

Sliding Window Maximum

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.

Input: `nums = [1,3,-1,-3,5,3,6,7]`, `k = 3`

Output: `[3,3,5,5,6,7]`

Explanation:

Window position	Max
-----------------	-----

-----	-----
[1 3 -1] -3 5 3 6 7	3
1 [3 -1 -3] 5 3 6 7	3
1 3 [-1 -3 5] 3 6 7	5
1 3 -1 [-3 5 3] 6 7	5
1 3 -1 -3 [5 3 6] 7	6
1 3 -1 -3 5 [3 6 7]	7

Input: `nums = [1,-1]`, `k = 1`

Output: `[1,-1]`

Input: `nums = [9,11]`, `k = 2`

Output: `[11]`

[1,3,-1,-3,5,3,6,7] k = 3

1,3,-1 => 3

3, -1, -5 => 3

-1, -3, 5 => 5

-3, 5, 3 => 5

5, 3, 6 => 6

3, 6, 7=> 7

[1,3,-1,-3,5,3,6,7] k = 3

[1[0],0[1],-1[2],-3[3],-5[4]], 3

OUTPUT : 1,0,-1

-5 -> -3 -> -1

9 8 7 10 5 4 3 6 N= 4

~ 2N

6 ->

ADD 4 + REMOVE 3

ADD 4 + REMOVE 3

14

[1,3,-1,-3,5,3,6,7] k = 3

BruteForce Approach : $O(nk)$



output1 :: [3, 3, 5, 5, 6, 7]
output2 :: [1, 0, -1]

Output : [3, 3, 5, 5, 6, 7]

$N - K + 1$

7[7] \rightarrow FRONT

I:4. K = 3
 $I - K = \text{FRONT}$
 $4 - 3 = 1$