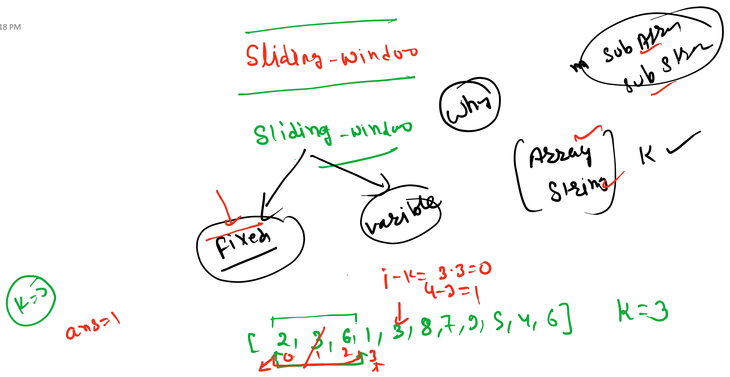
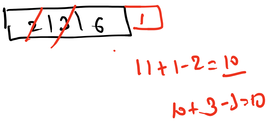
Lec-46

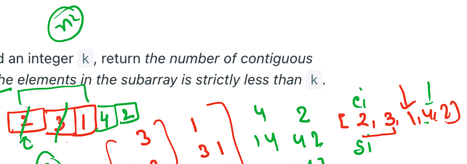
07/09/24 1:18 PM




```
public static int Maximum_Window(int [] arr,int k) {
       int sum=0;
       int ans=0;
       for (int i = 0; i <k; i++) {
           sum= sum+arr[i];
       }
       ans=sum;
       for (int i = k; i < arr.length; i++) {
           sum=sum+arr[i];//window grow
           sum=sum-arr[];// window shrink
           ans =Math.max(ans, sum);// ans update
       }
   }
}
```

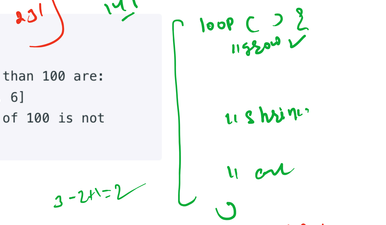


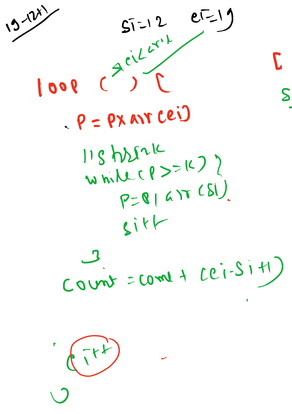
Given an array of integers `nums` and an integer `k`, return *the number of contiguous subarrays where the product of all the elements in the subarray is strictly less than* `k`.

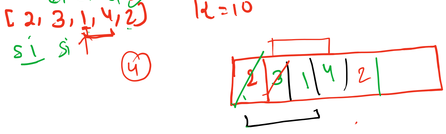


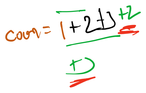
Example 1:

Input: nums = [10,5,2,6], k = 100
Output: 8
Explanation: The 8 subarrays that have product less than 100 are:
[10], [5], [2], [6], [10, 5], [5, 2], [2, 6], [5, 2, 6]
Note that [10, 5, 2] is not included as the product of 100 is not strictly less than k.



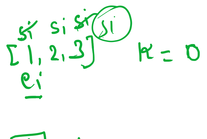


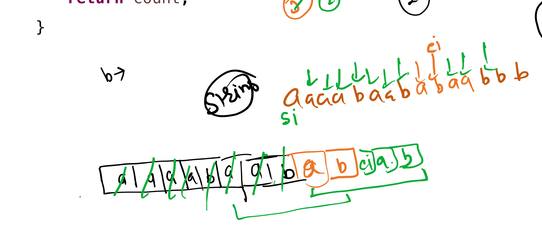


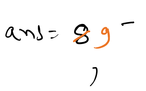


Product_Less_Than_K

```
public static int Product_Less_Than_K(int[] arr, int k) {
       int si = 0, ei = 0, count = 0, p = 1;
       while (ei < arr.length) {
           // grow
           p = p * arr[ei];

           // shrink i>=0
           while (p >= k) {
               p = p / arr[si];
               si++;
           }

           // ans update
           count += (ei - si + 1);

           ei++;
       }
       return count;
   }
```







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


```
public static int[] maxWindow(int[] arr, int k) {
    int n = arr.length;
    int[] max = new int[n - k + 1];
    int j = 0;
    Deque<Integer> dq = new LinkedList<>();
    for (int i = 0; i < k; i++) {
        while (!dq.isEmpty() && arr[dq.getLast()] <= arr[i]) {
            dq.removeLast();
        }
        dq.add(i);
    }
}
```