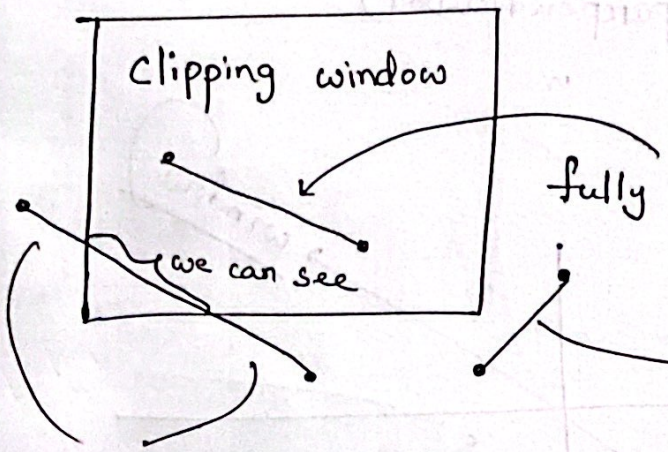
2 types

Clipping — Line and polygon.



Algorithms.

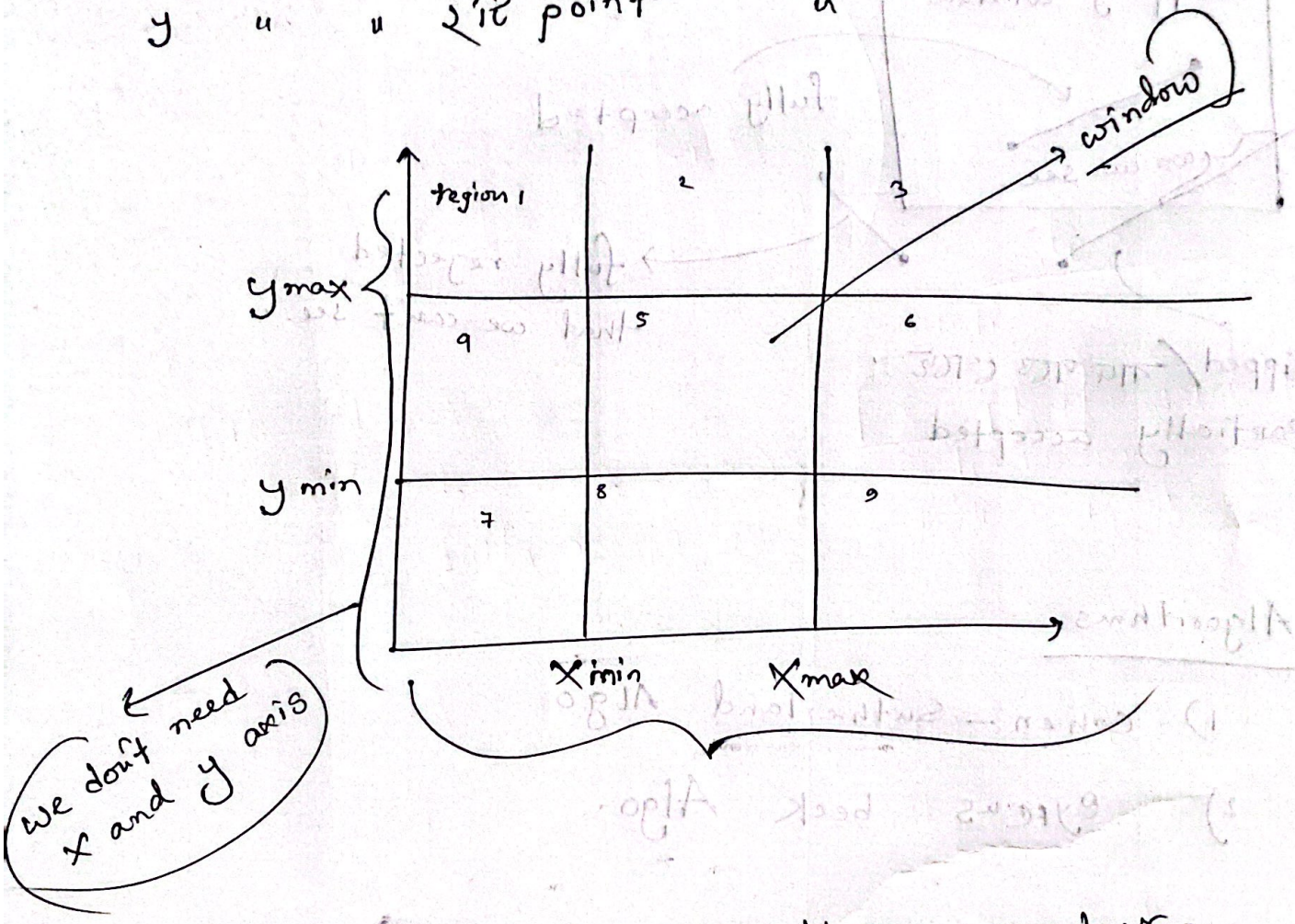
1) Cohen – Sutherland Algo
2) Cyrus beck Algo.

