Book: Baker

Computer Graphics & Multimedia

23/01/18

COMPUTER GRAPHICS

› Computer graphics is the field of visual computing where one utilize computers both to generate visual image and to integrate or alter visual and sampled information from the real world.

› Computer graphics is the discipline of producing picture or image using a computer including modelling like creation, manipulation and storage of geogramatic object and convert the screen to as an image or the process of transformation, reflection, shading and animation of the image.

Definition of program introduced by - 'Niklus Wirth'

Program = Algorithm + Data Structure.

Similarly, definition of computer graphics.

Computer graphics = Graphics algorithm + Data structure + languages

★ Data structure is suitable for computer graphics

★ Algorithm suitable for picture generaⁿ. and transformation

★ Language construct for generation of graphics object.

#

Uses of computer graphic

Type of object
- 2D
- 3D

Type of Interaction
- Controllable
- Non Controllable

Pictorial Representation
- line Drawing
- Black & white
- colour
- circle

Applications of Computer Graphics
1) CAD (Computer Aided Design) - Used for designing.
2) Presentation graphics (ppt, gif presentation)
3) Computer Art (video games)
4) Entertainment
5) Education and training
6) Visualization
7) Image processing.
8) Graphical User Interface (GUI)

29/01/18

Components of computer graphic.
1) Digital memory buffer
2) TV Monitor
3) Display controller.

CPU $\xrightarrow{\text{Update}}$ Frame Buffer $\xrightarrow{\text{Refresh}}$ Display Controller $\xrightarrow{\text{Pixel}}$ Video Monitor

Frame Buffer - Picture definition is stored in a M/y called frame buffer.

Display devices - The display system where the graphics are saved in the console screen of computer. It is responsible

Date

Role of picture

1) Used for representation
2) Used as a product design (Auto CAD)

Kind of picture
1) Symbolic   Realistic

for graphics display.
Mainly used in Monitor or TV.

4 TYPES OF VIDEO MONITORS -
~~types of Monitors (Display devices)~~

(*1) Raster Scan display - → They uses CRT for the display
(*2) Random Scan display -
3) Direct View storage tube
4) Flat Panel display.

→ They uses CRT

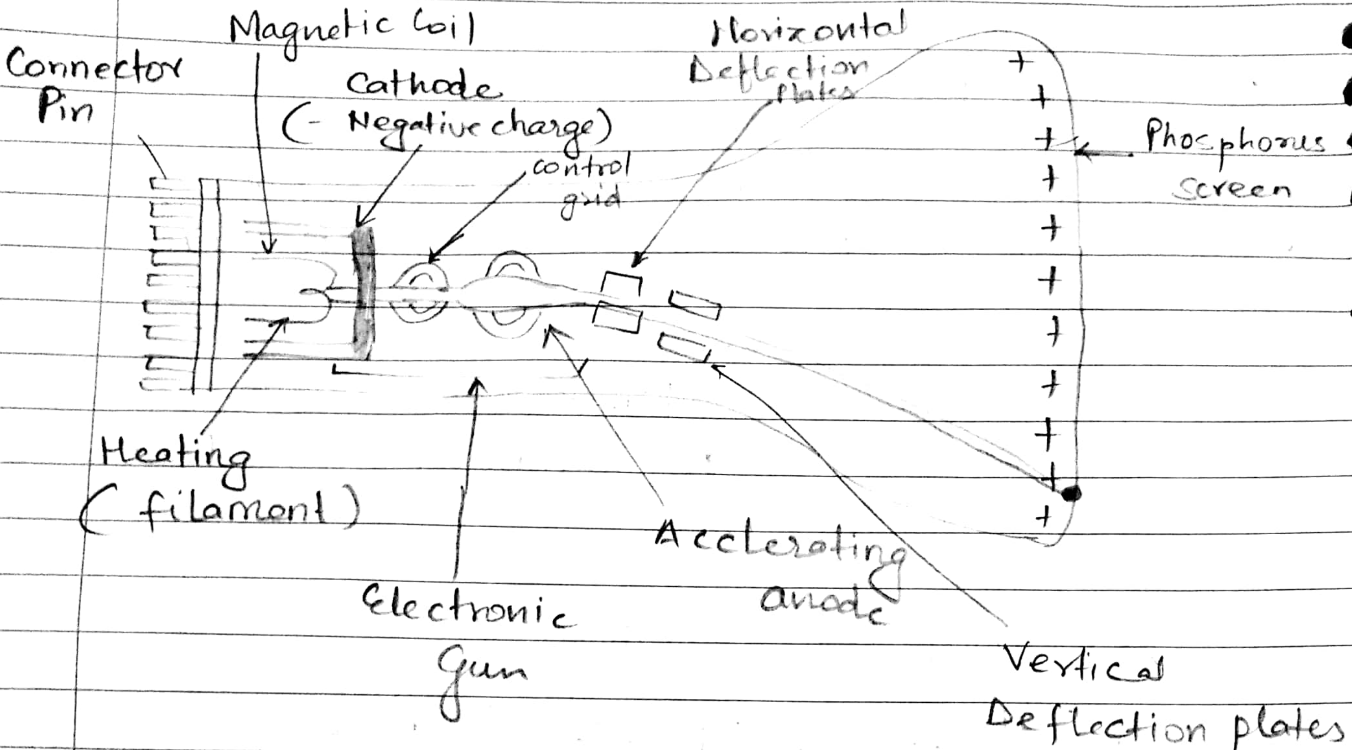
First - Monitor
Secondary - Printer

Display system are often referred to as video monitor or video display.
Most video monitors are based on cathode-ray tube and secondary solid-state monitors

Cathode-Ray Tube →

- Magnetic Coil → Heating filament (it generates heat).
- Cathode → Generates -ve electrons.
- Control Grid → It controls the voltage
- Accelerating anode → It provides +ve electrons
It monitors the path of electrons to the destination (Phosphorus screen).

In one second, a pixel gets refreshed as 60 times.



- Horizontal Deflecⁿ. Plates → It shows the left to right pixels.
- Vertical Deflection. Plates → It shows the ~~top~~ up to down pixels.

#

1) Heating filament → Heat is supplied to the cathode by directing a current through a coil of wire called filament. Here Magnetic field is generated. This causes electrons to be boiled of fully negative charged electrons are then accelerated.

2) Control Grid → Intensity of e⁻ beam is controlled by setting voltage ~~[illegible]~~ level. It controls whole ~~pole~~ architecture.

#__________

Scanned by CamScanner

charges . (black) B/W

3) Accelerating Anode → Positive charges give to focus and give a way to negative e^- to go on the phosphorus screen.

