Computer Graphics
Unit-1

Computer Graphics : The computer graphics is one of the most effective & commonly used way to communicate the process information to the user. It display the information in the form of graphics object such as pictures, charts, graphs & diagrams.

Computer is logical device which manipulate arithmetic & logic task. Graphics are visual presentation on some surface or pictorial representation & manipulation of data.

Types of Computer Graphics : There are mainly two categories of computer graphics :

1. Interactive Computer Graphics
2. Non-Interactive Computer Graphics

1. Interactive Computer Graphics : In interactive computer graphics both computer & user communicate with each other.
For eg : the video game controller of a particular game. This helps user to signal his request to the computer.

2. Non-Interactive Computer Graphics : It is also sometimes as passive computer graphics. Here user is not having any control over the image.

Uses Of Computer Graphics

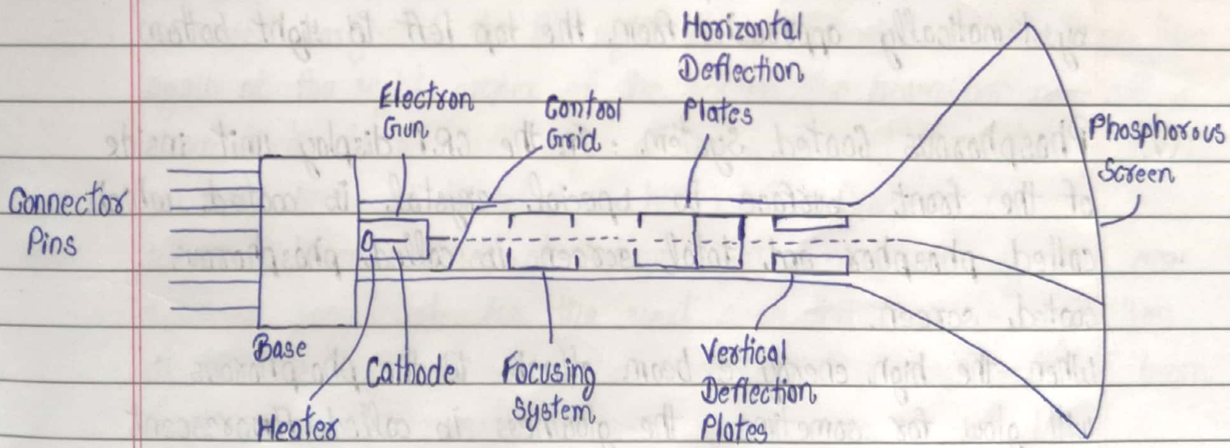
- Types of object
  - 2D
  - 3D
- Kind of Picture
  - Symbolic
  - Realistic
- Type of interaction
  - controller
  - Non-controller
- Pictorial Representation
  - line drawing
  - Black & white image
  - colour image
  - etc.

Pixel : It is a pictorial unit of graphics.
In computer graphics, pictures or graphic objects are presented as a collection of discrete pictures elements called pixels. The pixel is the smallest addressible screen element. It is the smallest piece of the display screen which we can control.

Resolution : It is defined as the number of pixels present in the unit of area.

Aspect Ratio : It is the ratio of number of pixels present horizontally and number of pixels present vertically.

Input Devices : Keyboard, Scanner, Joystick
Output Devices : CRT, LCD, Printer, Speaker
Display device are also known as output devices. The most commonly used output device in a graphic system is a video monitor. The operation of most video monitor is based on the standard (CRT design). Let us see the basic of CRT :



Basic three Accelerating Mode :

1. Pre-Accelerating
2. Focusing
3. Accelerating Anode

Provides the Path

It is used for video monitor. One electron beam will pass from left to right for which the pictorial unit will be displayed in the monitor. The basic function of individual component can be represented as follows:

(i) Electron Gun : A cathode surrounded by heater coil which is -ve charged. When the heater will be heat, the e^- will produce at the end of the gun which will be attacked by positively charged phosphor i.e. coated in the monitor. The e^- will flow in a sequential manner which looks like an e^- beam.

(ii) Control Electrode : It will control the flow of electron. It is connected with an amplifier which will be turned off and ON based upon the processing in the next level.

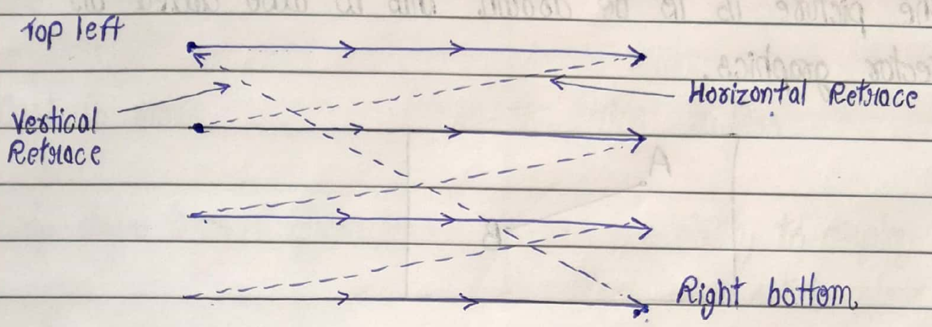
(iii) Focusing Electrode : It will help to create a clear picture by passing the electron beam in a narrow form.

(iv) Deflection Amplifier : It will bend the electron beam from which the electron beam will pass accross the monitor in a systematically approach from the top left to right bottom.

(v) Phosphorous Coated System : In the CRT display unit inside of the front surface is special crystal is coated which called phosphor and total screen is called phosphorous coated screen.
When the high energy e^- beam attacks to the phosphorous will glow for sometime, the glowness is called Fluorescent screen. The continuity of glow after sometime is called phosphorence. The duration of the phosphorence is called persistence.

* There are two techniques used for producing images on the CRT screen:
1. Raster Scan Display.
2. Random / Vector Scan Display.

1. Raster Scan Display: In this the e^- beam sweep across the screen one row at a time from the top left to right bottom. When it is reaching to the right bottom again it will be redirect to left top position & the same process will be continue for the raster scan we are using a buffer called frame buffer which store the actual picture definition.

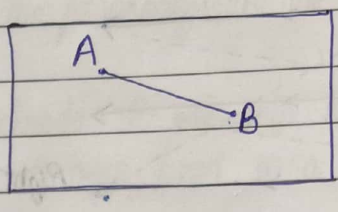


In raster scan a fixed path is followed by the e^- beam as shown in fig. At top left corner of the screen the e^- beam starts & moves horizontal to the right. It is referred as one scan line again at the right corner of the screen the beam becomes off & move back to the left edge of the screen at the starting point on next line. It is called as horizontal retrace.
When a beam reaches at bottom right corner we say that one scan is completed. For the next scan the beam is reposition at the top left corner of the screen. This movement of beam from bottom right corner to top left corner is called as vertical retrace.

Each point on the screen is called pixel. For the b/w system each pixel can be represented by one bit.
eg: 1 → white   is called bit map.
0 → black

If one pixel require more than one bit then it is called pixmap. which is use for the gray level colour.

2. Random / Vector Scan Display : In this CRT has a e⁻ beam which will be directed to the part of the screen where the picture can be drawn. In the random scan display generally use for the line drawing algorithm. Random scan display will be the high resolution than the raster scan display. "Display device with random scan operate by directing the e⁻ beam to only those parts of the screen where the picture is to be drawn this is also called as vector graphics.



If we want a line connecting point A with point B on the graphic display, the e⁻ beam to go directly from point A to B.

Advantages :
* High Resolution.
* Simple to animate
* Need less memory

Disadvantage :
* Less colour capability.
* Difficult for complex images.

Difference Between Raster scan Display and Random / Vector scan Display :

Random/Vector Scan Display	Raster Scan Display
(i) It operates by directly the e^- beam to only those part of the screen where the picture is to be drawn.	(i) The e^- beam starts at the top left corner of the screen and move to right horizontally.
(ii) Pen, Ploatters, DVST device are used.	(ii) CRT are used.
(iii) Cost is more.	(iii) Cost is less.
(iv) Only draw lines & characters.	(iv) It has ability to display area's filled with solid colour & pattern.
(v) Creation of diagram using random scan become easier so can be used in engineering & scientific drawing.	(v) It can be used in animations.

Colour CRT Monitor :
There are two technique used for producing colour display :

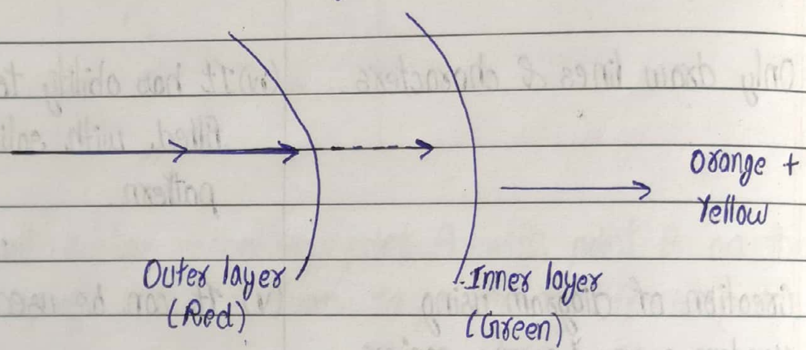
1. Beam Penetrating Technique.
2. Shadow Masking Technique.

1. Beam Penetrating Technique : This technique is used with Random scan monitor, in this technique the inside of CRT screen is coated with two layers of phosphour usually red & green. Display colour depends on how far e^- beam penetrates into the

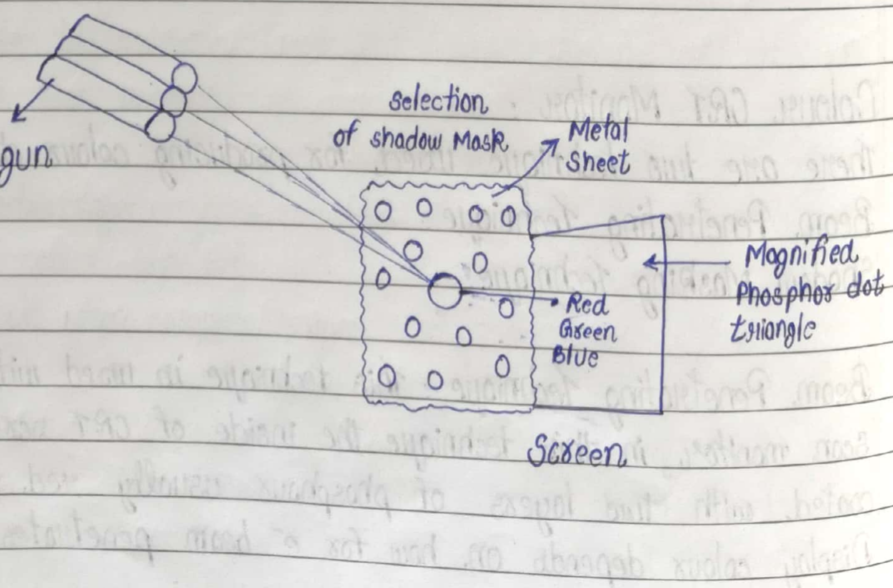
phosphor layer, outer is red phosphor layer and inner one is green. A beam of slow e^-'s excites only the outer red layer. A beam of fast e^-'s penetrates through red layer and exits green layer at intermediate beam speed combination of red & green ~~layer at intermediate beam speed~~ light are emitted two odd colour orange and yellow.

