Chapter 3 (10 Marks)

b

4. Clip the line with end point A (5, 30), B(20, 60) Against a clip window with lower most corner at P (10, 10) and upper right most corner at R (100, 100) using cohen sulter land line clipping algorithms. [4]
5. Derive the two dimensional reflection matrix through the line y = x+1. What are the finial co-ordinates of objects (2, 3) (4, 3),(4, 5) about line y = x+1? [6]

4. Prove that two successive rotation operations are additive in composite transformation. Find the clipped region in the window of diagonal vertex (10,10) and (100,100) for line P1 (5,120) and P2 (80,7) using Liang –Barsky Line Clipping Algorithms. [7]
5. What is the importance of window to view port transformation in computer Graphics? Explain two-dimensional viewing pipeline. [4]

b

3. Explain 2-D viewing pipeline. Obtain window to viewport transformation matrix with necessary steps and figures. Give example. [3+4+3]

circle algorithm. [2+6]

3. What do you mean by homogenous coordinates? By listing the steps involved, find out the final composite matrix for performing a rotation by 45 degrees about an arbitrary point (5, 5) in anti-clockwise direction. Use the obtained composite matrix to obtain the transformed coordinates of a triangle A(5, 6), B(6, 2) and C(4, 1). [1+5+2]

3. Derive the composite matrix for rotation about arbitrary point (a, b) in clockwise direction with angle (θ). Write an algorithm for Cohen Sutherland line clipping algorithm. [6+4]

4. Reflected the triangle ABC about the line 3X - 4Y + 8 = 0. The position vector of the coordinate ABC is given A(4, 1), B(5, 2) and C(4, 3). [8]

3. Define window and view port. Describe about two-dimensional viewing pipeline with matrix representation at each steps. [2+8]

3. What do you mean by homogeneous coordinates? Rotate a triangle A(5,6), B(6,2) and C(4,1) by 45 degree about an arbitrary pivot point (3,3). [2+6]

3. Write matrix for 2D reflection about axes. Derive the transformation matrix responsible for the reflection of 2D object about line y+x=0. [2+6]

3. Use Liang Barsky line clipping algorithm to clip a line starting from (6,100) and ending at (60, 5) against the window having its lower left corner at (10, 10) and upper right corner at (90, 90). [8]
4. Reflect the triangle ABC about the line 3X-4Y+8=0 the position Vector of coordinate ABC as A(4, 1), B(5, 2) and C(4, 3). [8]

3. Explain Sutherland-Cohen clipping algorithm with an example. [8]

b

3. Clip the line P1P2 with P1(-5,3) and P2(15,9) with clip window having diagonal coordinate (0,0) and (10,10) using Liang-Barskey line clipping method. [8]

b

3. What are the different steps of two dimensional world to screen viewing transformation? Describe with matrix representation at each steps. [5]

4. Obtain the end points of the line that connects P1(0,120) and P2(130,5) after cohen-sutheland clipping. The clip window has the following parameters. [5]

$x\omega_{min} = 0,\ y\omega_{min} = 0,\ x\omega_{max} = 150$ and $y\omega_{max} = 100$

3. What are the conditions for a point clipping? Find the clipped region of the line with endpoins (5, 130) and (50, 5) in a rectangular window with (10, 10) and (100, 100) diagonal vertices using Cohen-Sutherland line clipping algorithm. [10]

b

2. Write down the condition for point clipping. Find the clipped region in window of diagonal vertex (10,10) and (100,100) for line P_1 (5,120) and P_2 (80,7) using Liang-Barsky line clipping method. [2+8]

3. Find the transformation matrix the transforms that rectangle ABCD whose center is at (4,2) is reduced to half of its size, the center will remain same. The co-ordinate of ABCD are A(0,0), B(0,4), C(8,4) and D(8,0). Find Coordinate of new square. Also derive the transformation matrix to convert this rectangle to square. [10]

3. The reflection along the line y = x is equivalent to the reflection along the X-axis followed by counter clock wise rotation by α (alpha) Degree. Find the angle α. [10]

4. Write rotation matrix in clockwise direction with respect to x-axis, y-axis and z-axis. Rotate the object (0, 0, 0), (2, 3, 0), (5, 0, 4) about the rotation axis y =4. [3+7]

3. Given a clipping window A (10, 10), B (40,40), C(40,40) and D(10,40). Using cohen-sutherland line clipping algorithm find region code of each end points of lines P1P2, P3P4 and P5P6 where co-ordinates are P1 (5,15), P2(25,30), P3(15,15), P4(35,30), P5(5,8) and P6(40,15). Also find clipped lines using above parameters. [10]

4. Perform rotation of a line (10, 10, 10), (20, 20, 15) about Y-axis in clock wise direction by 90 degree. Explain about vector display. [6+4]

3. **The reflection along the line y = x is equivalent to the reflection along the X-axis followed by counter clock wise rotation by α (alpha) Degree. Find the angle α. [10]**

3. State the conditions of point clipping. Perform clipping operation for the following using Liang Barskey line clipping algotithm: [2+6]

Clipping window: (Xmin, Ymin) = (2,5) and (Xmin, Ymin) = (35,50)

Line: (x1, y1) = (-2,2) and (x2,y2) = (45,40)

3. find the composite transformation matrix for reflection about a line y = mx+c. [8]

3. Perform scaling transformation to the triangle with vertices A (6, 9), B (10, 5), C (4, 3) with scaling factors $S_x = 3$ and $S_y = 2$. [Show the necessary transformation matrix] [10]

5. Formulate a matrix that converts 2-D scene described in world coordinates to viewing coordinates. [10]

3. Which transformation converts a square to a rhombus? Obtain reflection matrix to reflect a point about the line y = x. [3+7]

3. Justify with necessary matrix operations that the two successive rotations in 2-D is additive.

Types of Transformations

- → Translation
- → Scaling
- → Rotation
- → Reflection
- → Shearing.

Translation / Shifting

To translate a point from co-ordinate position (x, y) to another (x_1, y_1) we add translation distances T_x and T_y to original cordinate

$$x_1 = x + T_x \qquad \begin{bmatrix} x_1 \\ y_1 \end{bmatrix} = \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix}$$
$$y_1 = y + T_y$$

→ (T_x, T_y) is called shift vector

matrix for translation → forwardward ma +ve
backward ma −ve

$x_1 = x + t_x$
↑
original position ma leung

$$\begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ t_x & t_y & 1 \end{bmatrix} \quad \text{or} \quad \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix}$$

$$\boxed{P' = P + T}$$

Scaling

→ alter or change the size of objects

$x' = x \cdot Sx$
$y' = y \cdot Sy$

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

$P' = S \cdot P$

If $Sx = Sy$ (uniform scaling) $Sx \neq Sy$ (differential scaling)

If $(Sx, Sy) < 1$ then size is reduced → ~~...~~
$= 1$ same
> 1 enlarged

If scaling factor is less than 1 then it moves the object closer to origin. If scaling factor is greater than 1, it move the objects away from origin.

Matrix for scaling:

$$S = \begin{bmatrix} Sx & 0 & 0 \\ 0 & Sy & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$Sx = Sy = 2$ (image enlarged 2 times)

Example:

Fixed point scaling

The location of the scaled object can be controlled by choosing a position called fixed point that is to remain unchanged after the scaling transformation. Fixed point (x_f, y_f) can be chosen as one of the vertices, centroid of the object, or any other position.

Steps:

1. Translate object so that the fixed point coincides with the co-ordinate origin.
2. Scale the object with respect to the co-ordinate origin.
3. Use the inverse translation of steps 1 to return the object to its original position.

