



# Stack - III

Foundation Course on Data Structures & Algorithms - Part II

→ Next smaller element :-

prev smaller

$a \rightarrow b$



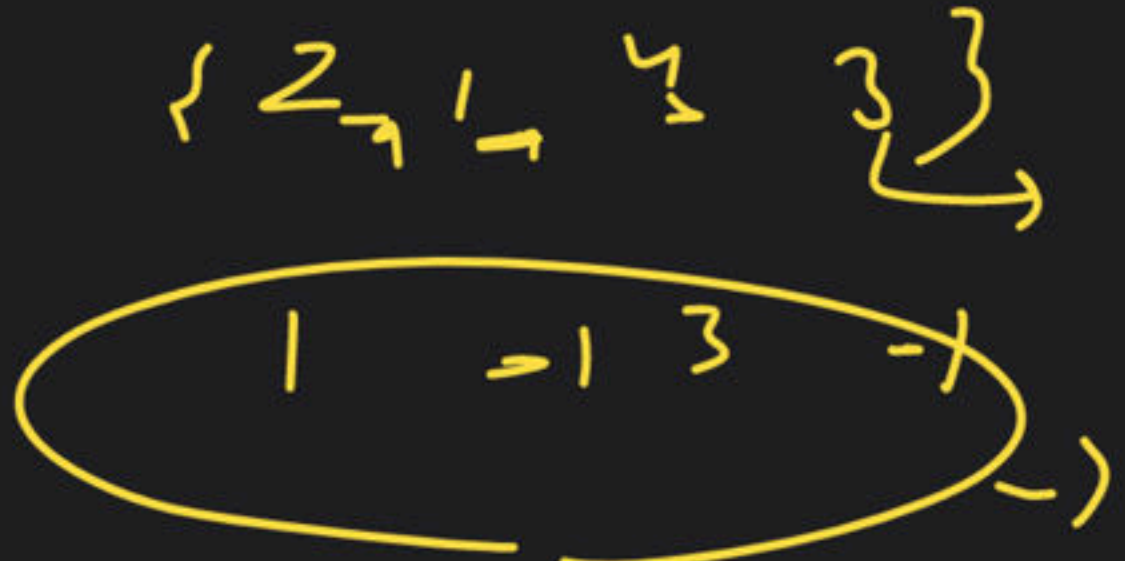
ans (

#1 → 2 for loops → T.C →  $O(n^2)$

#2 → stack

for ( $i = n - 1$ ;  $i \geq 0$ ;  $i--$ )

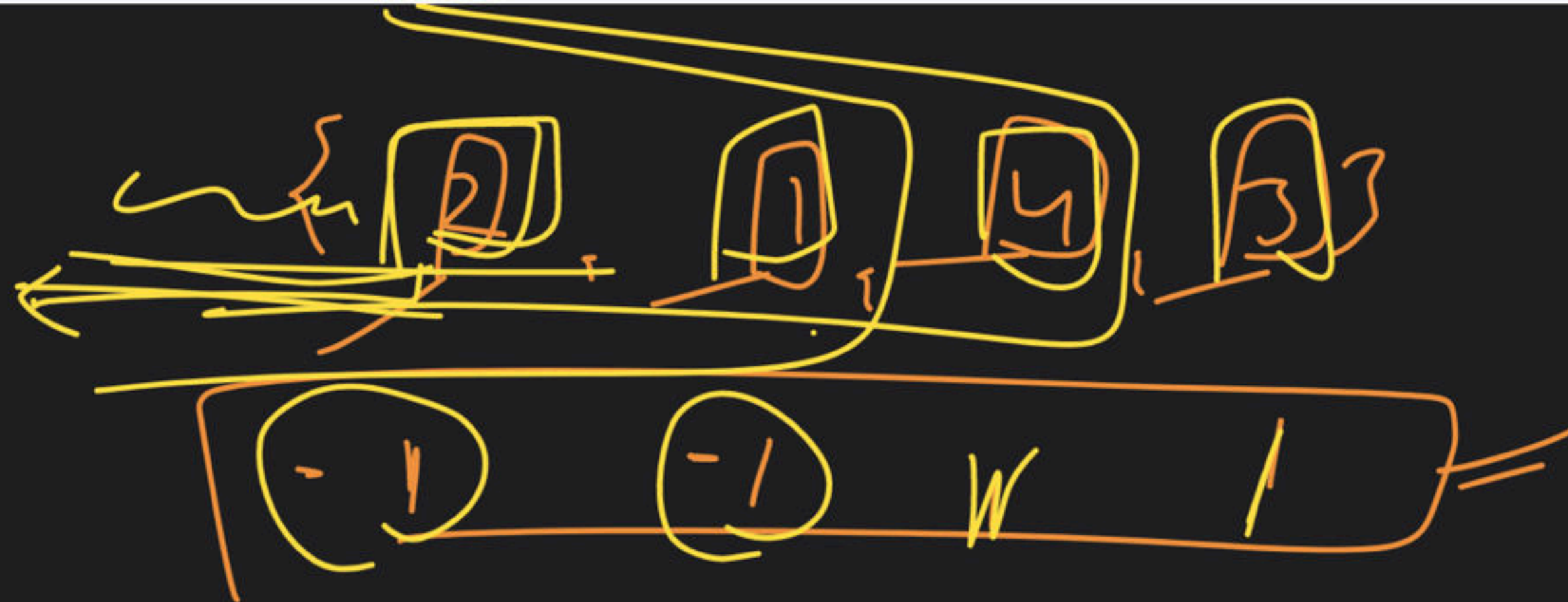
{  $ans[i] = item$ ;



while ( $s.top() \geq item$   
or  $s.top() == -1$ )

{  $s.pop()$

→  $ans[i] = s.top();$   
→  $s.push(item);$



-ve

?

→ Next Smaller  
→ Prev Smaller ]

