**Leetcode 239**

**sliding-window-maximum**

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

output = [3,3,5,5,6,7]

* **use deque to solve problem.**
* **T.C -> O(n)+O(n);**
* **S.C -> O(k)**

**Code**

class Solution {

    public int[] maxSlidingWindow(int[] nums, int k) {

        int[] ans = new int[nums.length - k + 1];

        int j = 0;

        Deque<Integer> q = new LinkedList<>();

        for (int i = 0; i < nums.length; i++) {

            if (!q.isEmpty() && q.peekFirst() < i - k + 1) q.pollFirst();

            while (!q.isEmpty() && nums[i] > nums[q.peekLast()]) q.pollLast();

            q.offer(i);

            if (i >= k - 1) ans[j++] = nums[q.peekFirst()];

        }

        return ans;

    }

}