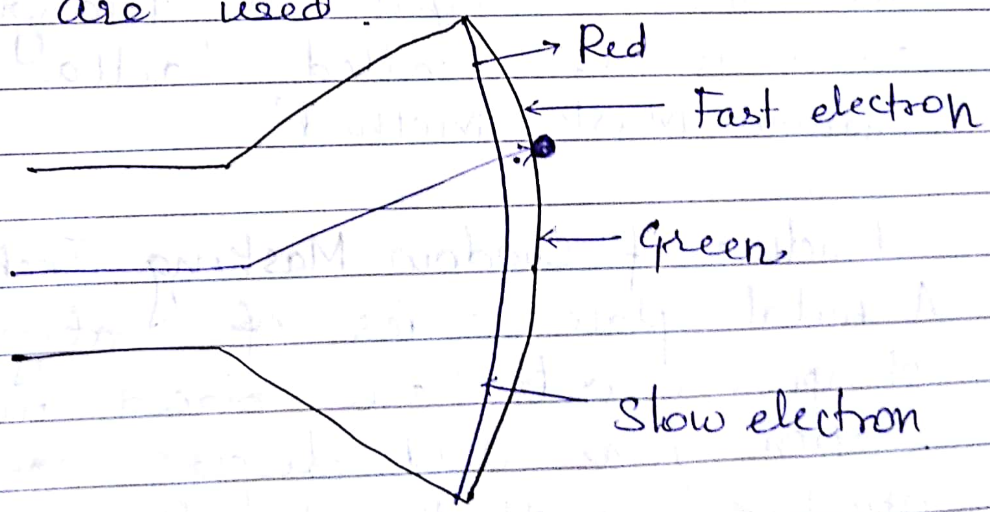
Refresh C.R.T. - One way to keep phosphorus glowing is to ~~der~~ redraw the picture repeatedly by quickly directing the e^- beam back over the same point. This type of display is called refresh CRT.

> One bit per second is in the form of 0 and 1 in Black & White CRT.

> In colour CRT, we have two types of mechanism - These are techniques

1) Beam Penetration → (Random Scale Display)
2) Shadow Masking (Delta Masking)
↳ (Raster Scale Display)

1) Beam Penetration -
For high resolution of TV, 24 bits colour patters are used.

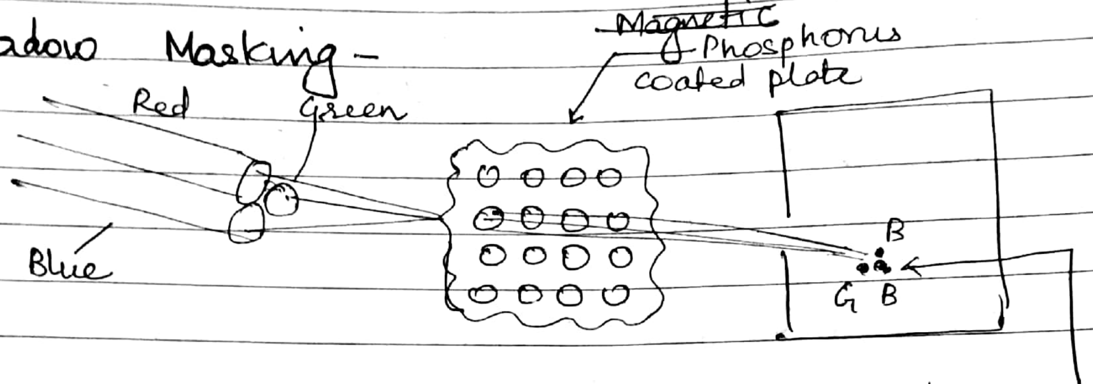


1) Beam Penetration technique or CRT are mostly used in Random-Scale Monitors.

Disadvantage → ~~Ther~~ We can make only some colors not all.

2) It will draw three dots in a pixel. 3) Two colors red & green coated to phosphorus screen.
4) Display colour depends upon how deep electron beam penetrate to phosphorus layer.

2) Shadow Masking -



It is also called delta masking.

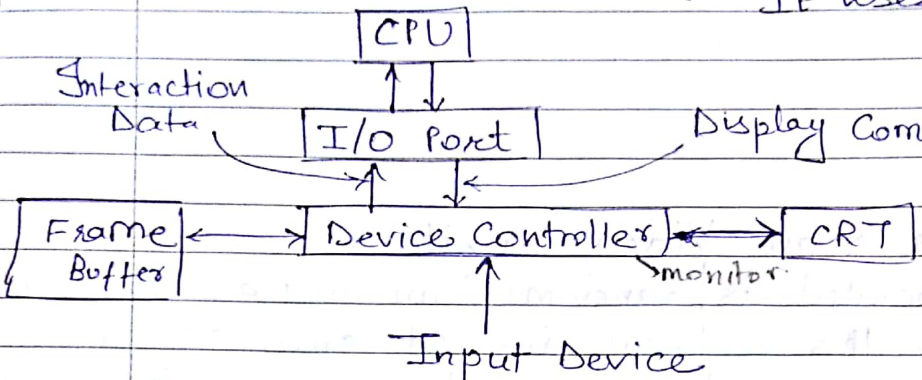
In CRT colour Monitors, we are using three E-guns to produce the colours. Single pixel is coated with red, green & blue dots. In shadow masking technique, we can generate wider range of colours. Three dots are arranged in triangular shape so it is also called 'delta ~~Masking~~ Shadow Mask Method'.

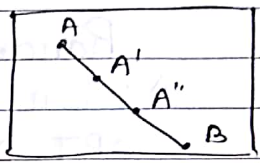
Structure of Shadow Masking Technique -
A metal plate series of aligned holes with phosphorus dots are placed in between electron gun and display screen. The objective is that when e⁻-beam penetrate the particular pixel, its near by pixel should not illuminate.

TYPES OF VIDEO MONITORS

① Random Scan display (Video Monitors)

↳ It uses Beam penetration CRT.

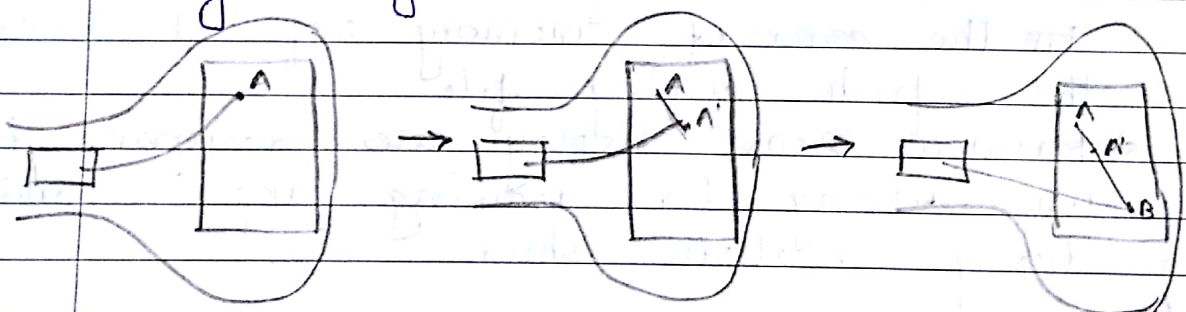




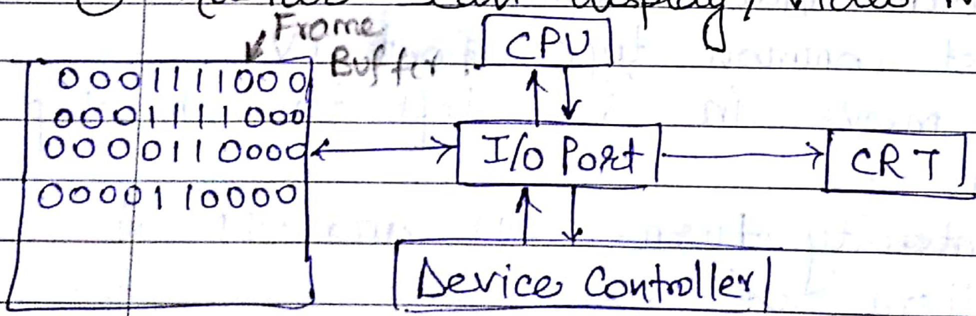
Video Monitor

- Device controller takes the i/p from i/p device and save it in frame buffer.
- Frame Buffer specifies what to draw, i.e. which pixel (or at what location) should glow.