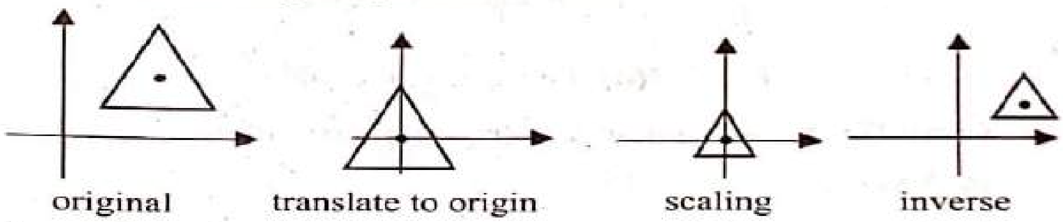


Fig 3.2: *Fixed point scaling of a triangle*

$$\begin{bmatrix} 1 & 0 & x_f \\ 0 & 1 & y_f \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & -x_f \\ 0 & 1 & -y_f \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} s_x & 0 & x_f(1-s_x) \\ 0 & s_y & y_f(1-s_y) \\ 0 & 0 & 1 \end{bmatrix}$$

(normally kehi bhany chaina bhane ~~anticlockwi~~ anticlock wise rotation)

So,

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

θ +ve $\rightarrow$ anticlockwise
(This derivation)

θ -ve $\rightarrow$ clockwise

For anticlockwise

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

$$R = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

For clockwise direction
θ = -ve

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

$$R = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

2. Rotation of a point about an arbitrary pivot position

1. Translation object so that pivot point is moved to co ordinate origin.
2. Rotate object about origin.
3. Translate object so that pivot point is returned to it original position.

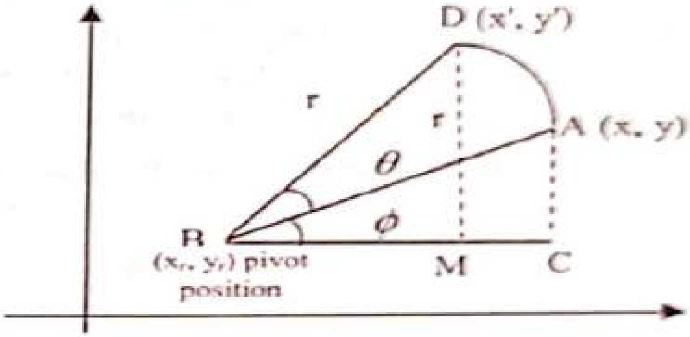


Fig. 3.3(b): Rotation of a point

Composite matrix, C.M. = T'R(θ)T

$$= \begin{bmatrix} 1 & 0 & x_r \\ 0 & 1 & y_r \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & -x_r \\ 0 & 1 & -y_r \\ 0 & 0 & 1 \end{bmatrix}$$

Homogeneous cordinates

→ Homogeneous co-ordinates are the co-ordinates that allows us to represent all ~~geom~~ geometric transformation equation as matrix multiplication.

In 2D position is represented with homogeneous co-ordinates $(x, y, 1)$ → 1 is used for convenience

eg: For translation

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

rotation

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

Scaling

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

not the one above is wrong

1. Translation

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

Date ____
Page ____

Rotate a line CD whose endpoints are (3,4) and C(12,15) about origin through 45° anticlockwise direction.

solution,

$$R = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \qquad \theta = 45^\circ$$

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

For A (3,4)

$$A' = \begin{bmatrix} 1/\sqrt{2} & -1/\sqrt{2} \\ 1/\sqrt{2} & 1/\sqrt{2} \end{bmatrix} \begin{bmatrix} 3 \\ 4 \end{bmatrix}$$

$$= \begin{bmatrix} -0.7071 \\ 4.9497 \end{bmatrix}$$

B (12,15)

$$B' = \begin{bmatrix} -2.121 \\ 19.081 \end{bmatrix}$$

Date ____
Page ____

Composite Transformations (Godhacha)

Two successive translations are additive

$$\begin{bmatrix} 1 & 0 & t_{x2} \\ 0 & 1 & t_{y2} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & t_{x1} \\ 0 & 1 & t_{y2} \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & t_{x1}+t_{x2} \\ 0 & 1 & t_{y1}+t_{y2} \\ 0 & 0 & 1 \end{bmatrix}$$

successive rotations are additive

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

Two successive scaling are multiplicative

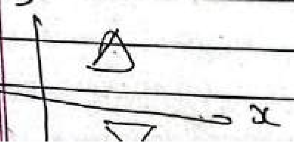
$$\begin{bmatrix} s_{x2} & 0 & 0 \\ 0 & s_{y2} & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} s_{x1} & 0 & 0 \\ 0 & s_{y1} & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} s_{x1}\cdot s_{x2} & 0 & 0 \\ 0 & s_{y1}\cdot s_{y2} & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Reflection

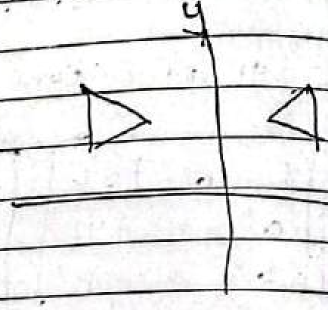
(i) Reflection about x-axis or about line $y = 0$

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

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



c ii) Reflection about y-axis or about line $x = 0$



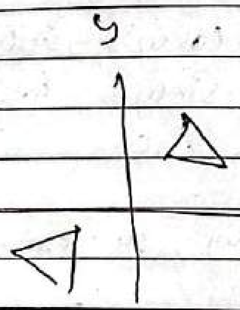
$x' = -x$

$y' = y$

$P' = R_{fy} P$

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

c iii) Reflection about origin.



$x' = -x$

$y' = -y$

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

c iv) Reflection about the line $y = x$

1. Rotate about origin in clockwise direction by $45°$
2. reflect against x-axis
3. Rotate in anti-clockwise direction by same angle.

$$R_{f(y=x)} = \underbrace{R(\theta)^{-1}}_{\text{anticlockwise}} \times \underbrace{R_{fx}}_{x\text{-axis}} \times \underbrace{R(\theta)}_{\text{clockwise}}$$

$(\theta = 45°)$

F

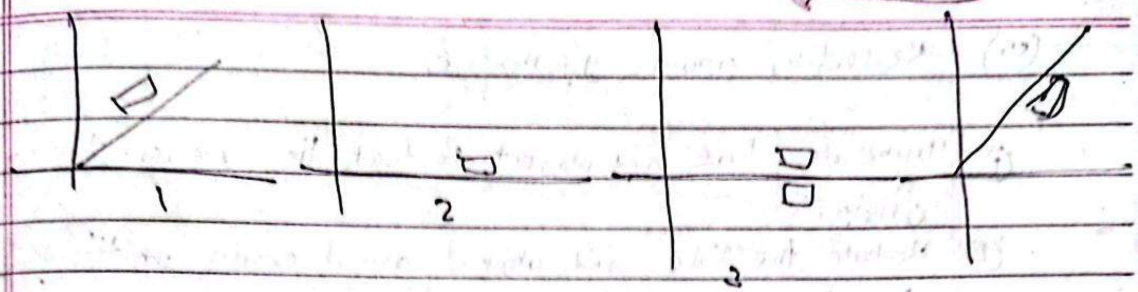


fig: reflection about $y = x$

$$R(\theta)^{-1} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad R_{fx} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$R(\theta) = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \sin\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

v) Reflection about the line $y = -x$

① Rotate about origin in anti-clockwise direction by 45°.
② ~~Take reflection in rotated line y=x to y-axis~~
② Take reflection against y-axis
③ Rotate in clockwise direction by same angle

$$R_f(y=-x) = R(\theta)^{-1} \times R_{fy} \times R(\theta) \qquad \theta = 45^\circ$$

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

(vi) Reflection about $y = mx + c$

① Translate line and object so that line passes through origin.

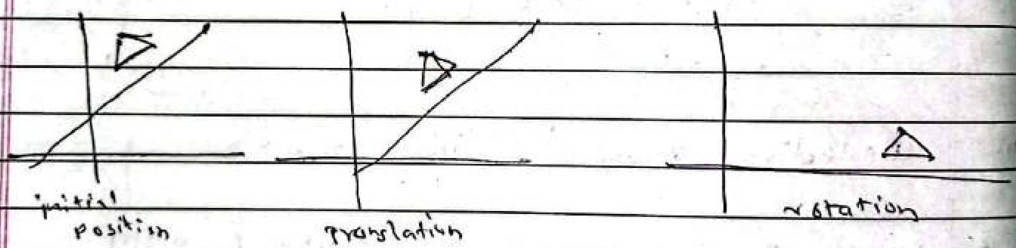
② Rotate the line and object about origin ~~until~~ until the line coincides with one of the co-ordinate axis.

