Interview Problems

Content

- Analysing constraints
- Problems

Avg. PSP of the batch

- Wednesday → Friday → Saturday → Monday

69.5 → 69.1 → 66.1 → 70%

Personal Goal → as close to 100%

If you are stuck at a problem {max time 25 mins}.

- Hint 1 → Hint 2 → Video solution
- TA {video call Help Request
- Post in WA group
- Reach out to me.

Analysing Constraints

1 sec $\longrightarrow$ $\approx 10^8$ iterations.

N	TC	# iteration	Result
10^5	$O(N^2)$	10^{10}	TLE
10^3	$O(N^2)$	10^6	Pass
20	$O(2^N)$	2^{20}	Pass
10^6	$O(N)$	10^6	Pass.

Q) Given a binary array of 0's & 1's. ***
Find the max # of consecutive 1's that can be obtained by updating atmost one 0 to 1

length A > 0

$A = [1\ 1\ \underset{1}{\cancel{0}}\ 1\ 1\ 0\ 1]$ ans = 5

$A = [1\ 1\ 0\ 1\ 1\ \underset{1}{\cancel{0}}\ 1\ 1\ 1]$ ans = 6

$A = [0\ 1\ 1\ 1\ \underset{1}{\cancel{0}}\ 1\ 1\ 0\ 1\ 1\ 0]$ ans = 6

$A = [1\ 1\ 1\ 1\ 1]$ ans = 5

Edge case : all ones.

→ count of ones == length of A

$A = [1\ 1\ \underset{1}{\cancel{0}}\ 0\ 1\ 1]$ ans = 3

$A = [1\ 1\ \underset{1}{\cancel{0}}\ 1\ 1]$ ans = 5

$A = [\underset{1}{\cancel{0}}\ 0\ 0\ 0\ 0]$ ans = 1

[1 1 ~~0~~ 1 1 0 1 1 1]

2 1 2

5

ans = ~~5~~ 6

2 3

[1 1 0 1 1 ~~0~~ 1 1 1]

1

6

2 0

ans = ~~4~~ 6

[0 1 1 1 0 1 1 0 1 1 ~~0~~]

1

3

Pseudocode

```
// handle the edge case      [1 1 1 1 1]
totalOnes = 0
for i ——> 0 to N-1 {
      if (A[i] == 1) {
            totalOnes ++
      }
}
if (totalOnes == N) {
      return totalOnes
}
```

```
ans = 0
                                                  TC : O(N)
for i ---> 0 to N-1 {                             SC : O(1)
    if ( A[i] == 0 ) {
        count = 1      // count of ones.
        // #ones on left
        for j ---> i-1 to 0 {
            if ( A[j] == 1 ) count ++
            else break
        }

        // #ones on right
        for j ---> i+1 to N-1 {
            if ( A[j] == 1 ) count ++
            else break
        }

        ans = max( ans , count )
    }
}

return ans
```

$$[\,0 \;\; 1 \;\; 1 \;\; 1 \;\; 0 \;\; 1 \;\; 1 \;\; 0 \;\; 1 \;\; 1 \;\; \overset{i}{\cancel{0}}\,]$$

1

count = 3

ans = $\cancel{4}$ 6

$$[\,0 \quad 1 \quad 1 \quad 1 \quad 0 \quad 1 \quad 1 \quad 0 \quad 1 \quad 1 \quad \overset{i}{\underset{1}{\cancel{0}}}\,]$$

Observation $\longrightarrow$ we are visiting every element atmost 3 times.

TC : $O(3*N) \longrightarrow O(N)$

$$\begin{array}{ccccc} 0 & 1 & 2 & 3 & 4 \\ \end{array}$$
$$[\,1 \quad 1 \quad 0 \quad 1 \quad 1\,]$$

i	j	# iterations
0	—	1
1	—	1
2	$1 \to 0$	5
	$3 \to 4$	
3	—	1
4	—	1

$O(N^2)$

$\simeq 9 \quad 2N \longrightarrow O(N)$

$A = [\,0 \quad 0 \quad 0 \quad 0 \quad 1 \quad 0 \quad 0\,]$

						Psum	Ssum	Sol"
	1	1	0	1	1			
PS	1	2	0	1	2			
	2	1	0	2	1	SS		

$2 + 1 + 2 = \underline{\underline{5}}$

Q> Given a binary array of 0's & 1's. ***
Find the max # of consecutive 1's that can be obtained by swap atmost one 0 to 1
length A > 0

A = [1 ~~0~~ 1 1   0 ~~1~~]   ans = 4
    1   0

A = [~~1~~ 1 ~~0~~ 1 1 1]   ans = 5
  0   1

NOTE : My ans can never exceed totalOnes in A.

[~~1~~ 1 0 1 1 ~~0~~ 1 1 1]   ans = 6   same as before
  0   1

[~~1~~ 1 ~~0~~ 1 1 1]   ans = 5   -1 from Before..
  0   1

Same as above solution

```
ans = prev solution
if ans > totalOnes {
        return ans - 1
}
return ans
```

Break
22:38

Q> Given an integer array, find the majority element.