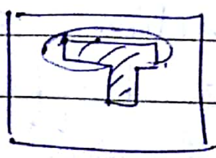
Limitation - Random scan display can draw only straight lines.



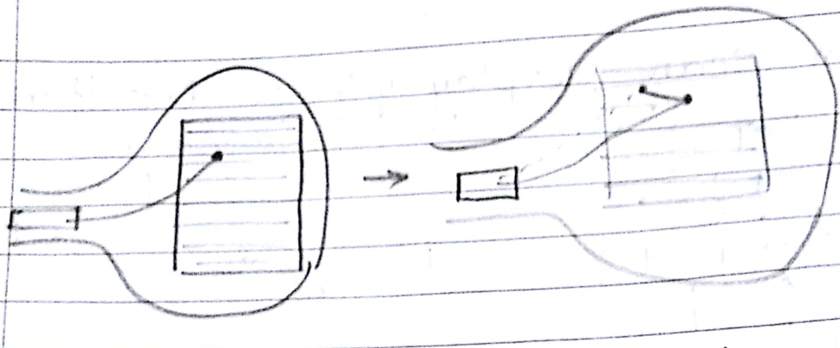
② Rastar Scan display / Video Monitor



It can draw any shape & curve.



1 → White/ON.
0 → Black/OFF



Random Scan Video Monitor-

1) When operated as a random Scan display Unit, a CRT has the e⁻ beam directed only to the parts of screen where a picture is to be drawn.
2) Random Scan monitors draw a picture 1 line at a time.
3) It is also referred as Vector Display, Stroke Writing and calligraphic display.
4) Refresh rate on random scan display detects the no. of Lines to be displayed.
5) Picture definition is now stored as a set of lines drawing commands in the area ~~In the area~~ of memory referred to as the refresh display file.
6) Random scan display are designed for line ~~display for~~ drawing app. & cannot display realistic shades.

1/2/18

Rastar Scan Video Monitor

1) The most common type of ont TV.
2) Electrons move in row left to right, top to bottom.
3) Beam Intensity turn ON and OFF to create pattern of.
4) Picture definition stored in a M/y area called frame buffer or refresh buffer

#

BITMAP - B/W
PIXMAP - COLOUR

Date

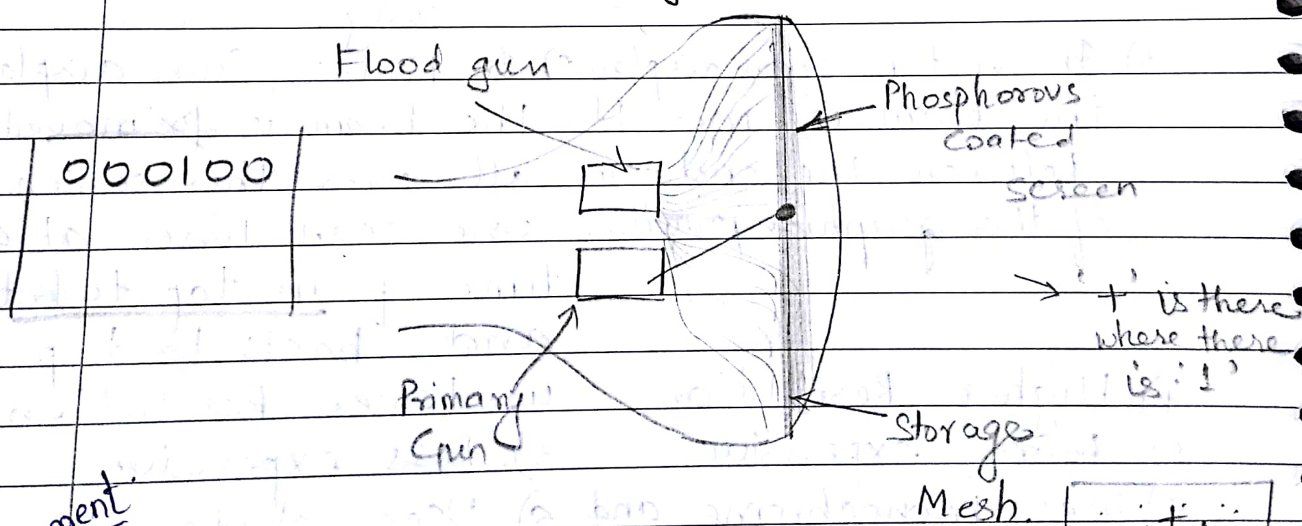
5) Stored Intensity value retreived from frame buffer to be painted; for black & white only 1 bit is used, zero for white when the beam intensity is ~~of~~ OFF and vice-versa. One for black when the intensity is ON.

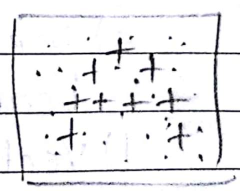
6) Additional bits are needed when coloured monitors are used, upto 24 bits per pixel in high quality system, depending upon resolution of the system.

(Vector) Random Scan Display	Rastar - Scan Display
1) Random Scan displays only draws lines and ~~control~~ character.	1) Rastar has ability to display areas filled with solid colours or patterns.
2) Don't use interlacing	2) It uses interla-cing.
3) In vector-scan display the beam is moved between the end points of the graphics primitives	3) In Rastar Scan display, the beam is ~~p~~ moved all over the screen, one scan line at a time from top to bottom and back to top
4) Higher Resolution	4) Lower Resolution
5) More expensive	5) Less expensive
6) Uses monochrome and beam penetration	6) Uses shadow masking
7) Vector display draws a continuous and straight line.	7) Rastar display can display mathematically smooth lines, polygons #

	and boundary of curve by primitives only by approximating them with pixel on the raster grid at point.
8) Editing is easy.	8) Editing is difficult
9) Refresh rate depends directly on picture complexity	9) Refresh rate ~~depends~~ is independent of picture complexity.
10) Scan conversion is not required.	10) Graphics primitives are specified in terms of their end points and must be scan converted into their corresponding pixels in the frame buffer.

Direct View Storage Tube (DVST).





Assignment

Flat Panel Display -

#

Scan Converting Techniques - It analyzes the difference b/w the pixels.

> Digital Differential Analyzer (DDA)

Algorithm

1) Read the lines 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~~ if ($\Delta X \geq \Delta Y$) then ~~length = Δ~~

length = ΔX

else

length = ΔY

4) Select the raster unit

$$\Delta X = \frac{(X_2 - X_1)}{\text{Length}}$$

$$\Delta Y = \frac{(Y_2 - Y_1)}{\text{length}}$$

5) $X = X_1 + 0.5 * \text{sign}(\Delta X)$

$Y = Y_1 + 0.5 * \text{sign}(\Delta Y)$

6) Now Plot the point

```
i = 1
while ( i ≤ length)
{
    plot (integer (x), integer (y));

    X = X + ΔY
    Y = Y + ΔY
    i = i+1
}
```

Date
#

$M = \frac{y}{x}$.

$\frac{1}{M}$ $\frac{y}{m}$

Date 5 02 18

Direct View Storage

An alternative method for maintaining a screen image is to store the picture info. inside this CRT instead of refreshing the screen. A DVST stores the picture information as charged distribution just behind the phosphorus coated screen. Two e⁻ guns are used in a DVST, one, the primary gun is used to store the picture pattern, second, the flood gun is used to maintain picture display.