③ Reflect the object through about that axis.

④ Apply inverse rotation about that axis.

⑤ Translate back to original location.

$$CM = T^{-1} . R(\theta)^{-1} . Rf . R(\theta) . T \qquad \theta = \tan^{-1}(m)$$



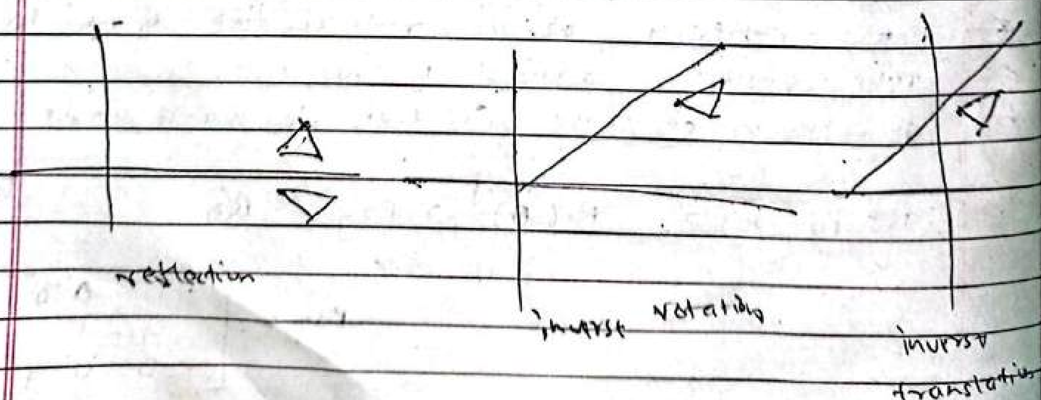


fig: Reflection about the line $y = mx + c$

$t_x = 0$
$t_y = c$

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

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

$t_x = 0$
$t_y = -c$

Scale an object (4,4), (3,2), (5,2) about a fixed point (4,3) by 2

$P' = CM \times P$

Translate the object so that the fixed point become origin
Scale the object
Translate the object back to its original position

$$C.M = \begin{bmatrix} 1 & 0 & 4 \\ 0 & 1 & 3 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & -4 \\ 0 & 1 & -3 \\ 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 0 & 4 \\ 0 & 1 & 3 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 2 & 0 & -8 \\ 0 & 2 & -6 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 2 & 0 & -4 \\ 0 & 2 & -3 \\ 0 & 0 & 1 \end{bmatrix}$$

$$P = \begin{bmatrix} 4 & 3 & 5 \\ 4 & 2 & 2 \\ 1 & 1 & 1 \end{bmatrix} \begin{array}{l} \rightarrow x\text{-coordinate} \\ \rightarrow y\text{-coordinate} \\ \end{array}$$

$$P' = CM \times P$$

$$= \begin{bmatrix} 2 & 0 & -4 \\ 0 & 2 & -3 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 4 & 3 & 5 \\ 4 & 2 & 2 \\ 1 & 1 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 4 & 2 & 6 \\ 5 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

final points $(4,5)$, $(2,1)$ and $(6,1)$

Rotate the triangle $(5,5)$ $(7,3)$ $(3,3)$ about fixed point $(5,4)$ in counter clockwise by 90°

$$C.M = T_{(5,4)}\, R_{90}\, T_{(-5,-4)}$$

$$= \begin{bmatrix} 1 & 0 & 5 \\ 0 & 1 & 4 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos 90^\circ & -\sin 90^\circ & 0 \\ \sin 90^\circ & \cos 90^\circ & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & -5 \\ 0 & 1 & -4 \\ 0 & 0 & 1 \end{bmatrix}$$

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

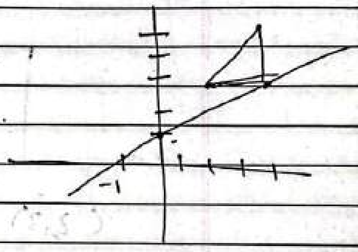
$$P' = C.M \times P$$

$$= \begin{bmatrix} 0 & -1 & 9 \\ 1 & 0 & -1 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 5 & 7 & 3 \\ 5 & 3 & 3 \\ 1 & 1 & 1 \end{bmatrix} = \begin{bmatrix} 4 & 6 & 6 \\ 4 & 6 & 2 \\ 1 & 1 & 1 \end{bmatrix}$$

(4,6), (6,6), (6,2) are new coordinates

(9) Reflect an object (2,3) (4,3) (4,5) about line $y = x+1$

$m = 1$, $c = 1$



C ⟹ $m = \tan\theta$

$1 = \tan\theta \quad \therefore \theta = 45^\circ$

① Translate with value of c
② Rotate by angle $\tan^{-1}(m)$
③ Reflect about x-axis
④ re rotate
⑤ Translate

$$C.M = T' \, R_\theta' \, R_{fx} \, R_\theta \, T$$

$$C.M = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos45^\circ & -\sin45^\circ & 0 \\ \sin45^\circ & \cos45^\circ & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos45^\circ & \sin45^\circ & 0 \\ -\sin45^\circ & \cos45^\circ & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & -1 \\ 0 & 0 & 1 \end{bmatrix}$$

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

$$P' = C.M \times P$$

$$= \begin{bmatrix} 0 & 1 & -1 \\ 1 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cancel{2\ 3\ 4}\ 2 & 4 & 4 \\ 3 & 3 & 5 \\ 1 & 1 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 2 & 2 & 4 \\ 3 & 5 & 5 \\ 1 & 1 & 1 \end{bmatrix}$$

(2,3), (2,5) and (4,5)

Reflect an object (2,3), (4,3) and (4,5) about $y = 2x + 1$

$m = 2$

$c = 1$

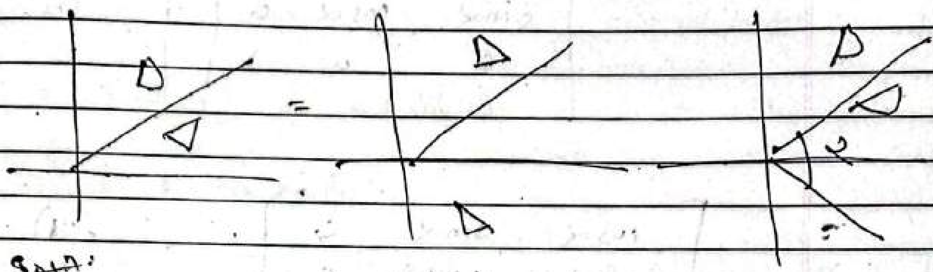
$\theta = \tan^{-1}(m) = \tan^{-1}(2) = 63.43^\circ$

$$C.M = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos 63.43^\circ & -\sin 63.43 & 0 \\ \sin 63.43^\circ & \cos 63.43^\circ & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & - \\ 0 & \end{bmatrix}$$

$$\begin{bmatrix} \cos 63.43^\circ & \sin 63.43 & 0 \\ -\sin 63.43^\circ & \cos 63.43 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & -1 \\ 0 & 0 & 1 \end{bmatrix}$$

Similar as above #

The reflection along the line $y = x$ is equivalent to reflection along x-axis followed by counter clockwise rotation by α degree? Find α



Soln:

~~(i) Rotate about origin in clockwise direction~~

Steps for reflection about line $y = x$

(i) rotate about origin in clockwise by 45°

(ii) reflection about x-axis

(iii) rotate in anticlock direction in 45°.

$$Rf(y=x) = R(\theta)^{-1} \times Rf_x \times R_\theta$$

$$= \begin{bmatrix} \cos 45 & -\sin 45 & 0 \\ \sin 45 & \cos 45 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos 45 & \sin 45 & 0 \\ -\sin 45 & \cos 45 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \text{ --- (i)}$$