Majority element $\longrightarrow$ freq $> N/2$

Note : If no majority element is present $\longrightarrow$ -1

A =	2	1	4
freq	1	1	1
	$1 > 1$	$1 > 1$	$1 > 1$
	False	False	False

$N = 3$ $\quad N/2 = 1$

ans = -1

	0	1	2	3	4	5	6	7
A =	3	4	3	2	4	4	4	4
freq	2	5	2	1	5	5	5	5

$2 \not> 4$ $\quad 5 > 4$ ✓

ans = 4

	0	1	2	3	4	5	6	7
A =	3	3	4	2	4	4	2	4
	2	2	4	2	4	4	2	4

ans = -1

	0	1	2	3	4	5	6	7	8	9	10
A =	3	4	3	6	1	3	2	5	3	3	3
	6	1	6	1	1	6	1	1	6	6	6

$6 > 5$

	0	1	2	3	4	5	6	7	8	9
A =	4	6	5	3	4	5	6	4	4	4
	5	2	2	1	5	2	2	5	5	5

$5 \not> 5$

ans = -1

How many majority element can be there?

Assume there are two majority elements

a b

$freq(a) > \frac{N}{2}$ $freq(b) > \frac{N}{2}$

$freq(a) + freq(b) > N$ ✗

Proof by contradiction. There is only one majority ele.

1 1 1 2 2 2 2

$a = 1$ $b = 2$

$freq(1) + freq(2) = N$

Lets assume m is majority element

$\longrightarrow$ $freq(m) > \frac{N}{2}$

$$\underset{> N/2}{freq(m)} + \underset{< N/2}{freq(!m)} = N$$

There is always only one majority element.

Bruteforce —

TC : $O(N^2)$

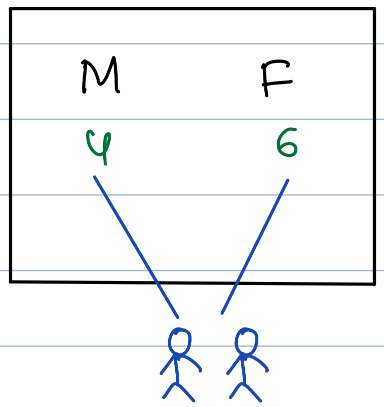
SC : $O(1)$

For each element u

Find its freq if $freq > N/2$ return u

return -1

Moore's Voting Algorithm.



M

M F

M F

M

F

Puru

Sujoy

Khushi

Aswin

Subham

Idea $\longrightarrow$ If we pair up any two distinct elements and remove them majority still remains the same.

	0	1	2	3	4	5	6	7
A =	3	4	3	2	4	4	4	4

major = $\cancel{3}$ $\cancel{3}$ 4

freq = $\cancel{1}$ $\cancel{0}$ $\cancel{1}$ $\cancel{0}$ $\cancel{1}$ $\cancel{2}$ $\cancel{3}$ 4

A = 1 2 3

major = ~~-1~~ ~~1~~ 3
freq = ~~0~~ ~~1~~ ~~0~~ 1 → how to verify?

Find the freq of 3 in A and check if freq > $\frac{N}{2}$
else
return -1

A = 4 4 4 4 5 2 3

major = ~~-1~~ 4
freq = ~~0~~ ~~1~~ ~~2~~ ~~3~~ ~~4~~ ~~3~~ ~~2~~ 1

freq(4) = 4 > 7/2 = ans = 4

A = 4 4 4 4 2 2 2 1

major = ~~1~~ 4
freq = ~~0~~ ~~1~~ ~~2~~ ~~3~~ ~~4~~ ~~3~~ ~~2~~ ~~1~~ 0

~~4 > 4~~ ans = -1

Pseudocode

```
major = -1        freq = 0
for i ⟶ 0 to N-1 {
        x = A[i]
        if ( freq == 0) {
                major = x
                freq = 1
        }
        else {
                if ( major == x ) {
                        freq++
                } else {
                        freq--
                }
        }
}
count = 0
for i ⟶ 0 to N-1 {
        if (major == A[i]) { count++ }
}

if (count > N/2 )   return major

return -1
```

TC : $O(N)$
SC : $O(1)$

Q> Given a 2D matrix, make all the elements in both row, column 0 if any cell in that row or column is equal to 0

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

$$A \begin{bmatrix} 1 & 0 & 2 \\ 3 & 4 & 5 \\ 6 & 7 & 8 \end{bmatrix} \qquad \begin{bmatrix} 0 & 0 & 0 \\ 3 & 0 & 5 \\ 6 & 0 & 8 \end{bmatrix}$$

Idea 1

$$\begin{array}{cc} \begin{array}{cccc} & & T & T \end{array} & \\ \begin{bmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 0 \\ 9 & 2 & 0 & 4 \end{bmatrix} & \begin{array}{c} \\ T \\ T \end{array} \end{array}$$

Mark a row or column for making zero later.

Idea 2

$$\begin{bmatrix} \overset{\text{inf}}{\cancel{1}} & \overset{\text{inf}}{\cancel{2}} & 0 & \overset{\text{inf}}{\cancel{4}} \\ \cancel{5}\,\text{inf} & \cancel{6}\,\text{inf} & \cancel{7}\,\text{inf} & 0 \\ \cancel{9}\,\text{inf} & \cancel{2}\,\text{inf} & 0 & \cancel{4}\,\text{inf} \end{bmatrix}$$

Whenever you find a 0 make the entire row and col = inf only if val != 0

Change all inf → 0

inf = INT_MAX

$$\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

---

why update to inf and not 0

$$\begin{matrix} 1 & 0 \\ 1 & 1 \end{matrix} \qquad \begin{matrix} 0 & 0 \\ 1 & 0 \end{matrix}$$

$$\begin{matrix} 0 & 0 \\ 0 & 0 \end{matrix}$$

Expected TC : O(N*M) SC (N*M)

SC O(N+M)

HW → SC O(1)

Doubt session