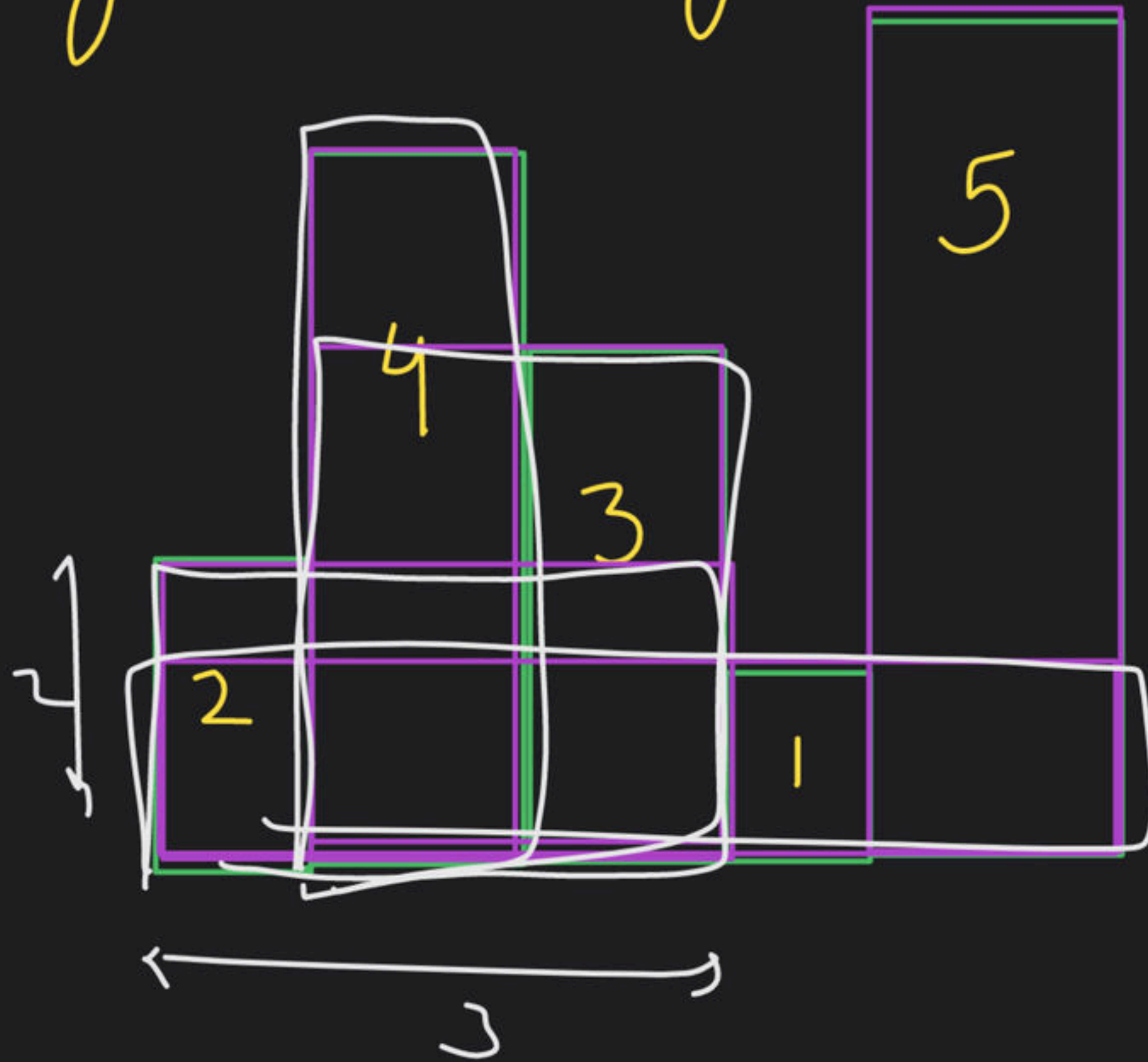




Q44 - Largest rectangular area in Histogram

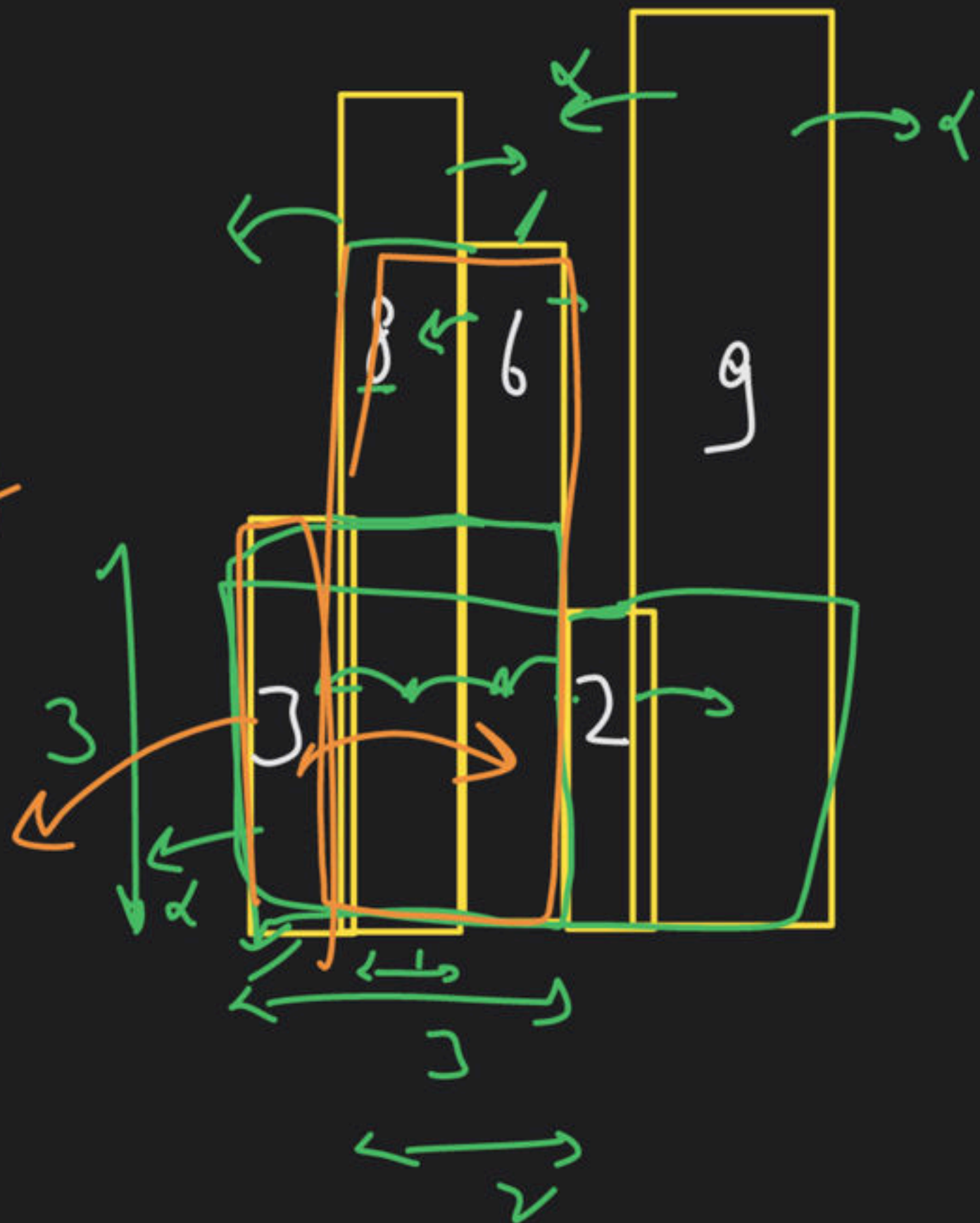
6  
4  
6  
5  
5

⑥



Largest R.A

9  
 8  
 6x2 = 12  
 2x5 = 10  
 5x1 = 5



#1

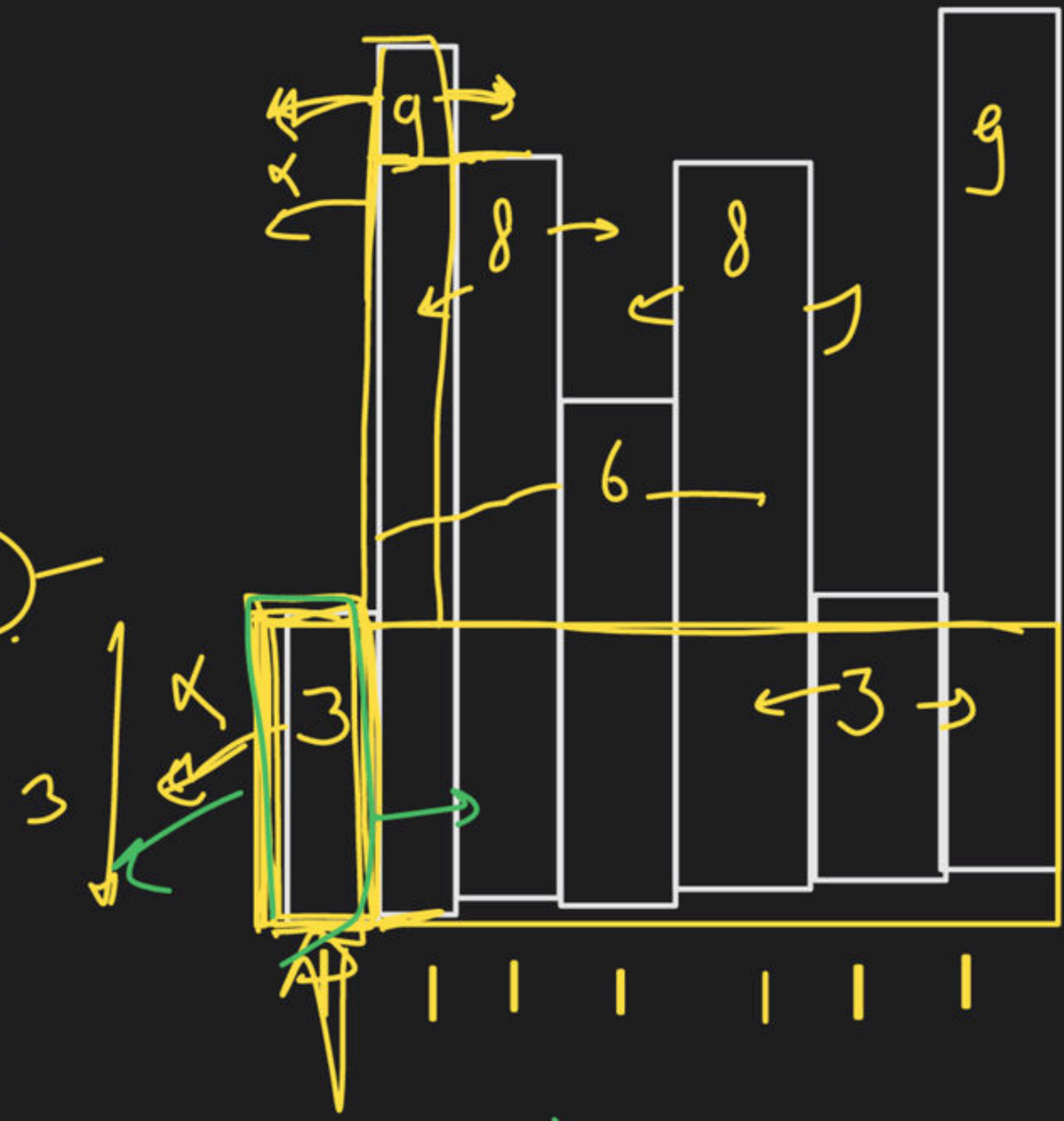


```

for (int i=0; i<n; i++)
{
  _
}
  
```



$7 \times 3 = 21$   
 $9 \times 1 = 9$   
 $1 \times 2 = 2$   
 $6 \times 4 = 24$   
 $1 \times 1 = 1$   
 $3 \times 7 = 21$   
 $9 \times 1 = 9$



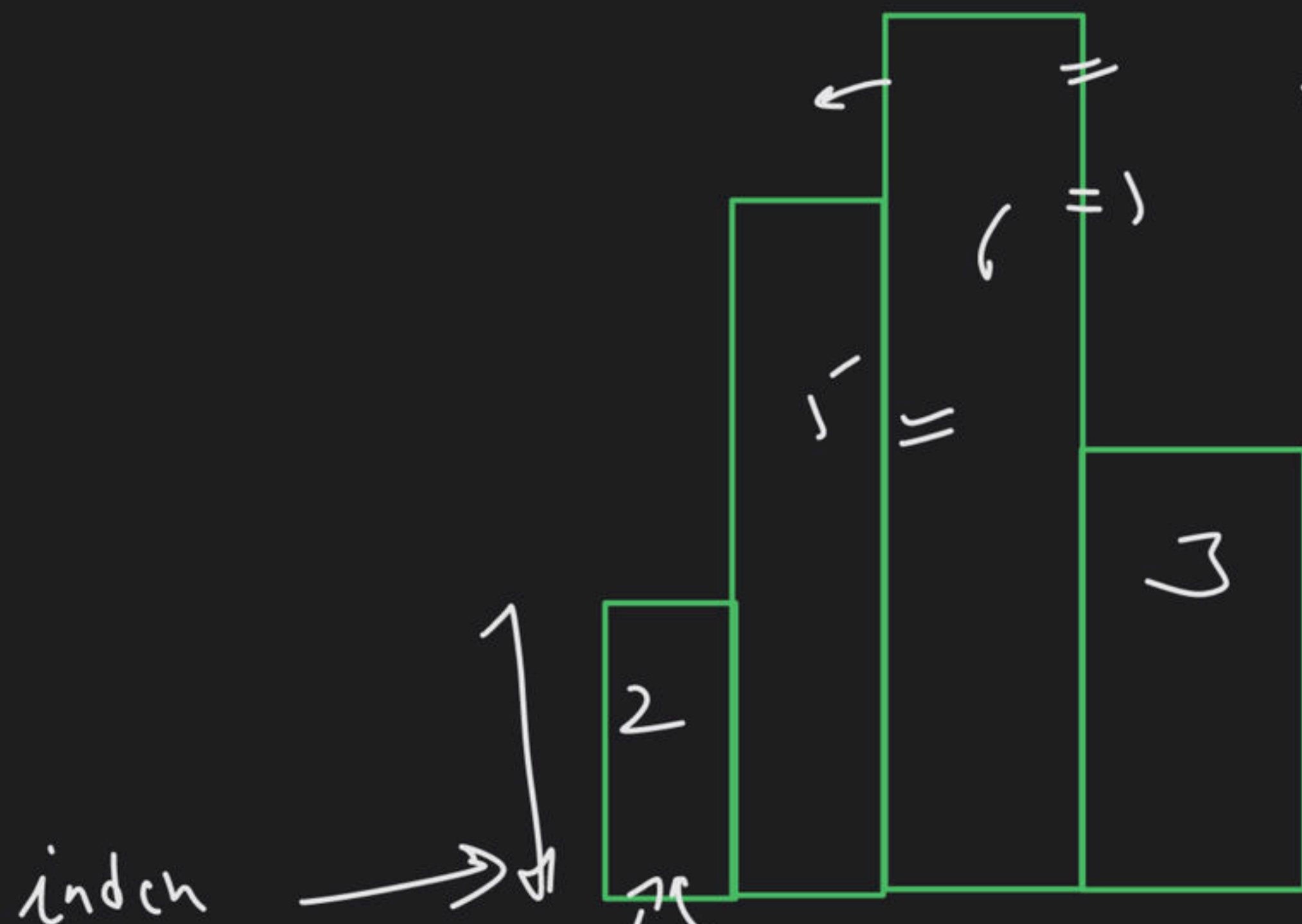
$O(n^2)$

```

for (int i = 0; i < n; i++)
{
    while (left structure)
    {
        // ...
    }
    while (right structure)
    {
        // ...
    }
}

```

new area =  
 max area = max(new area, current area)



