Two dimensional Viewing and Clipping

World coordinate system: Objects are placed into the scene by modeling transformations to a master coordinate system, commonly referred to as the world coordinate system.

Viewport: Used to control the placement of the clipping window within the display window.

Clipping window: A section of a two dimension scene that is selected for display.

(*) Window-to-viewport mapping: A window is specified by four world coordinates: wx_{min}, wy_{min}, wx_{max}, wy_{max}. Similarly, a viewport is described by four normalized device coordinates. vx_{max}, vx_{max}, vy_{min}, vy_{max}.

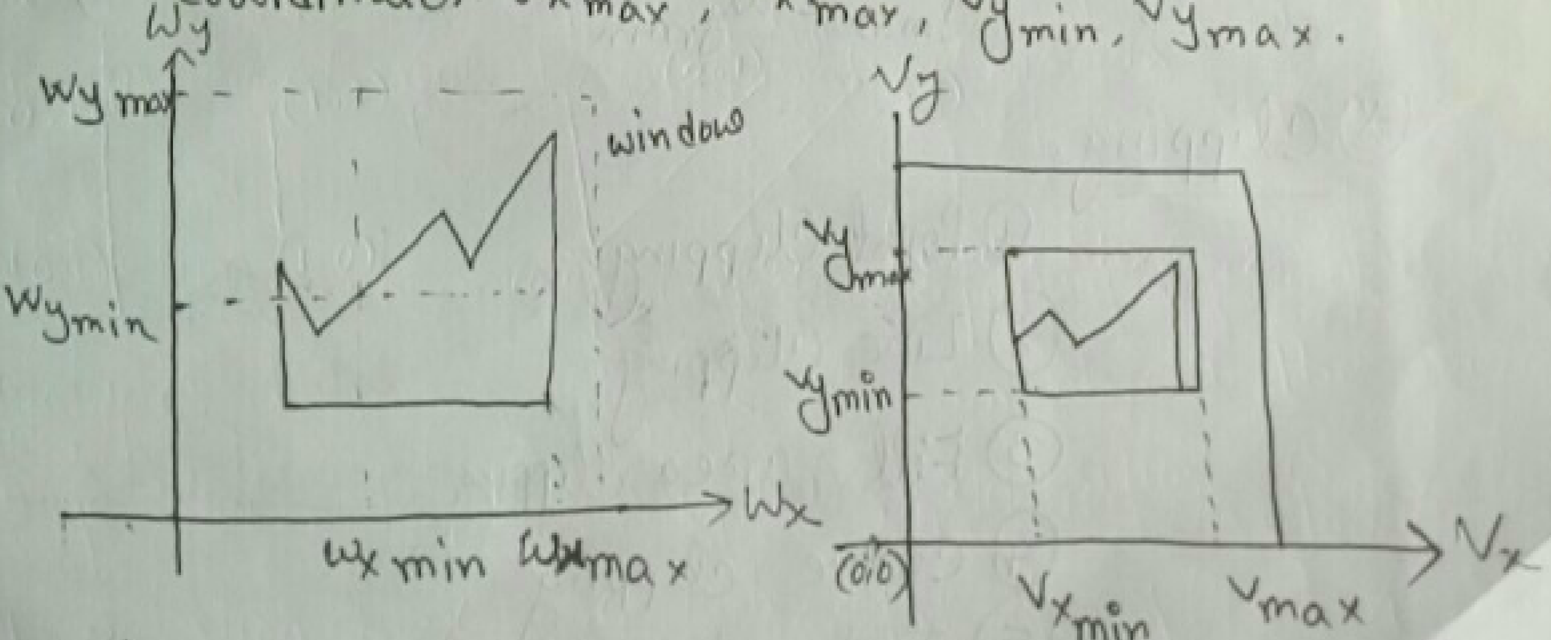

fig: window-to-viewport mapping

The objective of window-to viewport mapping is to convert the world coordinates (w_x, w_y) to normalized device co-ordinates (v_x, v_y). So,

$$\frac{w_x - w_{xmin}}{w_{xmax} - w_{xmin}} = \frac{v_x - v_{xmin}}{v_{xmax} - v_{xmin}} \text{ and}$$

$$\frac{w_y - w_{ymin}}{w_{ymax} - w_{ymin}} = \frac{v_y - v_{ymin}}{v_{ymax} - v_{ymin}}$$