Cohen Sutherland Algo

1) Clipping করতে হলে প্রথমে clipping window চেনা থাকতে হবে।

How to create clipping window :

x axis → २रो point . (perpendicular)
y " → २रो point "



→ Naming every region using binary number.

→ for 9 regions → 4 bit binanary and we'll call this binary as output .

Let's create 4 bits .

above	below	right	left

Above → we you're in a particular region.

Suppose, $y > y_{max}$

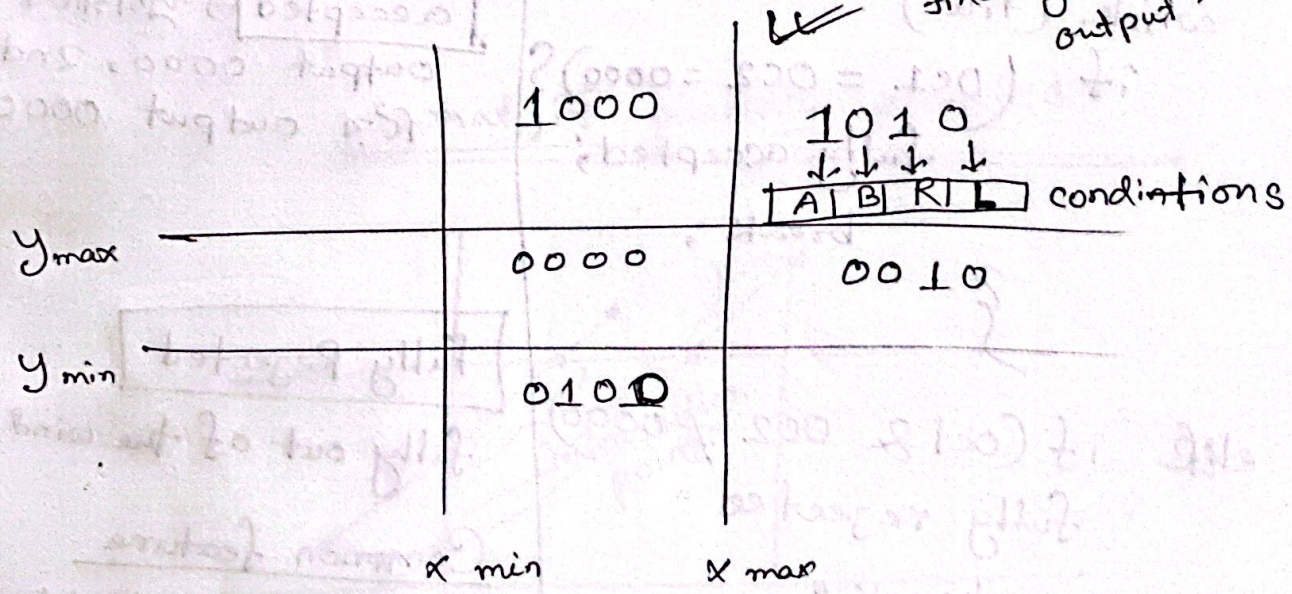
Below $y < y_{min}$

Right $x > x_{max}$

left → $x < X_{min}$

if condition correct then 1 otherwise 0

suppose we're finding this regions output.



* if the point is in axis,

suppose between 1010 and 0010

0010

choose the lowers one 0010,

where the number of bit "1" is less

select that.

Relates to the condition y, x

$y > y_{max}$, $y < y_{min}$

※ question

Cohen – Suther.

(x_1, y_1, x_2, y_2)

step

(1) create two outputs.