Again the matrix for reflection along x-axis followed by counter clockwise rotation by α degree is :-

$$C.m^{a} = \begin{bmatrix} \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} \cos\alpha & \sin\alpha & 0 \\ \sin\alpha & -\cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad \text{--- (ii)}$$

We know from qn that eq (i) and (ii) are same matrix

So, $\cos\alpha = 0$

$\alpha = \cos^{-1}(0)$

$\therefore \alpha = 90°$

Reflect the Δ ABC about the line $3x - 4y + 8 = 0$ the position vector of coordinate ABC as A(4,1), B(5,2) and C(4,3)

Soln:

$4y = 3x + 8$

$y = \frac{3}{4}x + 2$

$m = \frac{3}{4}$, $c = 2$

$\theta = \tan^{-1}(m) = 36.8698^\circ$

$C.M = T^{-1} R_\theta^{-1} R_{fx} R_\theta T$

$$= \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 2 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos 36.8698^\circ & -\sin 36.8698^\circ & 0 \\ \sin 36.8698^\circ & \cos 36.8698^\circ & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\begin{bmatrix} \cos 36.8698 & \sin 36.8698 & 0 \\ -\sin 36.8698 & \cos 36.8698 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & -2 \\ 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 0.28 & 0.96 & -1.92 \\ 0.96 & 0.28 & 2.56 \\ 0 & 0 & 1 \end{bmatrix}$$

$P' = C.M \times P$

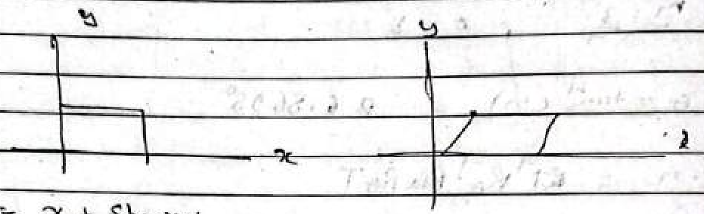
$$= \begin{bmatrix} 0.28 & 0.96 & -1.92 \\ 0.96 & 0.28 & 2.56 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 4 & 5 & 4 \\ 1 & 2 & 3 \\ 1 & 1 & 1 \end{bmatrix} = \begin{bmatrix} 0.16 & 1.4 & 2.08 \\ 0.12 & 6.8 & 5.56 \\ 1 & 1 & 1 \end{bmatrix}$$

$(0.16, 0.12)$ $(1.4, 6.8)$ $(2.08, 5.56)$

ii Shearing (not that important)

⇒ transformation that distorts the shape of the object.

(i) shearing toward x-direction relative to x-axis



$x' = x + sh_x \times y$

$y' = y$

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

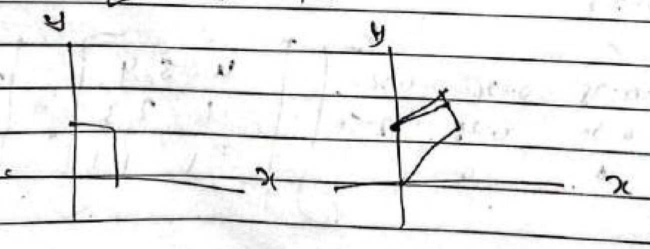
sh_x → +ve (right shearing)

sh_x → -ve (left shearing)

(ii) shearing towards y-direction relative to y-axis

$x' = x$, $y' = x \times sh_y + y$

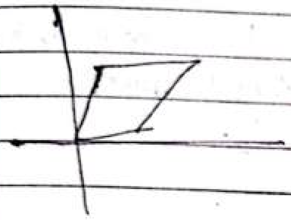
$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ sh_y & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$



(iii) Shearing in both direction is given by

$$x' = x + Shx \times y$$
$$y' = x \times Shy + y$$

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



(2015) (3 marks)

→ Square can be transformed into a rhombus by shearing it in both direction

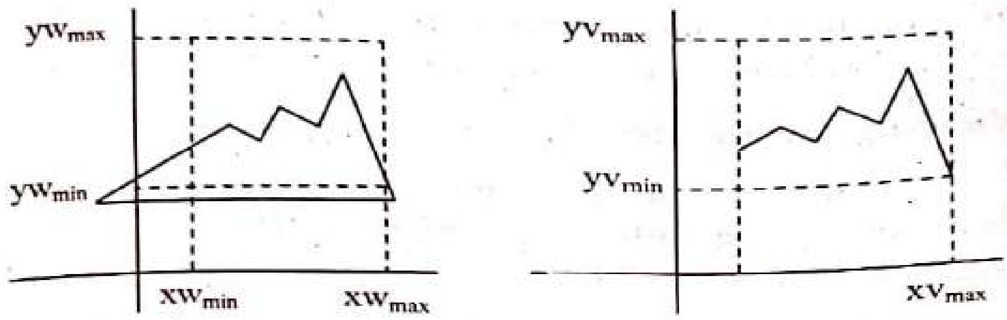


Fig 3.12: A viewing transformation using standard rectangles for the window and viewport.

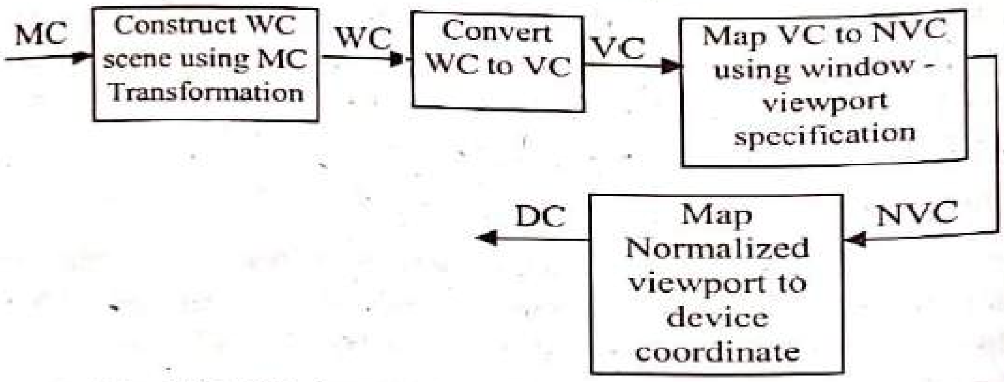


Fig. 3.13: 2D viewing transformation pipeline

Overview of the 2D Viewing Transformation Pipeline

The 2D viewing transformation pipeline is a sequence of coordinate transformations that prepare a 2D scene for display on a device like a computer screen. This pipeline ensures that objects are appropriately transformed from their original model space (defined by the creator) to the device space (where they will be displayed).

Step-by-Step Explanation

1. Model Coordinates (MC) to World Coordinates (WC):

- **Model Coordinates (MC):** This is the coordinate system used by the designer or modeler to define the objects in the scene. It is specific to each object and represents the object's local frame of reference. For example, a square might be defined with its center at (0, 0) in MC.
- **Transformation to World Coordinates (WC):** The first step is to transform the model coordinates into a common coordinate system called World

Coordinates (WC). This transformation involves:

- **Translation:** Moving the object to a new location in the world space.
- **Rotation:** Changing the orientation of the object.
- **Scaling:** Adjusting the size of the object.

The purpose of this transformation is to place all objects into a unified coordinate system, which represents the entire scene as a whole, taking into account the relative positions, orientations, and scales of different objects.

2. World Coordinates (WC) to Viewing Coordinates (VC):

- **World Coordinates (WC):** After the scene is constructed in the world coordinate system, the next step is to convert the world coordinates to viewing coordinates.

- **Viewing Coordinates (VC):** Viewing coordinates are defined relative to a viewer or camera's position and orientation. This step involves:
  - **Defining the View Window:** The window is a rectangular area in the world coordinate system that specifies the region of the world that the viewer wants to see. It is defined by its boundaries, typically denoted as (X_{Wmin}, Y_{Wmin}) and (X_{Wmax}, Y_{Wmax}).
  - **Transforming to VC:** The objects within the view window are transformed to a new coordinate system aligned with the viewer's or camera's perspective. This transformation aligns the window's center to the viewer's line of sight and reorients the axes to align with the viewing direction.