Advantages -

Because no refreshing is needed, very complex picture can be displayed at very high resolution without flicker.

Disadv.

Disadvantages of DVST system are that they ordinary do not display colour and ~~the~~ th selected parts of image cannot raise to eliminate of a picture section. The entire screen must be erased and modified picture redrawn.

> Flat Panel Display -

The term flat panel display refers to a class of video devices that have reduced volume, weight and power requirement compared to a CRT. A significant feature of FPD is that they are thinner than CRT and ~~we~~ we can hang them on wall or ~~we~~ wear

#

$M = \frac{y}{x}$

Date 5 02 18

Direct View Storage

An alternative method for maintaining a screen image is to store the picture info. inside this CRT instead of refreshing the screen. A DVST stores the picture information as charged distribution just behind the phosphorus coated screen. Two e⁻ guns are used in a DVST, one, the primary gun is used to store the picture pattern, second, the flood gun is used to maintain picture display.

Advantages –

Because no refreshing is needed, very complex picture can be displayed at very high resolution without flicker.

Disadv. Disadvantages of DVST system are that they ordinary do not display colour and ~~the~~ th selected parts of image cannot raise to eliminate of a picture section. The entire screen must be erased and modified picture redrawn.

› Flat Panel Display –

The term flat panel display refers to a class of video devices that have reduced volume, weight and power requirement compared to a CRT. A significant feature of FPD is that they are thinner than CRT and we can hang them on wall or wear

Date

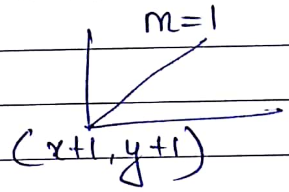
them on our ~~so~~ wrist.

Types of FPD:

① Emissive display - The emissive type of display system displays that convert electrical energy into light. Plasma Panel, Thin film, electroluminescent display ~~are~~ and light emitting diodes are the examples of emissive displays.

② Non-Emissive display - They use optical effects to convert sunlight or another light from some other source into graphics pattern. The most important example of non-emissive display is liquid crystal display (LCDs).

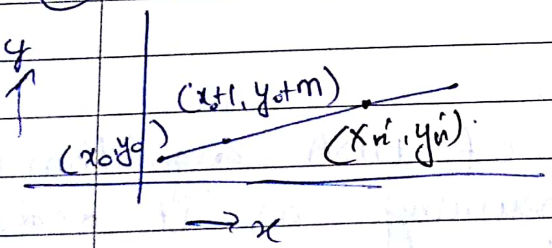
> Explaination of DDA algorithm -



① $m = 1$

New point → $(x+1, y+1)$

② $m < 1$



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

(m<1)

Next point

$x_n\ (X_{new}) = X_0 + 1$

$Y_n\ (Y_{new}) = Y_0 + m$

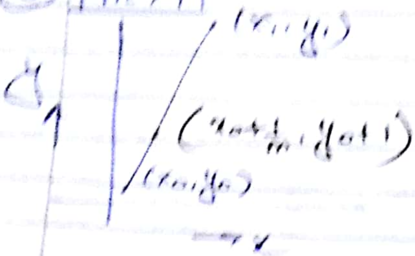
Plot $(x_n, \text{round } Y_n)$

(Next pixel of x_n) $x_n' = x_n + 1$ $\quad y_n' = y_n + m$

Repeat the same procedure.

Set pixel → (x_0, y_0) and (x_1, y_1)
Put pixel → we give all the next pixels.

③ $\boxed{m>1}$ Starting Point → x_0, y_0



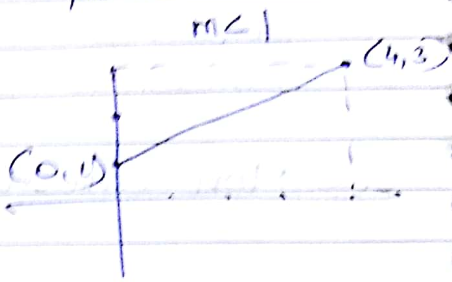
$x_n = x_{new} = x_0 + \frac{1}{m}$

$y_n = y_{new} = y_0 + 1$

$x_n' = x_n + \frac{1}{m}$, $y_n' = y_n + 1$

Example of DDA –

① $x_0 = 0$, $x_1 = 4$
$y_0 = 1$, $y_1 = 3$



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

$= \frac{3-1}{4-0} = \frac{2}{4} = 0.5 < 1$ i.e. $m<1$

X_n	Y_n	=	X_n'	Y_n'
1	1.5	=	1	2
1+1 = 2	1.5+0.5=2	=	2	2
2+1 = 3	2+0.5=2.5	=	3	3
3+1 = 4	2.5+0.5=3	=	4	3
4+1				

/2/18

Bresenham's line Algo.

One serious drawback of DDA algorithm is that it is very time consuming as it deals with a rounding off operation and floating point airthmetic. The algorithm developed by Bresenham is more accurate and efficient compared to DDA algo.

Date

because it avoids round function and scan converts lines using only incrementing integer calculation.

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 ~~$2\Delta Y -$~~ $P_0 = 2\Delta Y - \Delta X$ and obtain the starting values for the decision parameters.
4) At each X_K along the line starting at $K=0$ perform the following steps.

If $P_K < 0$ the next point to plot is (X_K+1, Y_K) and

$$P_{K+1} = P_K + 2\Delta Y$$

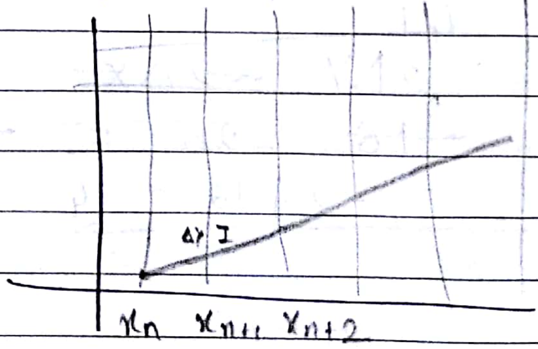
otherwise,

the next point to plot is (X_K+1, Y_K+1) and

$$P_{K+1} = P_K + 2\Delta Y - 2\Delta X$$

5) Repeat step 4 for Δx times.

① $m<1$



$Y = mX + c$

$$\boxed{m = \frac{\Delta Y}{\Delta X}}$$

#

Algorithm is used rather than the functions because with algorithm we get the perfect line.

eg: (20, 10) ; (30, 18)

$\Delta X = 30 - 20 = 10$; $\Delta Y = 18 - 10 = 8$

$2\Delta Y = 16$

$P_0 = 2\Delta Y - \Delta X$

$= 16 - 10 = \boxed{6}$

P_K	$X_K + 1$	$Y_K + 1$
6 (P_0)	20 (X_0)	10 (Y_0)
2	21	10+1 = 11
-2	22	11
14	23	11+1 = 12
10	24	12+1 = 13
6	25	14
2	26	15
-2	27	15
14	28	16
10	29	17
6	30	18

since, $P_K > 0$

$P_{K+1} = P_K + 2\Delta Y - 2\Delta X$

$= 6 + 16 - 20$

$= 2$

Now $P_K < 0$

$P_{K+1} = P_K + 2\Delta Y$ ~~$- 2\Delta X$~~

~~$2 + 16 - 20 = -2$~~

$= -2 + 16 = 14$

When $m > 1 \rightarrow$ then the value of y will always increase.

