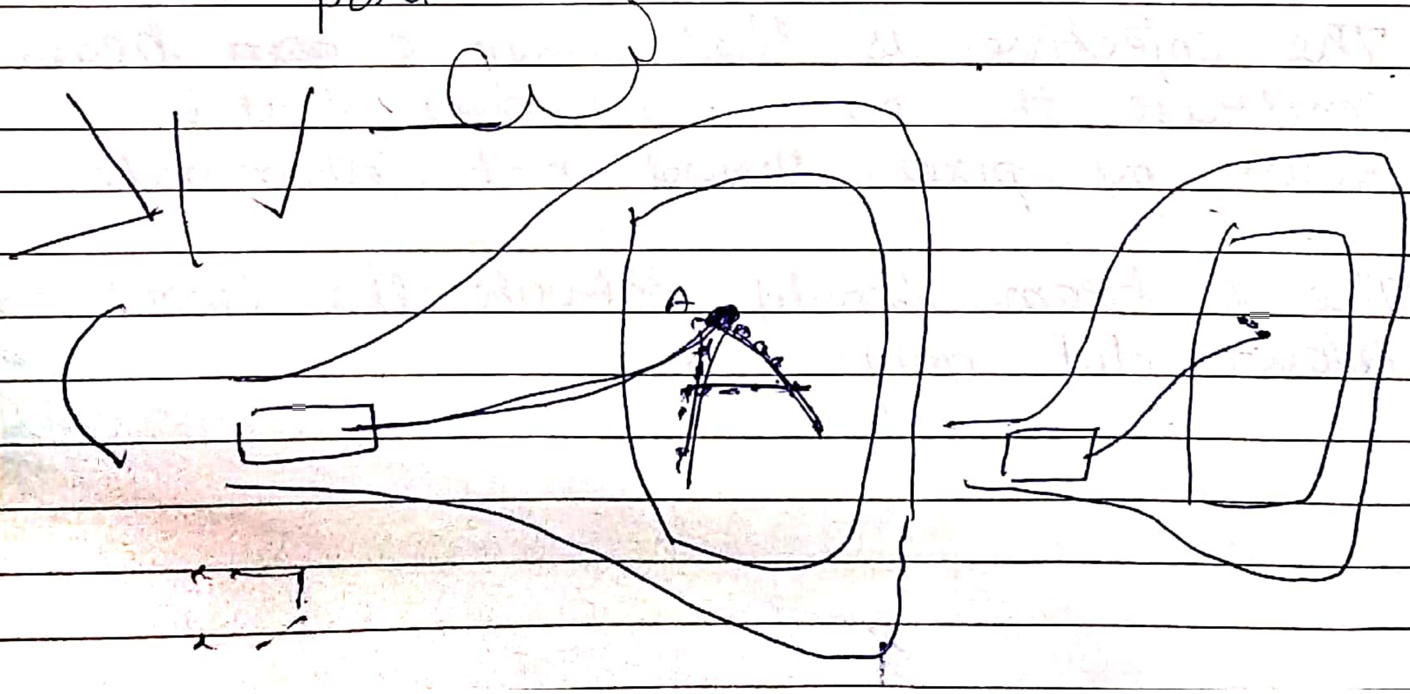


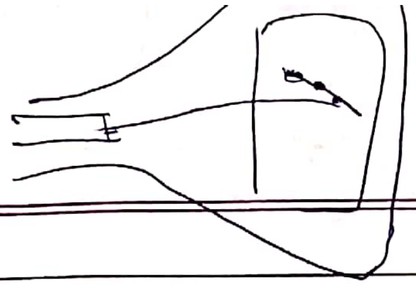




B to c off

" Video moniters are primary output port "





Date:______ Page no:

① When operated as a random-Scan display unit, a CRT has the electron beam directed only to the parts of the screen where a picture is to be drawn

② Random Scan monitors draw a picture on line at the time

③ it is also reffered as vector display stroke writing or calligraphic display.

④ Refresh rate on random-scan system depend on the number of lines to be displayed.

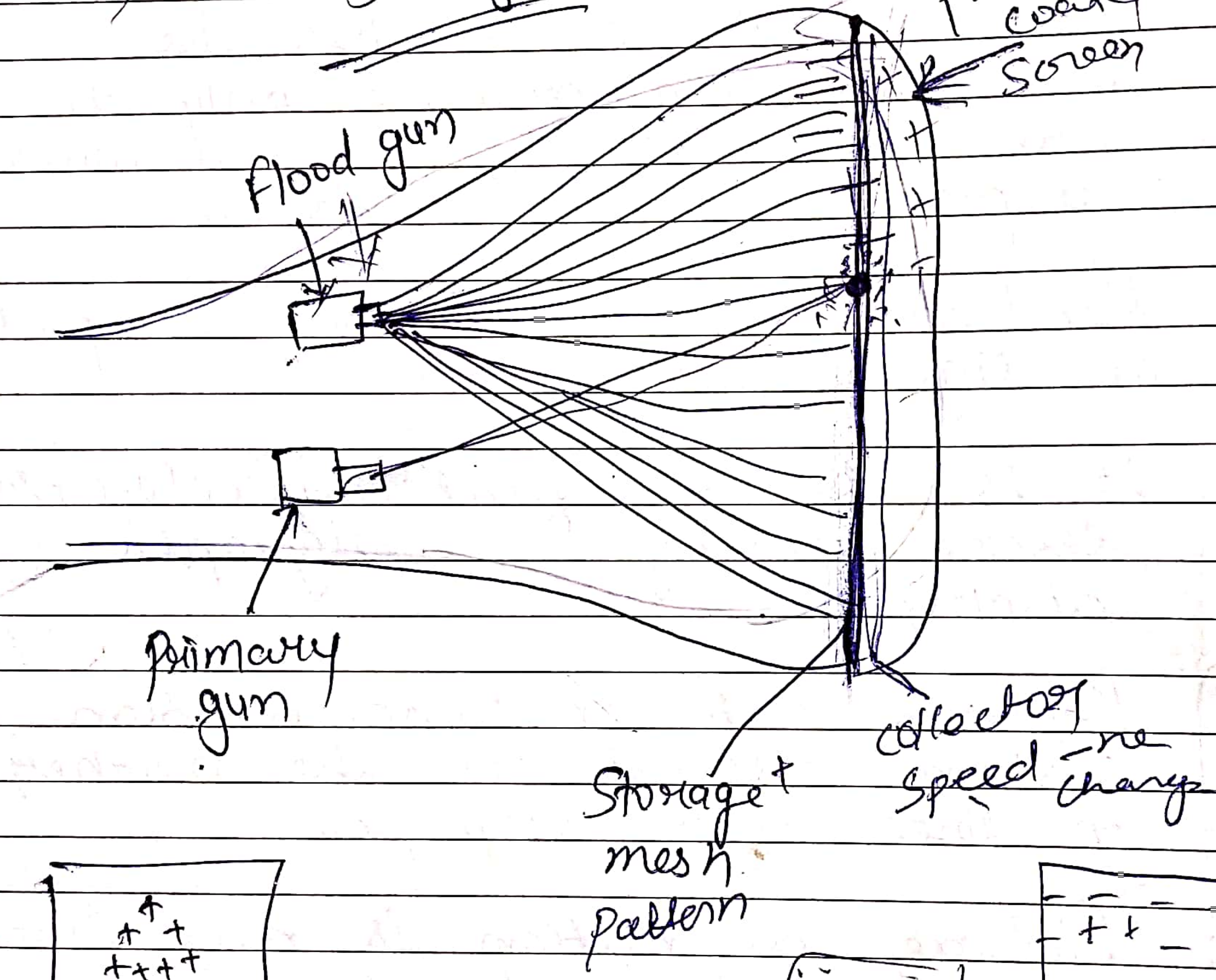
⑤ Picture defination is now stored as a set of line-drawing commands in an area of memory reffered to as the refresh display file.

