

$\checkmark \rightarrow 1$
 $\times \rightarrow 0$

none

j

	0	1	2	3	4
0	0	✓	0	0	0
1	0	0	0	0	0
2	0	✓	0	✓	0
3	0	✓	0	0	0
4	0	✓	0	0	0

I
 celebrity → 2 → 1 → 0 → 3 → 4 →

II
 1 → 2 → 3 → 4 →

Celebrity - ① ✓

- Test case analysis
- brute force
- optimisation $O(n)$

III.

	0	1	2	3	4
0	0	✓	0	0	0
1	0	0	0	0	✓
2	0	✓	0	✓	0
3	✓	0	0	0	✓
4	0	✓	0	0	0

0 - celebrity ?

0th person know 1

$$[2][1] = _$$

2 know 1's

	0	1	2	3	4
0	0	✓	0	✓	0
1	0	0	✓	0	0
2	0	0	0	✓	0
3	0	0	0	0	✓
4	0	0	✓	0	0

celebrity → }

{ → all persons knew celebrity
 → but celebrity don't know anyone

→ is it possible to have multiple celebrity?

→ 0 celebrity possible?

optimisation Approach.

* Elimination Approach.

matrix

	0	1	2	3	4	5
0	0	1	1	0	1	0
1	0	0	1	0	1	0
2	1	1	0	0	1	1
3	0	0	0	0	0	0
4	1	0	1	0	1	0
5	0	1	0	1	1	0

$$5 \rightarrow 11 \quad (1) \quad 3$$

$$3 - P_2$$

$(\text{matrix}[3][5] == 1) \{$
 ⑤ may be celebrity
 } else {
 eliminate ⑤
 ∵ 5 is guaranteed not a
 prob. celeb.
 }
 prob(3)

② $O(2n) \rightarrow O(n)$

matrix[P2][P1]

5 } pop from sta.

$(\text{matrix}[4][5] == 1) \{$
 4 knew 5
 eliminate ④ push(5)

$$P_1 \rightarrow 3$$

$$P_2 \rightarrow 2$$

$(\text{matrix}[2][3] == 1) \{$

2 knew 3.
 } then eliminate ② push(3)

$$P_1 = 3$$

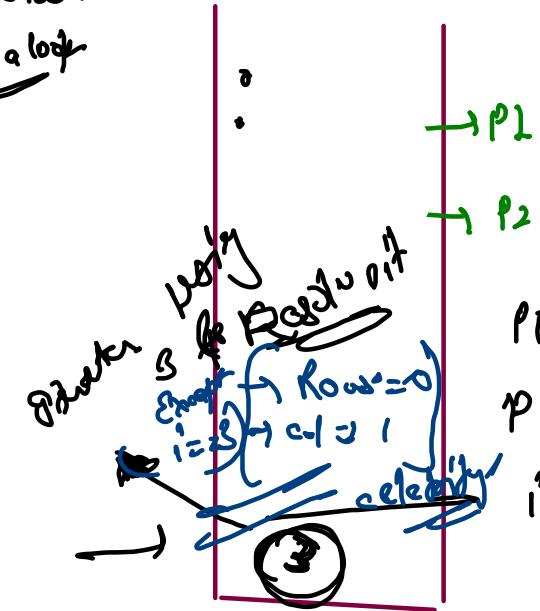
$$P_2 = 2$$

$\{ \text{if}(\text{matrix}[1][3] == 1) \{$

eliminate ①

∴ because I know 3 is guaranteed 1 is not celebrity

- ① → fill & stack
- ② remove a lock



$$P_1 = 3$$

$$P_2 = 0$$

$\{ \text{if}(\text{matrix}[0][3] == 1) \{$
 // Guessed 0 is celebrity
 curr node 0
 ③, prob(3)