#

Date 8 2 18

Differenciate between Bresenham's line algo. and DDA.

Bresenham's L.A.	DDA
1) Bresenham algorithm uses fixed points i.e. integer, airthmetic.	1) DDA uses floating point, i.e. real airthmetic
2) Bresenham algorithm uses only subtraction and addition in its operators.	2) DDA uses multiplica-tion and division on its operators.
3) Bresenham is faster than DDA.	3) DDA is rather slower than Bresenham algo.
4) Bresenham algo. is more efficient and much accurate than DDA algo.	4) DDA algo. is not as accurate & efficient as bresenham algo.
5) Bresenham algo. is less expensive than DDA.	
6) Bresenham algo. does not round off but takes the incremented values in its operation.	6) DDA algorithm round off the co-ordinates to integer i.e. nearest to the line, rounding off of the pixel position obtained by multiplican. or division causes &

continue

99
× 1.41

Date

acc

accumulation of error in the proceeding picture pixels.

Mid-Point Circle Drawing Algorithm

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 parameter as

$$P_0 = \frac{5}{4} - r \quad \left[\frac{5}{4} - r \approx 1 - r\right]$$

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

$$P_{K+1} = P_K + 2X_K + 2 \text{ and}$$

$$P_{K+1} = P_K + 2X_{K+1} + 1 - 2Y_{K+1}$$

$$2Y_{K+1} = 2Y_K - 2$$

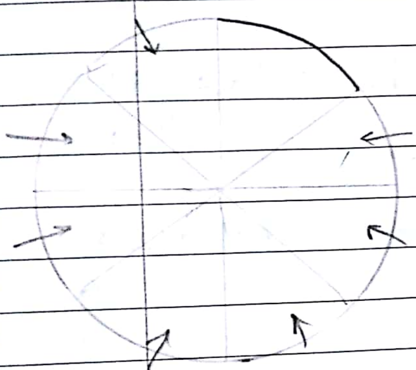
4. Determine symmetry points in the other seven octants.
5. Move each calculated pixel position (x,y) on

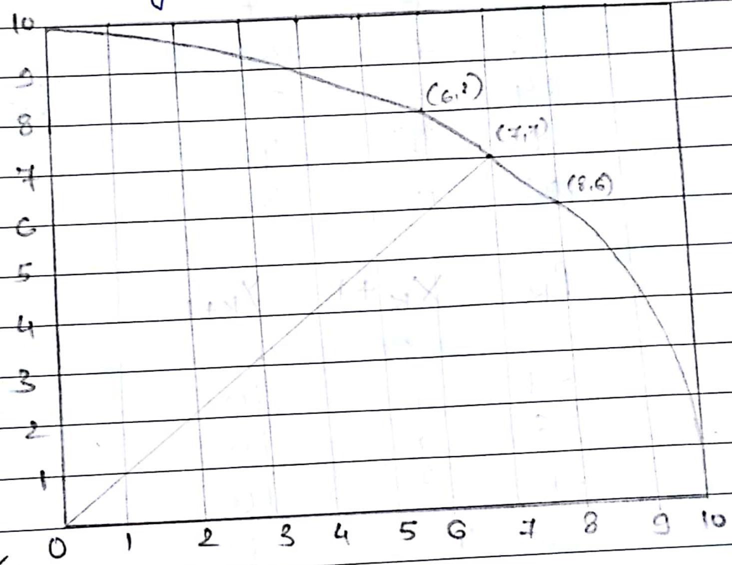
#

to the circular path centered on (x_c, y_c) and plot the coordinate values

$$X = X + X_c$$
$$Y = Y + Y_c$$

6. Repeat step 3 through 5 until $X \geq Y$.





Decision Maker/Parameter.

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

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

Q. Numerical

(0, r)

(0, 10)

Initial d.p.

$P_0 = 5/4 - r \approx 1 - r$

$= 1 - 10 = -9$

$P_0 < 0$

#

{

$x = x + xn;$

$y = y + yn;$

pixel ((int) x, (int) y, RED));

Teacher's Signature :

	P_k	X_k+1	Y_{k+1}
(P_0)	-9	0	10
	-9	1	10
	-6	2	10
	-1	3	10
	6	4	9
	16	5	8
	30	6	7
	46	7	6
	64	8	5
	84	9	4
	106	10	3

For $P_k < 0$

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

For $P_k > 0$

~~$P_{k+1} = P_k + 2X_{k+1} + 2$~~

$P_{k+1} = P_k + 2X_{k+1} + 1 - 2Y_{k+1}$

$8 + 12 + 1$
$- 16$

P_k	X_k+1	Y_{k+1}
	0	10
-9	1	10
-6	2	10
-1	3	10
6	4	9
-3	5	9
8	6	8
4	7	7

If centre was not on (0,0) then add the coordinates of centre to x & y.
Like (4,3) is centre → +4 to x_{k+1} and +3 to all y_{k+1} after solving.

Bresenham's circle drawing algorithm –

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 than until $(x < y)$ we have to

#

always initially put $x=0,\ y=r$.

Date

perform following steps.

if $(d<0)$ then

$d = d + 4x + 6$

$x = x+1$

$y = y$

else

$d = d + 4(x-y) + 10$

$y = y-1$

$x = x+1$

5. Plot (x,y)

6. Determine the symmetry points in other octants also.

7. Stop.

Midpoint ellipse 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 parameter in Region 1 as,

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

initialize dx and dy as,

$$dx = 2r_y^2 x$$

$$dy = 2r_x^2 y$$

#

Date

5. do

{ plot (x, y)

if ($P_1 < 0$)

{

$x = x+1$

$y = y$

$dx = dx + 2r_y^2$

$P_1 = P_1 + dx + r_y^2$

}

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 (y<0)

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

$$P2 = r_y^2\left(x+\frac{1}{2}\right)^2 + r_x^2(y-1)^2 - r_x^2 r_y^2$$

7\. do

{ plot (x, y)

if (P2 > 0)

{ $x = x$

$y = y-1$

$dy = dy - 2r_x^2$

$P_2 = P_2 - dy + r_x^2$

}

Date

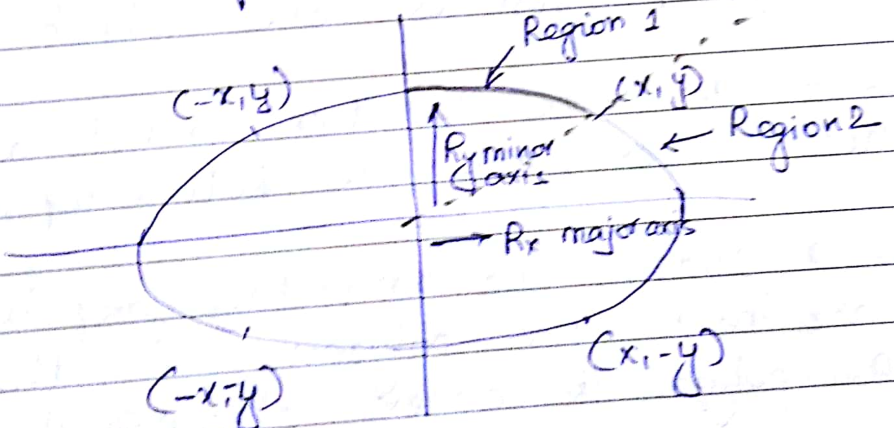
else
{
$x = x+1$
$y = y-1$
$d_x = d_x + 2r_y^2$
$d_y = d_y - 2r_x^2$

$P_2 = P_2 + d_x - d_y + r_x^2$
}
}
while $(y>0)$

8. Determine symmetrical points in other three quadrants
9. STOP.

13/2/18

Elipse Generation –
Basic equation → $\left(\frac{x}{a}\right)^2 + \left(\frac{y}{b}\right)^2 = 1$



Date

Polygon Filling Algorithm

A polygon may be represented as a no. of line segments connected end to end to form a closed figure. It may be represented as the points where the sides of polygons are connected. The line segment which make up the polygon boundary are called sides or edges. The end points of the size are called polygon vertices. The simplest polygon is the triangle, having 3 sides and 3 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 and all points on the line segment connecting them are also ~~on~~ inside the polygon.

