Facebook
Instagram → codestorywithMIK

CSwithMIK
(Twitter)

SLIDING WINDOW MECHANISM...

Video - 12

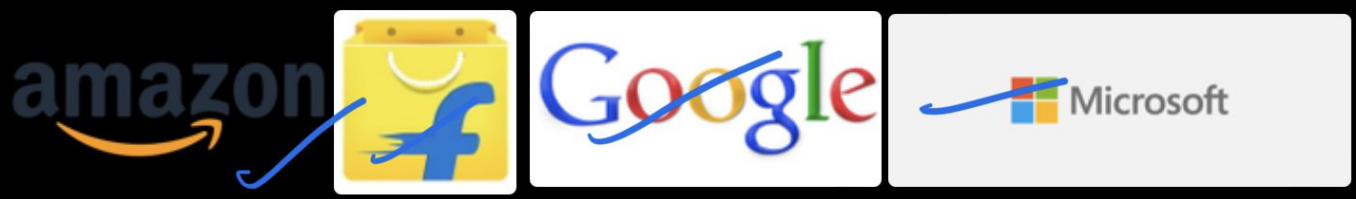
codestorywithMIK

Hard

Leetcode-239

Many Similar problems can be solved (even Hard ones)

Sliding Window Maximum



239. Sliding Window Maximum

Hard 15599 516 Add to List Share

You are given an array of integers `nums`, there is a sliding window of size `k` which is moving from the very left of the array to the very right. You can only see the `k` numbers in the window. Each time the sliding window moves right by one position.

Return *the max sliding window*.

Example :- nums = { 1, 3, -1, -3, 5, 3, 6, 7 } (0 1 2 3 4 5 6 7), K = 3

Output :- { 3, 3, 5, 5, 6, 7 }

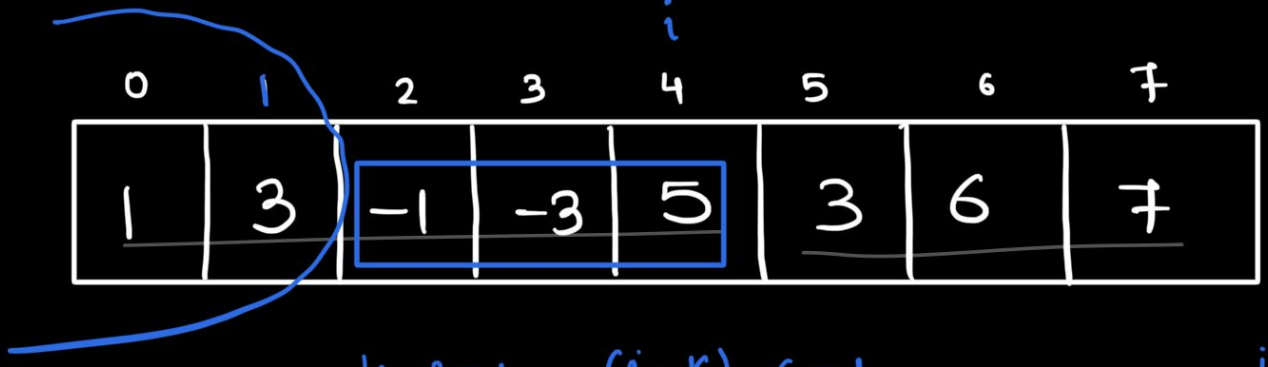
What we are going to learn today is one of the most important topics of Sliding Window.

"Monotonic Increasing / Decreasing"

stack
queue
deque

Intuition ...

K=3



4-3=1 (i-K) <=1

i-K=7-3
=4

idx 0 1 3 4 5 6 7

(X, 3, -1, -3, 5, X, X, 7)

{3, 3, 5, 5, 6, 7}

Decreasing order.

Monotonic Decreasing.

D.S. ?

Pop-back()

front()

push()

Deque.