  This step effectively "crops" the world to only include the relevant part that is visible in the defined window, and it reorients the scene so it is properly aligned from the viewer's perspective.

3. Viewing Coordinates (VC) to Normalized Viewport Coordinates (NVC):

- **Viewing Coordinates (VC):** Once we have a scene in viewing coordinates, the next step is to map it to a normalized viewport.
- **Normalized Viewport Coordinates (NVC):**

    - **Viewport Mapping:** The viewport is an area on the display device where the final image will be rendered. The normalized viewport coordinates are a standardized space, often ranging from (0,0) to (1,1) or (-1,-1) to (1,1).
    - **Transformation to NVC:** This step maps the rectangular area of the viewing window (in viewing coordinates) to the normalized viewport. This involves scaling and translating the objects to fit within the normalized space, ensuring the aspect ratio is maintained. This ensures that the portion of the scene visible through the view window is correctly scaled to fit within the viewport, preserving the scene's proportions.

  The normalized viewport serves as an intermediary, ensuring that the aspect ratio and orientation are maintained regardless of the device's resolution or size.

4. Normalized Viewport Coordinates (NVC) to Device Coordinates (DC):

- **Normalized Viewport Coordinates (NVC):** After mapping to the normalized space, the final step is to convert these coordinates to actual device coordinates.
- **Device Coordinates (DC):**
    - **Device-Specific Mapping:** The device coordinates refer to the actual pixel grid on the display device, such as a computer monitor or a smartphone screen. The device coordinates are specific to the resolution and size of the device.
    - **Transformation to DC:** This step involves scaling and translating the normalized viewport coordinates to the specific device's coordinate system. The normalized coordinates (ranging from (0,0) to (1,1)) are multiplied by the device's resolution to place the scene correctly on the screen.

  This step ensures that the scene is rendered correctly on the screen, with all objects appearing in the appropriate positions and scales according to the viewer's perspective.

Summary

The 2D viewing transformation pipeline ensures that objects created in their own local coordinate systems are properly placed, oriented, and scaled to appear correctly in the viewer's perspective on a specific device. This involves multiple transformations:

- from model coordinates to world coordinates (setting the scene in a global context),
- from world coordinates to viewing coordinates (aligning the scene with the viewer’s perspective),
- from viewing coordinates to normalized viewport coordinates (scaling and fitting the scene within a normalized display space),
- and finally, from normalized viewport coordinates to device coordinates (translating the scene to actual screen pixels).

By following this pipeline, we ensure a consistent and accurate representation of the 2D scene, regardless of the viewer's perspective or the device used for display.

Window to ViewPort Mapping / Transformation

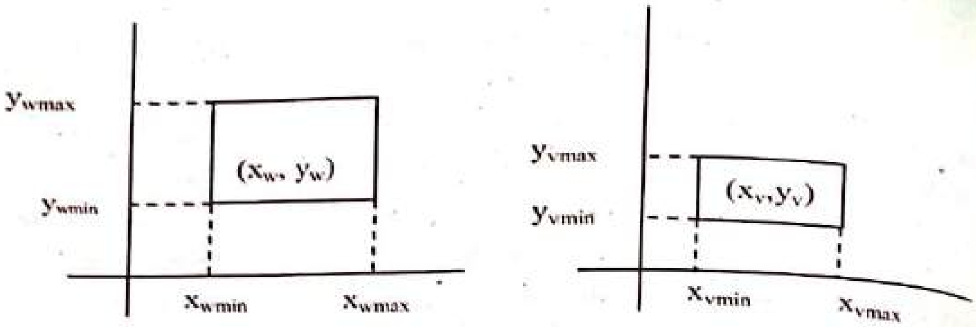


Fig. 3.15: Window to viewport mapping

A window is specified by four world co-ordinates x_{wmin}, x_{wmax}, y_{wmin}, y_{wmax}. A viewport is described by four device co-ordinate x_{vmin}, x_{vmax}, y_{vmin}, y_{vmax}. To maintain the same relative placement in the viewport as in window, we require.

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

And

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

Solving the equation for viewport

$x_v = x_{vmin} + (x_w - x_{wmin})s_x$

$y_v = y_{vmin} + (y_w - y_{wmin})s_y$

where,

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

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

Steps (for Window to Viewport Transformation):

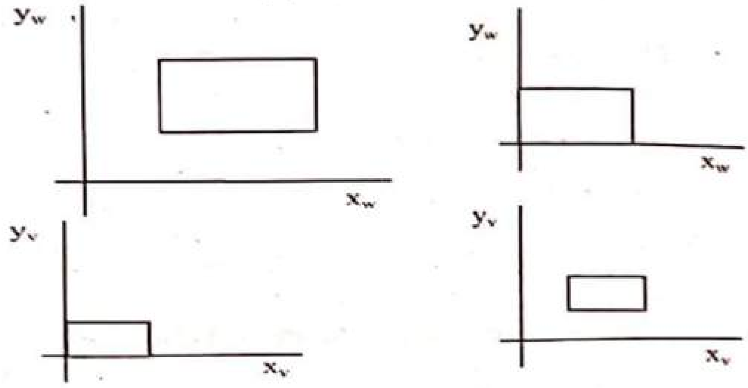


Fig. 3.16: Window to viewport transformation

1. The object together with its window is translated until the lower left corner of the window is at origin.
2. Object and window are scaled until window has dimension of viewport.

   Perform a scaling transformation using a fixed-point position of (x_{wmin}, y_{wmin}) that scales the window area to the size of the viewport
3. Again, translate to move viewport to its correct position.

Viewing Transformation:

1. Translate window to origin by

   $T_x = -x_{wmin}$

   $T_y = -y_{wmin}$
2. Scale window such that its size is matched to viewport.

$$S_x = \frac{x_{v\,max} - x_{v\,min}}{x_{w\,max} - x_{w\,min}}$$

$$S_y = \frac{y_{v\,max} - y_{v\,min}}{y_{w\,max} - y_{w\,min}}$$

3. Retranslate it by

   $T_y = y_{vmin}$ $T_x = x_{v\,min}$

Composite matrix (CM) = $T_v \times S_{wv} \times T_w$

T_w = Translate window to origin = $\begin{bmatrix} 1 & 0 & -x_{w\,min} \\ 0 & 1 & -y_{w\,min} \\ 0 & 0 & 1 \end{bmatrix}$

S_{wv} = Scaling of window to viewport

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

T_V = Translate viewport to original position

$$= \begin{bmatrix} 1 & 0 & x_{v\,min} \\ 0 & 1 & y_{v\,min} \\ 0 & 0 & 1 \end{bmatrix}$$

Example problem

(i) window port is given by (100, 100, 300, 300) and viewport is given by (50, 50, 150, 150). Convert the window port coordinate (200, 200) to the view coordinate.

