**[Count Subarrays Where Max Element Appears at Least K Times](https://leetcode.com/problems/count-subarrays-where-max-element-appears-at-least-k-times/)**

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

class Solution {

public:

    long long countSubarrays(vector<int>& nums, int k) {

        long long int max\_num = \*max\_element(nums.begin(),nums.end()),count = 0;

        long long int left = 0,right = 0,ans = 0;

        while(right<nums.size()){

            if(nums[right] == max\_num)count++;

            while(count>=k){

                if(nums[left]==max\_num)count--;

                left++;

                ans += nums.size()-right;

            }

            right++;

        }

        return ans;

    }

};

Link : <https://leetcode.com/problems/count-subarrays-where-max-element-appears-at-least-k-times/?envType=daily-question&envId=2024-03-29>