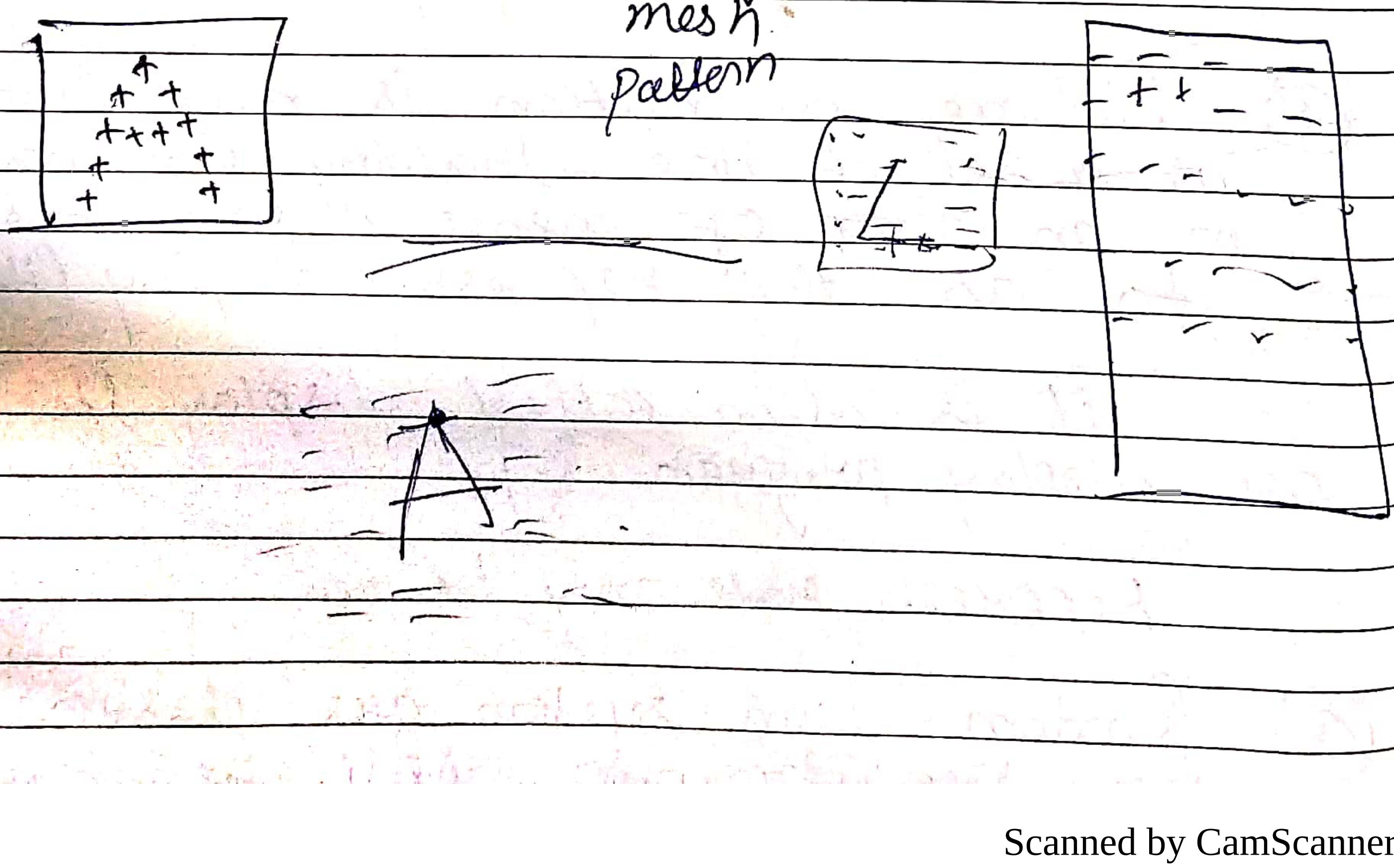
it is allso called display list or display program

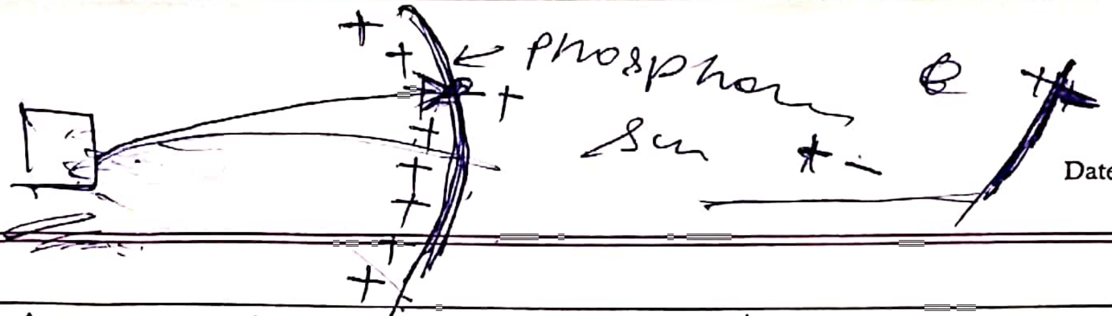
Refresh rate may be 60

⑥ Random-scan system are designed for line drawing application and can not display realistic shades

Direct view storage tube







Date:______ Page no:

Flat panel display :- used as both Input devices & output devices.

devices :- smart phones
TV , computers
Plasma
LCD
e-reader (amazon)

advantage :-

1. more energy efficient than CRT
2. Space saving.
3. better picture
4. Less eye strain (CRT sightly blurring)

two ~~types~~ categories :-

1. Volatile
2. Static

Volatile → constantly refresh even on stationary image

example →

1. TV — Plasma
2. com — LCD
3. smart Phone — LED

DVST :-

an alternative method for maintaining a screen image is to store the picture in-formation ~~is~~ inside the CRT instead of refreshing the screen. A direct-view storage tube (DVST) stores the picture information as a charge distribution just behind the phosphor-coated screen. Two electron guns are used in a DVST. one the primary gun is used to store the picture pattern, the second, the flood gun, maintains the ~~gun~~ picture display.

advantage :- because no refreshing is needed very complex pictures can be displayed at very high resolution without flicker.

disadvantages :- disadvantages of DVST systems are that they ordinarily do not display color and that selected parts of a picture cannot erased To eliminate of a picture section, the entire screen must be erased and the modified picture redrawn

MY ROUGH NOTE BOOK

Flat-panel ~~devic~~ displays

the term flat-panel display refers to a class of video devices that have reduced volume weight, and power requirments compared to a CRT. A significant feature of flat-panel displays is that they are are thinner than CRTs, and we can hang them on walls or wear them on our wrists

Types of flat planel displays.

① Emissive display :- displays that convert electrical energy into light. plasma panel, thin film electroluminescent display, and light-emmitting diodes are examples of emissive displays

② Non-Emissive display :- nonemmivssive displays (or no emitters) use optical effects to convert sunlight or light from some other source into graphics patters. the most important example of a non-emissive plat-panel display is a liquid crystal device.

Scan Converting techniques

(line, circle, ellips)

" scan and convert the line from the fram buffer into the screen.

line drawing :- given the specification for a straight line, find collection of addressable pixel which most closely approximates this line.

