

Max Consecutive Ones III:-

1004. Max Consecutive Ones III

Medium Topics Companies Hint

Given a binary array `nums` and an integer `k`, return the maximum number of consecutive 1's in the array if you can flip at most `k` 0's.

Example 1:

Input: `nums = [1,1,1,0,0,0,1,1,1,0]`, `k = 2`

Output: 6

Explanation: `[1,1,1,0,0,1,1,1,1,1]`

Bolded numbers were flipped from 0 to 1. The longest subarray is underlined.

Example 2:

Input: `nums = [0,0,1,1,0,0,1,1,1,0,1,0,0,0,1,1,1,1]`, `k = 3`

Output: 10

Explanation: `[0,0,1,1,1,1,1,1,1,1,0,0,0,0,1,1,1,1]`

Bolded numbers were flipped from 0 to 1. The longest subarray is underlined.

Constraints:

- $1 \leq \text{nums.length} \leq 10^5$
- `nums[i]` is either 0 or 1.
- $0 \leq k \leq \text{nums.length}$

at most k 0's can be flipped.

- This means variable sized sliding window having # of 0's $\leq k$.

- If # of 0's $> k$ shrink the window till 0's becomes less than k .

- record each valid window size in `max`

- return `max`.

```
C++ v Auto
1 class Solution {
2 public:
3     int longestOnes(vector<int>& a, int k) {
4         int ans=0;
5         int n=a.size();
6         int left=0, max=0;
7         for(int i=0; i<n; i++){
8             if(a[i]==0){
9                 max++;
10                while(left<=i && max>k){
11                    max--(a[left]==0)?max--:0;
12                    left++;
13                }
14            }
15            ans=max(i-left+1, ans);
16        }
17        return ans;
18    }
19 }
20 };
```

T.C: $O(n)$
S.C: $O(1)$