

1. 128. 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.

PROGRAM:-

```
def count_pairs(nums, k):
```

```
    index_map = {}
```

```
    count = 0
```

```
    for i, num in enumerate(nums):
```

```
        if num in index_map:
```

```
            for j in index_map[num]:
```

```
                if (i * j) % k == 0:
```

```
                    count += 1
```

```
            index_map[num].append(i)
```

```
        else:
```

```
            index_map[num] = [i]
```

```
    return count
```

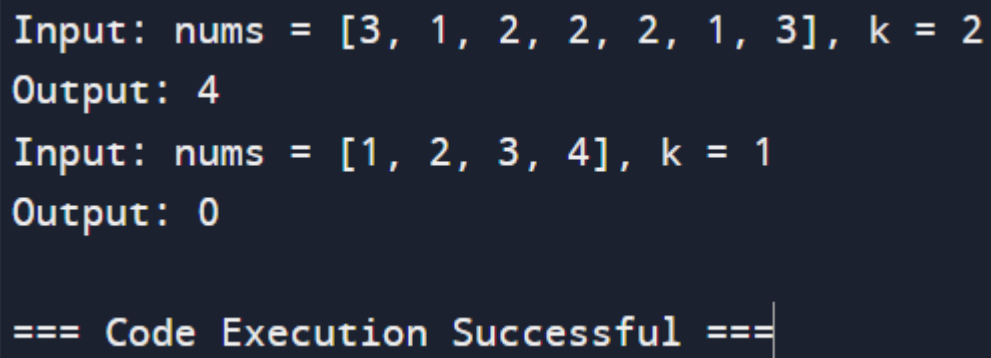
Example 1

```
nums1 = [3, 1, 2, 2, 2, 1, 3]
k1 = 2
output1 = count_pairs(nums1, k1)
print(f"Input: nums = {nums1}, k = {k1}\nOutput: {output1}")
```

Example 2

```
nums2 = [1, 2, 3, 4]
k2 = 1
output2 = count_pairs(nums2, k2)
print(f"Input: nums = {nums2}, k = {k2}\nOutput: {output2}")
```

OUTPUT:-

A screenshot of a code execution environment with a dark background. It shows the input and output for two examples. The first example has input nums = [3, 1, 2, 2, 2, 1, 3] and k = 2, resulting in an output of 4. The second example has input nums = [1, 2, 3, 4] and k = 1, resulting in an output of 0. At the bottom, it says '=== Code Execution Successful ===' with a cursor at the end.

```
Input: nums = [3, 1, 2, 2, 2, 1, 3], k = 2
Output: 4
Input: nums = [1, 2, 3, 4], k = 1
Output: 0

=== Code Execution Successful ===
```

TIME COMPLEXITY:- $O(n)$