1. ~~Straight line should be straight~~
2. line should start and end accurately

DDA :- digital differential analyzer

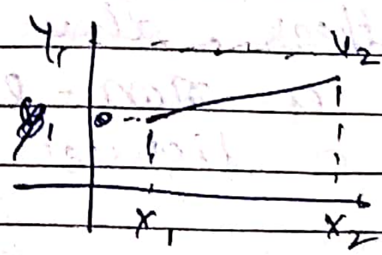
Pixel
~~Computi~~l

Slope intercept line equation

$$\boxed{y = mx + b}$$

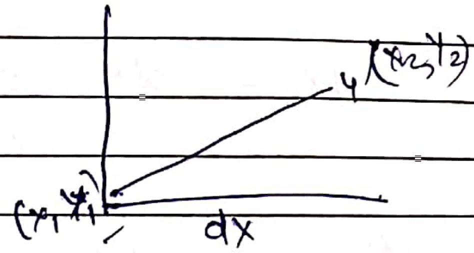
m → slope, b → y intercept

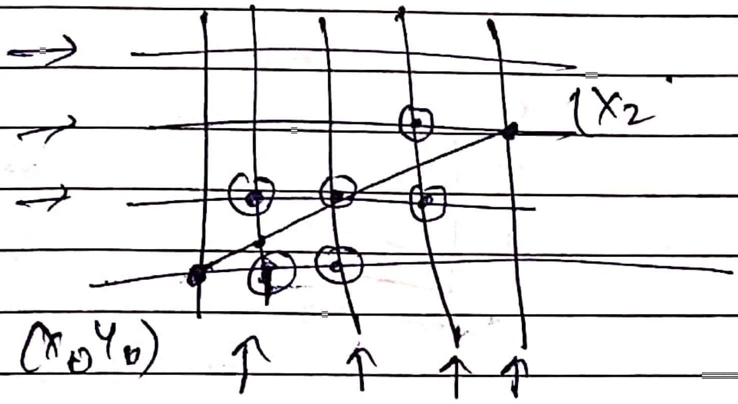
$$\boxed{m = \frac{dy}{dx}}$$



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

$$\boxed{b = y_1 - m x_1}$$

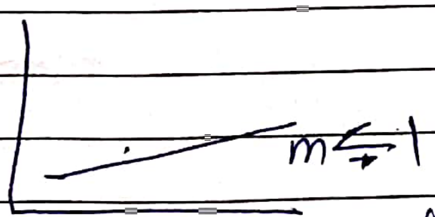


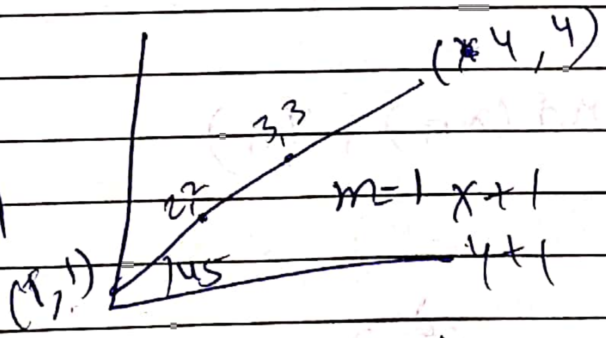


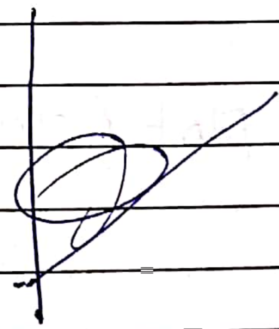
~~jo~~ initial points

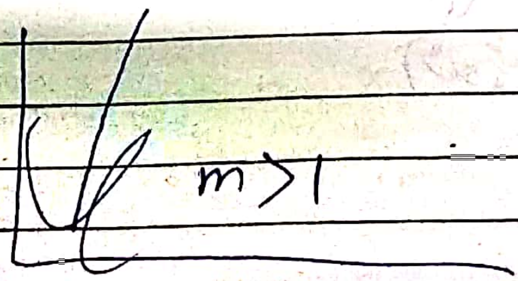
analyse diffrence bet^n point

~~X~~ $X_m =$









~~x~~ 1

Scanned by CamScanner

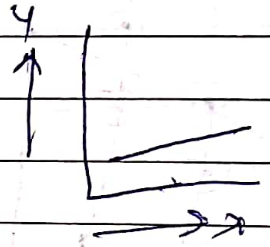
$$m = \frac{dy}{dx} = \frac{y_1 - y_0}{x_1 - x_0}$$

this differential equation can be used to obtain a rasterized straight line

Case 1

$m < 1$

$y = mx +$



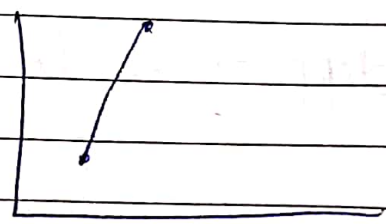
$x_n = x_0 + 1$

$y_n = y + m$ ← slope

Plot $(x_n, \text{round } y_n)$

Case 2

$m > 1$



$y_n = y_0 + 1$

$x_n = x_0 + \frac{1}{m}$ ← slope

Plot $(\text{round}(x_n), y_n)$

(round $3.5 = 4$, $3.4 = 3$)

~~example~~ :-

DDA algorithm

1. Read the line end points (x_1, y_1) and (x_2, y_2)

2. $\Delta x = |x_2 - x_1|$
$\Delta y = |y_2 - y_1|$

3. if $(\Delta x \geq \Delta y)$ then
length $= \Delta x$
else
length $= \Delta y$

4. Select the raster unit

$$\Delta x = \frac{(x_2 - x_1)}{\text{length}}$$

$$\Delta y = \frac{(y_2 - y_1)}{\text{length}}$$

5. $x = x_1 + 0.5 * \text{sign}(\Delta x)$
$y = y_1 + 0.5 * \text{sign}(\Delta y)$

(~~Sign function~~

6. Now plot the point

$i = 1$

while ($i \leq$ length)

{

plot (integer (x) , integer (y))

$x = x + \Delta x$

$y = y + \Delta y$

$i = i + 1$

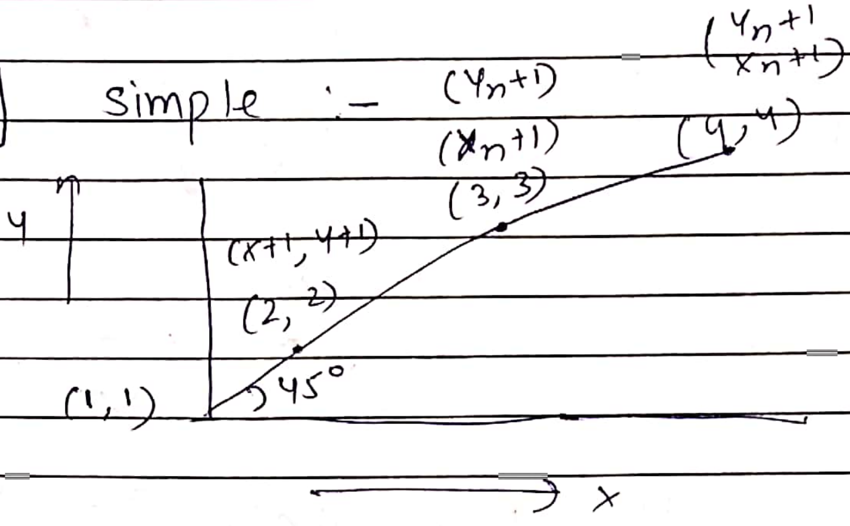
}

Scanned by CamScanner

DDA :-

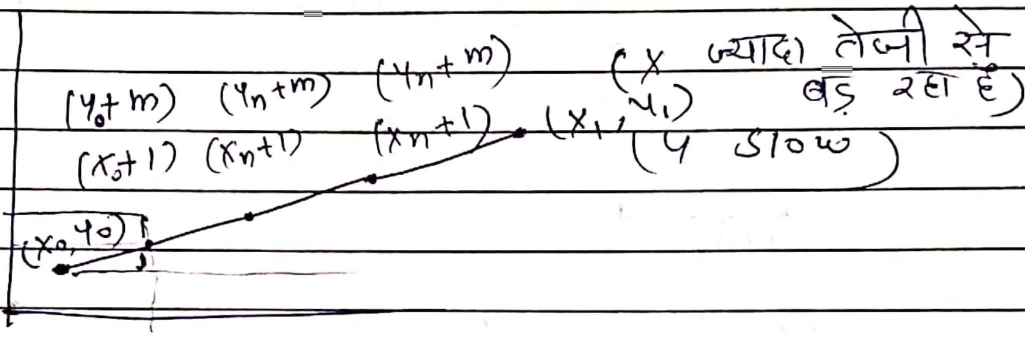
$\boxed{m=1}$ simple :-

~~Case 1 :-~~



Case 1 :-

$\boxed{m<1}$



Starting points → (x_0, y_0) (x_1, y_1)

↑ set pixel

$m < 1$

Next point :-

x_n (x_{new}) = x_0+1

y_n (y_{new}) = y_0+m plot (x_0+1, y_0+m) round

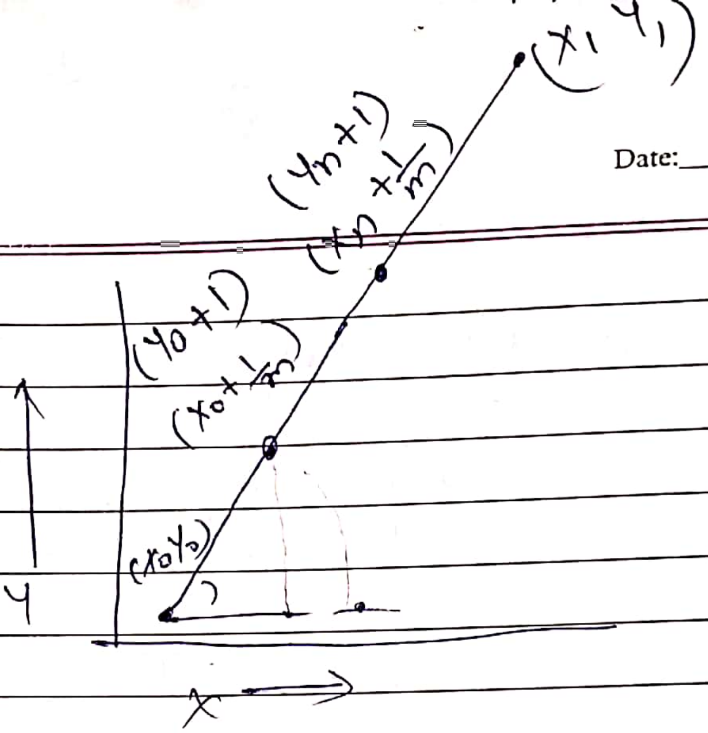
Plot (x_n, round of y_n)

→ (put pixel)

(condition through $x_{next} = x_1$

$y_n = y_1$)

$\boxed{m > 1}$



Starting points (x_0, y_0) (x_1, y_1)