Merits :
- Inexpensive technique to produce colour in random scan monitor

Demerits :
- Display only 4 colour.
- Quality of picture produce by this technique is not good as compared to other technique.



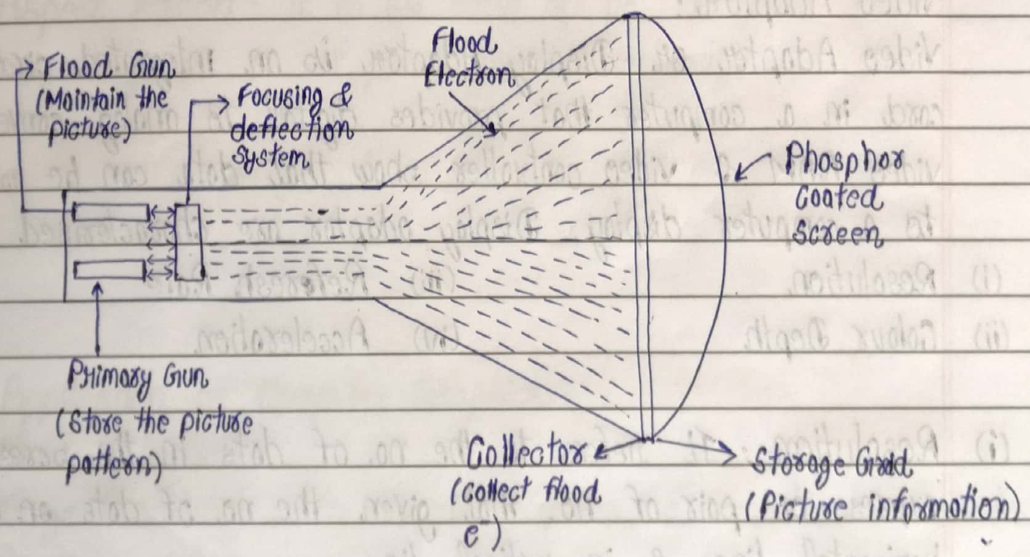
2. Shadow Masking Technique :



It produces a much wider range of colour then beam penetration technique. Hence this technique is commonly used Raster Scan display including colour TV. In this, CRT have 3 phosphor colour dot at each pixel position. One phosphor dot emit Red, 2nd → Green, 3rd → Blue show shadow mask CRT.

It has 3 e⁻ gun one of each colour dot and shadow mask grip just behind phosphor coated screen. 3 e⁻ beam are deflected & focused as a group on to the shadow mask and when they pass through a hole in shadow mask they excite a dot triangle. This consist of 3 small phosphor dot of RGB colour.

DVST (Direct View Storage Tube):



A DVST uses the storage grid which store the picture informative as +ve charge distribution just behind the phosphor coated screen. It consist of two e⁻ gun a primary gun & a flood gun. A primary gun store the picture pattern and flood gun maintain the picture display. A primary gun produce high speed e⁻ which strike on the storage grid to draw the picture pattern. The continuous low speed e⁻ flood gun pass through the control grid & are attracted to +ve charge area of storage grid.

Advantages :
- Referencing of CRT is not required.
- It has flat screen.

Disadvantages :
- They do not display colours & are available with single level of light intensity.
- The performance of DVST is somewhat inferior to CRT & was in failure to reference with CRT.

Video Controller :
It is often referred as video adaptor or graphic card, which is an important hardware component. It is used to generate graphics.

Video Adaptor :
Video Adaptor or Display Adaptor is an integrated circuit card in a computer that provides digital to analog converter, video RAM & video controller show that data can be send to a computer display. Display adaptor are characterised by:
(i) Resolution (iii) Refresh Rate
(ii) Colour Depth (iv) Acceleration

(i) Resolution : It refers to the no. of dots in the screen. It express as pair of no. that given the no. of dots on a horizontal lines & in vertical line.
640 x 480 1024 x 768
800 x 600 1280 x 1024

(ii) Colour Depth : (No. of colours) is determined by no. of bits assigned to hold colour value.
1 bit = 2 colour (Black & white)
2 bit = 4 colour
4 bit = 16 colour

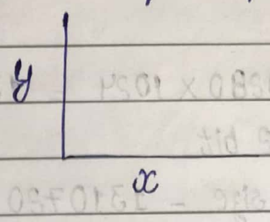
8 bit - 256 colour
16 bit - 32 thousand colour
24 bit - 16 million (high colour)
32 bit - latest

The display adaptor stores a value (4-32 bits) in memory for every dot on the screen.

(iii) Refresh Rate : It is nothing but the speed by which a particular dot on screen is getting printed.

(iv) Acceleration : This chip is an integrated circuit chip existing on the display adaptor. It is used to draw ready made chip line like drawing line & boxes.

(v) Aspect Ration : It is the ratio of no. of pixels present horizontally & no. of pixels present vertically.



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

Application of Computer Graphics :

- User Interface.
- Image Processing : In computer graphics, a computer is used to create a picture.
- Education & Training : Computer Graphics can be used to generate model of physical, financial & economic system. The model can be used as educational ads.
- CAD : To build structure of building..
- Plotting of graphics & charts .

Resolution, Frame Buffer And Aspect Ratio :

Q. Consider three different raster system with resolutions of 640 x 480, 1280 x 1024, 2560 x 2048. What size of frame buffer (in bytes) is needed for each of these system to store 12 bit per pixel. How much storage is required if 24 bit / pixel is to be stored.

Case (1) : 640 x 480

No. of Pixel = 640 x 480 = 307200

Per Pixel Store = 12 bit

Total no. of frame buffer size = 307200 x 12

or memory required = 3686400 bit

$= \frac{3686400}{8} = 460800$ bytes

$= \frac{460800}{1024} = 450$ KB

Case (2) : 1280 x 1024

No. of Pixel = 1280 x 1024 = 1310720

Per Pixel Store = 12 bit

Total no. of Frame buffer size = 1310720 x 12

= 15728640 bit

$= \frac{15728640}{8} = 1966080$ bytes

Case (3) : 2560 x 2048

No. of Pixel = 2560 x 2048 = 5242880

Per Pixel Store = 12 bit

Total no. of frame buffer size = 5242880 x 12

= 62914560 bit

$= \frac{62914560}{8} = 7864320$ bytes

Q Consider a raster system with resolution of 1024 x 1024 what is the size of raster (in bytes) needed to store 4 bit/pixel. How much storage is required if 8 bit/pixel are to be stored.

No. of Pixel = 1024 x 1024
= 1048576

Total no. of Frame buffer size = 1048576 x 4
= 4194304 bit

$= \frac{4194304}{8} = 524288$ bytes

Per Pixel store = 8 bit/pixel

Total no. of frame buffer size = 1048576 x 8 bits

$= \frac{1048576 \times 8}{8} = 1048576$ bytes.

* Access time/pixel = $\frac{1}{\text{Total Pixel Access Per Sec}}$

Q Consider two raster system with resolution of 640 x 480 & 1280 x 1024. How many pixels could be access per sec in each of these system by a display controller that refresh screen at a rate of 60 frame per sec. what is the access time of screen per pixel.

Case (1) : 640 x 480

No. of Pixel = 640 x 480 = 307200

Total no. of frame buffer size = 307200 x 60
= 18432000

Access time/pixel = $\frac{1}{\text{total pixel access per sec}} = 5.42 \times 10^{-8}$ sec/pixel.

Case (2) : 1280 x 1024

No. of Pixel = 1280 x 1024 = 1310720

Total no. of frame buffer size $= 1310720 \times 60$
$= 78643200$

Access time / pixel $= \frac{1}{\text{total pixel access per sec}} = 1.28 \times 10^{-8}$ sec/pixel

Q. A laser printer is capable of printing two pages (9x11) inch per sec at resolution of 600 pixels per inch. How many bits per sec does such device required.

$$\text{No. of Pixel} = 600 \times 600$$
$$= 360000$$
$$= 360000 \times 9 \times 11$$
$$= 35640000 \times 2$$
$$= 7.12 \times 10^{-7}$$

Q. How many K bytes does a frame buffer need in a 600x400 pixels.

$$\text{No. of pixel} = 600 \times 400 \times n \text{ bits}$$
$$= 240000\, n \text{ bits}$$
$$= \frac{240000\, n}{8} = 30000\, n \text{ bytes}$$
$$= \frac{30000\, n}{1024} = 29.29\, n$$

Q. Find out the aspect ratio of raster system using 8x10 inch screen and 100 pixel / inch.

$$\cancel{8 \times 10 \times 100 \times 100} = \cancel{8 \times 10^5}$$

$$= \frac{8 \times 100}{10 \times 100} = \frac{4}{5}$$

Scanned with CamScanner

Q How long would it take to load a 640 x 480 frame buffer with 12 bit per pixel if 10^5 bits can be transformed per sec. How long would it take to load a 24 bit per pixel frame buffer with a resolution of 1280 x 1024 using the same transfer rate.

No. of pixel = 640 x 480 = 307200

Total no. of frame buffer size = 307200 x 12 = 3686400

$$\text{Time} = \frac{3686400}{10^5} = 36.864 \text{ sec.}$$

$$\text{Time} = \frac{1280 \times 1024 \times 24}{10^5} = 314.5728 \text{ sec}$$

Q How much memory is needed for the frame of work to store a 640 x 400 display 16 gray level.

No. of pixels = 640 x 400 = 256000

$$16 \text{ gray level} = \frac{256000 \times 4}{8} \text{ byte} \qquad 16 \text{ gray} = 2^4 \quad 4 \text{ bit}$$

$$= \frac{128000}{1024} \text{ KB} = 125 \text{ KB}$$

Q How much time is spend scanning across each row of pixels during refresh on a raster system with resolution of 1280 x 1024 & refresh rate of 60 frames/sec.

~~No. of pixels = 1280 x 1024 = 1310720~~

Here resolution = 1280 x 1024

that means contains 1024 scan lines and each scan lines contains of 1280 pixel and refresh rate = 60 frame/sec.

that means 1 frame takes 1/60 sec.

Since Resolution = 1280 x 1024

∴ 1 frame consist of 1024 scan line

1024 scan lines takes 1/60 sec.

$$1 \text{ scan lines} = \frac{1}{60 \times 1024} \text{ sec} = 1.62 \times 10^{-5} \text{ sec}$$

Scanned with CamScanner

Point and Line :
Points is a fundamental elements of the picture representation. It is nothing but the position in a place define as either pairs or triplets of node depending on whether data are 2D or 3D. Thus (x, y) or (x_1, y_1, z_1) would represent a point in either in 2D or 3D space. Two points represents a line and collection of 3 or more points

Line Drawing Algorithm :
1. DDA (Digital Differential Analyzer) Algorithm.
2. Bresenham's Algorithm.

1. DDA Algorithm : It is the scan conversion for drawing the line. There are some following steps for drawing a line using DDA algorithm.