This $v_x - v_{xmin} = \dfrac{w_x - w_{xmin}}{w_{xmax} - w_{xmin}} (v_{xmax} - v_{xmin})$

$$\therefore v_x = \frac{v_{xmax} - v_{xmin}}{w_{xmax} - w_{xmin}} (w_x - w_{xmin}) + v_{xmin}$$

and, $v_y = \dfrac{v_{ymax} - v_{ymin}}{w_{ymax} - w_{ymin}} (w_y - w_{ymin}) + v_{ymin}$

Clipping:

1. Point Clipping
2. Line Clipping
3. Fill-area clipping
4. Curve clipping
5. Text clipping

Point Clipping: Point clipping is essentially the evalation of the following inequalities.

$$x_{min} \leq x \leq x_{max} \text{ and } y_{min} \leq y \leq y_{max}.$$

where x_{min}, x_{max}, y_{min} and y_{max} define the clipping window. A point (x, y) is considered inside the window area.

Line-clipping:- We devide the line clipping process into two phases.

(1) Identify those lines which intersect the clipping window and so need to be clipped and

(2) Perform the clipping.

All lines fall into one of the following clipping categories.

(1) **Visible**: Both endpoints of the line, lie within the window.

(2) **Not visiable**: The line definitely lies outside of the window. This will occur if $x_1, x_2 > x_{max}$

$$x_1, x_2 < x_{min}$$

and $y_1, y_2 > y_{max}$

$$y_1, y_2 < y_{min}$$

→

(iii) <u>Clipping candidate</u>: The line is in neither category 1 nor 2.

The algorithm employs an efficient procedure for finding the catagory of a line. It proceeds in two steps.

1. Assign a 4 bit region code to each endpoints of the line. The code is determined according to which of the following nine region of the plane the endpoint line in

1001	1000	1010
0001	0000	0010
0101	0100	0110

x_{min} x_{max}

Starting from the leftmost bit, 1 is true and 0 is false.

Bit 1 = endpoint is above the window = $sign(y - y_{max})$

Bit 2 = endpoint is below " " = $sign(y_{min} - y)$

Bit 3 = " " to the right " " = $sign(x - x_{max})$

Bit 4 = " " to the left of " = $sign(x - x_{min})$

here sign (a) = 1, if a is positive, otherwise of course a point with code 0000 is inside the window.

2. The line is visible if both region codes are 0000 and not ~~to~~ visiable if the bitwise logical AND of the codes is not 0000 and a candiate for clipping if the bitwise logical AND of the region codes is 0000.

Algorithm:

1. Given a line segment with endpoint $P_1 = (x_1, y_1)$ and $P_2 = (x_2, y_2)$

2. Complete the 4 bit codes for endpoints. If both codes are 0000 (bitwise OR of the codes yeilds 0000) lines lies completely inside the window. If the both codes have 1 in the same bit position (bitwise AND) of the codes is not 0000 the line lies outside the window.

3. If a line can't be trivially accepted or rejected at least one of the two endpoint must lie outside the window and the line segment cross

→

a windows edge. This line must be clipped.

(4) Examine one of the endpoints say $P_1 = (x_1, y_1)$. Read P_1 is 4 bit code in order: left-to right bottom-to-top

(5) If the bit 1 is 1, intersect with line, $y = y_{max}$

If " bit 2 is 1 " " " $y = y_{min}$

If " " 3 " 1 " " " $x = x_{max}$

If " " 4 " 1 " " " $x = x_{min}$

The coordinates of the intersection point are

$$\begin{cases} x_i = x_{min} \text{ or } x_{max} \\ y_i = y_1 + m(x_i - x_1) \end{cases} \text{(if the boundary line is vertical)}$$

or $$\begin{cases} x_i = x_1 + (y_i - y_1)/m \\ y_i = y_{min} \text{ or } y_{max} \end{cases} \text{if the boundary line is horizontal}$$

where, $m = \dfrac{y_2 - y_1}{x_2 - x_1}$, is the slope of line.