5-3-1  
 > 2

4/4 extent  
 > 2  
 Bad

RunTime  
 <  
 (40+1)

index

value

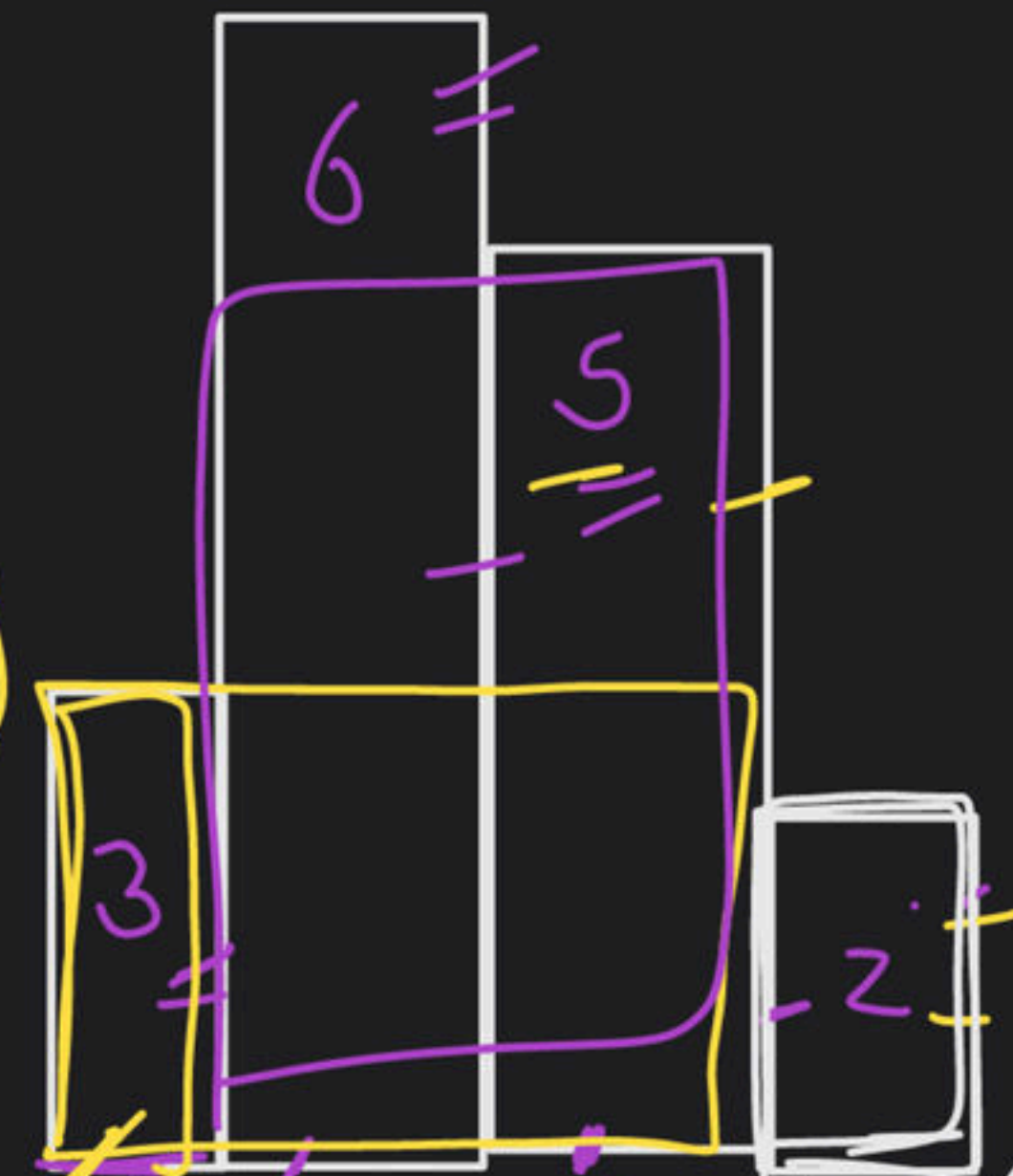
prev smaller  
 next smaller  
 2 → ps = -1  
 2 → ns = -1

length = arr[i] = 2  
 width = nexts - prevs + 1  
 = -1 - (-1) + 1  
 = 1



9

$l=2$   
 $b=3$



PS →

$-1$   $0$   $0$   $-1$

NS →

$3$   $2$   $3$   $-1$   $4$

$2 \times 4$   
 $= 8$

$l=2$   
 $b=1$

$1 + (-1) - 1$   
 $= -1$

$4 - (-1) - 1$   
 $= 4 + 1 - 1$

$l=6$   
 $b=2-0-1$   
 $= 1$

$l=5$   
 $b=3-0-1=2$   $(10)$

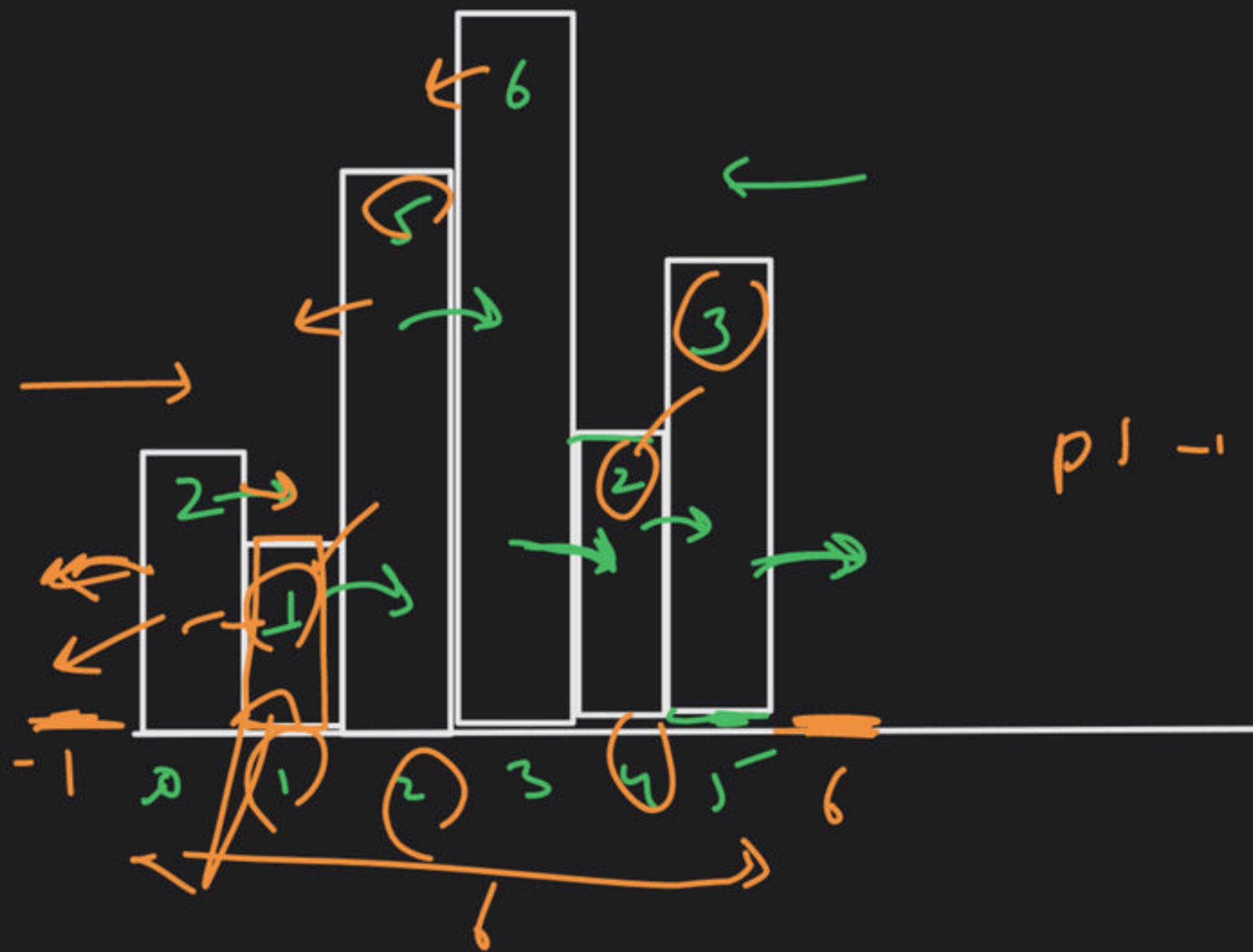
$l = arr[i] = 3$

$b = next - prev - 1$

$= 3 - (-1) - 1$

$= 4 - 1$

$= 3$



$$l = 1$$

$$b = -1 + (-1) - 1$$

① NS

$$\begin{bmatrix} 1 & \textcircled{-1} & 4 & 9 & -1 & -1 \end{bmatrix}$$

$$p1 = \begin{bmatrix} -1 & \textcircled{-1} & 1 & 2 & 1 & 4 \end{bmatrix}$$

$$l = 2$$

$$b = 1 - (-1) - 1 \rightarrow \boxed{2}$$

$$= 1$$

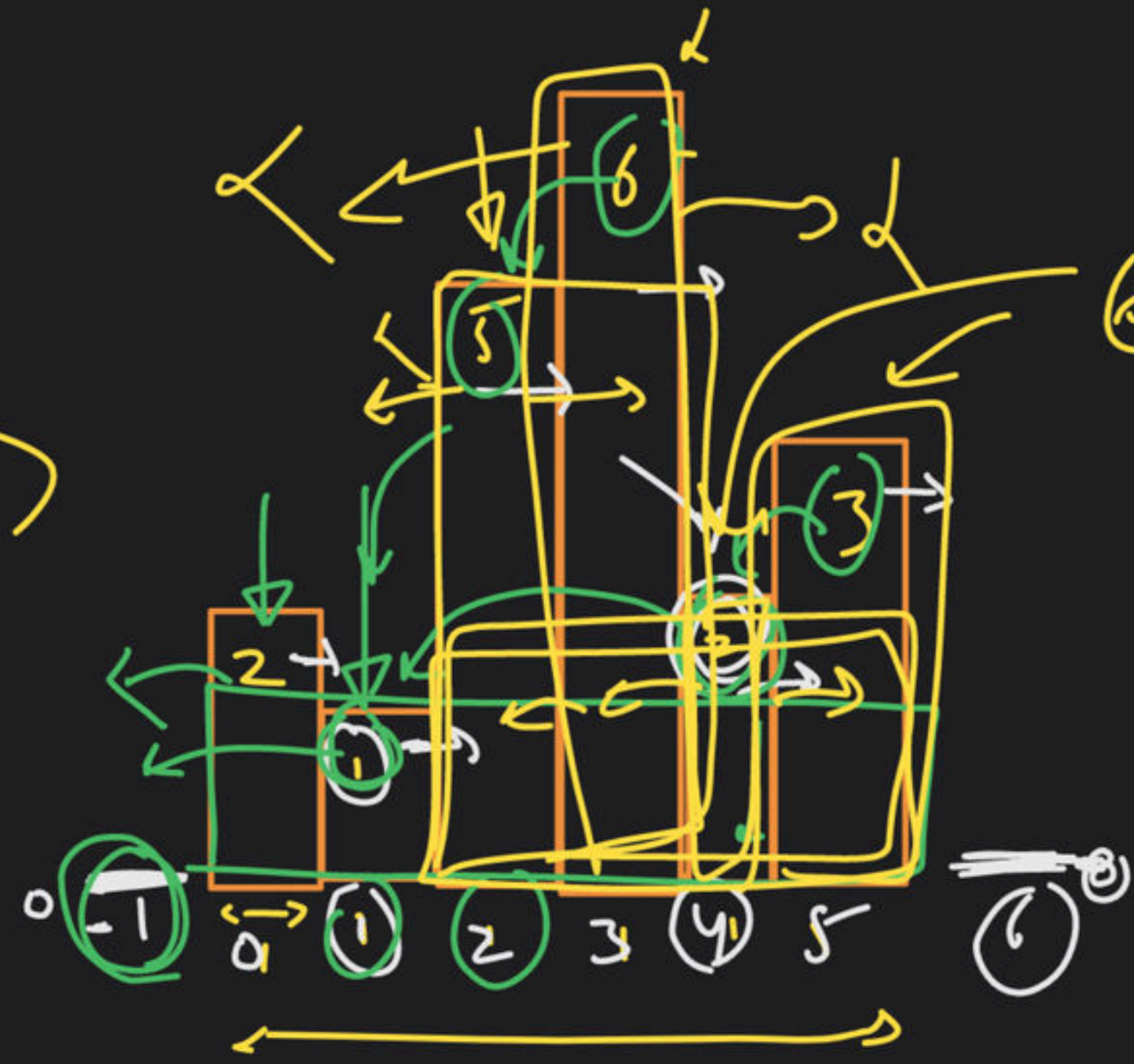
if (s.top() != -1)