Step 1 : We have to given two coordinates (x_1, y_1) & (x_2, y_2).
Step 2 : Find the $\Delta y = y_2 - y_1$ & $\Delta x = x_2 - x_1$
Step 3 : Find slope $m = \frac{y_2 - y_1}{x_2 - x_1}$.

Step 4 : If $m \leq 1$ and the original value of m with +ve gentle slope it means $x_1 < x_2$ and the line has to be drawn from left to right direction. In this condition for finding the next pixel.

$$x_{K+1} = x_K + 1$$
$$y_{K+1} = y_K + m$$

Step 5 : If $m > 1$ with positive steep slopes in this condition

$$y_{K+1} = y_K + 1$$
$$x_{K+1} = x_K + 1/m$$

Step 6 : If $m \leq 1$ with negative slope. In this condition $x_1 > x_2$ & line has to be drawn from right to left direction. In this case :

$$x_{K+1} = x_K - 1$$
$$y_{K+1} = y_K - m$$

Step 7: If $m > 1$ with negative slope. In this condition

$$y_{k+1} = y_k - 1$$
$$x_{k+1} = x_k - 1/m$$

Advantages:

(i) The DDA algo. is a faster method for calculating pixel positions then direct use of line equation $y = mx + c$.
(ii) It is very easy to understand.
(iii) Requires no special skills for implementation.

Disadvantages:

(i) Because of round of, errors are introduced and the causes calculated pixel position to lift away from the 2 line path.
(ii) Because of floating point operations, the algo is time consuming.

Cases in DDA Algorithm:

Case 1: $|m| \leq 1$

(a) $x_1 < x_2$ with (+ve) m $\rightarrow$ $x_{k+1} = x_k + 1$
$y_{k+1} = y_k + m$

(b) $x_1 > x_2$ with (+ve) m $\rightarrow$ $x_{k+1} = x_k - 1$
$y_{k+1} = y_k - m$

(c) $x_1 < x_2$ with (-ve) m $\rightarrow$ $x_{k+1} = x_k + 1$
$y_{k+1} = y_k + m$

(d) $x_1 > x_2$ with (-ve) m $\rightarrow$ $x_{k+1} = x_k - 1$
$y_{k+1} = y_k - m$

Case 2: $|m| > 1$

(a) $y_1 < y_2$ with (+ve) m $\rightarrow$ $x_{k+1} = x_k + 1/m$
$y_{k+1} = y_k + 1$

(b) $y_1 > y_2$ with (+ve) m $\rightarrow$ $x_{k+1} = x_k - 1/m$
$y_{k+1} = y_k - 1$

(c) $y_1 < y_2$ with (-ve) m $\longrightarrow x_{k+1} = x_k + 1/m$

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

(d) $y_1 > y_2$ with (-ve) m $\longrightarrow x_{k+1} = x_k - 1/m$

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

Q. Draw the line using DDA algo. using (0,0) to (4,6).

$x_1 = 0 \qquad x_2 = 4$

$y_1 = 0 \qquad y_2 = 6$

$\Delta x = x_2 - x_1 = 4 \qquad m = \frac{y_2 - y_1}{x_2 - x_1} = \frac{6}{4} = 1.5$

$\Delta y = y_2 - y_1 = 6$

Here, $m > 1$; $(1.5 > 1)$

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

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

K	x_K	y_K	Plot (x_{K+1}, y_{K+1})
0	0	0	(0,0)
1	·66	1	(1,1)
2	1·32	2	(1,2)
3	1·48	3	(2,3)
4	2·64	4	(3,4)
5	3·3	5	(3,5)
6	3·96	6	(4,6)

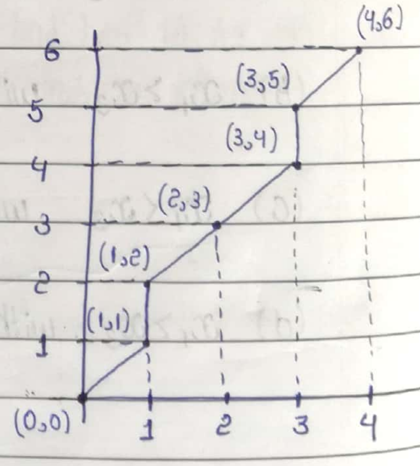
$x_1 = 0 + 1/1.5 = 0.66$

$y_1 = y_0 + 1 = 1$

$x_2 = x_1 + 1/m$

$= 0.66 + 0.66 = 1.32$

$y_2 = y_1 + 1 = 2$



$x_3 = x_2 + 1/m = 1.32 + 0.66 = 1.48$

$y_3 = y_2 + 1 = 2 + 1 = 3$

$x_4 = x_3 + 1/m = 2.64$

$y_4 = y_3 + 1 = 4$

$x_5 = x_4 + 1/m = 3.3$

$y_5 = y_4 + 1 = 5$

$x_6 = x_5 + 1/m = 3.96$

$y_6 = y_5 + 1 = 6$

Scanned with CamScanner

Q. Draw a line using DDA algo using (1,1) to (5,3)

$x_1 = 1 \quad x_2 = 5$

$y_1 = 1 \quad y_2 = 3$

$\Delta x = x_2 - x_1 = 4$

$\Delta y = y_2 - y_1 = 2$

$m = \frac{y_2 - y_1}{x_2 - x_1} = \frac{2}{4} = 0.5$

Here, $m \leq 1$

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

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

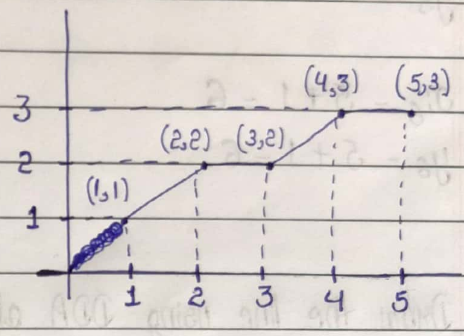
K	x_k	y_k	(x_{k+1}, y_{k+1})
0	1	1	(1,1)
1	2	1.5	(2,2)
2	3	2	(3,2)
3	4	2.5	(4,3)
4	5	3	(5,3)

$x_1 = x_0 + 1 = 2$

$y_1 = y_0 + m = 1.5$

$x_2 = x_1 + 1 = 3$

$y_2 = y_1 + m = 2$



$x_3 = x_2 + 1 = 4$

$y_3 = y_2 + m = 2.5$

$x_4 = x_3 + 1 = 5$

$y_4 = y_3 + m = 3$

Q. Draw the line from (0,0) to (6,6) using DDA algo.

$x_1 = 0 \quad x_2 = 6$

$y_1 = 0 \quad y_2 = 6$

$\Delta x = x_2 - x_1 = 6$

$\Delta y = y_2 - y_1 = 6$

$m = \frac{y_2 - y_1}{x_2 - x_1} = \frac{6}{6} = 1$

Here, $m = 1$

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

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

$x_1 = 0+1 = 1$
$y_1 = 0+1 = 1$

$x_2 = 1+1 = 2$
$y_2 = 1+1 = 2$

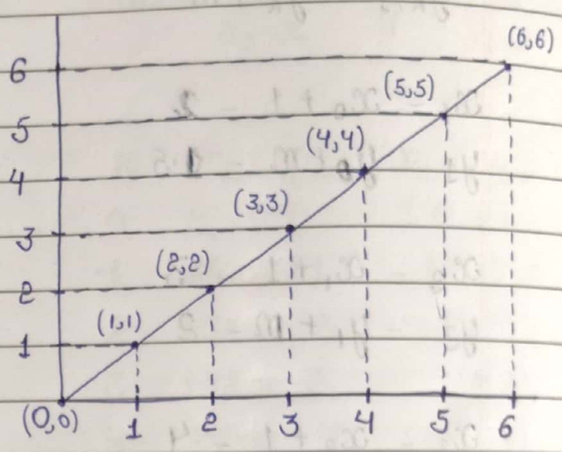
$x_3 = 2+1 = 3$
$y_3 = 2+1 = 3$

$x_4 = 3+1 = 4$
$y_4 = 3+1 = 4$

$x_5 = 4+1 = 5$
$y_5 = 4+1 = 5$

$x_6 = 5+1 = 6$
$y_6 = 5+1 = 6$.

K	x_K	y_K	(x_{K+1}, y_{K+1})
0	0	0	(0,0)
1	1	1	(1,1)
2	2	2	(2,2)
3	3	3	(3,3)
4	4	4	(4,4)
5	5	5	(5,5)
6	6	6	(6,6)



Q. Draw the line using DDA algo. (0,0) to (-6,-6)

Q. Draw the line using DDA algo. (30,18) to (20,10)

$x_1 = 30 \qquad x_2 = 20$

$y_1 = 18 \qquad y_2 = 10$

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

$\Delta y = y_2 - y_1 = 8$

$|m| = \left|\frac{y_2 - y_1}{x_2 - x_1}\right| = \left|\frac{8}{-10}\right| = \frac{8}{10} = 0.8$

$|m| < 1$

Here, $m < 1$, $(x_1 > x_2)$

$x_{K+1} = x_K - 1$

$y_{K+1} = y_K - m$

$x_1 = x_0 - 1 = 29$

$y_1 = y_0 - m = 17.2$

$x_2 = x_1 - 1 = 28$

$y_2 = y_1 - m = 16.4$

$x_3 = x_2 - 1 = 27$

$y_3 = y_2 - m = 15.6$

$x_4 = x_3 - 1 = 26$

$y_4 = y_3 - m = 14.8$

$x_5 = x_4 - 1 = 25$

$y_5 = y_4 - m = 14$

$x_6 = x_5 - 1 = 24$

$y_6 = y_5 - m = 13.2$

$x_7 = x_6 - 1 = 23$

$y_7 = y_6 - m = 12.4$

$x_8 = x_7 - 1 = 22$

$y_8 = y_7 - m = 11.6$

$x_9 = x_8 - 1 = 21$

$y_9 = y_8 - m = 10.8$

K	x_K	y_K	(x_{K+1}, y_{K+1})
0	30	18	(30, 18)
1	29	17.2	(29, 17)
2	28	16.4	(28, 16)
3	27	15.6	(27, 16)
4	26	14.8	(26, 15)
5	25	14	(25, 14)
6	24	13.2	(24, 13)
7	23	12.4	(23, 12)
8	22	11.6	(22, 12)
9	21	10.8	(21, 11)
10	20	10	(20, 10)

$x_{10} = x_9 - 1 = 20$

$y_{10} = y_9 - m = 10$

2. **Bresenham's line Drawing Algorithm:**

$|M| \leq 1 \quad (x_1 < x_2)$

Step 1: Input the two lines end points & store the left end point in (x_0, y_0).

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

Step 3: Calculate constants Δx, Δy, $2\Delta y$, $2\Delta x$ and obtain the starting value for the decision parameter $[P_0 = 2\Delta y - \Delta x]$.

Step 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 $[P_{K+1} = P_K + 2\Delta y]$.

Step 5: If $P_K \geq 0$ the next point to plot in (x_{K+1}, y_{K+1}) and $[P_{K+1} = P_K + 2\Delta y - 2\Delta x]$.