Problem: 1(a) Find the clipping categories for the line segments. fig: 1

1(b) Use the cohen-sutherland algorithm to clip the line segments in fig 1.

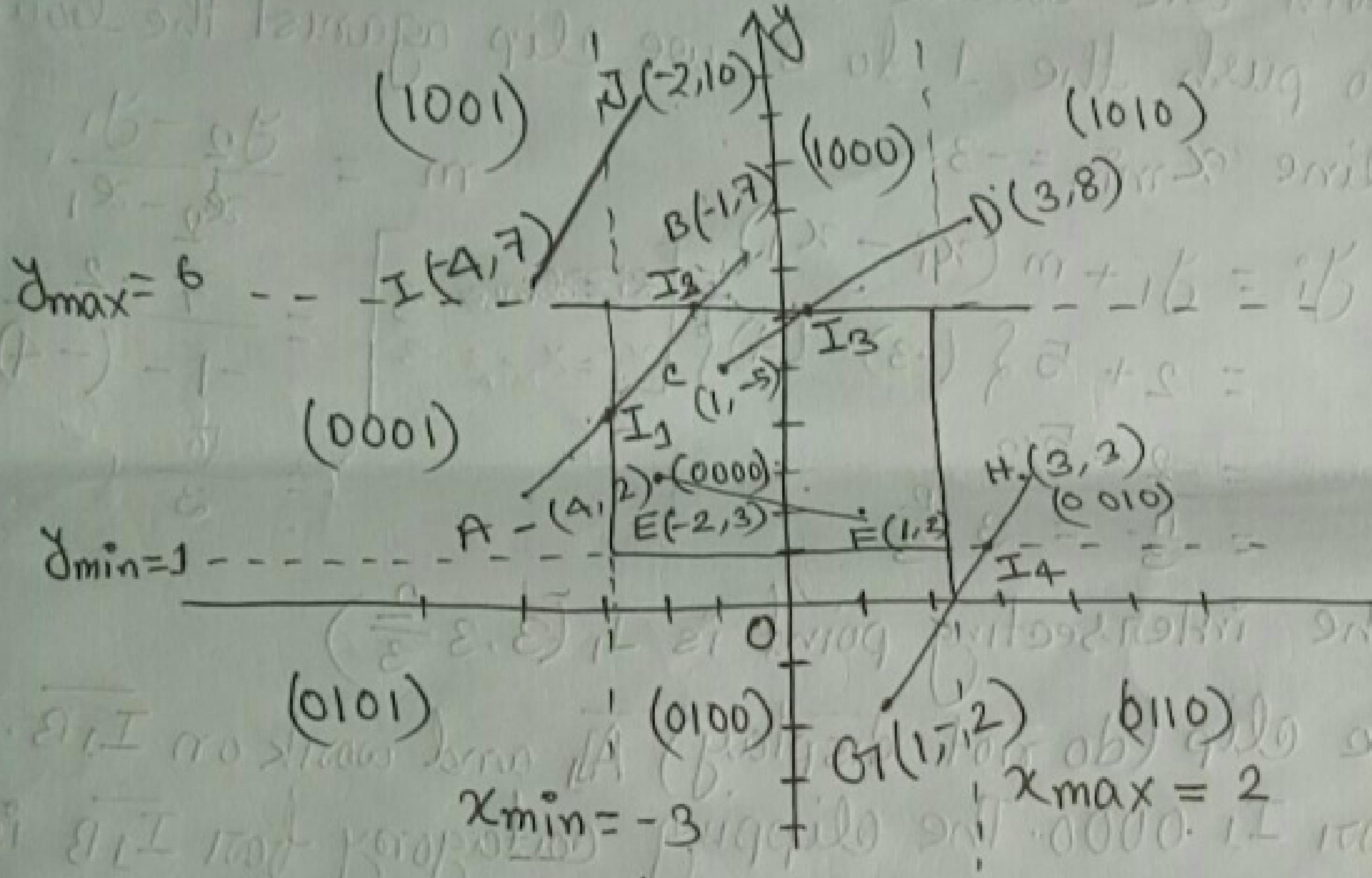

fig: 1

Solution: 1(a)

