6049. K Divisible Elements Subarrays

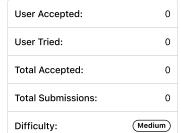
My Submissions (/contest/weekly-contest-291/problems/k-divisible-elements-subarrays/submissions/) Back to Contest (/contest/weekly-contest-291/)

Given an integer array nums and two integers k and p, return the number of distinct subarrays which have at most kelements divisible by p.

Two arrays nums1 and nums2 are said to be distinct if:

- They are of different lengths, or
- There exists at least one index i where nums1[i] != nums2[i].

A subarray is defined as a non-empty contiguous sequence of elements in an array.



strina-con

Example 1:

```
Input: nums = [2,3,3,2,2], k = 2, p = 2
Output: 11
Explanation:
The elements at indices 0, 3, and 4 are divisible by p = 2.
The 11 distinct subarrays which have at most k = 2 elements divisible by 2 are:
[2], [2,3], [2,3,3], [2,3,3,2], [3], [3,3], [3,3,2], [3,3,2,2], [3,2], [3,2,2], and [2,2].
Note that the subarrays [2] and [3] occur more than once in nums, but they should each be counted only once.
The subarray [2,3,3,2,2] should not be counted because it has 3 elements that are divisible by 2.
```

Example 2:

```
Input: nums = [1,2,3,4], k = 4, p = 1
Output: 10
Explanation:
All element of nums are divisible by p = 1.
Also, every subarray of nums will have at most 4 elements that are divisible by 1.
Since all subarrays are distinct, the total number of subarrays satisfying all the constraints is 10.
```

Constraints:

- 1 <= nums.length <= 200
- 1 <= nums[i], p <= 200
- 1 <= k <= nums.length

```
JavaScript
                                                                                                                              d c
1 \cdot | \text{const countDistinct} = (a, k, p) \Rightarrow \{
2
        let n = a.length, se = new Set();
3 •
         for (let i = 0; i < n; i++) {
4 ▼
             for (let j = i; j < n; j++) {
 5
                  let sub = a.slice(i, j + 1);
 6
                  if (ok(sub, k, p)) se.add(JSON.stringify(sub));
 7
             }
 8
9
         return se.size;
10
    };
11
    const ok = (a, k, p) \Rightarrow \{
12 ▼
13
         let cnt = 0;
14
        for (const x of a) {
15
             if (x \% p == 0) cnt++;
16
17
        return cnt <= k;
18
    };
```

United States (/region)

Custom Testcase

Use Example Testcases

PRun

Submission Result: Accepted (/submissions/detail/690642867/)

More Details ➤ (/submissions/detail/690642867/)

Share your acceptance!

Copyright © 2022 LeetCode

Help Center (/support) | Jobs (/jobs) | Bug Bounty (/bugbounty) | Online Interview (/interview/) | Students (/student) | Terms (/terms) | Privacy Policy (/privacy)