ans[i] = n

2



②  
 $l = 2$   
 $b = 2$   
 $2 \times 4 = 8$



① Next Smaller Element  
 $l = 6$   
 $b = 6 - 1 - 1 - 1 - 1 - 1 - 1 = 1$   
 $1 \rightarrow 3$

NS  $\rightarrow [1, 6, 4, 4, 6, 6]$

PS  $\rightarrow [-1, -1, 1, 2, 1, 4]$

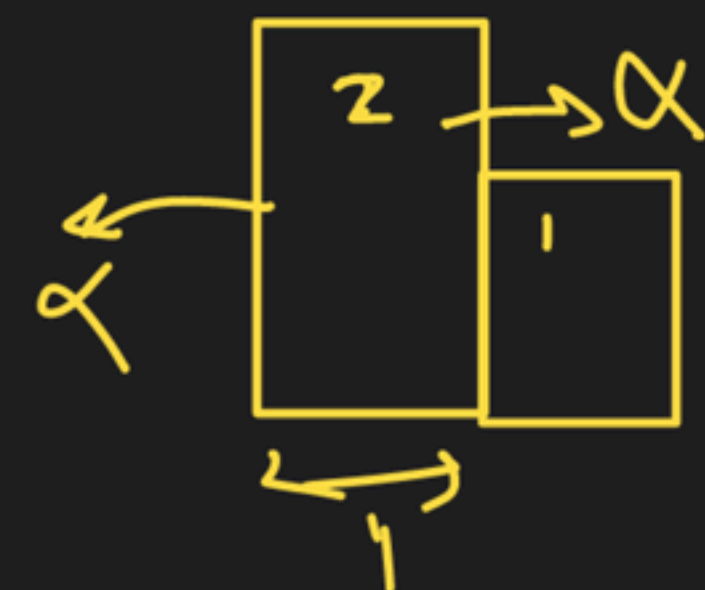
2  
 $l = 2$   
 $b = 1 - (-1) - 1 = 1$   
 $A = 2 \times 1 = 2$

①  
 $l = 1$   
 $b = 6 - (-1) - 1 = 6 + 1 - 1 = 6$   
 $A = 1 \times 6 = 6$

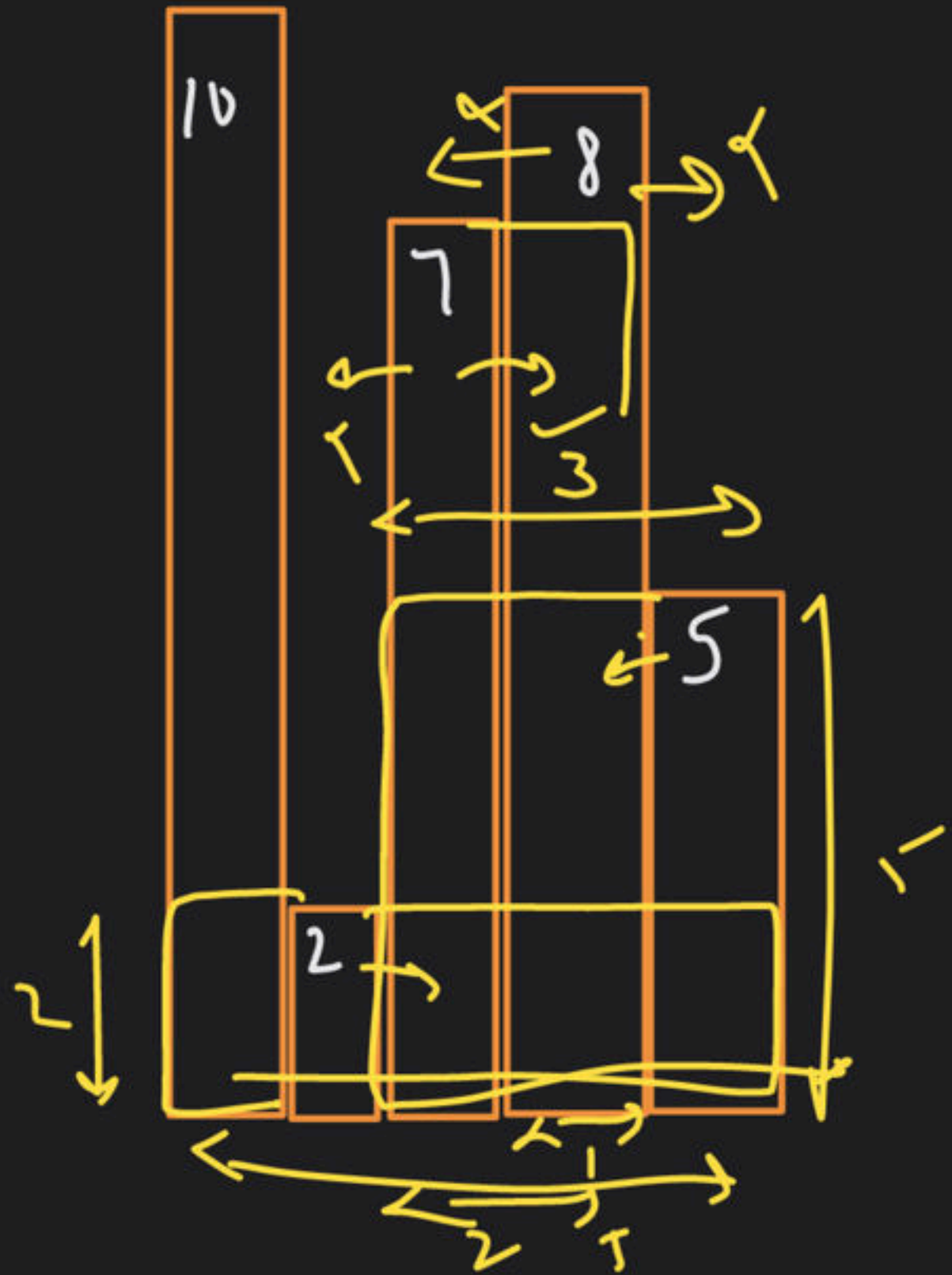
⑤  
 $l = 5$   
 $b = 4 - 1 - 1 = 2$   
 $A = 5 \times 2 = 10$

⑥  
 $l = 6$   
 $b = 7 - 2 - 1 - 1 - 1 - 1 = 1$   
 $A = 6 \times 1 = 6$





→ Largest Rectangular area in Histogram → (HARD)

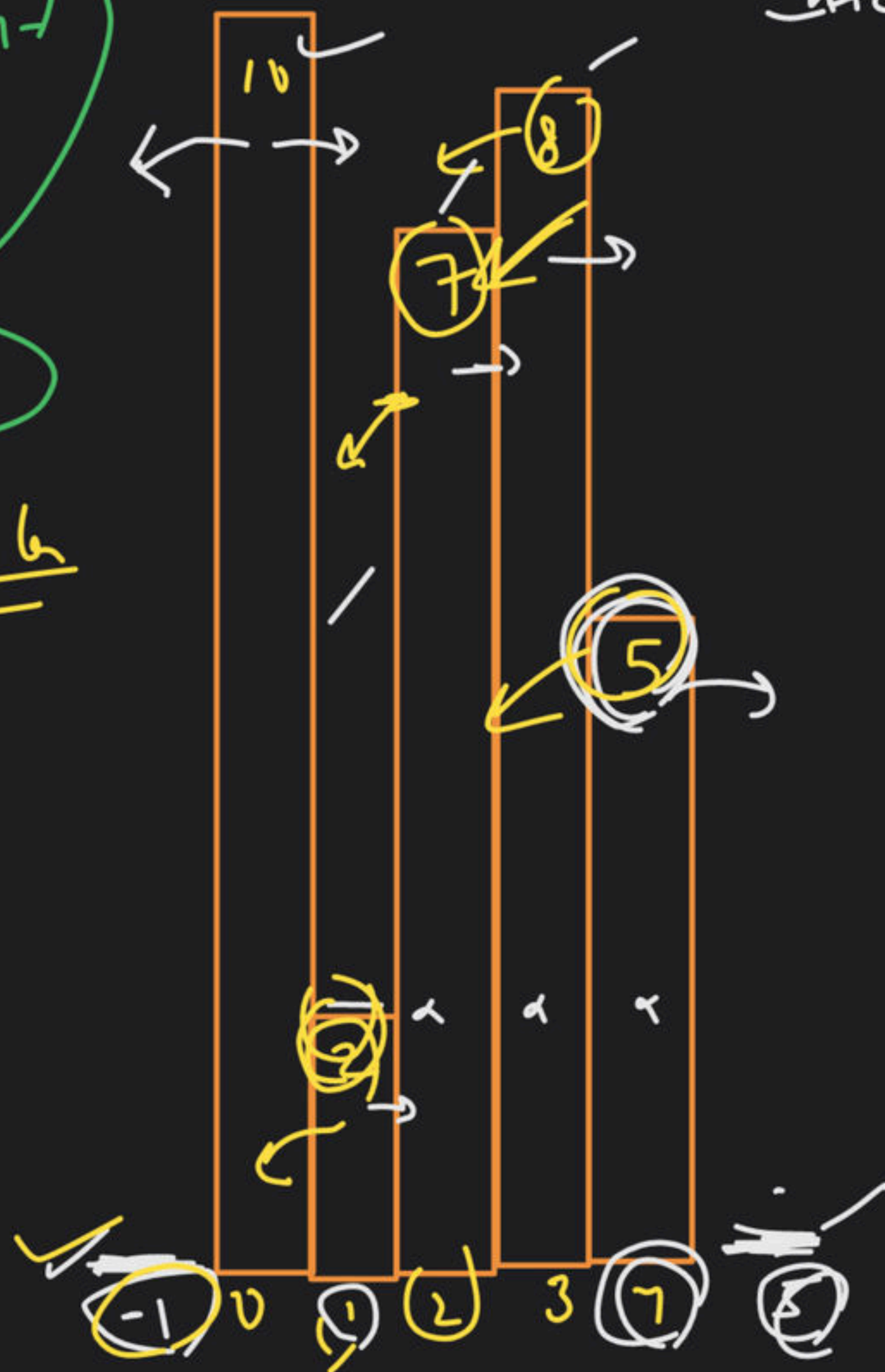
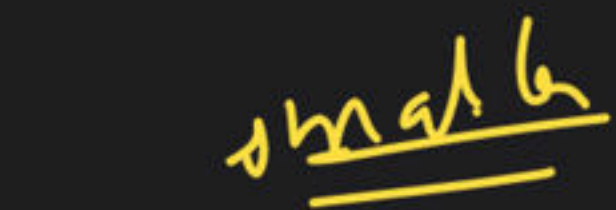