$OC1 = \text{from } (x_1, y_1)$

$OC2 = \text{from } (x_2, y_2)$

Code:

```
while (true)
    if (OC1 = OC2 = 0000){
        fully accepted;
        break;
    }
else if (oc1 & oc2 ≠ 0000)
    fully rejected
    break;
}
else{
    if(oc1 ≠0000){
        use non zero bit of OC1
        to find out the intsection
        point P1 and update OC1.
    }
    else{
        use non zero bit of ~~out~~ OC2
        to find the ~~out~~ intersection
        point P2 and update OC2.
    }
continue;
}
```

step 2

Conditions

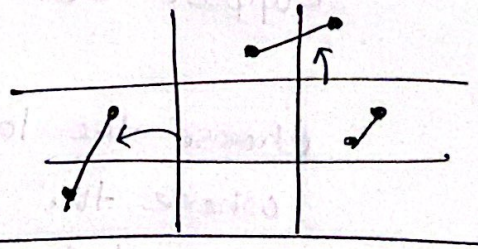
accepted → দুনাম্বারটির output 0000, 2nd টির output 0000.

Fully Rejected

fully out of the window

Common feature

eigther, right side/ left side / upper/below of same axis.



bitwise "and" of outcode

Suppose,

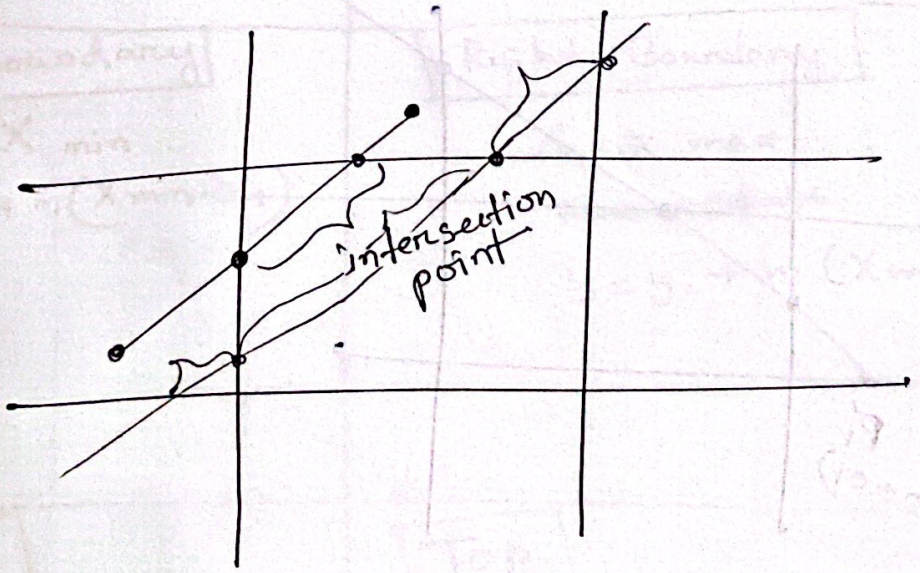
```
1000
1010
----
1000
```

and → multiplying

→ minimun 1, (at least a "1" then fully rejected)

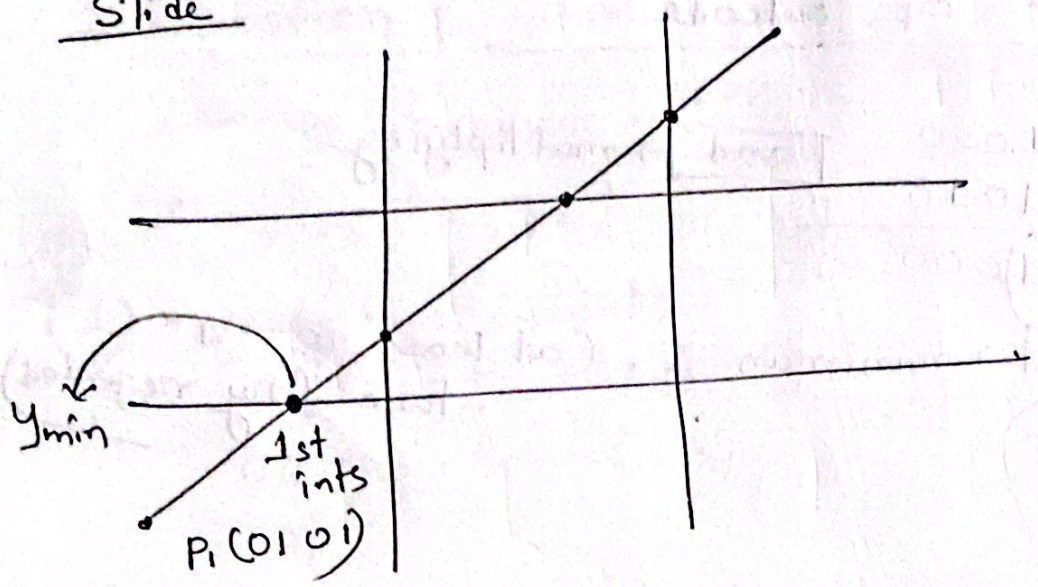
Rejecting process with condition explaination

