6293. Count the Number of Good Subarrays

My Submissions (/contest/weekly-contest-328/problems/count-the-number-of-good-subarrays/submissions/)	Back to Contest (/contest/weekly-contest-328/)

Given an integer array `nums` and an integer `k`, return *the number of* ***good*** *subarrays of* `nums`.

A subarray `arr` is **good** if it there are **at least** `k` pairs of indices `(i, j)` such that `i < j` and `arr[i] == arr[j]`.

A **subarray** is a contiguous **non-empty** sequence of elements within an array.

Example 1:

```
Input: nums = [1,1,1,1,1], k = 10
Output: 1
Explanation: The only good subarray is the array nums itself.
```

Example 2:

```
Input: nums = [3,1,4,3,2,2,4], k = 2
Output: 4
Explanation: There are 4 different good subarrays:
- [3,1,4,3,2,2] that has 2 pairs.
- [3,1,4,3,2,2,4] that has 3 pairs.
- [1,4,3,2,2,4] that has 2 pairs.
- [4,3,2,2,4] that has 2 pairs.
```

Constraints:

- $1 <= $ `nums.length` $<= 10^5$
- $1 <= $ `nums[i]`, `k` $<= 10^9$

User Accepted:	0
User Tried:	0
Total Accepted:	0
Total Submissions:	0
Difficulty:	Medium

JavaScript ▾   ⟨⟩ ↻ ⚙

```javascript
1  const addOneOrManyMap = (m, x, cnt = 1) => m.set(x, m.get(x) + cnt || cnt);
2  const removeOneOrManyMap = (m, x, cnt = 1) => { let occ = m.get(x); occ > cnt ? m.set(x, occ - cnt) : m.delete(x); };
3
4 ▾ const countGood = (a, k) => {
5      let res = subArraysAtLeastKPairsSame(a, k);
6      return res;
7  };
8
9 ▾ const subArraysAtLeastKPairsSame = (a, k) => {
10      let n = a.length, res = 0, same = 0, m = new Map(), l = 0;
11 ▾    for (let i = 0; i < n; i++) {
12          same += m.get(a[i]) || 0;
13          addOneOrManyMap(m, a[i]);
14 ▾        while (same >= k) {
15              removeOneOrManyMap(m, a[l]);
16              same -= m.get(a[l]) || 0;
17              l++;
18          }
19          res += l;
20      }
21      return res;
22  };
```

☐ **Custom Testcase**   (Use Example Testcases)

▶ Run   ☁ Submit

Submission Result: **Accepted** (/submissions/detail/878403295/) ❓   (More Details ❯ (/submissions/detail/878403295/))

Share your acceptance!