(10), (10)

$$7 \times 2 = 14$$

$$8 \times 1 = 8$$

~~$5 \times 3 = 15$~~



N-p- /

↘ NS → [1, 5, 4, 7], 5]

$$p_s = [-1, -1, 1, 2, 1]$$

index of prev smaller element

$B = [1, 5, 2, 3]$

$L = [10, 12, 7, 8, 5]$

↓

10 12 8

15

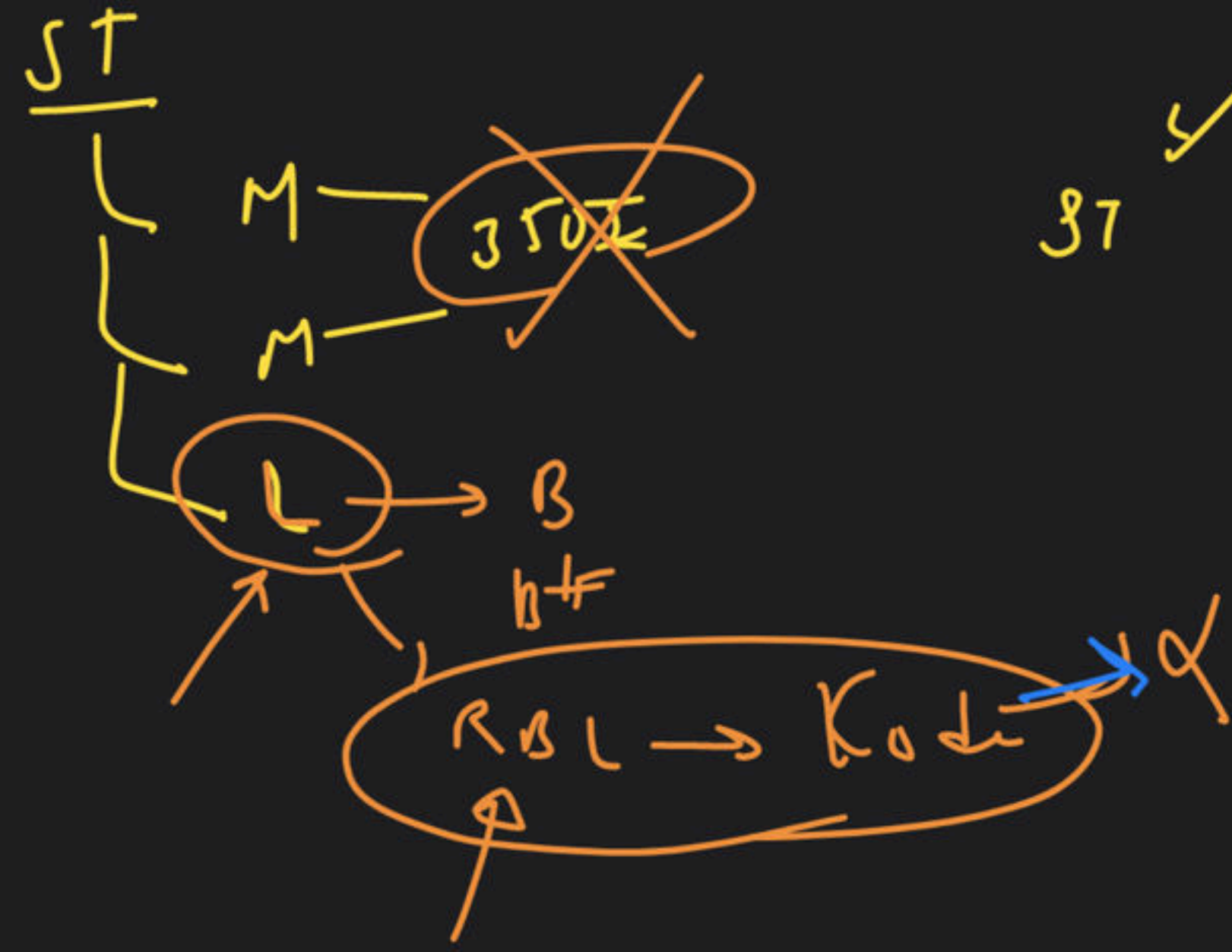


Paytm  $\rightarrow$  (10.5L)/A

M/M

1 Ladki

37



$\omega \rightarrow \text{stack} \rightarrow \text{index}$

arr  $(\text{s.top}())$

→ Amazon → final round → 1 question  
[Celebrity Problem]

Luck — CP  
LP  
CR  
FUMD → 15-20 min

↓  
GFG

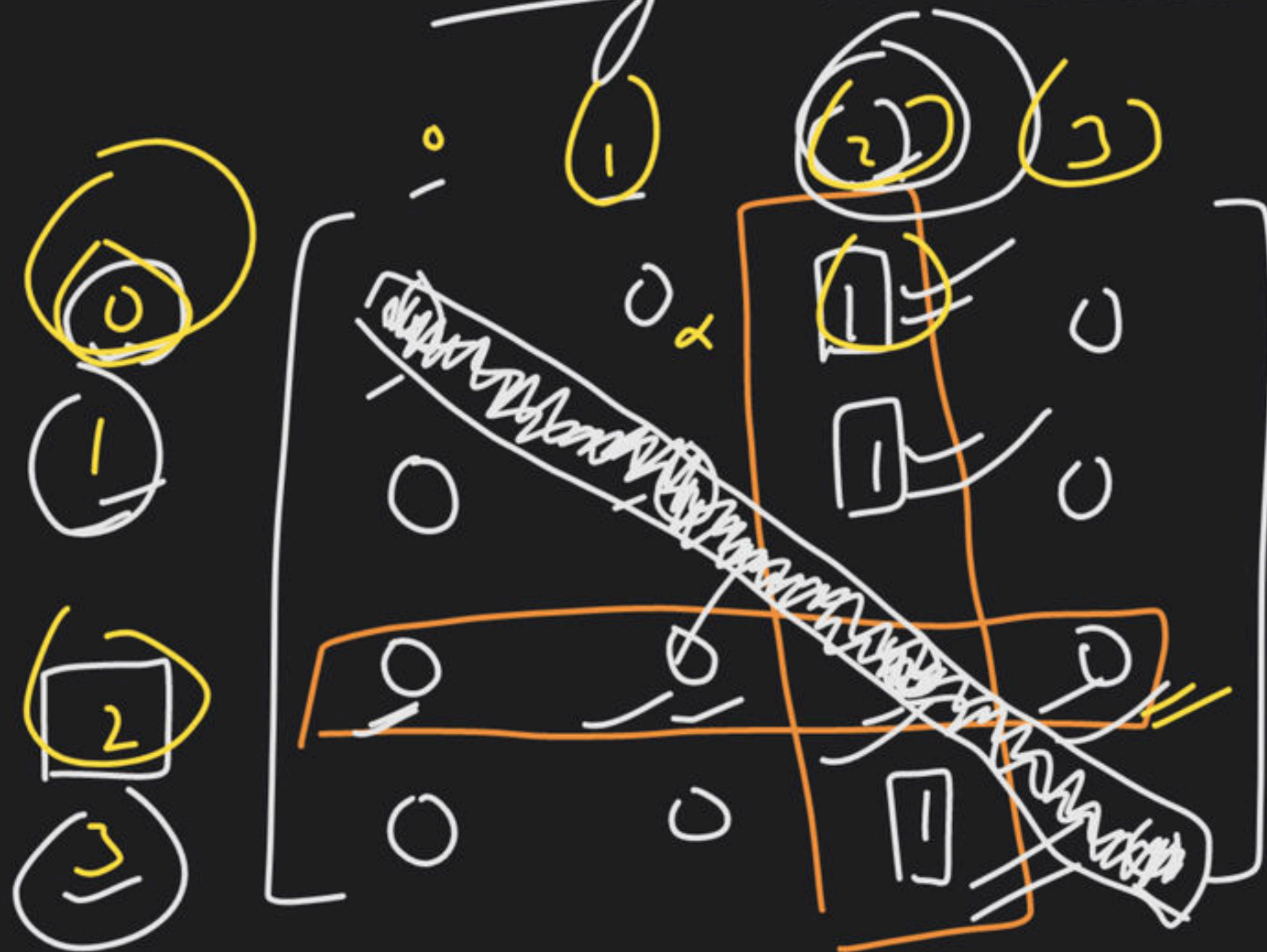
5-6 soln ✓  
1 soln → given

Int  
↓  
[C.P.]