what if partially accepted



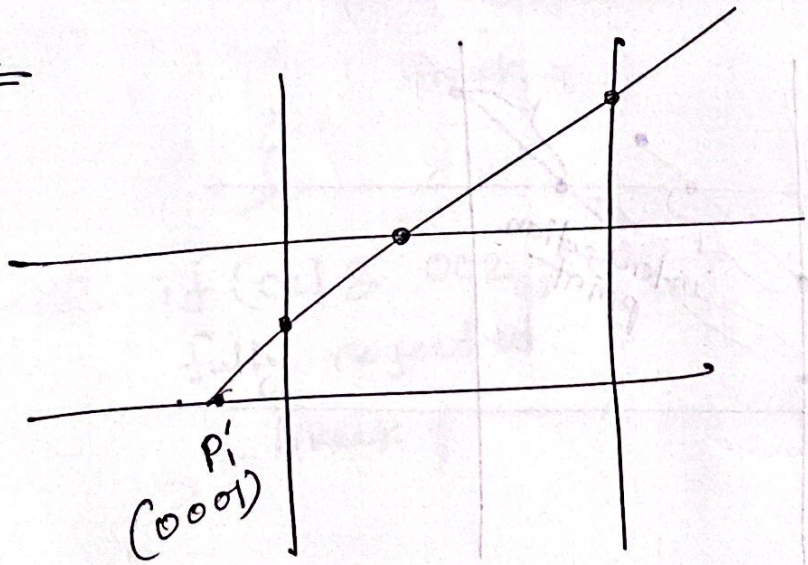
1) find the intersection point.

From Slide

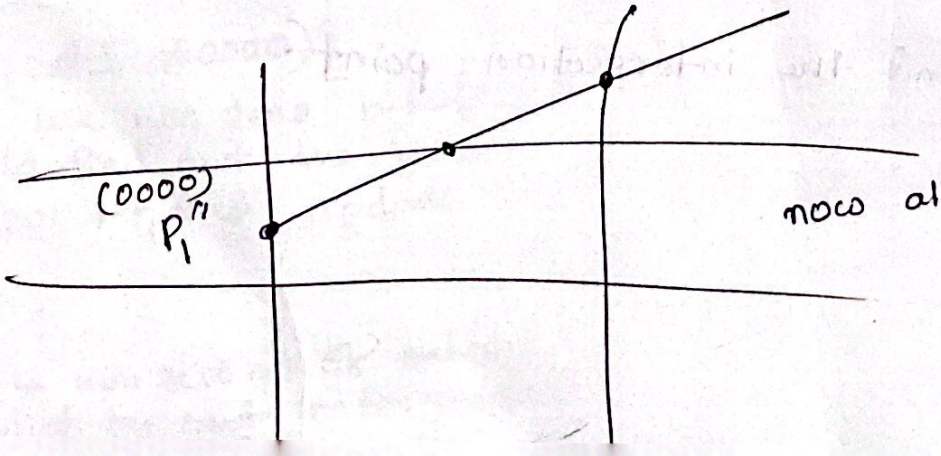


Started from condition $OC1 \neq 0000$

2

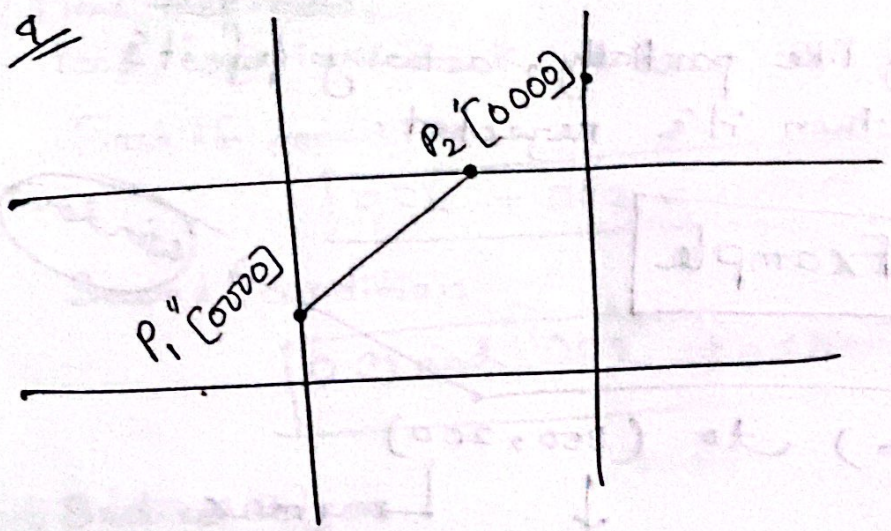


3



noco all are zero.