Step 6: Repeat step Δx times.

Q. Draw a line using Bresenham's algo (20, 10) and (30, 18).

$x_1 = 20$ $\quad$ $x_2 = 30$

$y_1 = 10$ $\quad$ $y_2 = 18$

$\Delta x = x_2 - x_1 = 10$ $\quad$ $m = \frac{\Delta y}{\Delta x} = \frac{8}{10} = 0.8 \leq 1$

$\Delta y = y_2 - y_1 = 8$ $\quad$ $(20 < 30)$

$2\Delta y = 16$, $2\Delta x = 20$

$2\Delta y - 2\Delta x = -4$

$P_0 = 2\Delta y - \Delta x$

$P_0 = 16 - 10 = 6$

$P_K \geq 0$ $\quad (x_{K+1}, y_{K+1})$

$P_{K+1} = P_K + 2\Delta y - 2\Delta x$

$P_1 = P_0 + (-4)$

$P_1 = 6 - 4 = 2$

$(P_K \geq 0)$

$P_{K+1} = P_K + 2\Delta y - 2\Delta x$

$P_2 = P_1 + (-4)$

$P_2 = 2 - 4 = -2$

$(P_K < 0)$

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

$P_3 = P_2 + 2\Delta y$

$P_3 = -2 + 16 = 14$

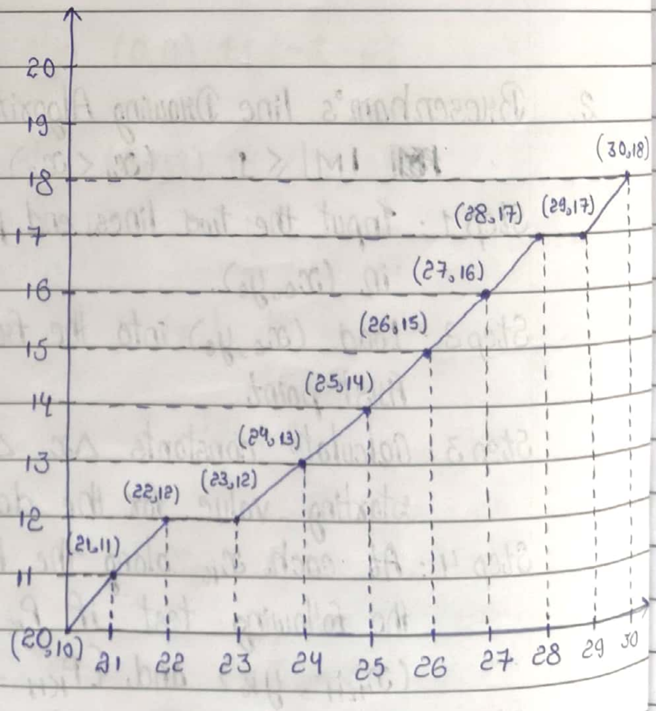
$(P_K \geq 0)$

$P_{K+1} = P_K + 2\Delta y - 2\Delta x$

$P_4 = P_3 + (-4)$

$P_4 = 14 - 4 = 10$

K	P_K	Plot (x_{K+1}, y_{K+1})
0	6	(21, 11)
1	2	(22, 12)
2	-2	(23, 12)
3	14	(24, 13)
4	10	(25, 14)
5	6	(26, 15)
6	2	(27, 16)
7	-2	(28, 17)
8	14	(29, 17)
9	10	(30, 18)



$(P_K \geq 0)$

$P_{K+1} = P_K + 2\Delta y - 2\Delta x$

$P_5 = P_4 + (-4)$

$P_5 = 10 - 4 = 6$

$(P_K \geq 0)$

$P_{K+1} = P_K + 2\Delta y - 2\Delta x$

$P_6 = P_5 + (-4)$

$P_6 = 6 - 4 = 2$

$(P_K \geq 0)$

$P_{K+1} = P_K + 2\Delta y - 2\Delta x$

$P_7 = P_6 + (-4)$

$P_7 = 2 - 4 = -2$

$(P_K < 0)$

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

$P_8 = P_7 + (16)$

$P_8 = -2 + 16 = 14$

$(P_K \geq 0)$

$P_{K+1} = P_K + 2\Delta y - 2\Delta x$

$P_9 = P_8 + (-4)$

$P_9 = 14 + (-4) = 10$

Case 2 : $|M| > 1$

$[P_0 = 2\Delta x - \Delta y]$

if $P_K < 0$: (x_K, y_{K+1}) $[P_{K+1} = P_K + 2\Delta x]$

if $P_K \geq 0$: (x_{K+1}, y_{K+1}) $[P_{K+1} = P_K + 2\Delta x - 2\Delta y]$

Repeat step Δy times

Q. Draw a line using Bresenham's algo (10, 20) and (18, 30).

$x_1 = 10$ $x_2 = 18$

$y_1 = 20$ $y_2 = 30$

$\Delta x = x_2 - x_1 = 8$

$\Delta y = y_2 - y_1 = 10$

$m = \frac{\Delta y}{\Delta x} = \frac{10}{8} = 1.25 > 1$

$2\Delta y = 20$, $2\Delta x = 16$

$2\Delta y - 2\Delta x = 4$

$2\Delta x - 2\Delta y = -4$

$P_0 = 2\Delta x - \Delta y$

$P_0 = 16 - 10 = 6$

$(P_K \geq 0)$

$P_{K+1} = P_K + 2\Delta x - 2\Delta y$

$P_1 = P_0 + (-4)$

$P_1 = 6 - 4 = 2$

$(P_K \geq 0)$

$P_{K+1} = P_K + 2\Delta x - 2\Delta y$

$P_2 = P_1 + (-4)$

$P_2 = 2 - 4 = -2$

$(P_K < 0)$

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

$P_3 = P_2 + 16$

$P_3 = -2 + 16 = 14$

$(P_K \geq 0)$

$P_{K+1} = P_K + 2\Delta x - 2\Delta y$

$P_4 = P_3 + (-4)$

$P_4 = 14 - 4 = 10$

$(P_K \geq 0)$

$P_{K+1} = P_K + 2\Delta x - 2\Delta y$

$P_5 = P_4 + (-4)$

$P_5 = 10 - 4 = 6$

$(P_K \geq 0)$

$P_{K+1} = P_K + 2\Delta x - 2\Delta y$

$P_6 = P_5 + (-4)$

$P_6 = 6 - 4 = 2$

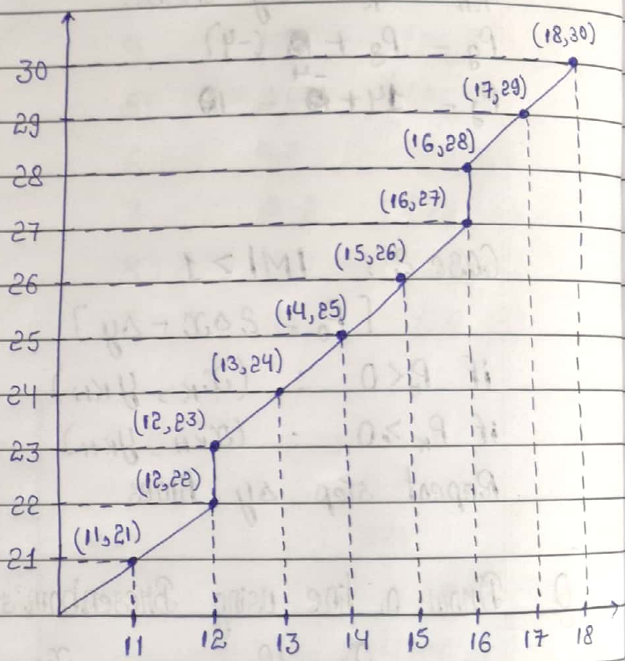
$(P_K \geq 0)$

$P_{K+1} = P_K + 2\Delta x - 2\Delta y$

$P_7 = P_6 + (-4)$

$P_7 = 2 - 4 = -2$

K	P_K	(x_{K+1}, y_{K+1})
0	6	(11, 21)
1	2	(12, 22)
2	-2	(12, 23)
3	14	(13, 24)
4	10	(14, 25)
5	6	(15, 26)
6	2	(16, 27)
7	-2	(16, 28)
8	14	(17, 29)
9	10	(18, 30)



$(P_K \cancel{<} 0)$

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

$P_8 = P_7 + 2\Delta x$

$P_8 = -2 + 16 = 14$

$(P_K \geq 0)$

$P_{K+1} = P_K + 2\Delta x - 2\Delta y$

$P_9 = P_8 + (-4)$

$P_9 = 14 - 4 = 10$

Scanned with CamScanner

Q Draw a line using Bresenham's algo (0,0) to (6,7).

$x_1 = 0 \qquad x_2 = 6$

$y_1 = 0 \qquad y_2 = 7$

$\Delta x = x_2 - x_1 = 6 \qquad m = \frac{\Delta y}{\Delta x} = \frac{7}{6} =$

$\Delta y = y_2 - y_1 = 7$

$2\Delta y = 14, \quad 2\Delta x = 12$

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

$P_0 = 2\Delta x - \Delta y$

$P_0 = 12 - 7 = 5$

K	P_K	(x_{K+1}, y_{K+1})
0	5	(1, 1)
1	3	(2, 2)
2	1	(3, 3)
3	-1	(3, 4)
4	11	(4, 5)
5	9	(5, 6)
6	7	(6, 7)

$(P_K \geq 0)$

$P_{K+1} = P_K + 2\Delta x - 2\Delta y$

$P_1 = P_0 + (-2) = 5 - 2 = 3$

$(P_K \geq 0)$

$P_{K+1} = P_K + 2\Delta x - 2\Delta y$

$P_2 = P_1 + (-2) = 3 - 2 = 1$

$(P_K \geq 0)$

$P_{K+1} = P_K + 2\Delta x - 2\Delta y$

$P_3 = P_2 + (-2) = 1 - 2 = -1$

$(P_K < 0)$

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

$P_4 = P_3 + 12 = -1 + 12 = 11$

$(P_K \geq 0)$

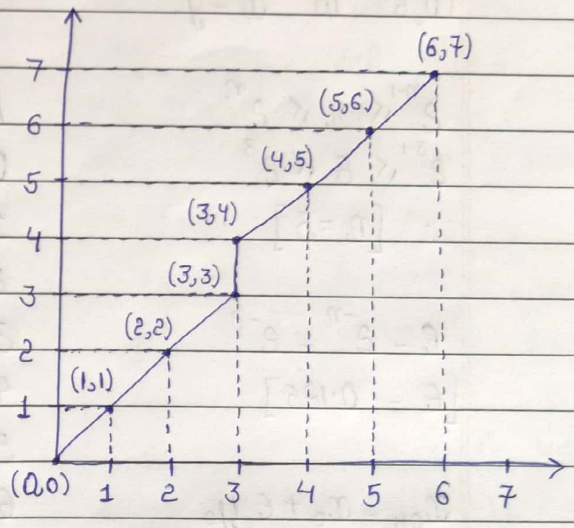
$P_{K+1} = P_K + 2\Delta x - 2\Delta y$

$P_5 = P_4 + (-2) = 11 - 2 = 9$

$(P_K \geq 0)$

$P_{K+1} = P_K + 2\Delta x - 2\Delta y$

$P_6 = P_5 + (-2) = 9 - 2 = 7$



Circle Generation Algorithm :

1. DDA Algorithm : It is the scanning of pixel which is generating in circle for calculating the pixel position using DDA algorithm. There are following some steps :

Step 1 : Let us given radius and end points of the circle.