↑ set pixel

$m > 1$

Next points :-

$$x_n = x_0 + \frac{1}{m}$$

$$y_n = y + 1$$

$$\text{Plot} = \left(\text{round } x_0 + \frac{1}{m}, y_0 + 1\right)$$

$$= (\text{round } x_n, y_n)$$

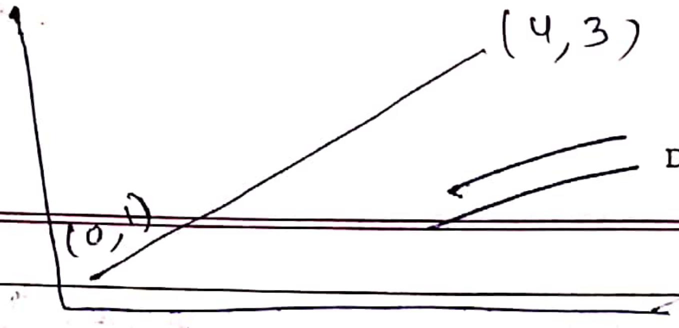
$$= \left(\text{round } x_n + \frac{1}{m}, y_n + 1\right)$$

$$= (\text{round } x_n, \cancel{\text{round}}\ y_n)$$

$$m = \frac{y_1 - y_0}{x_1 - y_0}$$

$$= \frac{3-1}{4-0}$$

$$= \frac{2}{4} \quad 0.5$$



example :-

$m = 0.5 \qquad m < 1$

Put	Pixel	X_n	Y_n
		0	1
1	2	0+1 = 1	1+0.5 = 1.5 = 2
2	3	2	~~2.05~~ 1.5+0.5 = 2
3	2.5	3	2.5+0.5 2.5

Y_n	X_n	X_n	Y_n
2	1	0	1
2	2	0+1 = 1	1+0.5 = 1.5
3	3	1+1 = 2	1.5+0.5 = 2
3	4	2+1 = 3	2.05
		3+1 = 4	2.5+0.5 = 3
~~4~~	~~[illegible]~~	~~4+1~~	3+3.5 3.05

X_n	Y_n
1	2
2	2
3	3
4	3

advantage & disadvantage book

Bresenham's Algorithm :-

① input the two endpoints and store the left end point in (x_0, y_0)

② load (x_0, y_0) into the frame buffer, that is, plot the first point.

③ Calculate constants Δx, Δy, $2\Delta y$ and ~~$\Delta 2\Delta y$~~ $2\Delta y - \Delta y$, and obtain the starting value for the decision parameters as

④ At each x_k along the line starting at $k=0$ perform the following test

if $P_k < 0$, the next point to plot is $(x_k + 1, y_k)$ and

$$P_k + 1 = P_k + 2\Delta y$$

otherwise, the next point to plot is $(x_k + 1, y_k + 1)$ and

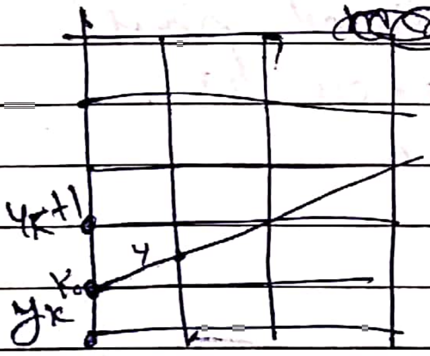
$$P_k + 1 = P_k + 2\Delta y - 2\Delta k$$

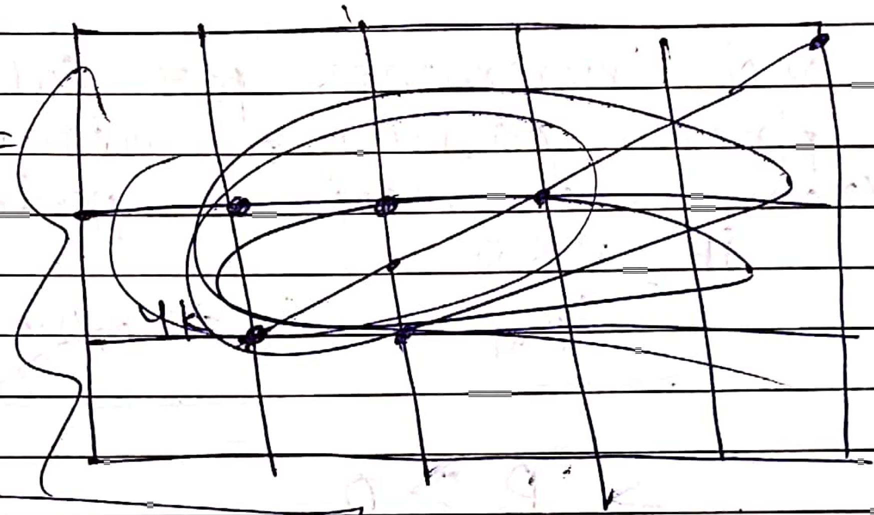
5 Repeat step 4 Δx times.

MY ROUGH NOTE BOOK

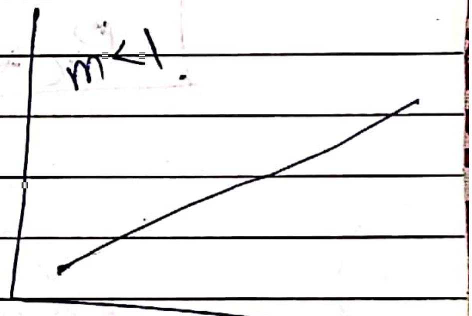
DDA – drawback (round function) to overcome

e= ~~Startpoint~~





$\boxed{\text{decission parameter}}$



$$y = mx + c$$

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

Calculate

Δx, Δy, $2\Delta y$, and $2\Delta y - \Delta y$

x – encrease

$m < 1$

example (20, 10) (30, 18)

$\Delta x = 30 - 20 = 10$

$\Delta y = 18 - 10 = 8$

initial decision parameter = $P_0 = 2\Delta - \Delta$

$$P_0 = 2\Delta y - \Delta x$$
$$= 2 \times 8 - 10$$
$$= 16 - 10$$
$$= 6$$

and increments for calculating successive pixel position along line path from the decision parameter as.

if $P_0 < 0$ $\quad P < 0$

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

$P_0 > 0$

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

~~$P_0 = 6$~~ $\quad P > 0 \quad$ initial param

~~$P_{k+1} = 6 + 16 - 16$~~ $\quad = 2\Delta Y - 2\Delta X$

$$= 16 - 20$$
$$= -4$$

$$P_{k+1} = P_k + 2\Delta y - 2\Delta x$$
$$= 6 - 4$$
$$= 2$$

$P_{k+1} = 2 - 4$ $\qquad P_{k+1} = P_k + 2\Delta y$

$= -2$ $\qquad = -2 + 16$

x_{k+1}

y_{k+1}

Date:______ Page no:

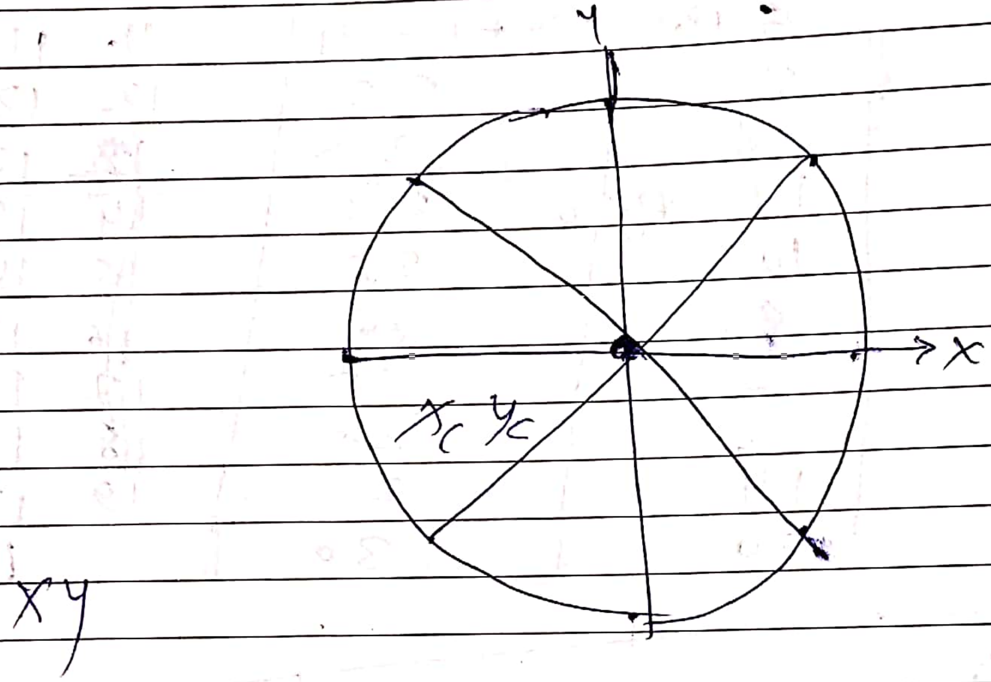
$P_k > 0$

iteration no.	P_k	X_{k+1} 20	Y_{k+1} 10	
0	6 (P_0)	20 + 1 = 21	11	11
1	2	22	12	12
2	−2	23	12	12 (y_k)
3	14 $P_k < 0$	24	14	13
4	10	25	15	14
5	6	26	16	15
6	+2	27	17	16
7	−2	28	18	16
8	14	29	19	17
9	10	30		18

if $P_k < 0$ $\quad y_k = y_{k+1}$

$P_k > 0$ $\quad y_k$

Mid point circle :-



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

$$y = \pm\sqrt{r^2 - x^2}$$

8 fold symmetry

Algo :-

① Input radius r and circle centre (x_c, y_c) and obtain the first point on the circumference of a circle centered on the origin as -

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

② Calculate the initial value of the decision parameters as.