E.g. Let us say

18

5 12

14 19

22 28

25 27

27 30

Then the output

✓ 112

✓ 14 19

✓ 22 30

$$\left. \begin{array}{l} \rightarrow \underline{L} - 8 \\ \rightarrow \underline{S} - 12 \\ \rightarrow \underline{14} - 19 \\ \rightarrow \underline{22} - 28 \\ \rightarrow \underline{\cancel{25}} - 27 \\ \rightarrow \underline{27} - 30 \end{array} \right\}$$

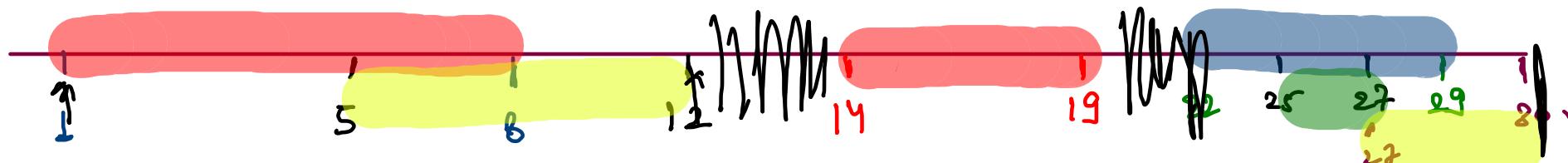
O/p. $\rightarrow \left. \begin{array}{l} \underline{L} - 12 \\ 14 - 19 \end{array} \right\}$ Marg.

$\left. \begin{array}{l} 22 - 30 \end{array} \right\}$ \rightarrow No freq'g.

18, 5 - 12

14 - 19

22 - 30 $\rightarrow 22 - 9 + 25 - 29 + 27 - 30$



$\checkmark 22 - 28 \rightarrow$ Range (on end)