q/



IS no page — slide

Formula

Left boundary

$X = X_{min}$

$y = y_1 + m(X_{min} - X_1)$

Right Boundary

$X_1 = X_{max}$

~~$X = X_1 + 1$~~

$y = y_1 + m(X_{max} - X_1)$

Bottom

$y = y_{min}$

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

Top

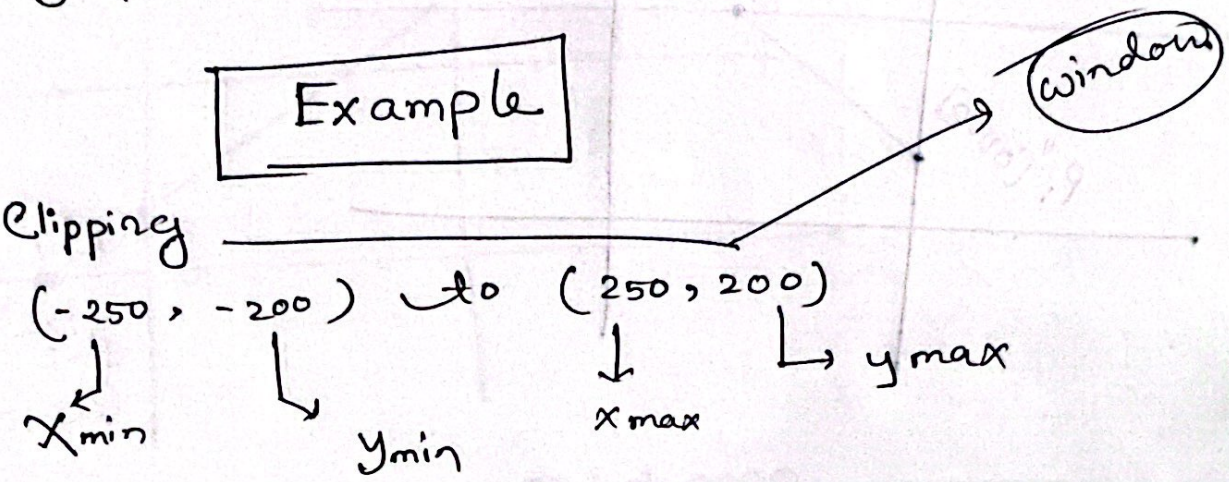
$y = y_{max}$

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

* There is nothing like partially, actualy if it's following partial then it's rejected.

Example

Given Clipping region (-250, -200) to (250, 200)



1) Clip the line from P(100, 50) to P2 (300, 100)

Step 1

P1 (100, 50) OC1
x y

OC1 = y > ymax | y < ymin | x > max | x < xmin

	50 > 200	50 < -200	100 > 250	100 < -250
	0	0	0	0
OC1=0000				
OC2 = 0010	100 > 200	100 < -200	300 > 250	300 > < -250
	0	0	1	0

OC1 → 0000
OC2 → 0010

※ Main task now,
loop conditions check

First if condition,

OC1 = OC2 = 0000 ✗

Second condition

OC1 and OC2 ≠ 0000 ✗

3rd condition

OC1 ≠ 0000

find non zero bit, intersection

4th condition

OC2 ≠ 0000 → find intersection ✓

∴ intersect P2 ⟶ Xmax

Formula → $x = x_{max} = 250$

$y = y_1 + m(X_{max} - X_1)$

$= 100 + m(250 - 300)$

$= 100 + \frac{100-50}{300-100}(-50)$

$= 100 + (0.25)(-50)$

$= 100 - 12.5$

$\therefore y = 87.5$ (Don't roundup now)

∴ intersection point P_2' (250, 87.5)
x y
new value

4 conditions matching

$87.5 \not> 200 \longrightarrow 0$

$87.5 \not< -200 \longrightarrow 0$

$250 \not> 250 \longrightarrow 0$

$250 \not< -250 \longrightarrow 0$

updated values, (0000)

again start from condition 1 from loop,

※ So show (100, 500) to P_2' (250, 87.5)

↓

now roundup the ans 88.

[Here coe just did it one time but it may need to play more than 1 time] → loop in cycle

+1.00

1) P_1 (−300, −350) to P_2 (400, 300)

2) P_1 (−250, −300) to (−100, 600)

Use DESMOS to verify your ans