> A concave polygon is one which is not convex.

Thus to determine whether or not a polygon is not convex, ask the following questions.

1) Does a straight line connecting any two points that are inside the polygon?, if the answer is NO, polygon is convex otherwise it is concave

Anticlockwise – Positive orientation
Clockwise – Negative orientation

#

Date

Polygon Filling Algorithm

A polygon may be represented as a no. of line segments connected end to form a closed figure. It may be represented as the points where the sides of polygons are connected. The line segment which make up the polygon boundary are called sides or edges. The end points of the size are called polygon vertices. The simplest polygon is the triangle, having 3 sides and 3 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 and all points on the line segment connecting them are also inside ~~on~~ the polygon.

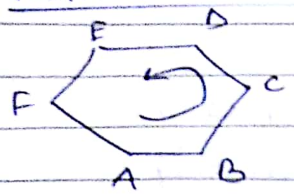
> A concave polygon is one which is not convex.

Thus to determine whether or not a polygon is not convex, ask the following questions.

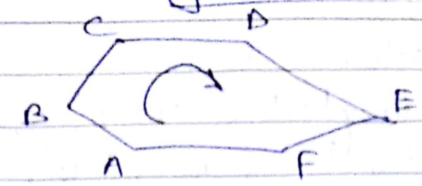
1) Does a straight line connecting any two points that are inside the polygon?, if the answer is NO, polygon is convex otherwise it is concave.

Anticlockwise – Positive orientation
Clockwise – Negative orientation

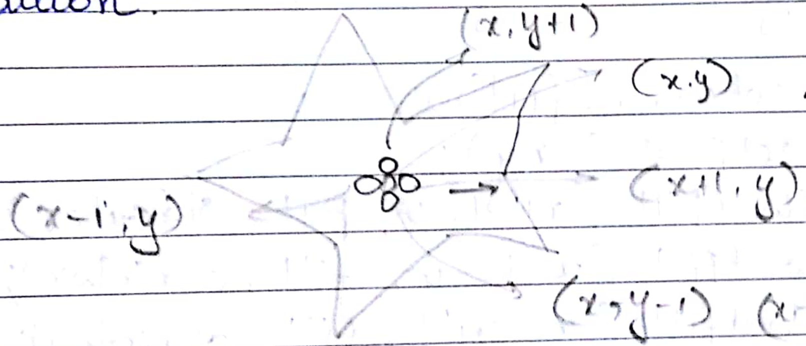
Positive Orientation



Negative Orientation

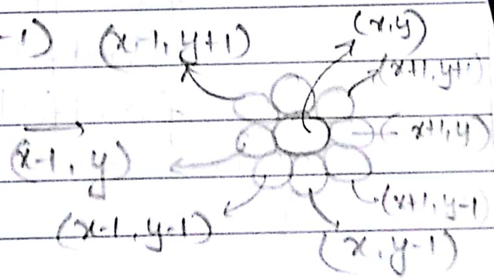


Polygon Filling - Filling is the process of colouring in a fixed area or region there are two basic approaches to area filling in Rastar System. › One way to fill an area is scan line fill algo.
› Another method for area filling is to start from a given interior position and point outward from this point until we encounter the specified boundary condition.



← 4-connected Boundary fill

8-connected Boundary fill



This also an algo. to fill colour inside any polygon

Boundary fill Algorithm -

The colour which is already filled (if any) — Colour of boundary

void fill(int x, int y, int fill, boundary) int boundary)
{
int current = get_pixel (x,y);
if ((current != boundary) & (current != fill))
{
setcolor (fill);

} x = ...
y = y+yn;
outpixel((int)x, (int)y, RED));

Teacher's Signature :

Date

```
setcolor (x, y);
boundary Fill (x+1, y, fill, boundary)
boundry Fill (x-1, y, fill, boundry)
boundry Fill (x, y+1, fill, boundry)
boundry Fill (x, y-1, fill, boundry)
}
}
```

Flood fill Algo.

oldcolor contains all the colors that is there on the polygon

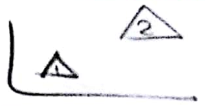
```
void floodfill (int x, int y, int fill, int color)
{
  if (getpixel (x,y) == old color)
  {
    setColor (fill);
    setPixel (x,y);
    floodFill (x+1, y, Fill, oldcolor);
    floodFill (x-1, y, fill, oldColor);
    floodfill (x, y+1, fill, oldcolor);
    floodfill (x, y-1, fill, oldcolor);
  }
}
```

(x,y)

2-D Transformation -

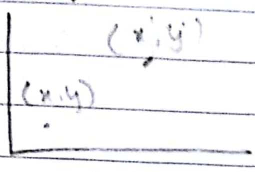
2-D transformation changes in the orientation, size and shape are accomplished with the geometric transformation that alter the co-ordinates description of object.

The basic geometric transformation are



Date

translation, rotation and scaling.



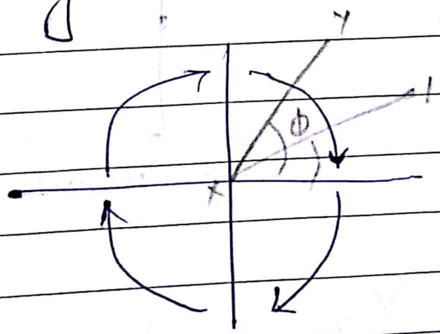
>> Translation - Translation consist of a shift of the object parallel to itself in any direction in the X-Y plane, any such shift can be accomplished by a shift in x direction + a shift in y direction.

If the amount of x-shift is called t_x and the amount of y-shift is called t_y. The translation of the point (x,y) into the point (x',y') is expressed by the formula -

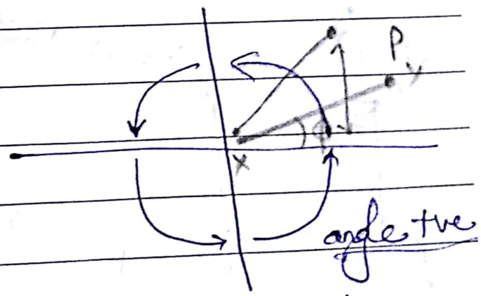
$$x' = x + t_x \qquad y' = y + t_y \rightarrow \begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix}$$

→ To translate any object with the points (x,y), we just translate each point individually.