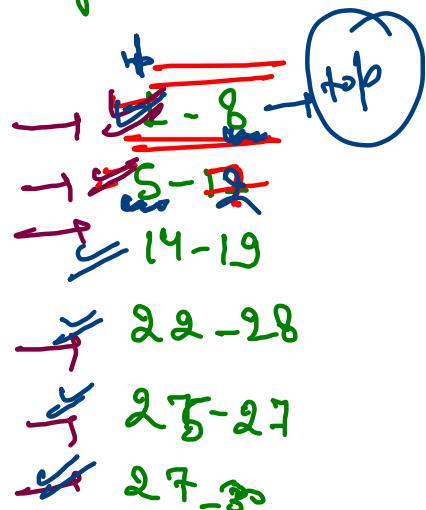
$\downarrow - 8$

$\checkmark 25 - 27$

$\checkmark 14 - 19$

$27 - 30$

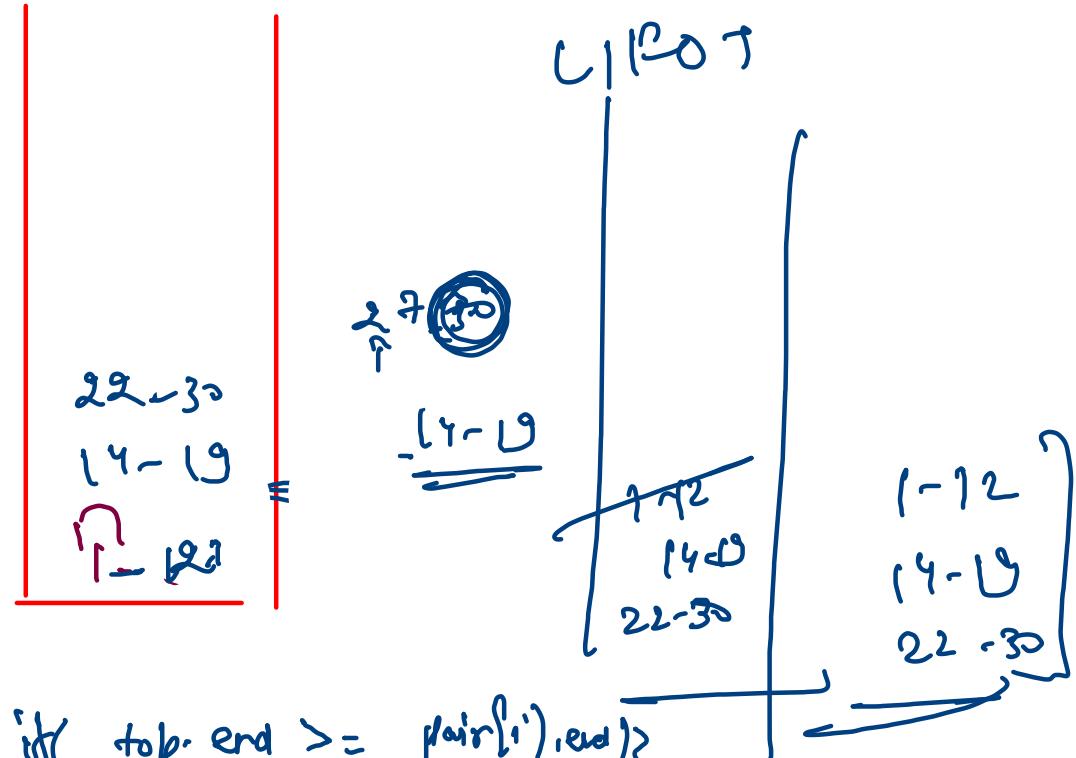
$\checkmark 5 - 12$



$22 - 30$
 $14 - 19$
 $5 - 12$

① sort on the basis of start time,
 ↗ how ??

②



if (top.end >= pair[i].end) {
 // merge } b

} else

2 digits

1 2 } →
pattern length: 2

$d^2 q$
 $\begin{pmatrix} 2 \\ 1 \end{pmatrix}$

d^2 + 1 → pattern length: 2
 ↓ (1, 9)
 ↓ smallest right
 ↓ (n+i) digit no. in New =
 ↓ (4, 3, 2, 1)

4 → 3 → 2 → 1 }
 ↓ ↓ ↓ ↓

2 ↓ x
 3 2 ↓ x
 4 3 2 1 // ↙

→ 1-9

→ unique digits

→ smaller No.