$P_0 = \frac{5}{4} - r \simeq 1 - r$

(3) at each x_k position, starting at $k = 0$, perform the following test if $P_k < 0$, the next point along the circle centered on $(0, 0)$ is (x_{k+1}, y) and

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

otherwise, the next point along the circle is $(x_k + 1, y_k - 1)$ and

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

where $2x_{k+1} = 2x_k + 2$ and $\cancel{2P_{k+1}} = 2$

$$2y_{k+1} = 2y_k - 2$$

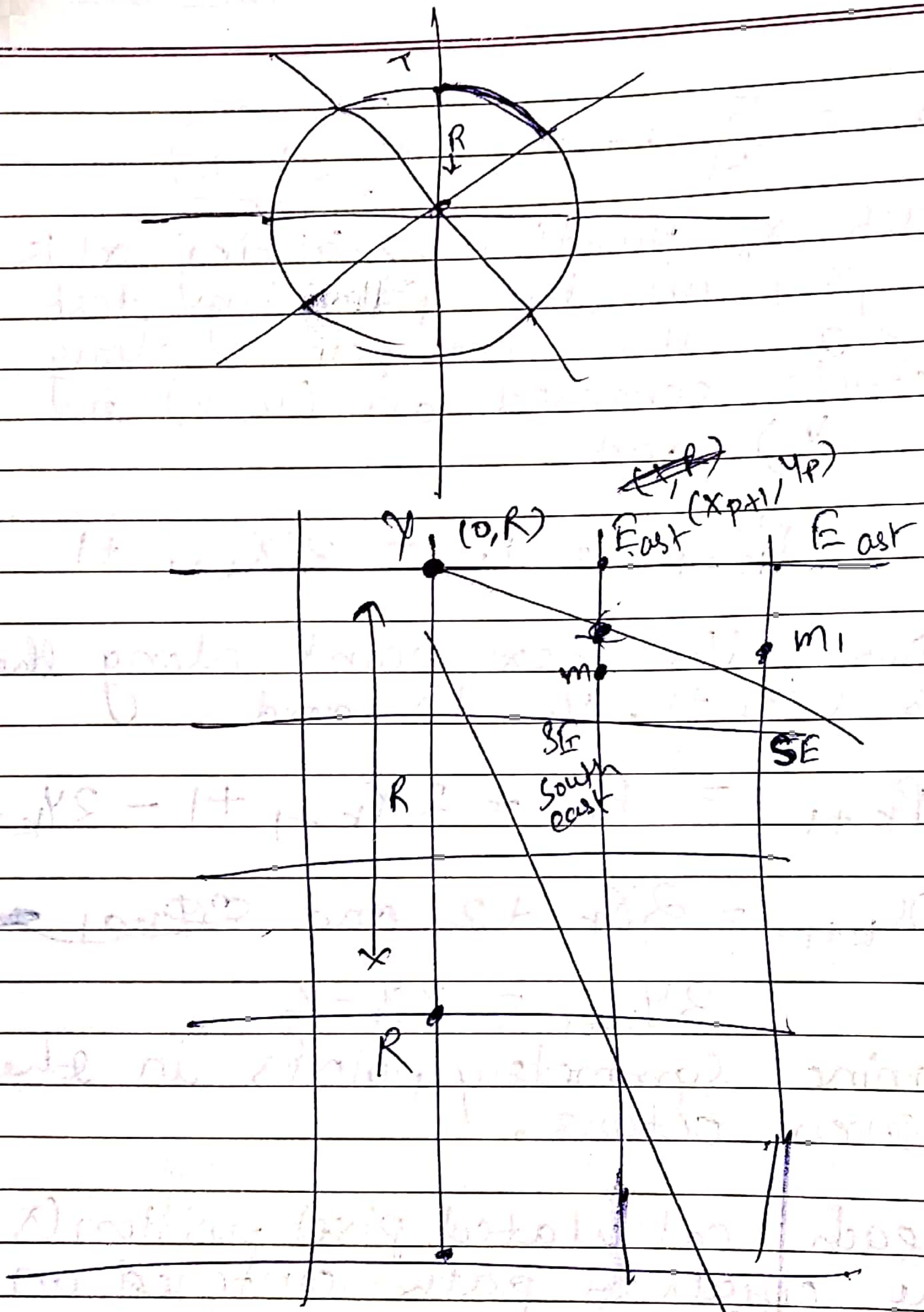
(4) Determine symmetry points in the other seven octans.

(5) move each calculated pixel position (x, y) onto the circular path centered on (x_c, y_c) and plot the coordinate values:

$$x = x + x_c$$

$$y = y + y_c$$

(6) Repeat steps 3 through 5 until $x \geq y$.



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

for given point (x', y')

$f(m)$ $f(x', y') = 0$ (pixel on the circle)

$f(m)$ $f(x', y') > 0$ (pixel outside the circle)

$f(x', y') < 0$ (pixel inside the circle

$m > 0$ (circle line near to east)

$f(x, y) = x^2 + y^2 - R^2 = 0$

evaluate $f(m)$ ~~outside~~ inside

if $f(m) < 0$ (~~inside~~ circle)

Choose SE ~~inside~~ outside

if $f(m) > 0$ (outside circle)

Choose E

decision point (x_p, y_p)

$d = f(m) = f(x_{p+1}, y_p - \frac{1}{2})$

$d = (x_p + 1)^2 + (y_p - \frac{1}{2})^2 - R^2$

when $d < 0$ (when E selected)

x की value बढ़ती

$d_{new} = (f m') = f(x_p + 2, y - \frac{1}{2})$

$d_{new} = (x_p + 2)^2 + (y_p - \frac{1}{2})^2 - R^2$

$\Delta E = d_{new} = 2x_p + 3$

when SE selected (x की बढ़ेगी y की 1 pixel decrease है)

$$d_{new} = F(m') = F\left(x+2,\ y_p - \frac{3}{2}\right)$$

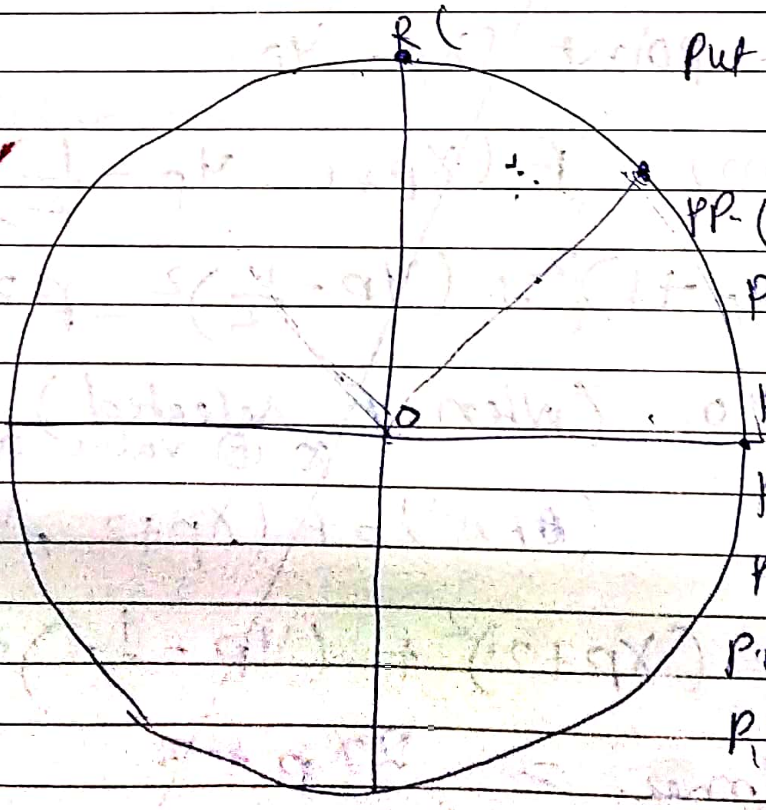
$$\Delta SE = d_{new} - d$$

$$= 2x_p - 2y_p + 5$$

if $x = 0$ $\cancel{R=}$ $y = R$

$$d = (0+1)^2 + \left(R - \frac{1}{2}\right)^2 - R^2$$

$$= \frac{5}{4} - R$$



put pixel (xcenter + x,
ycenter + y);

pp (xcenter − x, ycenter + y);

pp (" + x, " − y);

pp (" − x, " − y);

pp (" + y, " + x);