>> Rotation -



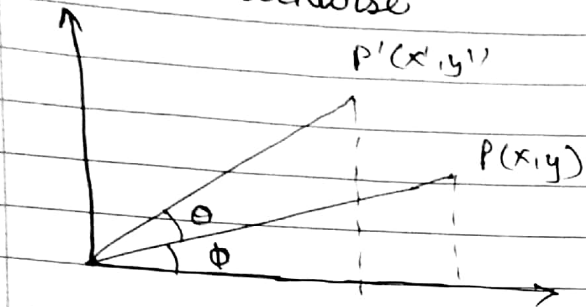
Angle is -ve
Clockwise



Counter clockwise

#

Counter - clockwise



$$x = r\cos\phi$$
$$y = r\sin\phi$$

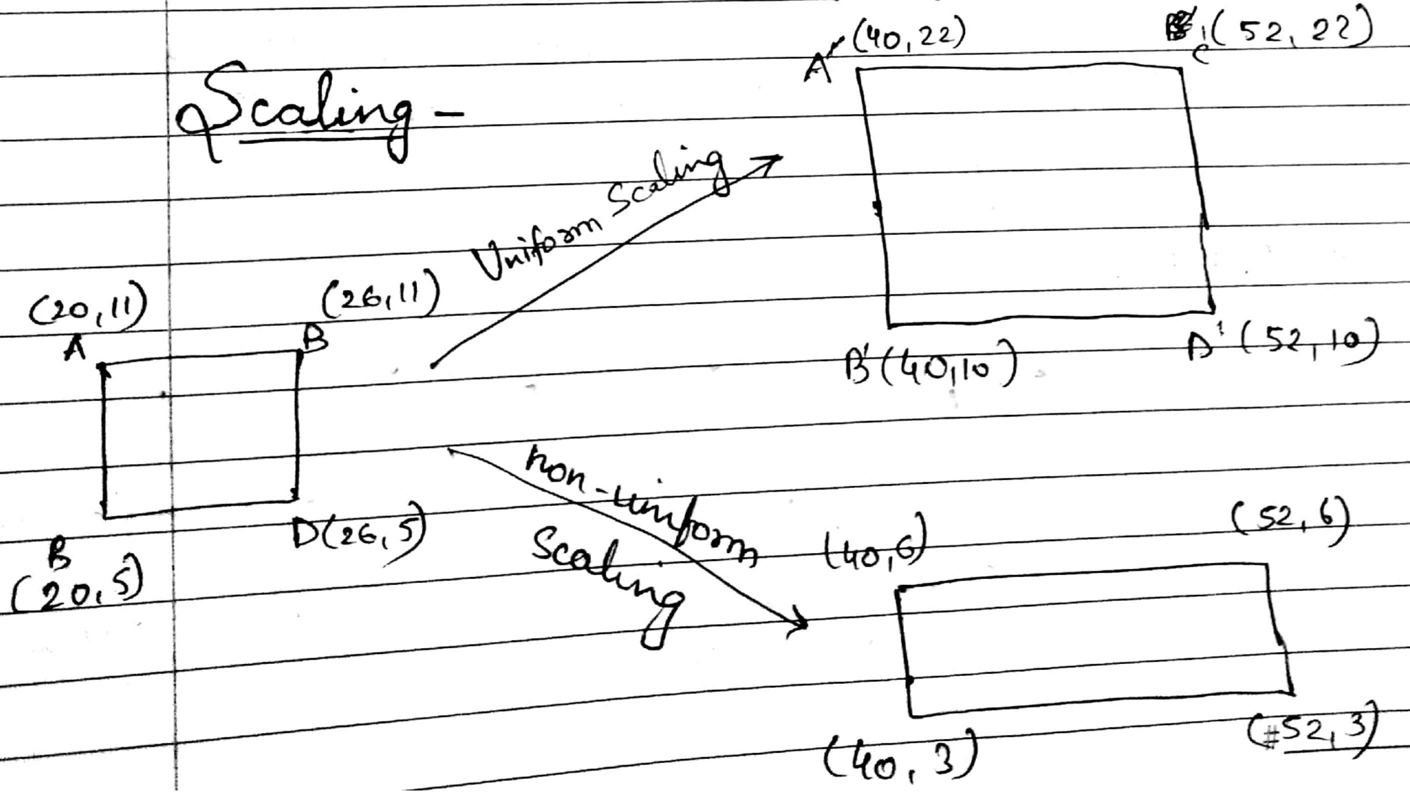
$$x' = r\cos(\phi+\theta)$$
$$= r\cos\phi\cos\theta - r\sin\theta\sin\phi$$
$$\boxed{x' = x\cos\theta - y\sin\theta}$$

$$y' = r\sin(\theta+\phi)$$
$$= r\cos\phi\sin\theta + r\sin\phi\cos\theta$$
$$\boxed{y' = x\sin\theta + y\cos\theta}$$

anticlockwise $R = \begin{bmatrix} \cos\theta & -\sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix}$

$$[x'] = [T][X]$$

Scaling -



Date

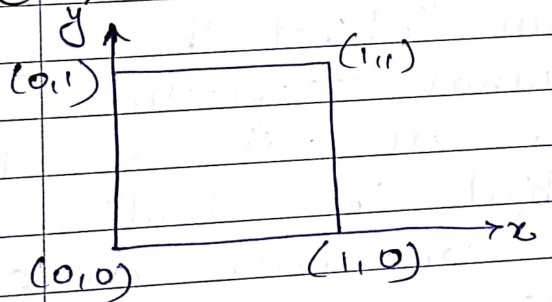
A scaling transformation alters the size of an object. This operation can be carried out on polygons by multiplying the co-ordinates value (x,y) of each vertex by scaling factor S_x and S_y,
To produce the transform co-ordinates (x', y').

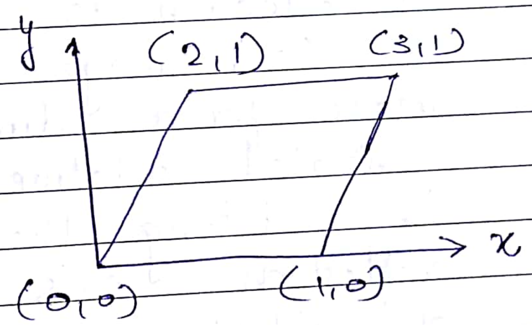
$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} S_x & 0 \\ 0 & S_y \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

$$[X'] = [T][X]$$
$$[Y'] = [T][Y]$$

Shearing -

① X- Shear



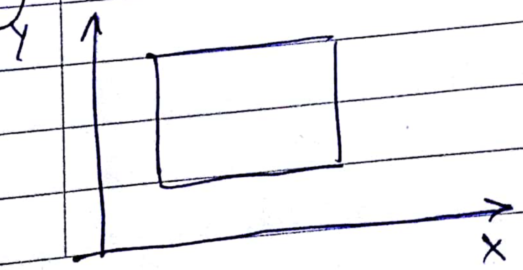


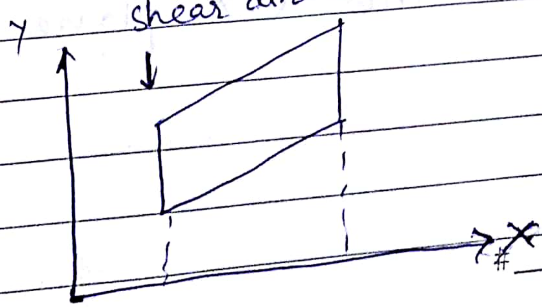
$$[X' . Y] = [X, Y] * \begin{bmatrix} 1 & 0 \\ b & 1 \end{bmatrix}$$

a and b are

$$[X', Y] = [X + Yb, Y]$$

② Y- Shear





x = ... u+yn; ... (int)y, RED));

Date

$$[x, y'] = [x, y] * \begin{bmatrix} 1 & a \\ 0 & 1 \end{bmatrix}$$
$$= [x, xa + y]$$

The transformation that distorts the shape of an object such that the transformed shape appears as if the object was composed of internal layes that had been caused to slide ~~layer~~ over each other is called shear.