→ pattern length: 8

digit 9

Example

Simplest

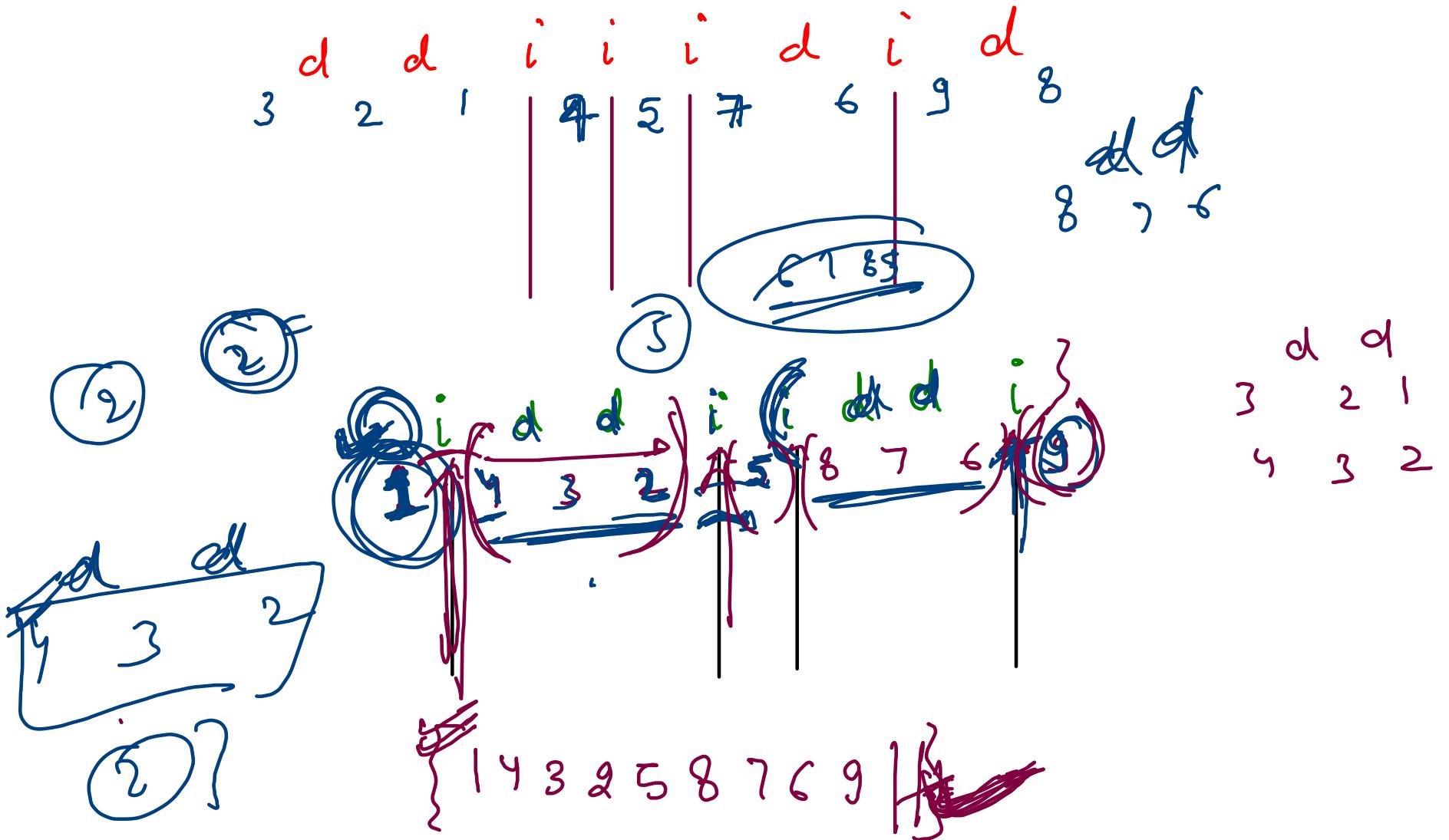
log. - II

i i i }
 ↓ ↓ ↓ ↓ ↓ ↓

→ $d^2 d^1 i \rightarrow d^5 d^4 i^8 d^7$
 ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓

3 2 1 → 6 5 4 - 8 7 // ↙

28





$(\text{charAt}(i) == 'a')$ {
cont → print
ignore}

d d i d i i d ·
3 2 1 5 4 6 8 7 //

3 2 1 5 4 6 8 7 //

cont at 2: 3 4 8 6 7 8 ⑤

$\text{charAt}(i) == 'i'$

↑ { print stack -
after removal)

d d i d i i d
↑ ↑ ↑ ↑ ↑ ↑ ↑

After loop → stack print

→ push(c#),
→ ignore c#

J f i J i d d d i
 d d i i d d d i
 3. 2 1 . 4 8 7 6 5 9



3 2 1 4 8 7 6 5 9

Count = 2 3 4 5 6 7 8 9 10

$\text{count} = \text{d}$
 $\text{push}(\text{count})$
 $\text{count} + 1;$

$\text{count} = \text{i}$
 $\text{print Stack (from removal)}$
 push count
 $\text{count} + 1;$

After loop - print Stack

Desired function:

Stack →

using array

push →
pop
top
size.
display.

Stack

class



↑
top = -1
→ top

for

ref = 30;
tos

st

capacity

char [top + 1] = 10;

push 10
push 20
push 30

top > stack over

top + 1

30 20 10

for

0

Mong.

Top
data,

Stack overflow → over capacity

Stack Underflow → st.size = 0
& st.pop();