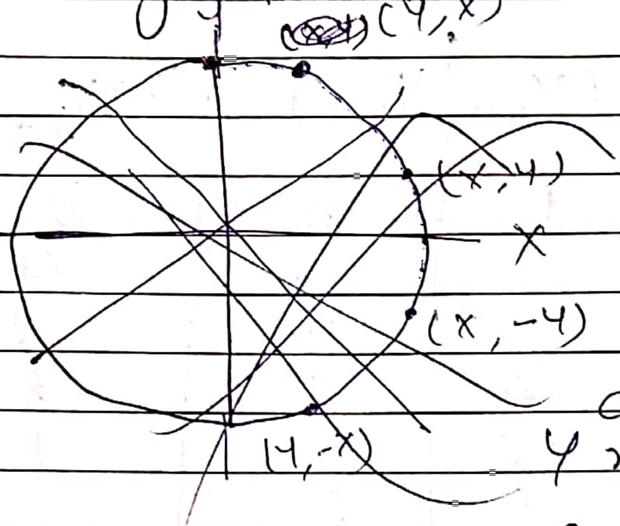
pp (" − y, " + x);

pp (" + y, " − x);

pp (" − y, " − x);

Scanned by CamScanner

mid point algorithm

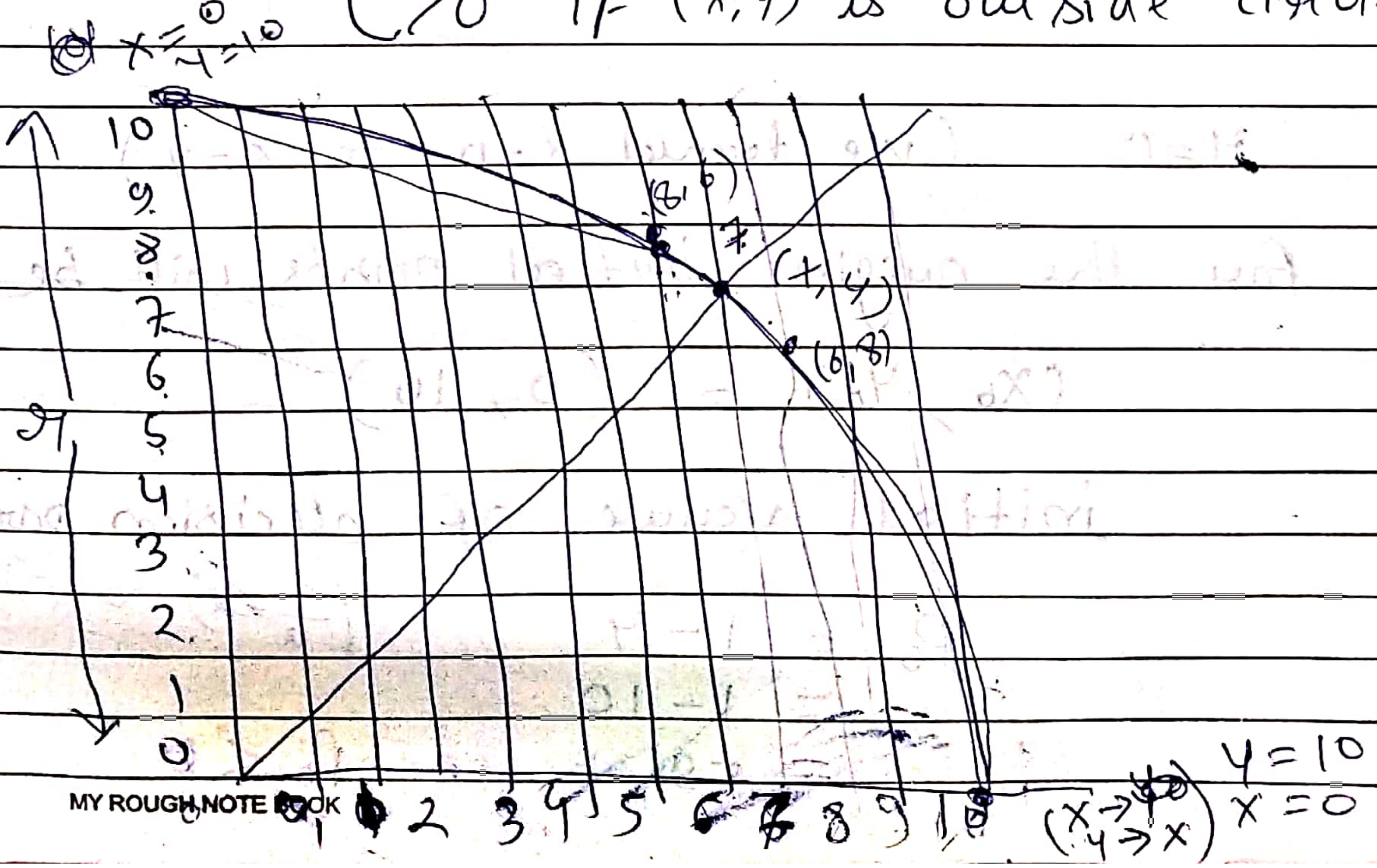


It moves 90° to 45°

and x moves to positive direction and y moves in -ve

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

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



neumerical :-

K	P_K	$(X_{k+1},$	$Y_{k+1})$	$2X_{k+1}$	$2Y_{k+1}$
0	-9	1	10	2	20
1	-6	2	10	4	20
2	-1	3	10	6	20
3	6	4	9	8	18
	-3	5	9	10	18
	6	6	8	12	16
	3	7	7	14	14

$r=0$ (we decided $x=0$ to $x=y$)

For the origin initial points will be

$$(x_0, y_0) = (0, 10)$$

initial value of decision parameter

$$P_0 = 1 - r \qquad \left(\frac{5}{4} - r\right) \approx (1 - r)$$
$$= 1 - 10$$
$$= -9 \qquad P_0 < 0$$

and $2x_0 = 0$ $2y_0 = 20$

① P_k $(P_0) < 0$ (-9) So

$x_{k+1} = 0+1 = 1$

$y_{k+1} = y_k = 10$

$\boxed{2x_{k+1} = 1 \times 1 = 2}$

$\boxed{2y_{k+1} = 2 \times 10 = 20}$

② because $P_k < 0$

So

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

$= -9 + 2 + 1$

$\boxed{P_{k+1} = -6}$

$P_{k+1} < 0$ (-6) So

So $x_{k'+1} = 1+1$
$= 2$

$y_{k'+1} = y_k = 10$

$2x_{k+1} = 2 \times 2 = 4$

$2y_{k+1} = 10 \times 2 = 20$

(3) $P_k < 0$

$$P'_{k+1} = P_k + 2X_{k+1} + 1$$
$$= -6 + 4 + 1$$
$$\boxed{P_{k+1} = -1}$$

$$X_{k+1} = 2+1 = 3$$

$$Y_{k+1} = Y_k = 10$$

$$2X_{k+1} = 2 \times 3 = 6$$

$$2Y_k + 1 = 2 \times 10 = 20$$

(4) $P_k < 0$

$$P_{k+1} = P_k + 2X_{k+1} + 1$$
$$= -1 + 2 \times 3 + 1$$
$$= -1 + 7 = 6$$
$$\boxed{P_{k+1} = 6}$$

$$X_{k+1} = 3+1 = 4 \qquad 2X_{k+1} = 8$$

$$Y_{k+1} = 10 - 1 = 9 \qquad 2Y_{k+1} = 18$$
$$= 9$$

(5) $P_k > 0$

$$P_{k+1} = P_k + 2X_{k+1} + 1 - 2Y_{k+1}$$
$$= 6 + 8 + 1 - 18$$
$$= 15 - 18 = -3$$

$= -3 + 10 + 1$ $\quad 6 + 12 + 1 - 16$
$= 6$ $\quad = 3$

10:30

$x_{k+1} = 4 + 1 = 5$

$y_{k+1} = y_k = 9$ $\quad$ (∵ $P_k < 0$)

$2x_{k+1} = 10$ $\quad$ $2y_{k+1} = 18$

(6) till (7,7)

