

# # Weekly Contest (LC) - 439 :-

A] 3471. Find the largest among missing integer :- ✓

B] 3472. Longest Palindromic Subsequence After at most k ops :- ✓

→ Approach - 1 :-

① Subsequence Palindrome

$\text{? ? a ? a ?}$  any other case  
 ↳ Palin. Subseq. of same element

Best case → happy case :  $[a \ b \ c \ c \ b \ a]$

↳ entire string is already a palind.

$\text{? a b b ? a}$

↳ any other case  
 ↳ Palindrome subseq. of different elements

② Longest

• comparing string str. :  $\_ \_ \_ \_ \_ \_$

↳ we can only find Pal. substring ⇒ can't skip (2/w)  
 but we need Pal. subseq. ⇒ we can skip elements (b/w)

• no need to think which character need to be changed to what.

$b \rightarrow c \rightarrow d$

2 ops. ⇒  $b \rightarrow d$

$\begin{matrix} (1) & & (3) \\ a & \rightarrow & c \end{matrix}$

cost =  $|c - a|$ , 2 ops.

$\begin{matrix} (1) & & (26) \\ a & \rightarrow & z \end{matrix}$

cost =  $|a - z|$ , 25

↳ Forward cost

Reverse =  $26 - ab(z - a) = 26 - 25 = 1$

cost ↳ Reverse op.

$\begin{matrix} & i & & j \\ & \boxed{a} & b & c & e & \boxed{d} \\ & \downarrow & & & & \downarrow \end{matrix}$

①  $\text{? } b \ c \ e \ d$   
 skip(L)

②  $a \ b \ c \ e \ ?$   
 skip R

③ convert  $a \rightarrow d$   
 cost = 3  
 $d \ b \ c \ e \ d$

Now ①, ②, ③ have 3 options:-

Left Skip  
 Right Skip  
 Convert L or R

states req. for DP  $\rightarrow [i][j][k]$   $\rightarrow$  max op. we can apply  
 $TC = O(200 \times 200 \times 200) = 8 \times 10^6$

c) 3473. Sum of K subarrays with length at least m ✓

→ Approach -1:- Rem. ~~some~~ arrays to be taken

- Targets:  $K$  subarrays (non-overlapping)
- subarr len.  $\geq m$   $m \in [1, 3]$  constraint
- At any point of arr  $\rightarrow$  see options

nums = [1, 2, -1, 3, 3, 4]

$\Rightarrow$  RECURSION

start a sub-array

how do we know

it is already started?

hasStarted    ?

OR canExtend    -----

Both means same thing

nums [1, 2, -1, 3, 3, 4]

mem = 2

canExtend = 0

(Starting from  $i=0$ )

canExtend = 0

take the subarray

means subarray hasn't started yet

skip the curr idx (go to  $i+1$ )

$\hookrightarrow$  solve( $i+1$ , mem, canExtend = Remains as is)  
 (0 here)

but len  $\geq m$



$i$                    $i+m$   
 $\downarrow$                    $\downarrow$

nums = [1, 2, -1, 3, 3, 4]

prpr. ele. taken here

[Your sum from  $i$  to  $i+m-1$ ]  $\rightarrow$  currSum + solve( $i+m$ , mem.-1, canExtend)

YES  $\uparrow$

- Range sum of  $m$  elements
- can leverage prefix sum even though ( $m \leq 3$ )

$i$   
 $\downarrow$   
 nums[] = [1, 2, -1, 3, 3, 4]

mem = 1

canExtend = Yes

Extend it forward

nums = [1, 2, -1, 3, 3, 4]

$num[i] + \text{solve}(i+1, \text{mem},$   
 $\text{canExtend})$   
 YES

skip & start afresh

nums [1, 2, -1, 3, 3, 4]

$\text{solve}(i, \text{mem}, \text{canExtend} = 0)$   
 not going fwd, starting again