Step 2 : Satisfy the condition $2^{n-1} \leq r \leq 2^n$
For this equation find the value of n.

Step 3 : Find the decision parameter $\epsilon = 2^{-n}$

Step 4 : For calculating the next pixel in X and Y direction

$$x_{K+1} = x_K + \epsilon Y_K$$

$$y_{K+1} = y_K - \epsilon x_{K+1}$$

Q. Calculate the pixel position along the circle path with radius $r = 6$, centre on the origin using DDA algo from point $(0,6)$ to $x = y$.

$2^{n-1} \leq r \leq 2^n$

$2^{3-1} \leq 6 \leq 2^3$

$\therefore [n = 3]$

$\epsilon = 2^{-n} = 2^{-3}$

$[\epsilon = 0.125]$

K	x	y	Plot (x, y)
0	0	6	(0,6)
1	0.75	5.90	(1,6)
2	1.48	5.715	(2,6)
3	2.25	5.71	(2,6)
4	3	5.62	(3,6)
5	3.75	5.53	(4,4)
6	4.5	5.43	(4,4)

$x_{0+1} = x_0 + \epsilon y_0$

$x_1 = 0 + (0.125)(6) = 0.75$

$y_{0+1} = y_0 - \epsilon x_{0+1}$

$y_1 = 6 - (0.125)(0.75) = 5.90$

$x_{1+1} = x_1 + \epsilon y_1$

$x_2 = 0.75 + (0.125)(5.90) = 1.48$

$y_{1+1} = y_1 - \epsilon x_{1+1}$

$y_2 = y_1 - \epsilon x_2$

$y_2 = 5.90 - (0.125)(1.48) = 5.715$

Q Calculate the pixel position along the circle path with radius $r=8$, centre on the origin using DDA algo from point $(0,8)$ to $x=y$.

$2^{n-1} \leq r \leq 2^{n}$

$2^{4-1} \leq 8 \leq 2^{4}$

$\therefore [n=4]$

$\epsilon = 2^{-n} = 2^{-4}$

$[\epsilon = 0.0625]$

$x_1 = 0 + (0.0625)(8) = 0.5$

$y_1 = 8 - (0.0625)(0.5) = 7.96$

$x_2 = 0.5 + (0.0625)(7.96) = 0.9$

$y_2 = 7.96 - (0.0625)(0.9) = 7.90$

$x_3 = 0.9 + (0.0625)(7.90) = 1.39$

$y_3 = 7.90 - (0.0625)(1.39) = 7.81$

k	x	y	(x, y)
0	0	8	(0,8)
1	0.5	7.96	(1,8)
2	0.9	7.90	(1,8)
3	1.39	7.81	(1,8)
4	1.87	7.69	(2,8)
5	2.35	7.54	(2,8)
6	2.82	7.36	(3,7)
7	3.28	7.15	(3,7)
8	3.72	6.91	(4,7)
9	4.15	6.65	(4,7)
10	4.62	6.30	(5,6)
11	5.01	5.9	(5,6)
12	5.48	5.5	(5,5)

$x_4 = 1.39 + (0.0625)(7.81) = 1.87$

$y_4 = 7.81 - (0.0625)(1.87) = 7.69$

$x_5 = 1.87 + (0.0625)(7.69) = 2.35$

$y_5 = 7.69 - (0.0625)(2.35) = 7.54$

$x_6 = 2.35 + (0.0625)(7.54) = 2.82$

$y_6 = 7.54 - (0.0625)(2.82) = 7.36$

$x_7 = 2.82 + (0.0625)(7.36) = 3.28$

$y_7 = 7.36 - (0.0625)(3.28) = 7.15$

$x_8 = 3.28 + (0.0625)(7.15) = 3.72$

$y_8 = 7.15 - (0.0625)(3.72) = 6.91$

$x_9 = 3.72 + (0.0625)(6.91) = 4.15$

$y_9 = 6.91 - (0.0625)(\cancel{6.91}\ 4.15) = 6.65$

$x_{10} = 4.15 + (0.0625)(6.65) = 4.62$

$y_{10} = 6.65 - (0.0625)(4.62) = 6.30$

$x_{11} = 4.62 + (0.0625)(6.30) = 5.01$

$y_{11} = 6.30 - (0.0625)(5.01) = 5.9$

$x_{12} = 5.01 + (0.0625)(5.9) = 5.48$

$y_{12} = 5.9 - (0.0625)(5.48) = 5.55$

Mid Point Circle Generation Algorithm :
When the circle includes the radius r and its centre coordinate (x_c, y_c) then the circle is generated calculating the pixel from starting point to end point. Their are following steps :

Step 1 : Input radius 'r' and circle centre (x_c, y_c) and obtain the first point on the circumference of a circle centre on the horizon as $(x_0, y_0) = (0, r)$.

Step 2 : Calculate the initial value of the decision parameter as :

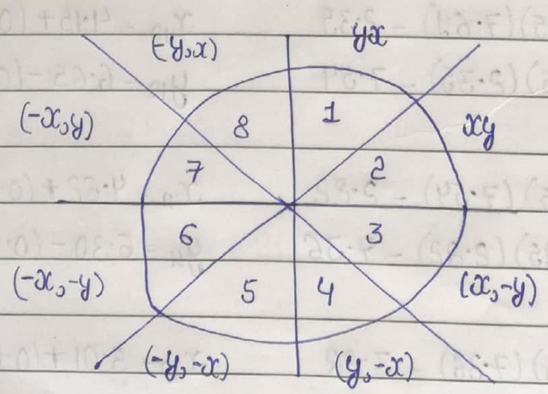
$$\left\{ P_0 = 5/4 - r \quad \text{or} \quad P_0 = 1.25 - r \quad \text{or} \quad P_0 = 1 - r \right\}$$

Step 3 : At each P_K position starting at $K = 0$ perform the following test :
if $P_K < 0$, the next point along the circle centre $(0,0)$ is (x_{K+1}, y_K) and $[P_{K+1} = P_K + 2x_{K+1} + 1]$
if $P_K > 0$, 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 $[2y_{K+1} = 2y_K - 2]$.

Step 4 : Determine semmetric points in the other seven octants.

Step 5 : Move each calculate pixel position (x, y) on to the circular path centre on (x_c, y_c) and plot the coordinate values as $x = x + x_c$ & $y = y + y_c$.

Step 6 : Repeat step 5 until $x >= y$.



Q. Plot the circle with centre origin and $r = 10$ from the point $(0, 10)$ to $x = y$.

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

$r = 10$

P_K	x_K	y_K	(x_{K+1}, y_{K+1})
-8.75	0	10	(0, 10)
-5.75	1	10	(1, 10)
-0.75	2	10	(2, 10)
6.25	3	10	(3, 10)
-2.75	4	9	(4, 9)
8.25	5	9	(5, 9)
5.25	6	8	(6, 8)
6.25	7	7	(7, 7)

$P_0 = 1.25 - r = 1.25 - 10$

$= -8.75$

$P_0 < 0$

$P_{K+1} = P_K + 2x_{K+1} + 1$

$P_1 = P_0 + 2x_1 + 1$

$= -8.75 + 2 \times 1 + 1$

$= -5.75$

$P_1 < 0$

$P_2 = P_1 + 2x_2 + 1$

$= -5.75 + 4 + 1$

$= -0.75$

$P_2 < 0$

$P_3 = P_2 + 2x_3 + 1$

$= -0.75 + 6 + 1$

$= 6.25$

$P_3 > 0$

$P_4 = P_3 + 2x_4 + 1 - 2y_4$

$P_4 = 6.25 + 8 + 1 - 18$

$= -2.75$

$P_4 < 0$

$P_5 = P_4 + 2x_5 + 1$

$P_5 = -2.75 + 10 + 1$

$= +8.25$

$P_5 > 0$

$P_6 = P_5 + 2x_6 + 1 - 2y_6$

$= 5.25$

$P_6 > 0$

$P_7 = P_6 + 2x_7 + 1 - 2y_7$

$= 6.25$

The Bresenham's Circle Drawing Algorithm:
This algorithm works on the principle of bresenham line draw algorithm as it takes only integer values in comparision to DDA and mid point circle algorithm.
In the Bresenham's algorithm their are some following steps to draw the circle in 1^{st} octant and further octants are calculated as well as.

Step 1: Let us given the radius r and the initial value $(x_0, y_0) = (0, r)$.

Step 2: Find the decision parameter:
$[P_0 = 3 - 2r]$

Step 3: If $P_K \leq 0$, then the next pixel will be generated according to (x_{K+1}, y_K).
$[P_{K+1} = P_K + 4x_K + 6]$

Step 4: Otherwise $P_K > 0$, the next pixel will be generated as (x_{K+1}, y_{K-1})
$[P_{K+1} = P_K + 4x_K - 4y_K + 10]$

Step 5: This process will be continue until $x \leq y$.

Q Draw the circle with radius $r = 10$.

$r = 10$.
$(x_0, y_0) = (0, r) = (0, 10)$
$P_0 = 3 - 2r$
$= 3 - 2 \times 10 = -17$
$P_0 \leq 0$
$P_1 = P_0 + 4x_0 + 6$
$= -17 + (4 \times 0) + 6$
$= -11$
$P_1 < 0$
$P_2 = P_1 + 4x_1 + 6$
$= -1$
$P_2 \leq 0$
$P_3 = P_2 + 4x_2 + 6$
$= 13$

P_K	x_K	y_K	(x_{K+1}, y_{K+1})
-17	0	10	(0, 10)
-11	1	10	(1, 10)
-1	2	10	(2, 10)
13	3	10	(3, 10)
-5	4	9	(4, 9)
17	5	9	(5, 9)
11	6	8	(6, 8)
13	7	7	(7, 7)

Mid Point Ellipse Algorithm:

Step 1: Input r_x, r_y and ellipse centre (x_c, y_c) & obtain the first point on an ellipse centre on the origin as $(x_0, y_0) = (0, r_y)$.

Step 2: Calculate the initial value of decision parameter in region 1 as $P_0 = r_y^2 - r_x^2 . r_y + \frac{1}{4} r_x^2$

Step 3: At each x_k position in region 1 starting at $K=0$ perform the following test if $P_k \leq 0$ the next point along ellipse centre on $(0,0)$.

