

# Problem 2962: Count Subarrays Where Max Element Appears at Least K Times

## Problem Information

**Difficulty:** Medium

**Acceptance Rate:** 62.41%

**Paid Only:** No

**Tags:** Array, Sliding Window

## Problem Description

You are given an integer array `nums` and a \*\*positive\*\* integer `k`.

Return \_the number of subarrays where the\*\*maximum\*\* element of `nums` \_appears\*\*at least\*\* `k` \_times in that subarray.\_

A \*\*subarray\*\* is a contiguous sequence of elements within an array.

**Example 1:**

**Input:** nums = [1,3,2,3,3], k = 2 **Output:** 6 **Explanation:** The subarrays that contain the element 3 at least 2 times are: [1,3,2,3], [1,3,2,3,3], [3,2,3], [3,2,3,3], [2,3,3] and [3,3].

**Example 2:**

**Input:** nums = [1,4,2,1], k = 3 **Output:** 0 **Explanation:** No subarray contains the element 4 at least 3 times.

**Constraints:**

\* `1 <= nums.length <= 105` \* `1 <= nums[i] <= 106` \* `1 <= k <= 105`

## Code Snippets

**C++:**

```
class Solution {  
public:  
    long long countSubarrays(vector<int>& nums, int k) {  
  
    }  
};
```

**Java:**

```
class Solution {  
public long countSubarrays(int[] nums, int k) {  
  
}  
}
```

**Python3:**

```
class Solution:  
    def countSubarrays(self, nums: List[int], k: int) -> int:
```