Bresnham's algorithm for circle drawing :-

1. Read the radius r of the circle.
2. Initialize the decision variable
3. $d = 3 - 2 * r$
   $x = 0$
   $y = r$

4. if we are using octants symmetry to plot the pixels then until $(x < y)$ we have to perform following steps,

if $(d < 0)$ then
$d = d + 4x + 6$
$x = x + 1$
$y = y$

else ~~$d = d + 4x + 6$~~
~~$x = x + 1$~~
$d = d + 4(x - y) + 10$

$d \geq 0$

Date:_______ Page no:

$y = y - 1$

$x = x + 1$

$y = y - 1$

$x = x + 1$

$x = x + 1$

$y = y$

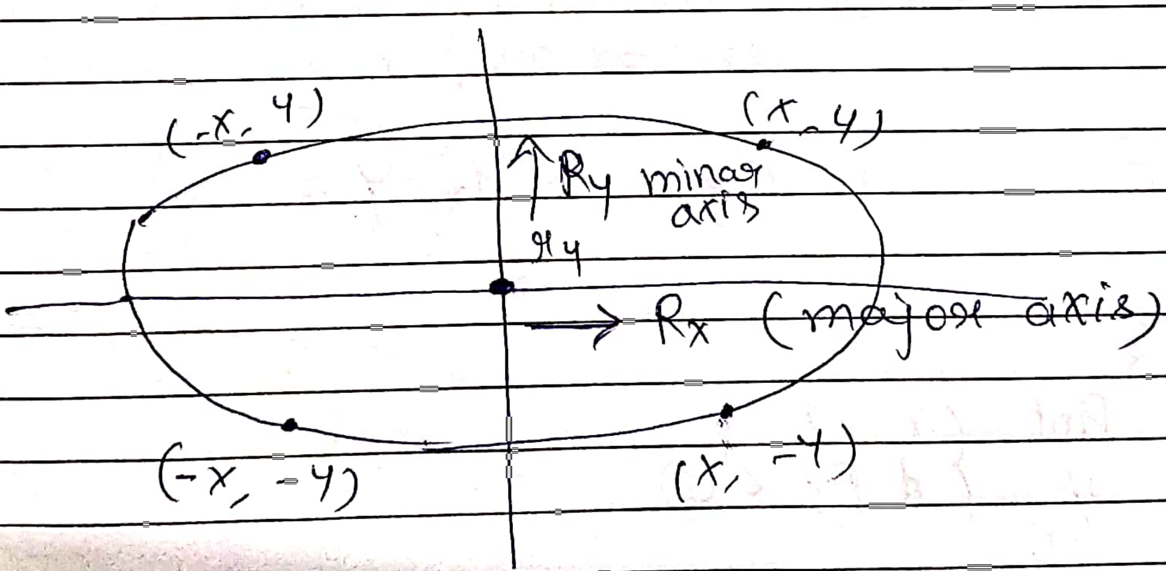
5. Plot (x, y)
6. Determine the symetry points in other octants also
7. ~~6~~ Stop

B. Algo	DDA Algo
1. Bresnham algo. uses fixed points, that is integer arithmetic	1. DDA uses floating points that is real arithmetic
2. B. uses only subtraction and addition in it's operators	2. DDA uses multipication and division in its
3. Bresenham is faster than DDA	3. DDA is rather slower than B. Algo.
4. B. algo. is more efficient and much accurate than DDA Algo	4. DDA algo. is not as accurate and efficient as B. A
5. B. algo. is less expensive than DDA algo	5. DDA algorithm uses an enormous number of floating point multipication so it is ex-pensive

(6) B. algo does not round off but takes the incremental value in it's operation	(6) DDA algo. round off the cordinates to integer that is nearest to the line. Rounding off of the pixel position obtained by multiplication or division causes an accumulation of error in the proceeding pixels

elips generation :-

$$\text{Basic equation} \Rightarrow \left(\frac{x}{a}\right)^2 + \left(\frac{y}{b}\right)^2 = 1$$



4 symmetri

2a = major axis
2b = minor axis

Mid point ellips algorithm

① Read radii r_x and r_y

② initialise starting point as

$$x = 0$$
$$y = r_y$$

Plot (x, y)

③ Calculate the initial value of decision parameter in region 1 as,

$$P_1 = r_y^2 - r_x^2 r_y + \frac{1}{4} r_x^2$$

④ ~~init~~ initialize dx and dy as,

$$dx = 2r_y^2 x$$

$$dy = 2r_x^2 y$$

⑤ do
{

Plot (x, y)
if $(P1 < 0)$
{
$x = x+1$
$y = y$
$dx = dx + 2r_y^2$
$P1 = P1 + dx + r_y^2$
}

MY ROUGH NOTE BOOK

$(P_1 > 0)$

else {

$x = x+1$

$y = y-1$

$dx = dx + 2r_y^2$

$dy = dy - 2r_x^2$

$P_1 = P_1 + dx - dy + r_y^2$

}

}

while

$(dx \leq dy)$

6. Calculate the initial value of decision parameter in region 2 as

$$P_2 = r_y^2 (x + \tfrac{1}{2})^2 + r_x^2 (y-1)^2 - r_x^2 r_y^2$$

7. do

{

Plot (x, y)

if $(P_2 > 0)$

{

$x = x$

$y = y-1$

$dy = dy - 2r_x^2$

$P_2 = P_2 - dy + r_x^2$

else

{

$x = x + 1$

$y = y - 1$

$dy = dy - 2 r_x^2$

$dx = dx + 2 r_y^2$

$p2 = p2 + dx - dy + r_x^2$

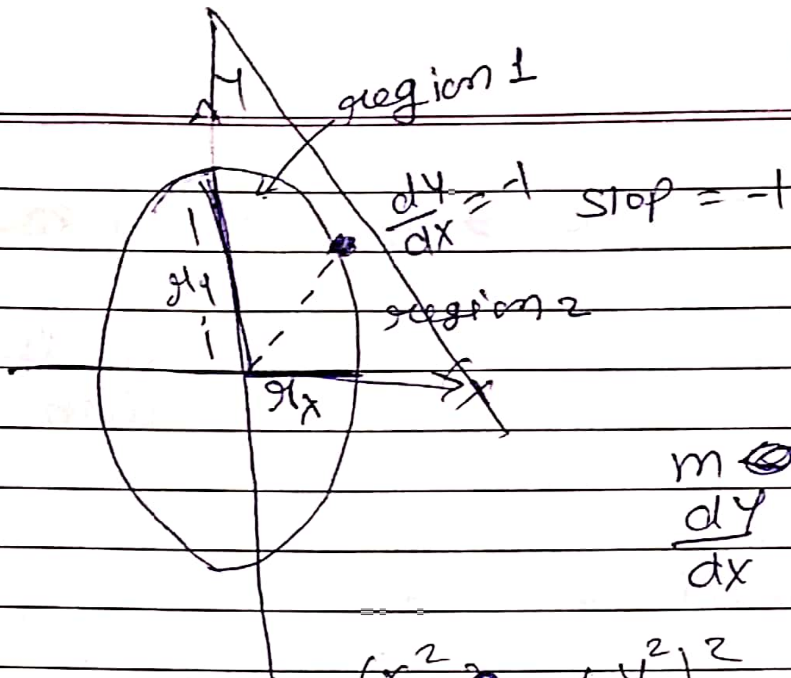
}

}

while ($y > 0$)

8. determine symmetrical points in other three quadrants
9. Stop.

-1 +1



$m\,\cancel{o}l = -1$

$\frac{dy}{dx} = -1$

elips equastion $\left(\frac{x^2}{r_x^2}\right)^{2} + \left(\frac{y^2}{x_y^2}\right)^2 = 1$

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

$$x^2 r_y^2 + y^2 r_x^2 = r_x^2 r_y^2 \begin{cases} < 0 & \text{implies } (x,y) \text{ inside the ellipse} \\ = 0 & \text{implies } (x,y) \text{ on the ellips} \\ > 0 & \text{implies } (x,y) \text{ outside the ellipse} \end{cases}$$

origin's 1's ending point will become origin2's starting point.

$$F(x,y) = x^2 r_y^2 + y^2 r_x^2 - r_x^2 r_y^2$$

$\frac{dy}{dx} = -1$ (region 1 to 2 shifting)

$-\frac{2r_x^2 y^2}{2r_y^2 x^2} = -1$

$2r_y^2 x = 2r_x^2 y$ [

[before this we increase x then y increase]

Neumerical :-

$r_x = 8 \quad r_y = 6$

$P_1 = r_y^2 - r_x^2 \cdot r_y + \frac{1}{4} r_x^2$

$= 6^2 - 8^2 \times 6 + \frac{1}{4} 8^2$

$= 36 - 64 \times 6 + \frac{1}{4} 64$

~~= 36 -~~

Polygon filling :-

Introduction :-

A polygon may be ~~represted~~ represented as a number of line segments connected, end to end to form a closed figure. alternatively, it may be represented as the points where the sides of the polygon connected. the lines segment which make up the polygon boundry are called "Sides or edges. the end points of the sides are called the polygon "Vertices". the simplest polygon is the triangle, having three sides and three vertex points.