↓  
[knows(A, B)]

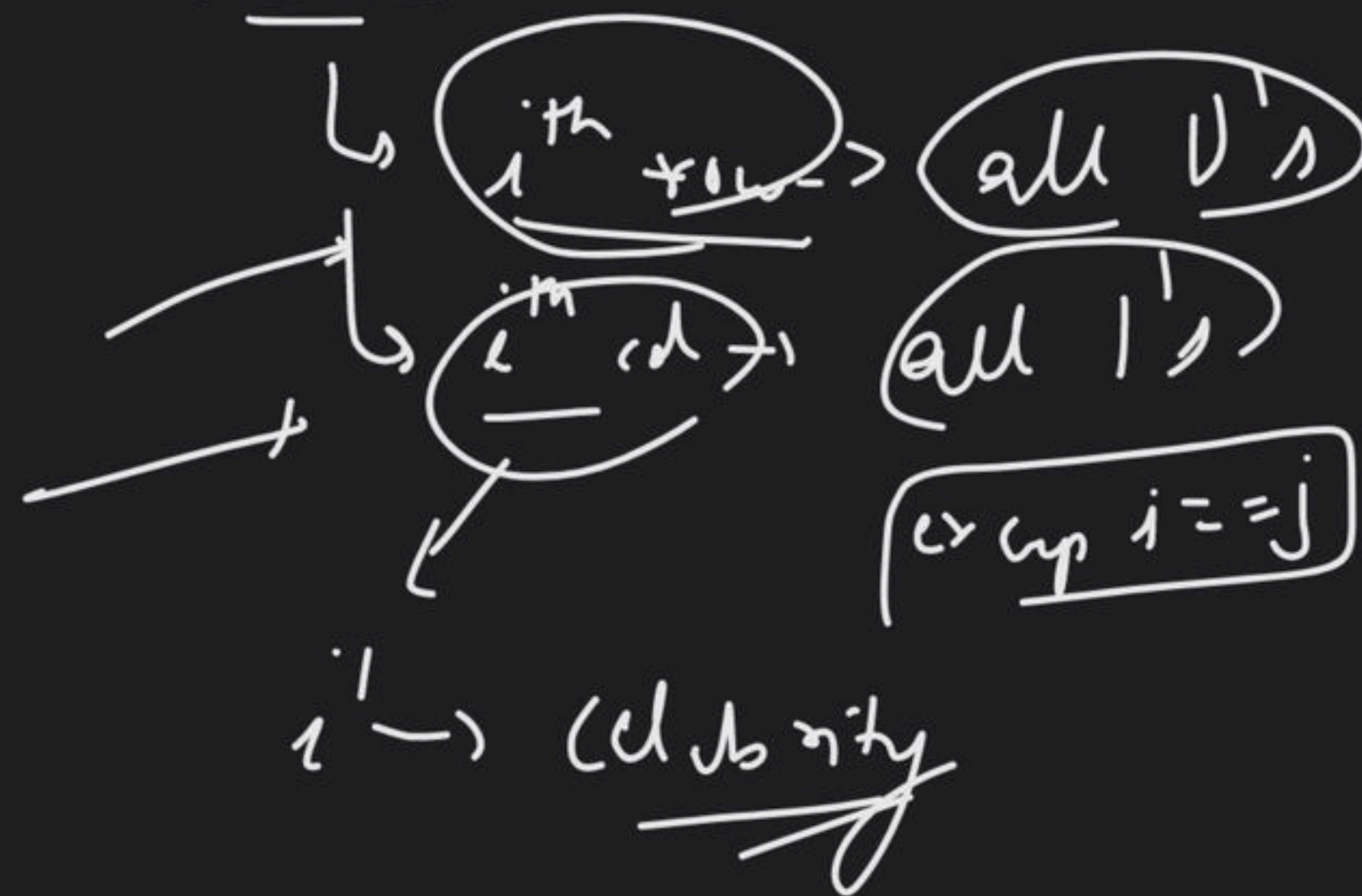


# Celebrity Problem



$i == j$  -> ignore

obs:-



#1 →

```
for (int i=0 ; i<n; i++)
```

```
{
```

```
// check row
```

```
for (int j=0 ; j<n; j++)
```

```
mat[j][i] →  
+
```

```
// check
```

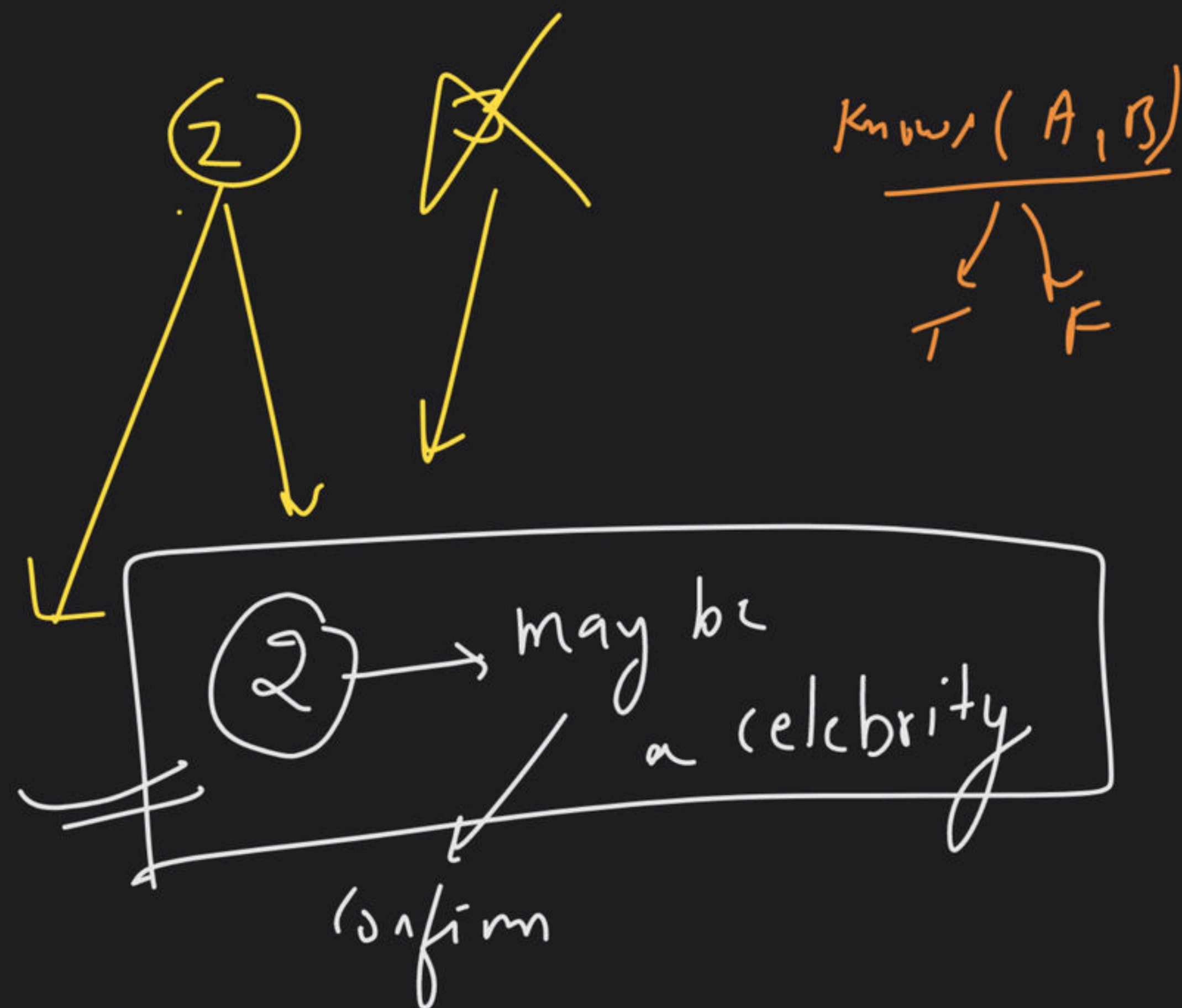
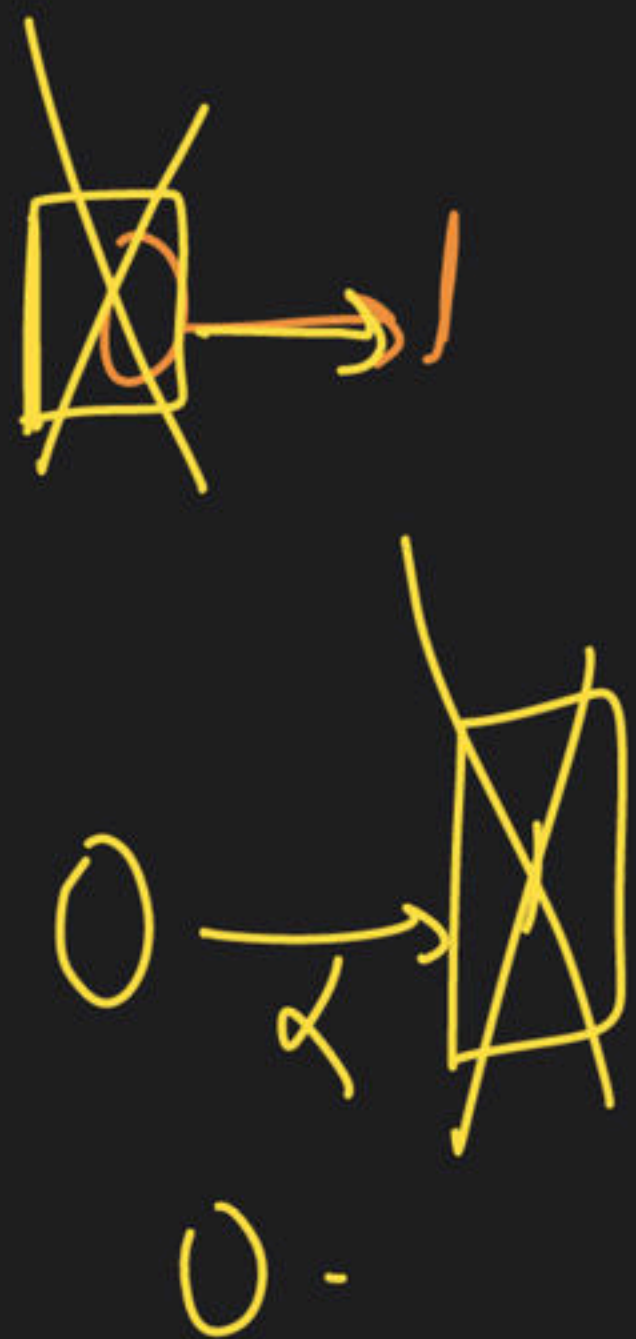
```
for ( )
```

```
mat
```