(x_{k+1}, y_k) $[P_{k+1} = P_k + 2r_y^2 . x_{k+1} + r_y^2]$

$P_k > 0$:

(x_{k+1}, y_{k-1}) $[P_{k+1} = P_k + 2r_y^2 . x_{k+1} - 2r_x^2 y_{k+1} + r_y^2]$

Step 4: Continue until $2r_y^2 x \geq 2r_x^2 y$

Step 5: Calculate the initial value of decision parameter in region 2 using the last point (x_0, y_0) calculated in Region 1 as:

$P_0 = r_y^2 (x_0 + 1/2)^2 + r_x^2 (y_0 - 1)^2 - r_x^2 . r_y^2$

Step 6: At each y_k position in region 2 starting at $K=0$ perform the following test $P_k > 0$, the next point along ellipse centre on $(0,0)$ is (x_k, y_{k-1}) &

$[P_{k+1} = P_k - 2r_x^2 . y_{k+1} + r_x^2]$

Otherwise $P \leq 0$, the next point along the circle as (x_{k+1}, y_{k-1}) & $[P_{k+1} = P_k + 2r_y^2 x_{k+1} - 2r_x^2 y_{k+1} + r_x^2]$

Step 7: Using the same incremental condition for (x, y) as in region 1.

Step 8: Determine symmetry points in the other three quadrants.

Step 9: Move each calculated pixel position x & y on to the ellipse path centre on (x_c, y_c) & plot the coordinates value $x = x + x_c$, $y = y + y_c$.

Step 10: Repeat the step for region 1 untill $2r_x^2 > 2r_y^2$.

Q Calculate the pixel position given the radius of 8 & 6 using of mid-point ellipse algorithm. centre (0,0).

$r_x = 8$ $r_y = 6$

$r_x^2 = 64$ $r_y^2 = 36$

Now, initial coordinates are :

$(x_0, y_0) = (0, r_y) = (0,6)$

$2r_x^2 = 128$, $2r_y^2 = 72$

Region 1 :

$P_0 = r_y^2 - r_x^2 . r_y + 1/4 r_x^2$

$= 36 - 64 . 6 + 1/4 . 64$

$= -332$

$P_0 \leqslant 0$

$P_1 = P_0 + 2r_y^2 . x_1 + r_y^2$

$= -332 + 72 \cdot 1 + 36 = -224$

$P_1 \leqslant 0$

$P_2 = P_1 + 2r_y^2 . x_2 + r_y^2$

$= -44$

$P_2 \leqslant 0$

$P_3 = P_2 + 2r_y^2 . x_3 + r_y^2$

$= -44 + 216 + 36 = 208$

$P_3 > 0$:

$P_4 = P_3 + 2r_y^2 . x_4 - 2r_x^2 y_4 + r_y^2$

$= 208 + 288 - 640 + 36 = -108$

$P_4 \leqslant 0$

$P_5 = P_4 + 2r_y^2 . x_5 + r_y^2$

$= -108 + 360 + 36 = 288$

K	P_K	x_0	y_0
0	-332	0	6
1	-224	1	6
2	-44	2	6
3	208	3	6
4	-108	4	5
5	288	5	5
6	244	6	4
7	400	7	3

$P_5 > 0$

$P_6 = P_5 + 2r_y^2 . x_6 - 2r_x^2 y_6 + r_y^2$

$= 288 + 432 - 512 + 36$

$= 244$

$P_6 > 0$

$P_7 = P_6 + 2r_y^2 . x_7 - 2r_x^2 y_7 + r_y^2$

$= 244 + 504 - 384 + 36$

$= 400$

$2r_y^2 . x \geqslant 2r_x^2 y$

$504 \geqslant 384$

Region 2 :

$P_0 = r_y^2 (x_0 + 1/2)^2 + r_x^2 (y_0 - 1)^2 - r_x^2 . r_y^2$

$= 36 (7 + 1/2)^2 + 64 (3-1)^2 - 64 . 36 = -23$

$P_0 \leqslant 0$

$P_1 = P_K + 2r_y^2 x_1 - 2r_x^2 y_1 + r_x^2 = 361$

K	P_K	x_0	y_0
0	-23	7	3
1	361	8	2

Parallel line Drawing Algorithm:

The parallel version of line drawing algorithm depends on the line style, line thickness and line width. When we draw the line using end points the line can be completed, dotted or in the brightness and intensity form.

For line style we use the command set linestyle(), and the function can be any number which determines the different style as continuous or dotted for different line width we use the command set linewidth() and in this function we use the value 1. It means the pixel of the line display with background colour.

Unit - 2

2-D Transformation :

It is a technique by which the object is moved about the horizon, about axis or about the line and the shape position and direction of the object is changed with background of translation, rotation, scaling and reflection. The transformation has two categories :

(i) Geometric Transformation

(ii) Coordinate Transformation

(i) Geometric Transformation : In this the object is moved about any axis with relative to the stationary coordinate system.

Translation :

In the translation the object is displayed in the direction, their are two translation parameters :

(i) T_x - in x direction

(ii) T_y - in y direction

Let us consider an object P on the coordinate system with coordinate (x, y) and after translating the new coordinate points (x', y') so we determine the new coordinates P' by the formula :

$$[P' = T_V . P] \longrightarrow \text{column measure order}$$

where 'T_V' is Translation vector

The value is in matrix format

$$T_V = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix}$$

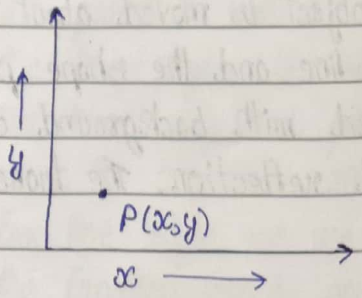
and the x' value, $x' = x + t_x$

and $y' = y + t_y$

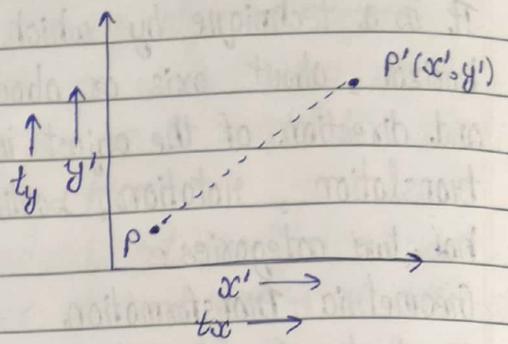
$$P' = T_V . P$$

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

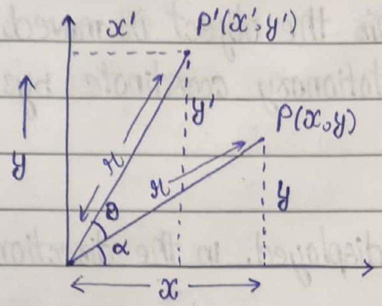
Before Translation



After Translation



Rotation :



$\cos\alpha = x/r$

$[x = r\cos\alpha]$

$\sin\alpha = y/r$

$[y = r\sin\alpha]$

$\cos(\theta+\alpha) = x'/r$

$[x' = r\cos(\theta+\alpha)]$

$\sin(\theta+\alpha) = y'/r$

$[y' = r\sin(\theta+r)]$

$x' = r\cos\theta.\cos\alpha - r\sin\theta.\sin\alpha$

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

$y' = r\sin\theta\cos\alpha + r\cos\theta.\sin\alpha$

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

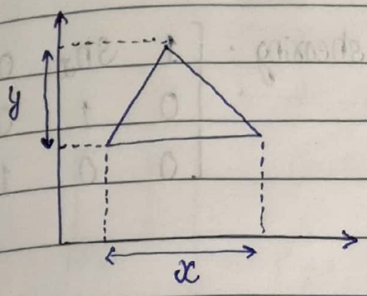
$[P' = R_\theta . P]$ (anticlock wise) $\qquad [P' = R_{(-\theta)}.P]$ (clockwise)

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

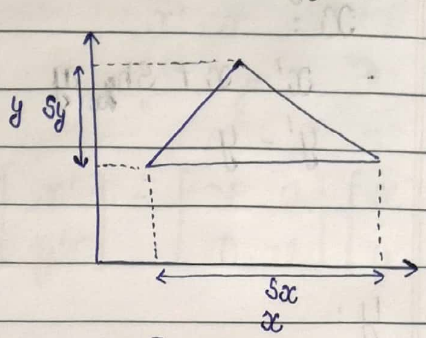
Scaling :

It is the process of changing the shape of the object with stationary coordinate system.

Before Scaling



After Scaling



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

There are two scaling factors : S_x, S_y.

$S_x = S_y = 1$ is called Homogeneous.

$S_x = S_y > 1$ is called Magnification

$S_x = S_y < 1$ is called Reduction.

Scaling Factor : $P' = S_{x}S_{y} P$

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

~~Reflection~~:

Homogeneous Coordinate & Matrix Representation :

In design & picture formation process many times we may require to perform translation, rotation & scaling to fit the picture components into their proper position. Any transformation to represent general expressions :

$$[P' = M_1 . P + M_2]$$

where M_1 = Multiplicative Matrix (Identity)

M_2 = Additive Matrix

Shearing : One coordinate fixed.

x :

$x' = x + Sh_x \cdot y$

$y' = y$

x shearing : $\begin{bmatrix} 1 & Sh_x & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$

y :

$y' = y + Sh_y \cdot x$

$x' = x$

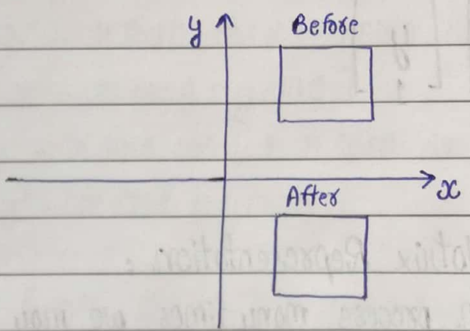
y shearing : $\begin{bmatrix} 1 & 0 & 0 \\ Sh_y & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$

Reflection (Ref or M) :

In this the coordinate system works as an image & the object is reflected or has its image infront of

Let us consider the object P with coordinates (x, y)

Case 1 : About x-axis :



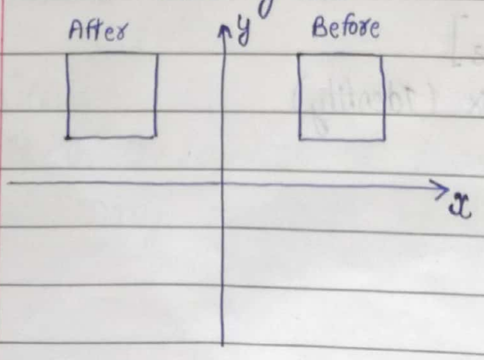
$x' = x$

$y' = -y$

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

$[P' = M_x \cdot P]$

Case 2 : About y-axis



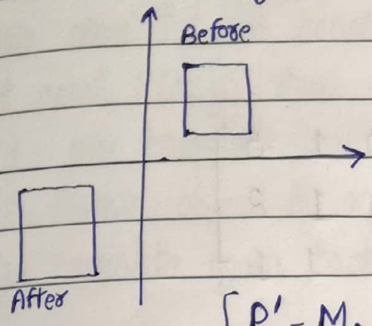
$x' = -x$

$y' = y$

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

$[P' = M_y \cdot P]$

Case 3 : About origin.