Given,

$(Xw_{min}, Yw_{min}) = (100, 100)$ $(Xw_{max}, Yw_{max}) = (200, 300)$

$(Xv_{min}, Yv_{min}) = (50, 50)$ $(Xv_{max}, Yv_{max}) = (150, 150)$

$(Xw, Yw) = (200, 200)$

Then,

$$S_x = \frac{Xv_{max} - Xv_{min}}{Xw_{max} - Xw_{min}} \qquad S_y = \frac{Yv_{max} - Yv_{min}}{Yw_{max} - Yw_{min}}$$

$$\therefore S_x = \frac{150 - 50}{300 - 100} \qquad S_y = \frac{150 - 50}{300 - 100}$$

$$= 0.5 \qquad = 0.5$$

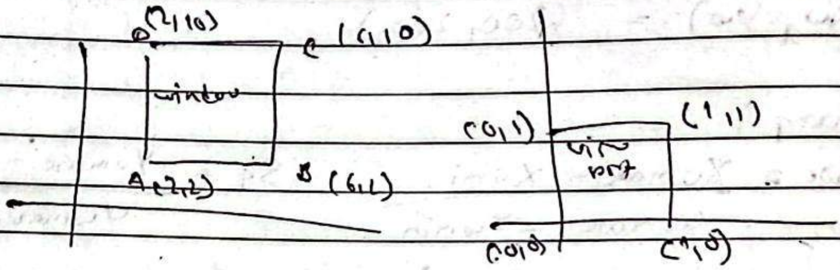
Now,

$$\begin{bmatrix} Xv \\ Yv \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 50 \\ 0 & 1 & 80 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0.5 & 0 & 0 \\ 0 & 0.5 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & -100 \\ 0 & 1 & -100 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 200 \\ 200 \\ 1 \end{bmatrix}$$

$$\therefore \begin{bmatrix} Xv \\ Yv \\ 1 \end{bmatrix} = \begin{bmatrix} 100 \\ 100 \\ 1 \end{bmatrix}$$

$\therefore (Xv, Yv) = (100, 100)$

Find the normalization transformation for window to viewport which uses the rectangle whose lower left corner is at (2,2) and upper right corner is at (6,10) as a window and the viewport that has lower left corner at (0,0) and upper right corner at (1,1)



$$S_x = \frac{X_{vmax} - X_{vmin}}{X_{wmax} - X_{wmin}} \qquad S_y = \frac{Y_{vmax} - Y_{vmin}}{Y_{wmax} - Y_{wmin}}$$

$$= \frac{1-0}{6-2} = 0.25 \qquad = \frac{1-0}{10-2} = 0.125$$

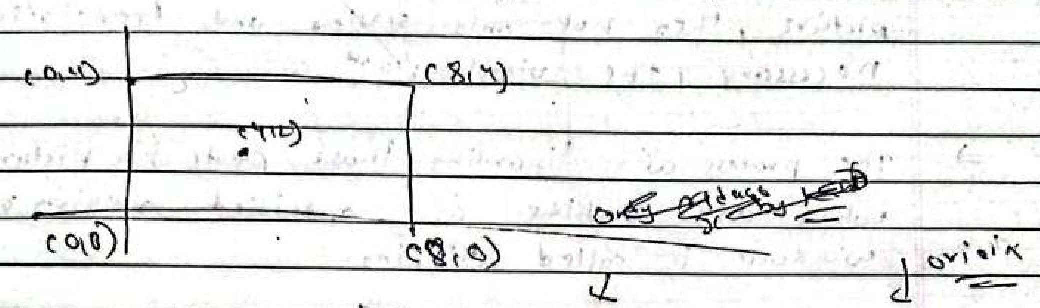
The transformation matrix is

$$T = \begin{bmatrix} S_x & 0 & 0 \\ 0 & S_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix}$$

window to viewport only in this case

$$= \begin{bmatrix} 0.25 & 0 & 0 \\ 0 & 0.125 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & -2 \\ 0 & 1 & -2 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 0.25 & 0 & -0.5 \\ 0 & 0.125 & -0.25 \\ 0 & 0 & 1 \end{bmatrix}$$

Find transformation matrix that transforms rectangle ABCD whose center at (4,2) is reduced to half. co-ordinate are A(0,0), B(0,4), C(8,4), D(8,0)



$$T = \begin{bmatrix} 1 & 0 & 4 \\ 0 & 1 & 2 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0.5 & 0 & 0 \\ 0 & 0.5 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & -4 \\ 0 & 1 & -2 \\ 0 & 0 & 1 \end{bmatrix}$$

$$T = \begin{bmatrix} 0.5 & 0 & 2 \\ 0 & 0.5 & 1 \\ 0 & 0 & 1 \end{bmatrix}$$

$$A' = TA = \begin{bmatrix} 2 \\ 1 \\ 1 \end{bmatrix} \quad (2,1)$$

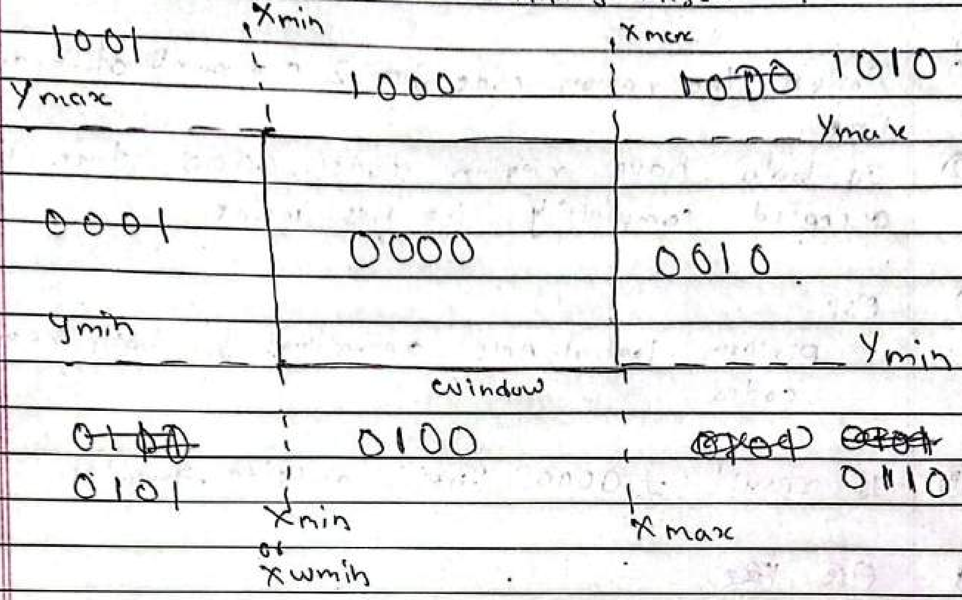
Similarly find out other point

$B' = TB$

$C' = TC$

$D' = TD$

4 Cohen sutherland line Clipping Algorithm



(1) generate TBRL (Region code) → 4 bit code
↓
TOP buttom ~~left~~ right

if $x < x_{min}$ → left part of window
$x > x_{max}$ → point will right of wind.
$y < y_{min}$ → bottom of window
$y > y_{max}$ ⇒ top of window

Algorithm

1. Assign the region code for 2 end points of a given line
2. If both have region codes 0000 then line accepted completely i.e lies inside
3. else
   perform logical AND operation for both region codes
4. If result ≠ 0000 line is outside → invisible
5. else ~~line~~
   result = 0000, then line is partially inside

   (i) choose an ~~en poi~~ endpoint of the line that is outside the given rectangle
   (ii) find ~~the~~ intersection point
   (iii) replace endpoint with intersection point and update region code
   (iv) repeat step from 2 ~~until line~~ until line is trivially accepted or rejected

6. Repeat step (2) for other lines as well

Also:

(i) assign region code to both end points say C0 and C1

(ii) if C0 OR C1 = 0000 then accepted completely (inside window)

else if

C0 AND C1 ≠ 0000 reject it

else (C0 AND C1 = 0000) → partially inside

clip if line crossed Xmin or Xmax

then,

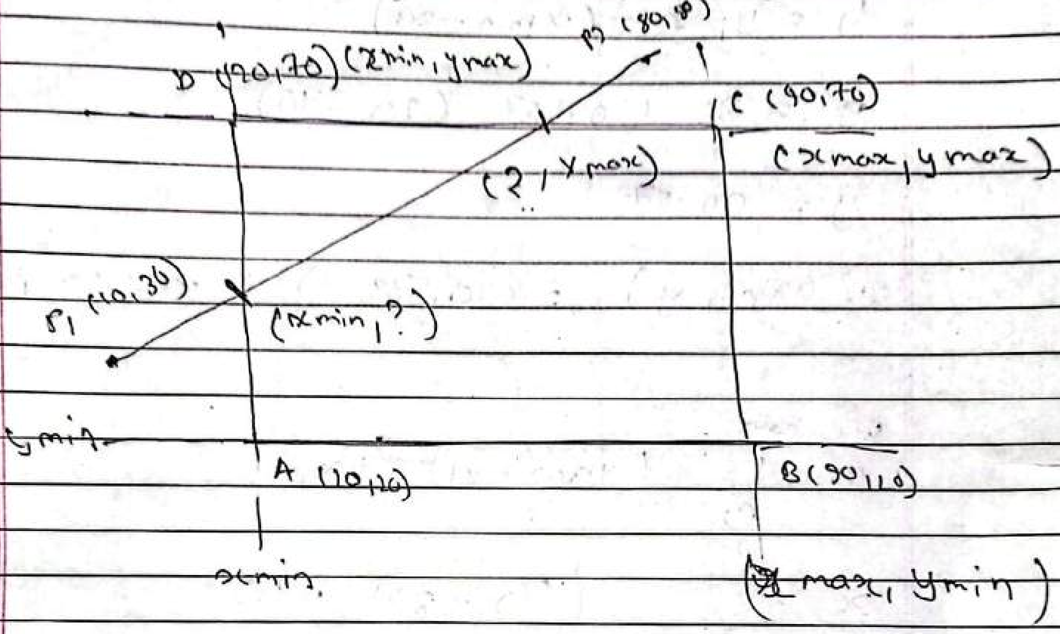
$$y = y_1 + m(x - x_1) \quad \left(x = x_{max} \text{ or } x_{min} \right)$$