7.62  
↓  
 $O(n^2)$

3





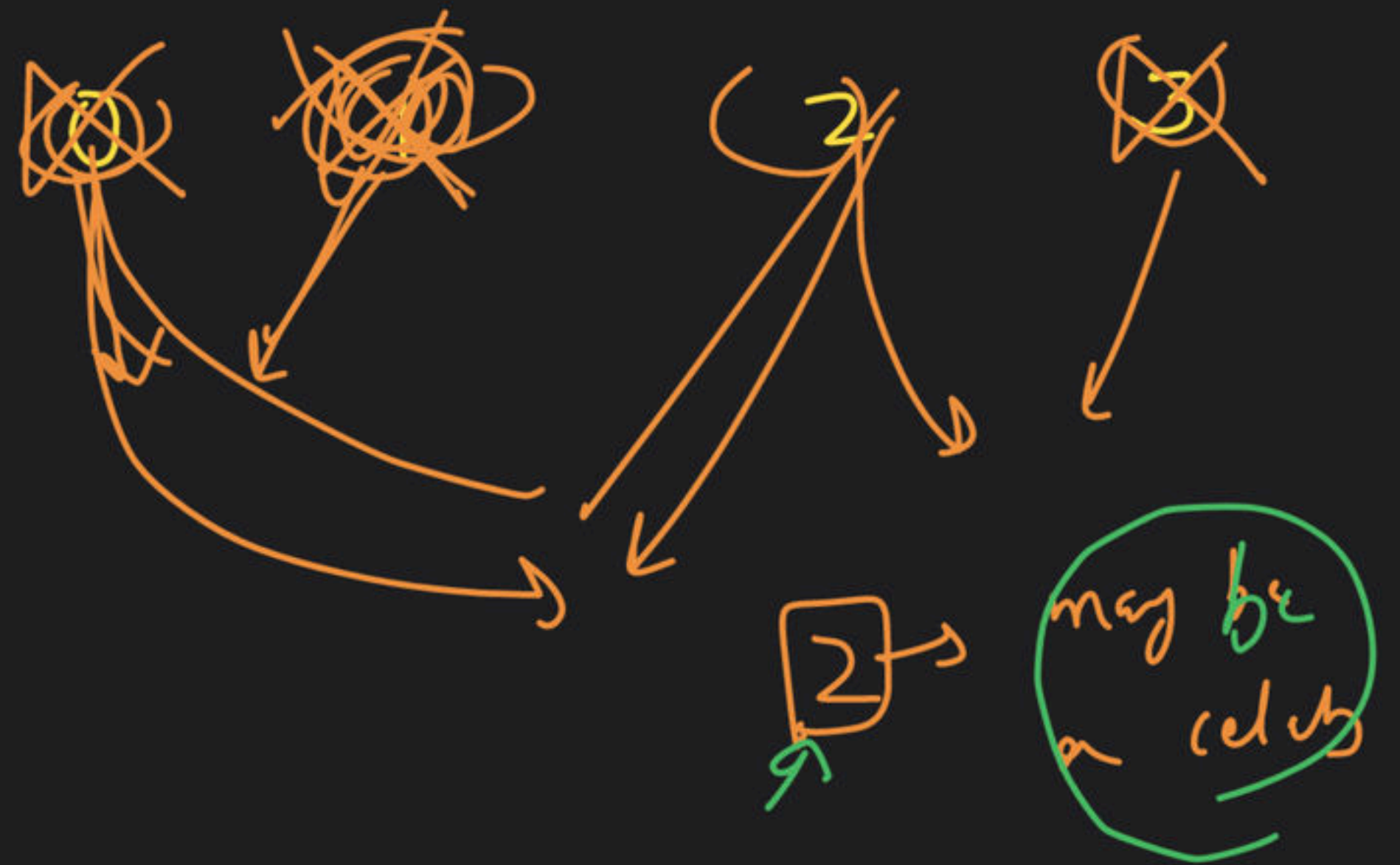
for ( —————> row

for ( —————> col



0  
 1  
 2  
 3

	0	1	2	3
0	<del>0</del>	0	1	0
1	0	<del>0</del>	1	0
2	0	0	<del>0</del>	0
3	0	0	1	<del>0</del>



obs

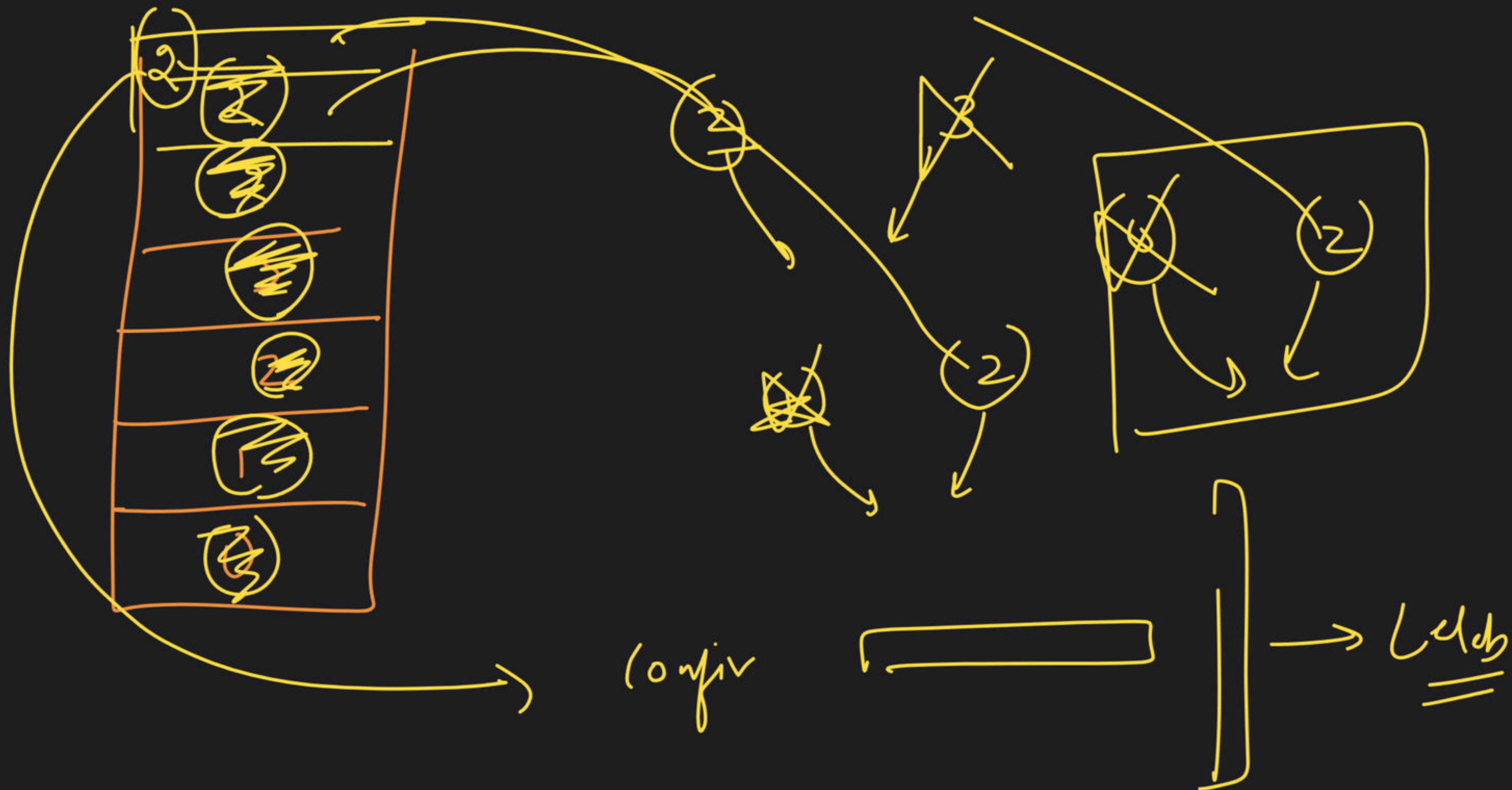
↳ all 0's → row → True

↳ all 1's col → True

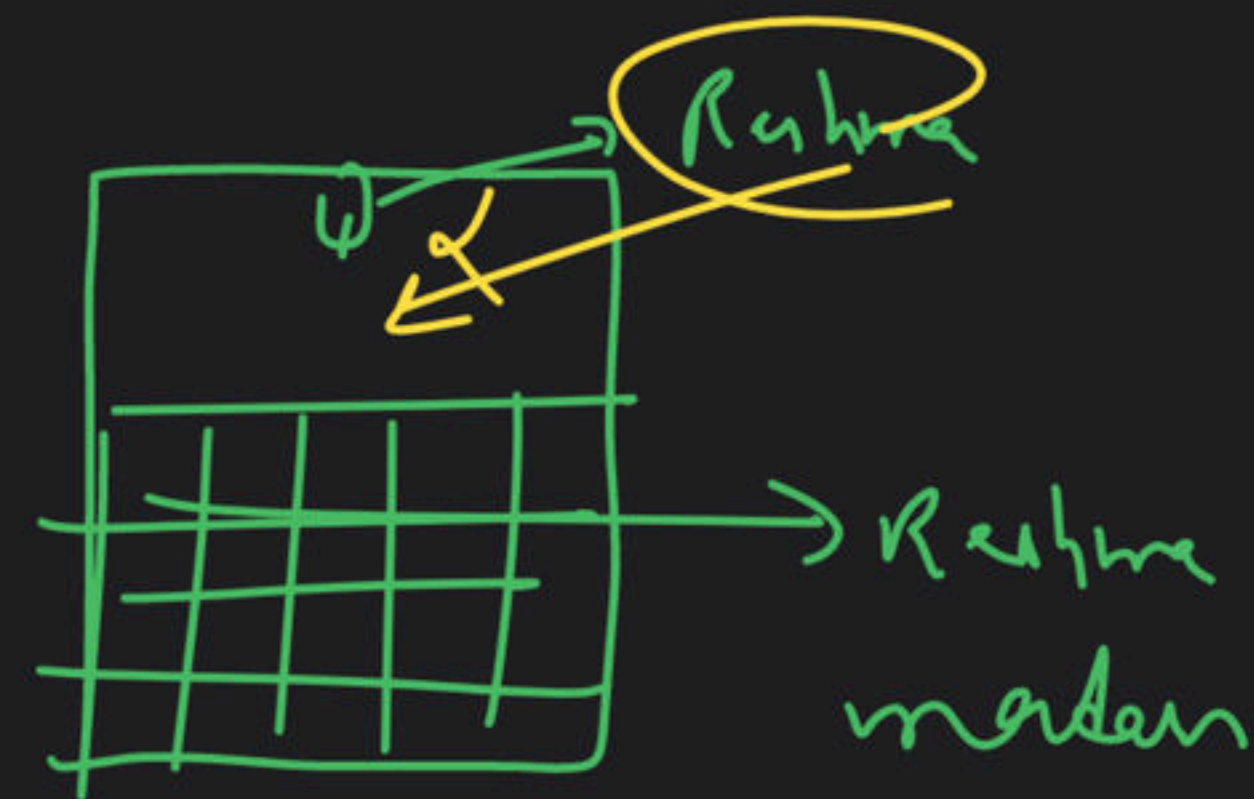
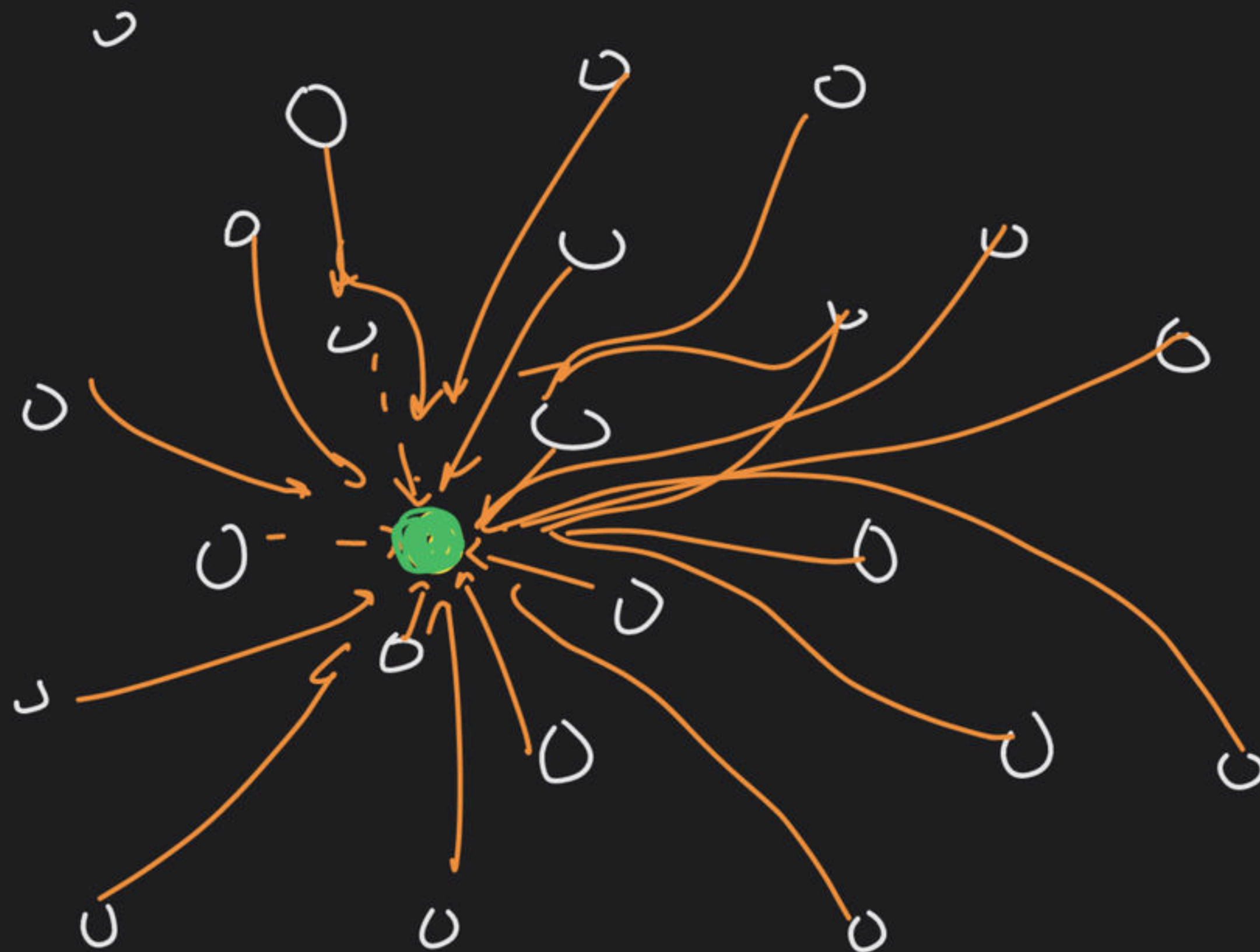
ji haan