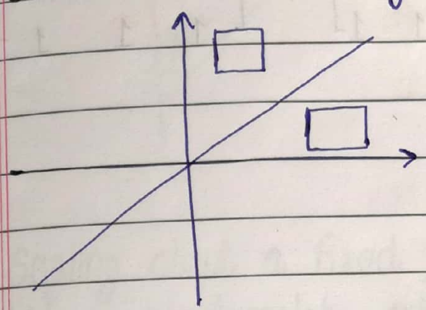
$x' = x$

$y' = -y$

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

$[P' = M_0 . P]$

Case 4 : About line $(y = x)$



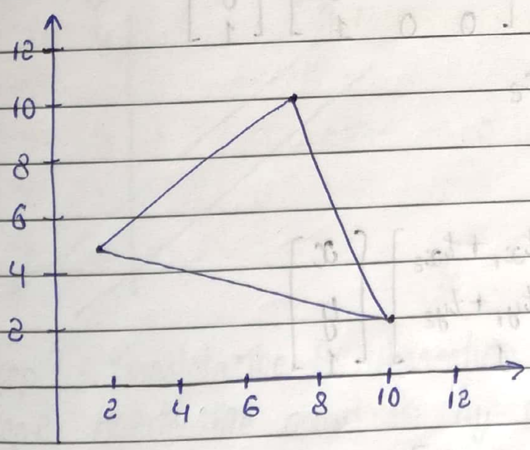
$x' = y$

$y' = x$

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

Q. Translate a polygon with coordinate : A (2,5), B (7,10), C (10,2) by 3 units in x-direction & 4 units in y-direction.

Before Translation :



$[P' = T_v . P]$

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 3 \\ 0 & 1 & 4 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 2 & 7 & 10 \\ 5 & 10 & 2 \\ 1 & 1 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 5 & 10 & 13 \\ 9 & 14 & 6 \\ 1 & 1 & 1 \end{bmatrix}$$

A'(5,9), B'(10,14), C'(13,6)

Case 5 : About line $(y = -x)$

$x' = -y$

$y' = -x$

Q. Consider a coordinate $A(0,0)$, $B(1,1)$, $C(5,2)$. Rotation of $45°$ at origin.

$$[P' = R_\theta . P]$$

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 1 & 5 \\ 0 & 1 & 2 \\ 1 & 1 & 1 \end{bmatrix}$$

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1/\sqrt{2} & -1/\sqrt{2} & 0 \\ 1/\sqrt{2} & 1/\sqrt{2} & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 1 & 5 \\ 0 & 1 & 2 \\ 1 & 1 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 3/\sqrt{2} \\ 0 & \sqrt{2} & 7/\sqrt{2} \\ 1 & 1 & 1 \end{bmatrix}$$

Composite Transformation :

2 Successive Translation :

$$[P' = T_v . P]$$

$$(t_{x_1}, t_{y_1}) \quad (t_{x_2}, t_{y_2})$$

$$P' = [T(t_{x_1}, t_{y_1})(t_{x_2}, t_{y_2})] P$$

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & t_{x_1} \\ 0 & 1 & t_{y_1} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & t_{x_2} \\ 0 & 1 & t_{y_2} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

$$x' = x + t_{x_1} + t_{x_2}$$

$$y' = y + t_{y_1} + t_{y_2}$$

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & t_{x_1} + t_{x_2} \\ 0 & 1 & t_{y_1} + t_{y_2} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

$$[P' = T(t_{x_1}, t_{x_2}, t_{y_1}, t_{y_2}) . P]$$

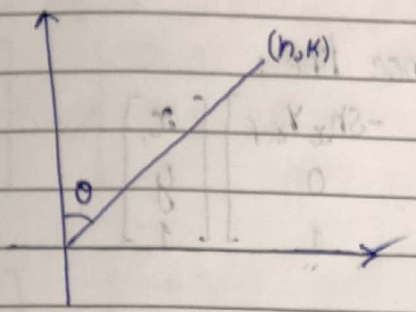
Rotation about any point (arbitary point).

In this case let us consider the arbitary point (h, K) and the object point $P(x, y)$ there are following some steps:

Step 1: We translate arbitary point (h, K) to origin.

Step 2: Rotate object by angle θ.

Step 3: Translate again back to arbitary point (h, K).



$$[P' = T_V . R_\theta . T_{-V} . P]$$

Scaling about a fixed point:

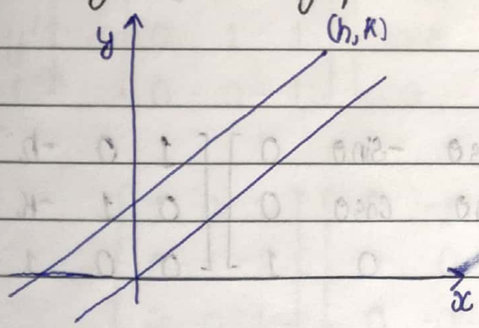
Step 1: We translate arbitary point (h, K) to origin.

Step 2: Scale object (~~x, y~~) at origin.

Step 3: Translate back to arbitary point (h, K).

$$[P' = T_V . S_{sx sy} . T_{-V}].P]$$

Reflecting about any point (arbitary point):



$$[P' = M_L . P]$$

$$[P' = T_V R_{(-\theta)} M_y R_\theta T_{(-V)} . P]$$

Step 1: Translate the P intersection point (0, c) to the origin.

Step 2: Rotate the coordinate by θ degree in anticlock wise direction.

Step 3: Mirror reflection about y-axis.

Step 4: Rotate back $(-\theta)$ degree.

Step 5: Translate back to the intersection point (0, c) to (h, K).

Shearing Transformation with respect to any point:
The shearing transformation w.r.t the point (h, K), the matrix form is written as:

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & sh_x & -sh_x Y_{ref} \\ sh_y & 1 & -sh_y X_{ref} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

x shear with y reference line:

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

y shear with x reference line:

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

Q. Write a transformation matrix for rotation by $\pi/2$ and rotate anticlockwise direction about point (1, 1).

$$P' = T_V R_\theta T_{(-V)}$$

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

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

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

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

Q Find the new coordinate of the point $P(2,-4)$ after rotation $\theta = \pi/2$.

$P' = R_\theta . P$

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 2 \\ -4 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 2 \\ -4 \\ 1 \end{bmatrix}$$

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 4 \\ 2 \\ 1 \end{bmatrix}$$

$P(4,2)$

Q Find the new coordinate of the point $P(2,-4)$ after rotation $\pi/2$ about point $P(1,1)$.

$P' = T_v R_\theta T_{(-v)} . P$

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & -1 \\ 0 & 1 & -1 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 2 \\ -4 \\ 1 \end{bmatrix}$$

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

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 6 \\ 2 \\ 1 \end{bmatrix}$$

$P(6,2)$

Q Perform 60° rotation of a triangle with vertices $A(0,0)$, $B(1,1)$, $C(-1,-1)$ about origin & about the point $(0,1)$.

About origin :

$$P' = \cancel{T_v}\, R_\theta\, \cancel{T_{(-v)}} \cdot P$$

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

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

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

About point $(0,1)$:

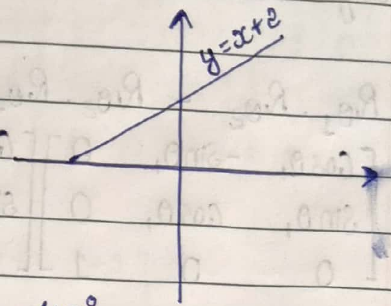
$$P' = T_v\, R_\theta\, T_{(-v)} \cdot P$$

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

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

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

Q. Reflect the triangle with vertices $A(-1,0)$, $B(0,-2)$, $C(5,0)$ above the line $y = x+2$.



$y = x + 2$

w.r.t $y = mx + c$

$m = 1$, $c = 2$

so,

$m = \tan\theta \Rightarrow \theta = \tan^{-1} m = \tan^{-1} 1 = 45^\circ$

Here, $(h, k) = (0, 2)$, $x' = -x$, $y' = y$

$$P' = T_v \cdot R_{(-\theta)} M_y R_\theta T_{(-v)} \cdot P$$

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

$$\begin{bmatrix} 1 & 0 & -h \\ 0 & 1 & -k \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} -1 & 0 & 5 \\ 0 & -2 & 0 \\ 1 & 1 & 1 \end{bmatrix}$$

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

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & -2 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} -1 & 0 & 5 \\ 0 & -2 & 0 \\ 1 & 1 & 1 \end{bmatrix}$$

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1/\sqrt{2} & 1/\sqrt{2} & 0 \\ -1/\sqrt{2} & 1/\sqrt{2} & 2 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} -1/\sqrt{2} & 1/\sqrt{2} & 0 \\ 1/\sqrt{2} & 1/\sqrt{2} & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} -1 & 0 & 5 \\ -2 & -4 & -2 \\ 1 & 1 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 2 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} -1 & 0 & 5 \\ -2 & -4 & -2 \\ 1 & 1 & 1 \end{bmatrix} = \begin{bmatrix} -2 & -4 & -2 \\ 1 & 2 & 7 \\ 1 & 1 & 1 \end{bmatrix}$$

Q. Prove that the 2D transformation with rotation about the origin is commutative $R_1 . R_2 = R_2 . R_1$

$$R_{\theta_1} . R_{\theta_2} = R_{\theta_2} . R_{\theta_1}$$

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

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

Q. Reflect the polygon with vertices $A(-1,0)$, $B(0,-2)$, $C(0,1)$, $D(0,2)$ about the line:

(i) $y = 2$ (ii) $x = 2$ (iii) $y = x+2$

(i) $y = 2$

$P' = T_v \, M_y \, T_{(-v)} . P$, $(h, K) = (0, 2)$, $x' = x$, $y' = -y$

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

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

$$= \begin{bmatrix} -1 & 0 & 0 & 0 \\ 4 & 6 & 3 & 2 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

$A'(-1,4)$, $B'(0,6)$, $C'(0,3)$, $D'(0,2)$

(ii) $x = 2$

$P' = T_v M_y T_{(-v)} . P$, $(h, K) = (2, 0)$, $x' = -x$, $y' = y$

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

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

$$= \begin{bmatrix} 5 & 4 & 4 & 4 \\ 0 & -2 & 1 & 2 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

$A'(5, 0)$, $B'(4, -2)$, $C'(4, 1)$, $D'(4, 2)$

(iii) $y = x + 2$

Here, $m = 1$, $c = 2$, $\theta = 45^\circ$, $(h, K) = (0, 2)$, $x' = -x$, $y' = y$

$P' = T_v R_{(-\theta)} M_y R_\theta T_{(-v)} . P$

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & h \\ 0 & 1 & K \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & -h \\ 0 & 1 & -K \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} -1 & 0 & 0 & 0 \\ 0 & -2 & 1 & 2 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

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

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

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 2 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} -1 & 0 & 0 & 0 \\ -2 & -4 & -1 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

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

$A'(-2,1)$, $B'(-4,2)$
$C'(-1,2)$, $D'(0,2)$

Q. Reflect the ~~the~~ $\triangle ABC$ about the line $3x-4y+8=0$. The position vector of the coordinate ABC is given as $A(4,1)$, $B(5,2)$, $C(4,3)$.