else,

$$x = x_1 + \frac{1}{m}(y - y_1) \quad \left(y = y_{max} \text{ or } y_{min} \right)$$

(iii) verify $x_{min} \le x \le x_{max}$, $y_{min} \le y \le y_{max}$ } if it doesn't satisfy then repeat

① let ABCD be the rectangular window with A (20,20), B (90,20), C (90,70) and D (20,70). Use cohen sutherland algo to clip line P_1P_2 when P_1 (10,30) and P_2 (80,90)



(i) find region code TBRL code

P_1 = 0 0 0 1
P_2 = 1 0 0 0

P_1 OR P_2 = 0000 (visible) line

P_1 AND P_2 = ~~0000~~ 0000, line is partially visible slope $m = \frac{y_2 - y_1}{x_2 - x_1} = \frac{90 - 30}{80 - 10} = \frac{60}{70} = 0.857$

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

$y = 30 + 0.857 \, (20 - 10)$

$y = 30.57$

$\therefore (x_{min}, y) = (20, 38.57)$ is intersection point

we have

$x = \frac{1}{m}(y_{max} - y_1) + x_1$

$x = \frac{1}{0.857}(70 - 30) + 10$

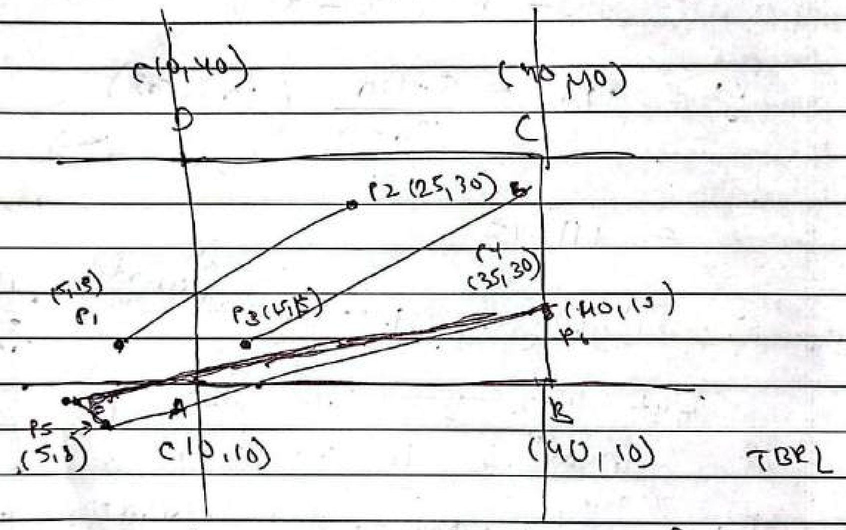
$= 56.57$

$\therefore$ ($\cancel{56.57}$, x, y_{max}) is 2^{nd} intersection point

($\cancel{x, 70}$ 56.57, 70) #

Date ______
Page ______

\# Given a clipping window A(10,10), B(40,40), ~~C(4,4)~~ C(40,40), D(10,40) using cohen sutherland line clipping algorithm find region codes of lines P_1P_2, P_3P_4 and P_5P_6 whose co-ordinates are $P_1(5,15)$ $P_2(25,30)$, $P_3(15,15)$ $P_4(35,30)$ $P_5(5,8)$ and $P_6(40,15)$ also clip the line.



For, $P_1(5,15)$ code → 0001 $P_2(25,30)$ code → 0000

P_3 → 0000 P_4 → 0000

P_5 → 0101 P_6 → 0000

For P_1 and P_2

P_1 OR P_2 = 0001 → line is not completely inside

P_1 AND P_2 = 0000 → line is partially visible

Now, finding intersection

$$m = \frac{y_2 - y_1}{x_2 - x_1} = \frac{30-15}{25-5} = \frac{3}{4}$$

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

$$y = 8 + \frac{3}{4}\,(10 - 5) = 18.75$$

$P_1' = (x_{min}, y) = (10, 18.75)$

P_2 (25, 30)

For P_3 and P_4

P_3 OR P_4 = 0000 (It lies inside)

For P_5 and P_6

P_5 OR P_6 = 0101 → not completely inside

P_5 AND P_6 = 0000 → partially inside

$m = \frac{y_2 - y_1}{x_2 - x_1} = \frac{15-8}{40-5} = \frac{1}{5}$

P5 and P6

0101   0000

TBRL

bottom left

intersection bottom boundary

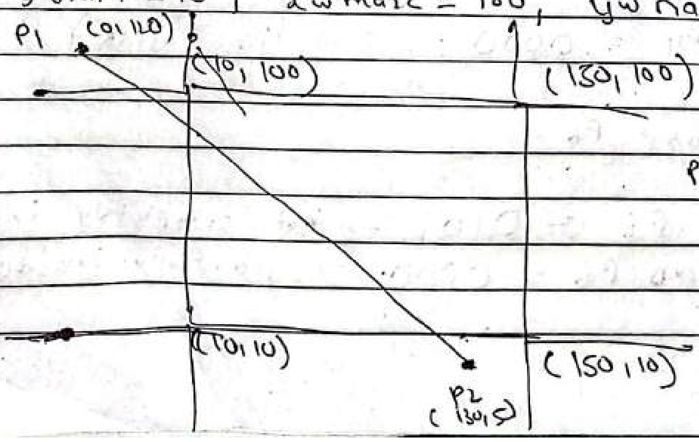
$x = x_1 + \frac{1}{m}(y_{min} - y_1)$

$x = 5 + \cancel{6}\,5\,(10 - 8)$

$x = 15$

$P_3'(x, y_{min}) = P_3'(15, 10)$

P_1 (0, 120) and P_2 (130, 5) ($x_{wmin} = 10$, $y_{wmin} = 10$, $x_{wmax} = 150$, $y_{wmax} = 100$)



TBRL

$P_1 = 1001$

$P_2 = 0100$

P1 OR P2 = 1101 ≠ 0000 so line is not completely inside

P1 AND P2 = 0000 → partially visible needs clipping

For P_1'

$$y = y_1 + m(x - x_1) \qquad (x = x_{min})$$

$$y' = 120 + \frac{5-120}{130-0}\left(10 - \overset{0}{\cancel{0}}\right)$$

→ mistake ayo bhane [illegible] dine

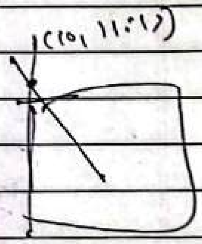
$$= 111.15$$

$$\therefore P_1' (10, 111.15)$$ → not satisfied

~~but max y is 100 so~~

max y is is 100 so,

again, we have to find intersection point



$$x = x_1 + \frac{1}{m}(y - y_1) \qquad y = y_{max}$$

$$= 10 + \frac{1}{\frac{5 - 111.15}{130 - 10}}(100 - 111.15)$$

$$= 22.60$$

$$\therefore P_1' = (x, y_{max}) = (22.60, 100)$$

For P_2'

$$x = x_1 + \frac{1}{m}(y - y_1) \quad (y = y_{min})$$

$$= 0 + \frac{1}{\frac{5-120}{130-0}}(10 - 120)$$

$$= 124.34$$

$\therefore P_2' (124.34, 10)$

Line Barsky

Algorithm :