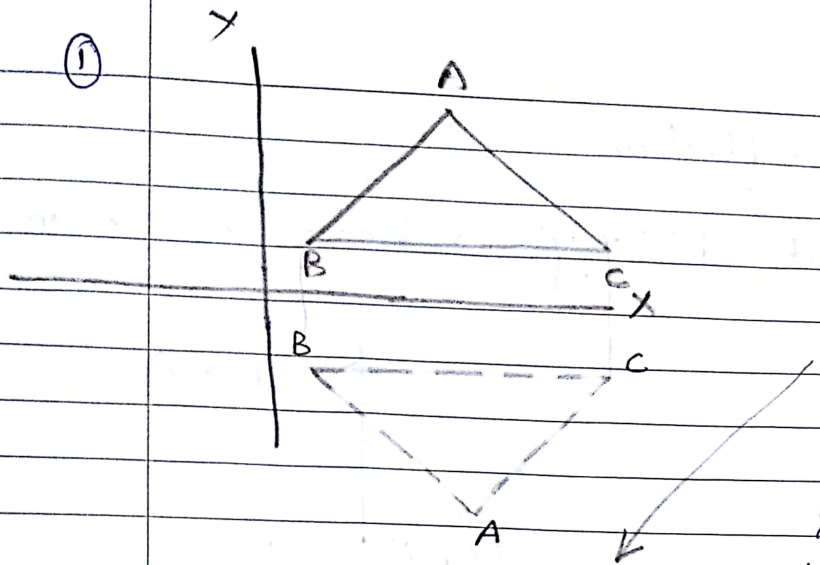
Two common shearing transformation are those that shift x values and those that shift y values. Shearing can be x-directional or y-directional.

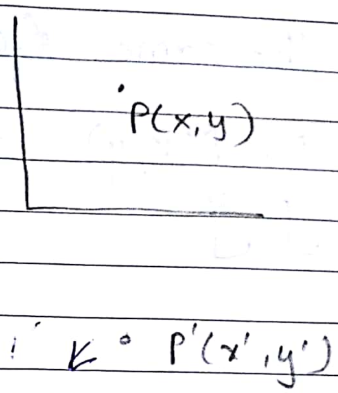
Reflection -

A reflection is a transformation that produces a mirror image of an object. The mirror image for a dimensional reflection is generated relative to an axis of reflection by rotating the object 180° about a reflection axis. We can choose an axis of reflection in the x-y plane or perpendicular to the x-y plane. when the reflection axis is a line in x-y plane, the rotation path about this axis is in a plane perpendicular to the x-y plane.

Date

(1)





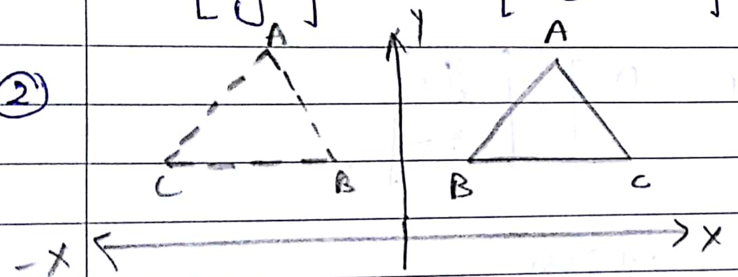
$x' = x$

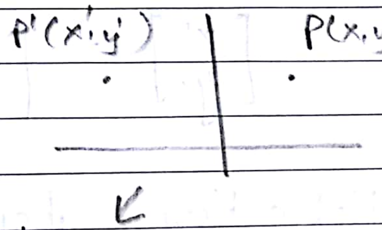
$y' = -y$

⇒ x-axis

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

(2)





$x' = -x$ and $y' = y$.

(i) Reflection about x-axis ($y=0$)

The basic principle define reflection transformation are -

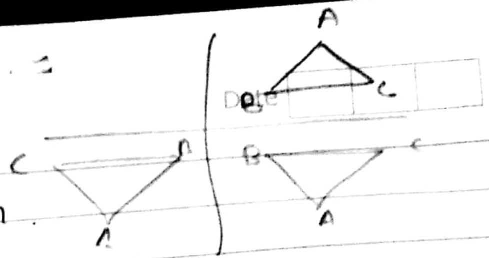
a) The image of an object is formed on the sides opposite to where the object is lying mirror line.

b) The perpendicular distance of the object from the mirror line is same to the distance of the reflected image.

(ii) Reflection about Y-axis ($x=0$)

A reflection about y-axis ($x=0$) flip x co-ordinates while y-coordinates remains

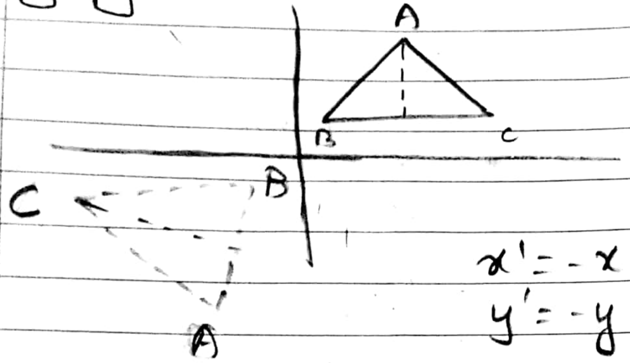
#

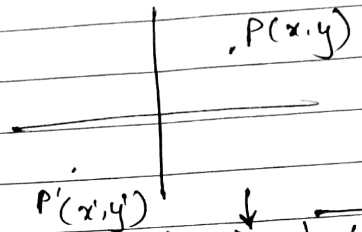


the same. ~~f~~for reflection.

For $P(x, y)$ and $P'(x', y')$ where $x' = -x$ and $y' = y$.

(3)





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

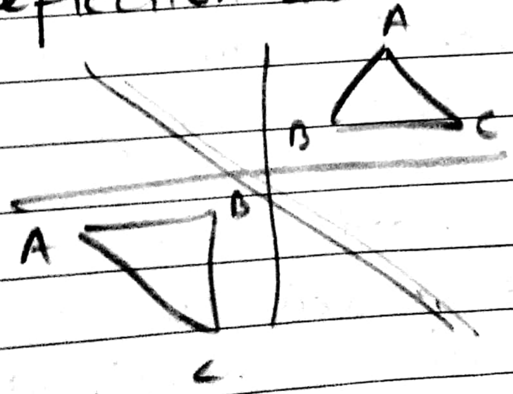
$\cancel{x' = -x}$ $\cancel{y' = -y}$ $\cancel{x' = -y}$ $\cancel{y' = -x}$

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

(iii) Reflection about origin

In this case, actually the mirror line has the axis perpendicular to the X-Y plane in passing through the origin. After reflection, both the x and y co-ordinates of the object point is flipped that x' becomes $-x$ and y' becomes $-y$.

(4) Reflection about $x = -y'$



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

#