

Count Equal and Divisible Pairs in array

2176. Count Equal and Divisible Pairs in an Array

Easy Topics Companies Hint

Given a 0-indexed integer array `nums` of length `n` and an integer `k`, return the number of pairs (i, j) where $0 \leq i < j < n$, such that `nums[i] == nums[j]` and $(i * j)$ is divisible by `k`.

Example 1:

Input: `nums = [3,1,2,2,2,1,3]`, `k = 2`
Output: 4
Explanation: There are 4 pairs that meet all the requirements:
 - `nums[0] == nums[6]`, and $0 * 6 == 0$, which is divisible by 2.
 - `nums[2] == nums[3]`, and $2 * 3 == 6$, which is divisible by 2.
 - `nums[2] == nums[4]`, and $2 * 4 == 8$, which is divisible by 2.
 - `nums[3] == nums[4]`, and $3 * 4 == 12$, which is divisible by 2.

Example 2:

Input: `nums = [1,2,3,4]`, `k = 1`
Output: 0
Explanation: Since no value in `nums` is repeated, there are no pairs (i, j) that meet all the requirements.

Constraints:

- $1 \leq \text{nums.length} \leq 100$
- $1 \leq \text{nums}[i], k \leq 100$

`nums = [3,1, 2,2,2,1,3]` , `k = 2`

`idx = [0,1,2,3,4,5,6]`

[]

n^2

Approach 1 :- Hashmaps :-

```
C++ Auto
1 class Solution {
2 public:
3     int countPairs(vector<int>& nums, int K) {
4         int ans=0;
5         unordered_map<int, vector<int>> mp;
6         for(int i=0;i<nums.size();i++){
7             mp[nums[i]].push_back(i);
8         }
9         for(auto &it:mp){
10             vector<int> v=it.second;
11             if(v.size()==1)continue;
12             for(int i=0;i<v.size()-1;i++){
13                 for(int j=i+1;j<v.size();j++){
14                     if(((v[i]*v[j])%K==0)ans++;
15                 }
16             }
17         }
18         return ans;
19     }
20 };
```

$S.C \cdot O(n)$

Approach 2: Loops :-

```
C++ Auto
1 class Solution {
2 public:
3     int countPairs(vector<int>& nums, int k) {
4         int n = nums.size();
5         int res = 0; // number of pairs meeting the requirements
6         for (int i = 0; i < n - 1; ++i) {
7             for (int j = i + 1; j < n; ++j) {
8                 if ((i * j) % k == 0 && nums[i] == nums[j]) {
9                     ++res;
10                }
11            }
12        }
13        return res;
14    }
15 };
```

$T.C \cdot O(n^2)$

$S.C \cdot O(1)$