Story :-

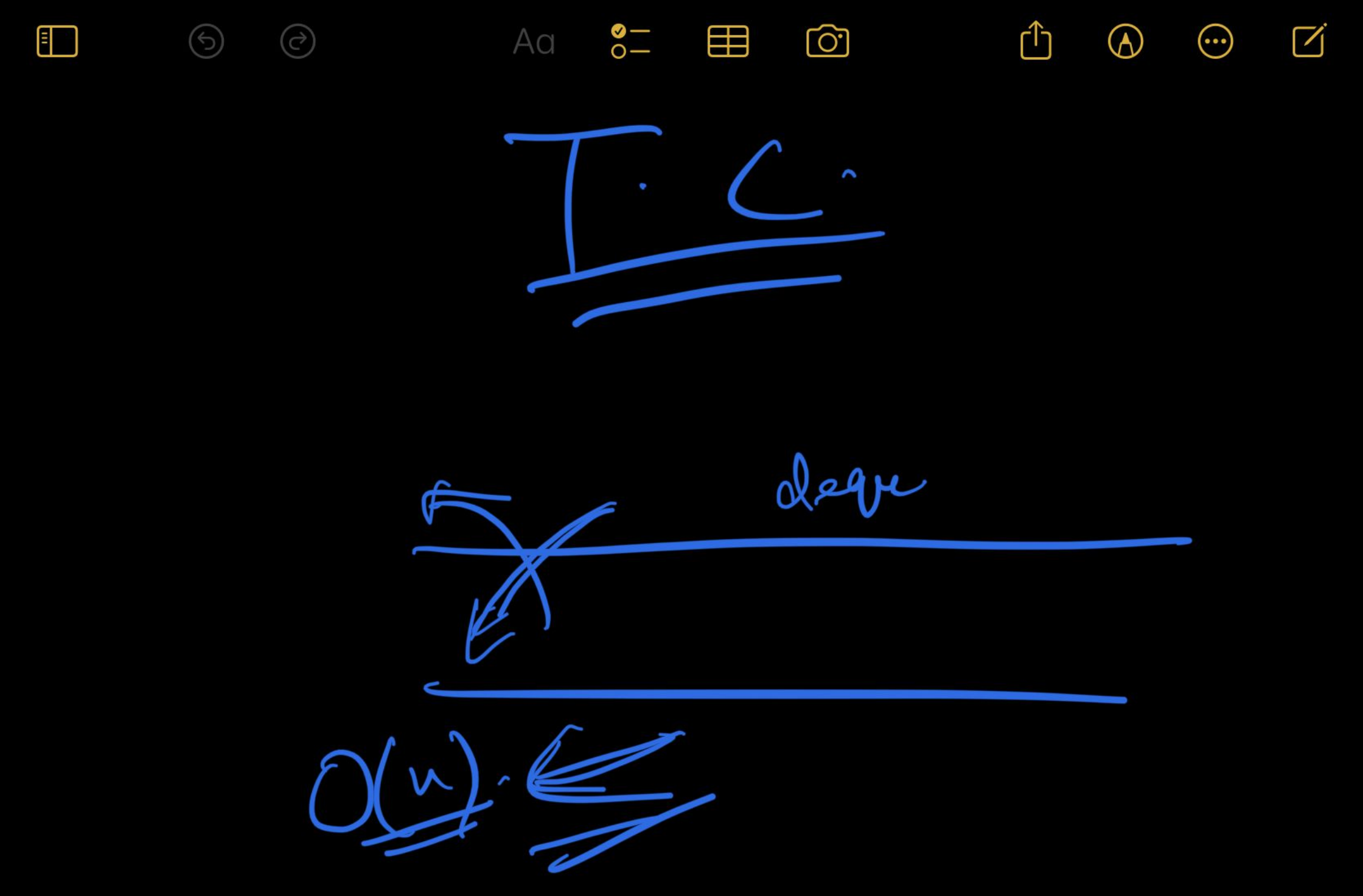
① i, (Size window check)

② greater than nums[i] > deq. element
pop_back.

③ push nums[i]

④ (i >= K-1)

result.push(nums[deq.front()]);


T. C.
deque
O(n)