(clarity

mai hunn



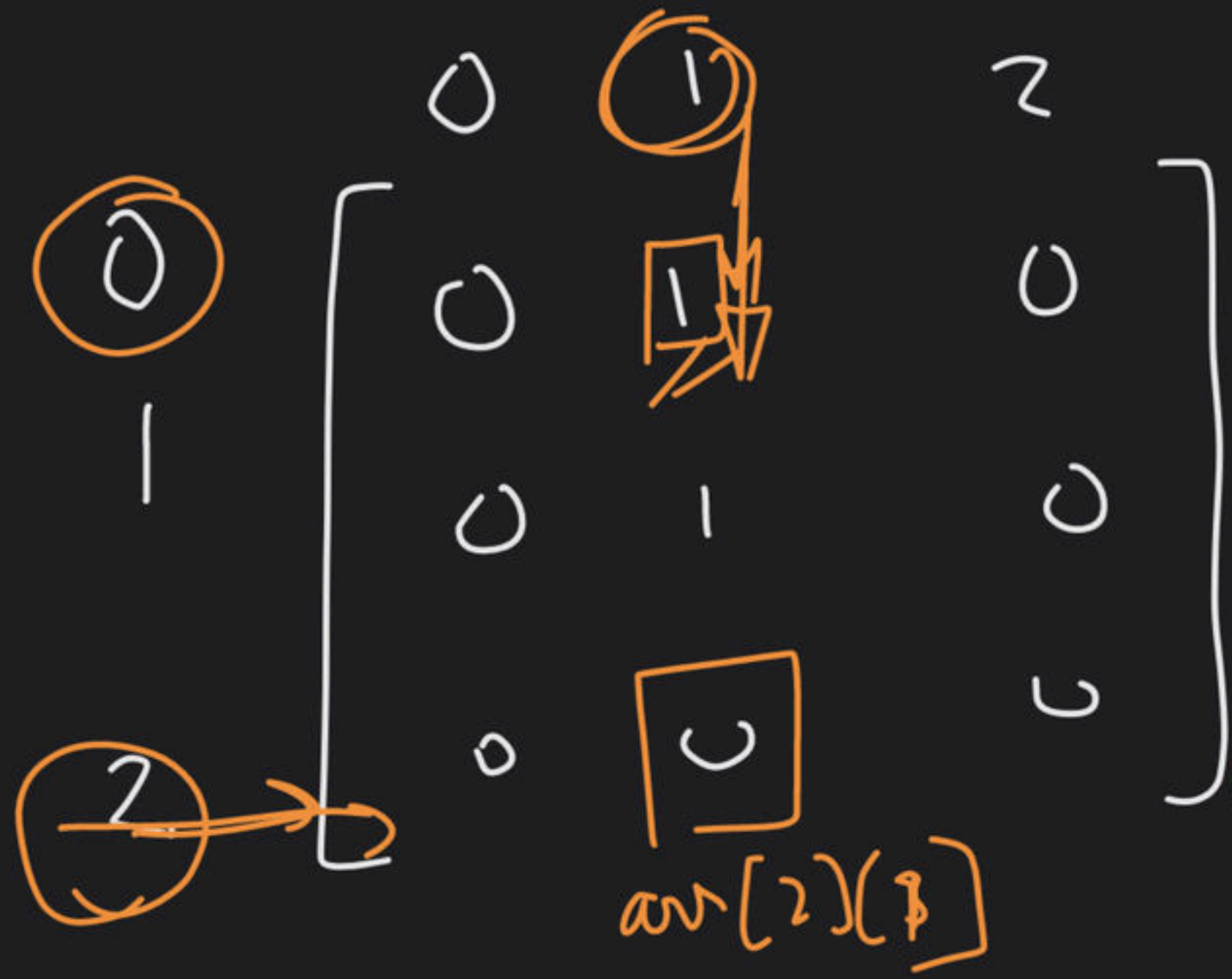








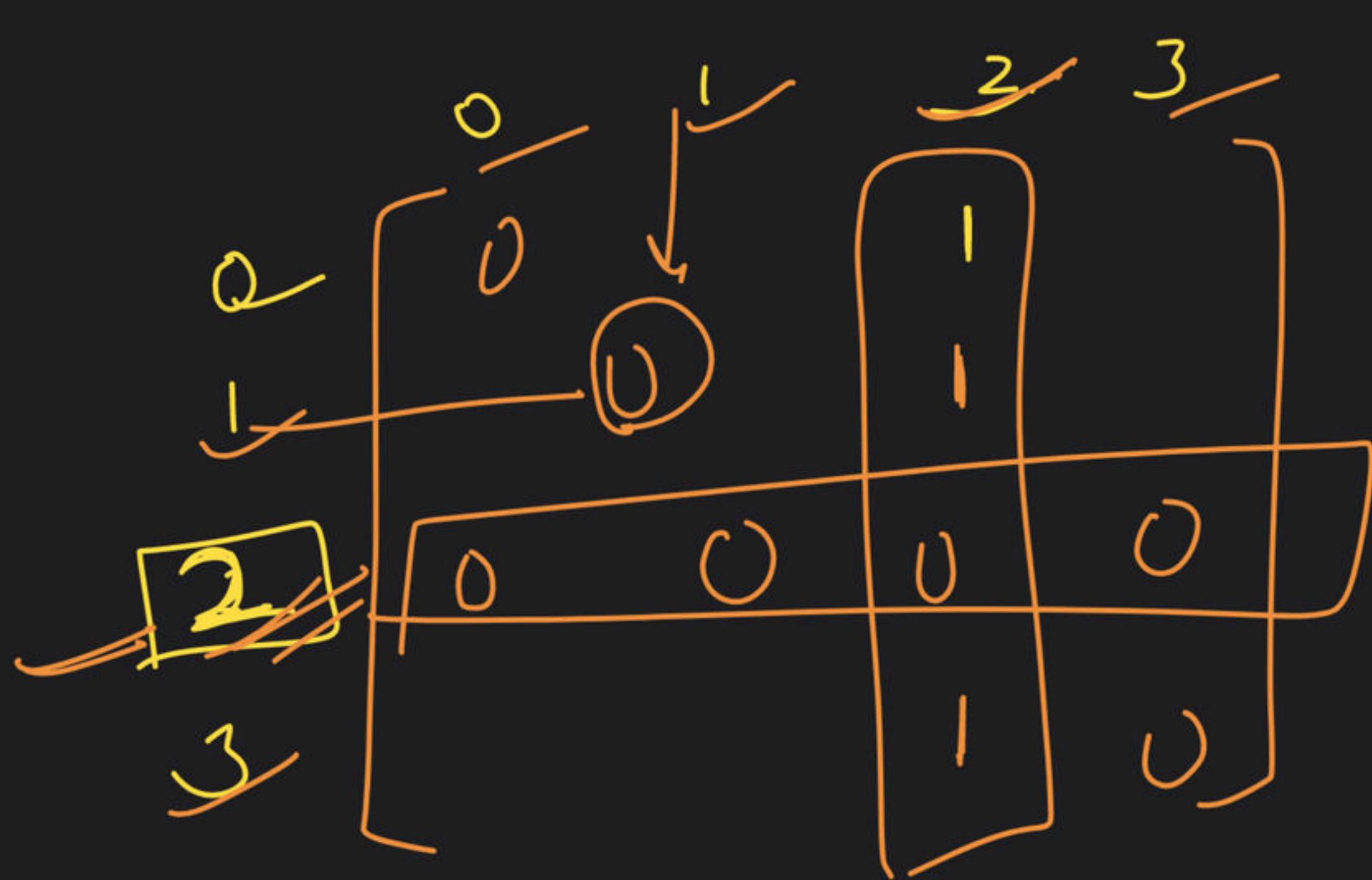




arr [0] [1] → 1  
↓  
Ja

1 → Jaantz hai  
0 → nahin jaantz

2 1



CP  
LRA

g4.1.

Code

- ↳ stack me din
- ↳ stack me 2 pop, 1 push
  - ↳ size
- ↳ verify
  - ↳ row
  - ↳ col
- ↳ row/col, ans

arr[1][1]

fln



→ naam mat le

→ don't take name

4:30pm