Given : $3x-4y+8=0 \Rightarrow 4y = 3x+8 \Rightarrow y = 3/4\, x + 2$

$m = 3/4$, $C = 2$, $(h,K) = (0,2)$

$\theta = \tan^{-1} m = \tan^{-1} 3/4 = 36.87$, $x' = -x$, $y' = y$

$P' = T_V\, R_{(-\theta)}\, M_y\, R_\theta\, T_{(-V)} \cdot P$

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & h \\ 0 & 1 & K \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$
$$\begin{bmatrix} 1 & 0 & -h \\ 0 & 1 & -K \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 4 & 5 & 4 \\ 1 & 2 & 3 \\ 1 & 1 & 1 \end{bmatrix}$$

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 2 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0.8 & 0.6 & 0 \\ -0.6 & 0.8 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0.8 & -0.6 & 0 \\ 0.6 & 0.8 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$
$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & -2 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 4 & 5 & 4 \\ 1 & 2 & 3 \\ 1 & 1 & 1 \end{bmatrix}$$

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 0.8 & 0.6 & 0 \\ -0.6 & 0.8 & 2 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} -0.8 & 0.6 & 0 \\ 0.6 & 0.8 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 4 & 5 & 4 \\ -1 & 0 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} -0.28 & 0.96 & 0 \\ 0.96 & 0.28 & 2 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 4 & 5 & 4 \\ -1 & 0 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} -2.08 & -1.4 & -0.16 \\ 5.56 & 6.8 & 6.12 \\ 1 & 1 & 1 \end{bmatrix}$$

Q. Find the transformation that transform a given square ABCD to half its size with centre still remaining at same position. The coordinates of square are A(1,1), B(3,1), C(3,3), D(1,3) at the point (2,2).

Here, $S_x = 1/2$, $S_y = 1/2$

$P' = T_V \, S_{s_x s_y} \, T_{(-V)} \cdot P$

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

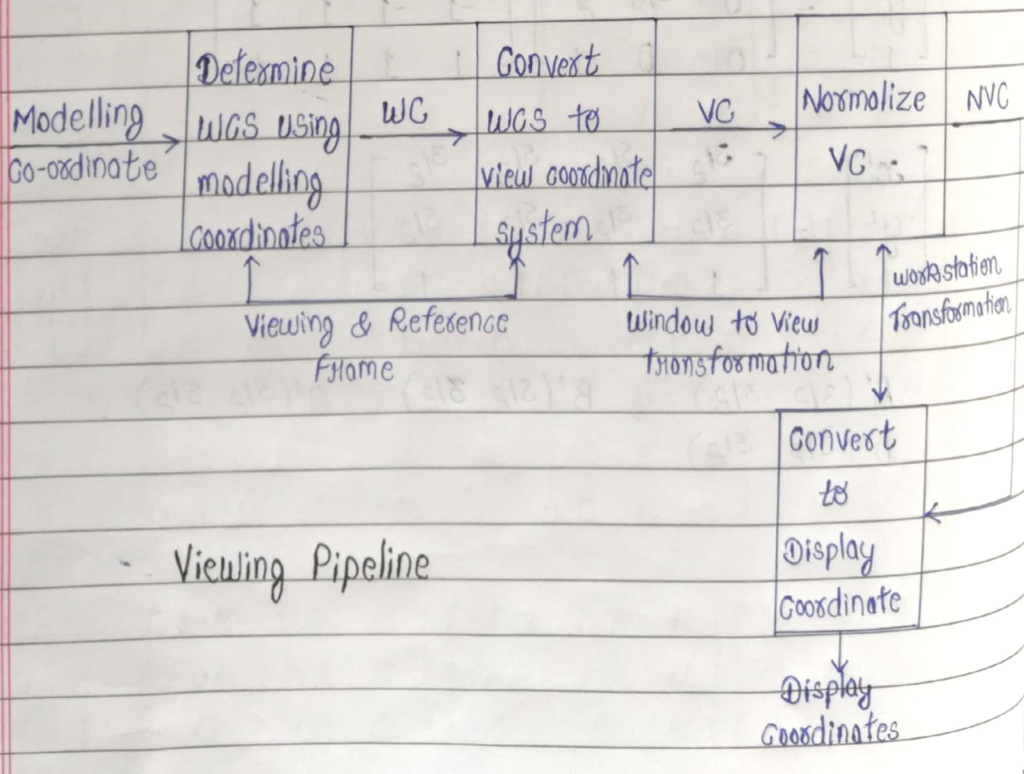
$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1/2 & 0 & 2 \\ 0 & 1/2 & 2 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} -1 & 1 & 1 & -1 \\ -1 & -1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 3/2 & 5/2 & 5/2 & 3/2 \\ 3/2 & 3/2 & 5/2 & 5/2 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

$A'(3/2, 3/2)$, $B'(5/2, 3/2)$, $C'(5/2, 5/2)$, $D'(3/2, 5/2)$

Windowing & Clipping:

The image is stored in computer memory according to the world coordinate system & that image is to be displayed on the display device according to the physical device coordinate system. The mapping of world coordinate system into the physical device coordinate system for displaying the images, it is called viewing transformation:
The 2D viewing transformation is also called window to view code transformation, when we select the graphical images with different views this process is called windowing. And the images is divided into visible & unvisible format on the device, the unvisible device is discarded. This process is called clipping.
The finite area of the world coordinate system for viewing transformation is called window & images is to display on a display device that area is called view port. In other words window defines what part of the image is to be displayed & view port defines where the image is to be displayed.
The overall process is called viewing pipeline.



Viewing Pipeline

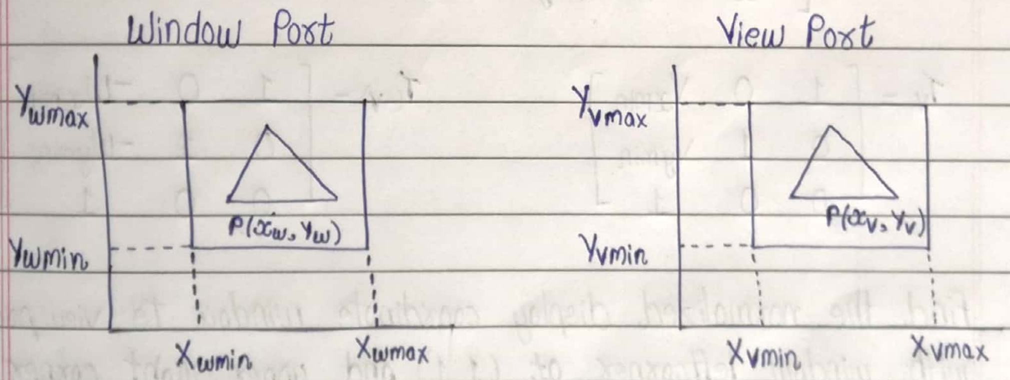
Step 1: Convert the world coordinate system (WCS) to viewing coordinate (VC) system.
Step 2: The viewing coordinate (VC) is again converted into normalize viewing coordinate (NVC).
Step 3: The normalize viewing coordinate (NVC) is converted into display device coordinate.

Viewing Reference Frame:
It is established for converting the WCS to VCS. In this both of coordinate system (WCS & VCS) by the rotation technique in anti-clockwise direction, so this process is called viewing reference frame and transform matrix is written as

$$[N_{WCVC} = TR]$$

where, T - Translation of VC to origin
R - Rotation of WC & VC.

Window to View Port Transformation:



Relative position is same in both:

$$\frac{x_w - x_{wmin}}{x_{wmax} - x_{wmin}} = \frac{x_v - x_{vmin}}{x_{vmax} - x_{vmin}}$$

$$\frac{y_w - y_{wmin}}{y_{wmax} - y_{wmin}} = \frac{y_v - y_{vmin}}{y_{vmax} - y_{vmin}}$$

Find out view port coordinate :

$$x_v = \left(\frac{x_w - x_{wmin}}{x_{wmax} - x_{wmin}}\right) \times (x_{vmax} - x_{vmin}) + x_{vmin}$$

$$x_v = (x_w - x_{wmin})\left(\frac{x_{vmax} - x_{vmin}}{x_{wmax} - x_{wmin}}\right) + x_{vmin}$$

$$\begin{bmatrix} x_v = (x_w - x_{wmin}) . S_x + x_{vmin} \\ y_v = (y_w - y_{wmin}) . S_y + y_{vmin} \end{bmatrix}$$

Work Station Transformation :

When the normalized VC is generated, convert to display device coordinate. This process is called WST.

Their are some following steps for WST :

Step 1 : Translate the WC to the origin according to view port coordinate.

Step 2 : Scale both objects and window by the scaling function factor S_x & S_y.

Step 3: Translate back to the view port position according to WC.

$$[N_{DC} = T_V . S_x S_y . T_{-V}]$$

$$T_V = \begin{bmatrix} 1 & 0 & V_{xmin} \\ 0 & 1 & V_{ymin} \\ 0 & 0 & 1 \end{bmatrix} \qquad T_{(-V)} = \begin{bmatrix} 1 & 0 & -W_{xmin} \\ 0 & 1 & -W_{ymin} \\ 0 & 0 & 1 \end{bmatrix}$$

Q. Find the normalized display coordinate window to view port with window leftcorner at (1,1) and upper right corner at (3,5) on to view port with lower left corner at (0,0) & upper right corner at (1/2, 1/2)

$x_{wmin} = 1$ $\qquad$ $x_{wmax} = 3$

$y_{wmin} = 1$ $\qquad$ $y_{wmax} = 5$

$x_{vmin} = 0$ $\qquad$ $x_{vmax} = 1/2$

$y_{vmin} = 0$ $\qquad$ $y_{vmax} = 1/2$

$$S_x = \frac{1/2 - 0}{3 - 1} = \frac{1/2}{2} = \frac{1}{4}$$

$$S_y = \frac{1/2 - 0}{5 - 1} = \frac{1/2}{4} = \frac{1}{8}$$

$$N_{DC} = T_V \cdot S_x \cdot S_y \; T_{(-V)}$$

$$N_{DC} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1/4 & 0 & 0 \\ 0 & 1/8 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & -1 \\ 0 & 1 & -1 \\ 0 & 0 & 1 \end{bmatrix}$$

$$N_{DC} = \begin{bmatrix} 1/4 & 0 & -1/4 \\ 0 & 1/8 & -1/8 \\ 0 & 0 & 1 \end{bmatrix}$$

Q. Find the normalize transformation window to view code with window left corner (1, 1) & upper right corner (3, 5) onto the normalize view code design screen.

$x_{wmin} = 1$ $\quad$ $x_{wmax} = 3$

$y_{wmin} = 1$ $\quad$ $y_{wmax} = 5$

$x_{vmin} = 0$ $\quad$ $x_{vmax} = 1$

$y_{vmin} = 0$ $\quad$ $y_{vmax} = 1$

$$S_x = \frac{1-0}{3-1} = \frac{1}{2} \qquad S_y = \frac{1-0}{5-1} = \frac{1}{4}$$

$$N_{DC} = T_V \; S_x \cdot S_y \; T_{(-V)}$$

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

$$N_{DC} = \begin{bmatrix} 1/2 & 0 & -1/2 \\ 0 & 1/4 & -1/4 \\ 0 & 0 & 1 \end{bmatrix}$$