

Problem 2475: Number of Unequal Triplets in Array

Problem Information

Difficulty: Easy

Acceptance Rate: 73.00%

Paid Only: No

Tags: Array, Hash Table, Sorting

Problem Description

You are given a **0-indexed** array of positive integers `nums`. Find the number of triplets `(i, j, k)` that meet the following conditions:

`0 ≤ i < j < k < nums.length` `nums[i]`, `nums[j]`, and `nums[k]` are **pairwise distinct**.
In other words, `nums[i] != nums[j]`, `nums[i] != nums[k]`, and `nums[j] != nums[k]`.

Return `the number of triplets that meet the conditions.`

Example 1:

Input: `nums = [4,4,2,4,3]` **Output:** `3` **Explanation:** The following triplets meet the conditions: - `(0, 2, 4)` because `4 != 2 != 3` - `(1, 2, 4)` because `4 != 2 != 3` - `(2, 3, 4)` because `2 != 4 != 3` Since there are 3 triplets, we return 3. Note that `(2, 0, 4)` is not a valid triplet because `2 > 0`.

Example 2:

Input: `nums = [1,1,1,1,1]` **Output:** `0` **Explanation:** No triplets meet the conditions so we return 0.

Constraints:

`3 ≤ nums.length ≤ 100` `1 ≤ nums[i] ≤ 1000`

Code Snippets

C++:

```
class Solution {  
public:  
    int unequalTriplets(vector<int>& nums) {  
  
    }  
};
```

Java:

```
class Solution {  
    public int unequalTriplets(int[] nums) {  
  
    }  
}
```

Python3:

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