Polygon can be divided into two classes :- (1) convex (2) concave

"A convex polygon is a polygon such that for any two points inside the polygon all points on the line segment connecting them are also inside the polygon."

A concave polygon is one which is not convex.

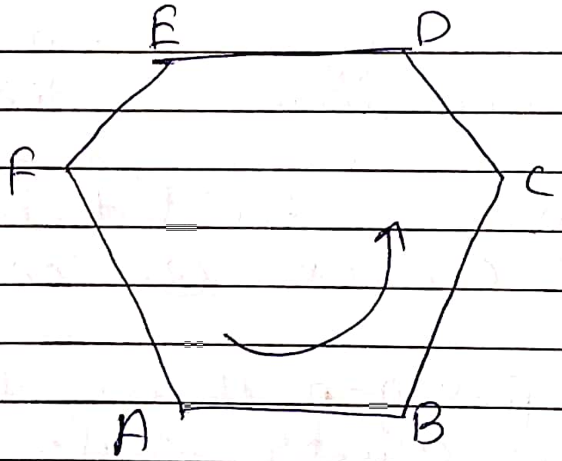
Thus to determine whether or not a polygon is convex ask the fallowing question

Does a straight line connecting any two points that are inside the polygon?

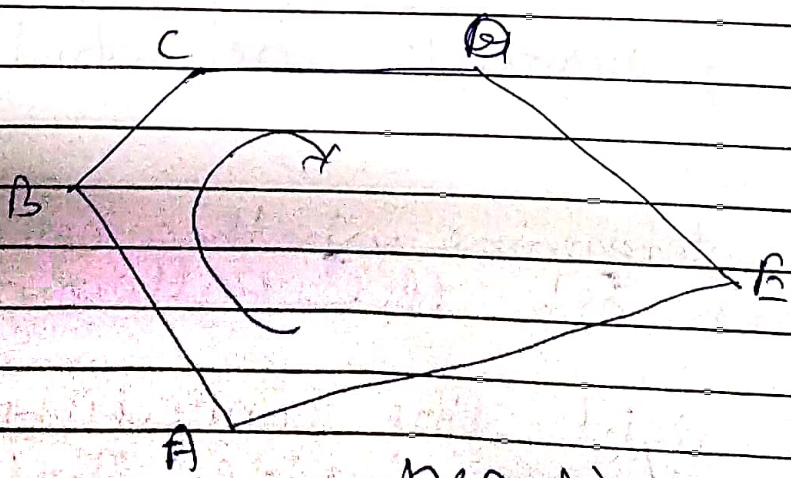
Date:______ Page no:

if the answer is no, the polygon is convex otherwise it is concave.

We have a normal convention of writing the polygon as P vertices, P_1, P_2... P_n. A polygon is said to be "positively oriented" if visiting of all the vertices in the given order produces a counter clockwise circuit, otherwise it is said to be negative-oriented polygon.



Positive orientation



negative orientation

MY ROUGH NOTE BOOK

Scanned by CamScanner

~~Polygon Representation:-~~

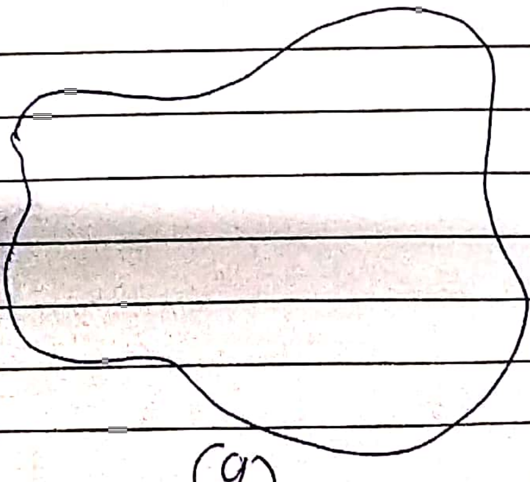
Polygon filling :-

Filling is the process of "colouring in" a fixed area of or a region. there are two basic approaches to area filling in raster system. one way to fill an area is Scan line fill algorithm another method for area filling is to start from a given interior position and point outward from this point, until we encounter the specified boundry condition.

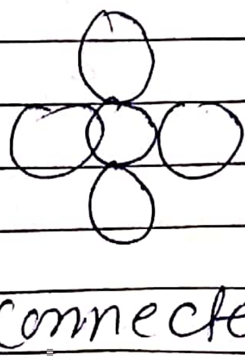
1. Boundry fill } seed fill
2. Flood fill }
3. Edge fill
4. Fence Fill

Boundry fill algorithm

(1)



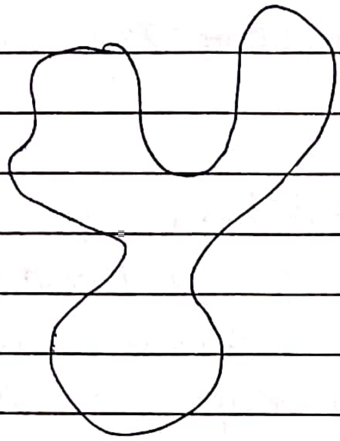
(a)

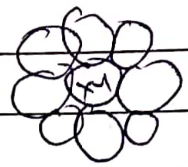


4 connected pixel

Date:______ Page no:

(2)





8- Connected pixel

colour inside

boundry colour

```
void fill (int x, int y, int fill, int boundry)
{
int current = get pixel (x, y);
if ((current != boundry) & current != fill)
 {
  Set Color (fill);
  Setpixel (x, y);

  boundryFill (x+1, y, fill, boundry);
  boundryFill (x-1, y, fill, boundry);
  boundry fill (x, y+1, fill, boundry);
  boundry fill (x, y-1, fill, boundry);
 }
}
```

Flood fill algorithm :-

this algorithm also begins with a seed (starting pixel) inside the region. it checks to see if the pixel has the region original colour. if the answer is yes, it returns to the caller.

This method shares great simarities in it's operating principle with the boundry fill also.

```
Void FloodFill (int x, int y, int fill, int color)
{
If (get Pixel (x, y) == old color)
{
set Color (Fill);
set Pixel (x, y);
floodfill (x+1, y, fill, old color);
Flood Fill (x-1, y, Fill, old color);
Flood Fill (x, y-1, fill, old color);
}
}
```

MY ROUGH NOTE BOOK

2nd Unit

2-D transformation :- (translation, Rotation

Flood fill	boundry fill
① Area filling is started from a point	Byte code Secure Pointer की जगह

Date: Page no:

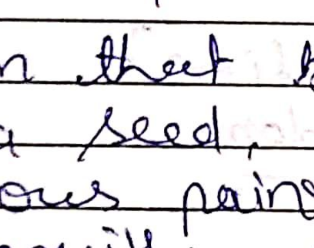
and 8 - connected

4 - Connected :- there are techniques in which pixels are considered as connected to each other. in 4 - connected method the pixels may have upto 4 neighbouring pixels, as right, above, left and below of the current pixel

Similarly, in 8 - connected method the pixels may have upto 8 neighbouring pixels.

Boundry fill algorithm:-

This is recursive algorithm that begins with a starting pixel called a seed, inside a region and continuous painting toward the boundry the algorithm checks to see if this pixel ~~is a~~ boundry (a) pixel or has already filled, if answer is No, it fills the pixel and makes



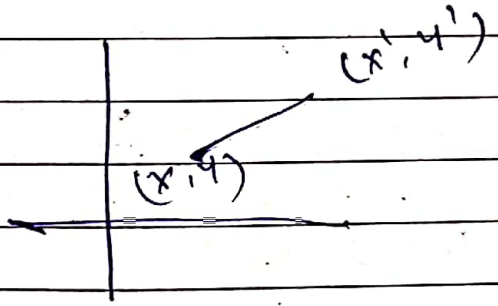
2-D transformation, translation, Rotation.

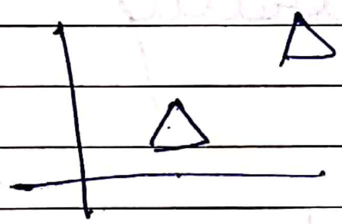
Geometric transformation :-

Change in orientation, size and shape are accomplished with geometric transformation that alter the co-ordinates descripition of objects.

The basic geometric transformation are translation, rotation and scaling

Translation :-





Translation consist of a shift of the object parellel to itself in any direction in the (x,y) plane, any such shift can be accomplished by a shift in x-direction plus a shift in y direction.

if the amount of x-shift