① If $P_k = 0$ for some k, the line is parallel to the clipping boundary now test q_k

If one $q_k < 0$ for these k, then line is outside
If all $q_k \geq 0$ for these k, then some portion of the line is outside

② For all $P_k < 0$ (i.e line proceeds from outside to inside boundary)

calculate $u_1 = \max(0, r_k)$

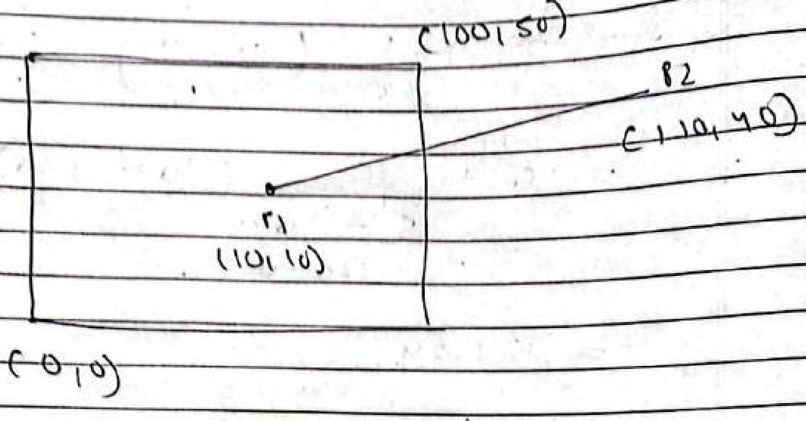
③ For all $P_k > 0$ i.e line proceeds from inside to outside boundary calculate
$u_2 = \min(1, r_k)$

④ If $u_1 > u_2$ discard line
⑤ The line is now between $[u_1, u_2]$.

$x_1' = x_1 + u_1 \Delta x$
$y_1' = y_1 + u_1 \Delta y$

$x_2' = x_1 + u_2 \Delta x$
$y_2' = y_1 + u_2 \Delta y$

Line Biansky Numerical



k	pk	qk	rk
1	$-\Delta x$ $= -(110-10)$ $= -100$ i.e. $pk < 0$	$x_1 - x_{wmin}$ $= 10 - 0$ $= 10$	$r_1 = \frac{10}{-100} = \frac{-1}{10}$ = coast or u_1
2	Δx $= 100$ $pk > 0$	$x_{wmax} - x_1$ $100 - 10$ $= 90$	$r_2 = \frac{90}{100} = 0.9$ = floats at u_2
3	$-\Delta y$ $= -(40-10)$ $= -30$ $pk < 0$	$y_1 - y_{wmin}$ $= 10 - 0$ $= 10$	$r_3 = \frac{10}{-30}$ $= -\frac{1}{3}$ = coast for u_1

(f) $\Delta y = 30$ $\quad y_{max} - y_1 = 50 - 10 = 40$ $\quad r_4 = 40/30 = 4/3$ copy of u_2

Now,

$$u_1 = max\left(0, -\frac{1}{10}, -\frac{1}{3}\right) = 0$$

$$u_2 = min\left(1, 0.9, \frac{4}{3}\right) = 0.9$$

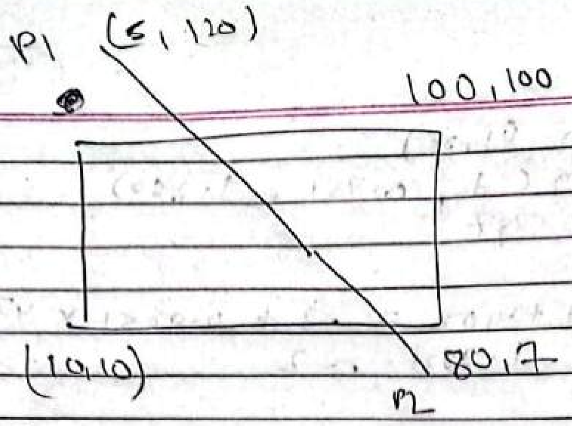
Clipped line,

$x_1' = \cancel{10} \; x_1 + u_1 \Delta x$
$= 10 + 0$
$= 10$

$y_1' = y_1 + u_1 \Delta y$
$= 10 + 0$
$= 10$

$x_2' = x_1 + u_2 \times \Delta x$
$= 10 + 0.9 \times 100 = 100$

$y_2' = y_1 + u_2 \times \Delta y$
$= 10 + 0.9 \times 30 = 37$



k	pk	qk	rk
1	$-\Delta x$ $-(80-5)$ $= -75$ $p_k < 0$	$x_1 - x_{wmin}$ $5 - 10$ $= -5$	$\frac{-5}{-75}$ $= \frac{1}{15}$ (u_1)
2	Δx $= 75$ $p_k > 0$	$x_{wmax} - x_1$ $100 - 5$ $= 95$	$\frac{95}{75}$ (u_2)
3	$-\Delta y$ $= -(7-120)$ $= 113$ $p_k > 0$	$y_1 - y_{wmin}$ $120 - 10$ 110	$\frac{110}{113}$ (u_2)
4	Δy $= -113$ $p_k < 0$	$y_{wmax} - y_1$ $100 - 120$ $= -20$	$\frac{20}{113}$ (u_1)

$$u_1 = \max\left(0, \frac{1}{15}, \frac{120}{113}\right)$$

$$= \frac{20}{113}$$

$$u_2 = \min\left(1, \frac{35}{75}, \frac{100}{113}\right)$$

$$= \frac{100}{113}$$

$$x_1' = x_1 + u_1 \Delta x$$

$$= 5 + \frac{20}{113} \times 75 = 18.27$$

$$y_1' = y_1 + u_1 \Delta y = 120 + \frac{20}{113} \times -113$$

$$= 100$$

$$x_2' = x_1 + u_2 \Delta x = 5 + \frac{100}{113} \times 73$$

$$= 78$$

$$y_2' = y_1 + u_2 \Delta y = 120 + \frac{100}{113} \times -113$$

$$= \cancel{20}\ 10$$

~~$P_1'(x, y)$~~ = P_1'

P_1' (18.27, 100)

P_2' (78, 10)

30

$(x_{min}, y_{min}) = (7,5)$

$(x_{max}, y_{max}) = (35,50)$

line $(x_1, y_1) = (-2,2)$ $\quad (x_2, y_2) = (45,40)$

$\Delta x = x_2 - x_1 = 45+2 = 47$

$\Delta y = y_2 - y_1 = 40 - 2 = 38$

K	P_K	q_K	$r_K = \frac{q_K}{P_K}$
1	$-\Delta x = -47$, $P_K < 0$	$x_1 - x_{min} = -4$	0.0851 (u_1)
2	$\Delta x = 47$, $P_K > 0$	$x_{max} - x_1 = 37$	0.787 (u_2)
3	$-\Delta y = -38$, $P_K < 0$	$y_1 - y_{min} = -3$	0.0789 (u_1)
4	$\Delta y = 38$, $P_K > 0$	$y_{max} - y_1 = 48$	1.263 (u_2)

$u_1 = \max(0, r_k)$

$= \max(0, 0.0851, 0.0789)$

$= 0.0851$

$u_2 = \min(0_1, r_k)$

$= \min(1, 0.787, 1.285)$

$= 0.787$

$x_1' = x_1 + u_1 \Delta x = -2 + 0.0851 \times 47$

$= 1.9997 \approx 2$

$y_1' = y_1 + u_1 \Delta y = 2 + 0.0851 \times 38$

$= 5.2238$

≈ 5

$x_2' = x_1 + u_2 \Delta x = -2 + 0.787 \times 47$

$= 34.989 \approx 35$

$y_2' = y_1 + u_2 \Delta x = 2 + 0.787 \times 38$

$= 31.906 \approx 32$

$\therefore$ Required points (2, 5) and (35, 32) #