Category 1 (visiable): $\overline{EF}$ since the region code for both endpoint is 0000

" 2 (Not visiable): $\overline{IJ}$ since (1001) and (1000) = 1000 (which is not 0000)

" 3 (candidate for clipping) $\overline{AB}$ since (0001) AND (1000) = 0000

$\overline{CD}$ " (0000) " (1010) = 0000

$\overline{GH}$ " (0100) " (0010) = 0000

Solution: 1 (b)

From solution 1 (a), the candidates for clipping are $\overline{AB}$, $\overline{CD}$ and $\overline{GH}$.

For clipping $\overline{AB}$ we can start with either A or B since both are outside the window. The code for A is 0001, to push the 1 to 0, we clip against the boundary line $x_{min} = -3$.

$$y_i = y_1 + m(x_i - x_1) \qquad m = \frac{y_2 - y_1}{x_2 - x_1}$$

$$= 2 + 5\{(-3) - (4)\} \quad [x_i = x_{min} = -3] \qquad = \frac{7-2}{-1-(-4)}$$

$$= 2 + 5/3$$

$$= \frac{11}{3} = 3\frac{2}{3} \qquad = \frac{5}{3}$$

The intersecting point is $I_1(3, 3\frac{2}{3})$

We clip (do not display) $\overline{AI_1}$ and work on $\overline{I_1B}$. The code for I_1 0000. The clipping catagory for $\overline{I_1B}$ is 3 since (0000) AND (1000) is (0000). Now B is outside the window we use push to 1 to a 0 by clipping aganist the line $y_{max} = 6$. Hence,

$$x_i = x_1 + \frac{y_i - y_1}{m}$$

$$= -4 + \frac{6-2}{5/3}$$

$$= -4 + (4 \times \frac{3}{5}) = -4 + \frac{12}{5} \quad [y_i = y_{max} = 6]$$

$$= -\frac{8}{5} = -1\frac{3}{5}$$

The intersecting point is $I_2(-1\frac{3}{5}, 6)$. Thus $\overline{I_2B}$ is clipped. The code for I_2 is 0000. The remaining segment I_1I_2 is displayed since both endpoints lie in the window (i,e, their codes are 0000)

For $\overline{CD}$ we start with D since it is outside the window. Its code is 1010. We push the first 1 to a 0 by window clipping aganist the line $y_{max} = 6$.

$$x_i = x_1 + \frac{y_i - y_1}{m} \qquad m = \frac{y_2 - y_1}{x_2 - x_1}$$

$$= -1 + \frac{6-5}{3/4} \left[\because y_i = y_{max} = 6\right] \qquad = \frac{8-5}{3-(-1)}$$

$$= -1 + \frac{4}{3} = \frac{-3+4}{3} \qquad = \frac{3}{4}$$

$$= \frac{1}{3}$$

The intersecting point is $I_3(1/3, 6)$ and its code is (0000). $\overline{I_3D}$ is clipped and the remaining segment $\overline{CI_3}$ is displayed since both endpoints lie in the window

For clipping $\overline{GH}$, we can start with either G or H since both are outside the window. The code for G is 0100 and we push the 1 to a 0 by clipping aganist the line $y_{min} = 1$.

$$x_i = x_1 + \frac{y_i - y_1}{m} \qquad m = \frac{y_2 - y_1}{x_2 - x_1}$$

$$= 1 + \frac{1-(-2)}{5/2} \left[y_i = y_{max} = 1 \right] \qquad = \frac{3-(-2)}{3-1}$$

$$= 1 + 3 \times 2/5 \qquad = \frac{5}{2}$$

$$= 1 + \frac{6}{5} = \frac{11}{5}$$

$$= 2\frac{1}{5}$$

The intersection point is $I_4(2\frac{1}{5}, 1)$ and its code is 0010. We clip $\overline{GI_4}$ and work on $\overline{I_4H}$.
Segment $\overline{I_4H}$ is not displayed since (0010) AND (0010) = 0010

---o---