Count Triplets ☆

Problem Submissions Leaderboard Editorial

You are given an array and you need to find number of tripets of indices (i, j, k) such that the elements at those indices are in geometric progression for a given common ratio r and $i < j < k$.

For example, $arr = [1, 4, 16, 64]$. If $r = 4$, we have $[1, 4, 16]$ and $[4, 16, 64]$ at indices $(0, 1, 2)$ and $(1, 2, 3)$.

Function Description

Complete the countTriplets function in the editor below. It should return the number of triplets forming a geometric progression for a given r as an integer.

countTriplets has the following parameter(s):

- arr: an array of integers
- r: an integer, the common ratio

Input Format

The first line contains two space-separated integers n and r, the size of arr and the common ratio.
The next line contains n space-seperated integers $arr[i]$.

Constraints

- $1 \leq n \leq 10^5$
- $1 \leq r \leq 10^9$
- $1 \leq arr[i] \leq 10^9$

Output Format

Return the count of triplets that form a geometric progression.

Sample Input 0

```
4 2
1 2 2 4
```

Sample Output 0

```
2
```

Explanation 0

There are 2 triplets in satisfying our criteria, whose indices are $(0, 1, 3)$ and $(0, 2, 3)$

Sample Input 1

```
6 3
1 3 9 9 27 81
```

Sample Output 1

```
6
```

Explanation 1

The triplets satisfying are index $(0, 1, 2)$, $(0, 1, 3)$, $(1, 2, 4)$, $(1, 3, 4)$, $(2, 4, 5)$ and $(3, 4, 5)$.

Sample Input 2

```
5 5
1 5 5 25 125
```

Sample Output 2

```
4
```

Explanation 2

The triplets satisfying are index $(0, 1, 3)$, $(0, 2, 3)$, $(1, 3, 4)$, $(2, 3, 4)$.

Change Theme C++

```
8
9   // Complete the countTriplets function below.
10  long countTriplets(vector<long> arr, long r) {
11      int n = arr.size();
12      long count=0;
13      int nexti,nextj,j,k;
14
15      for(int i =0; i<=n-3; i++){
16          nexti = arr[i]*int(r);
17          j=i+1;
18          while((arr[j]<=nexti)&&(j<(n-1))){
19              if(arr[j]==nexti){
20              nextj=arr[j]*int(r);
21              k=1;
22                  while((arr[j+k]<=nextj)&&((j+k)<n)){
23                      if(arr[j+k]==nextj) {
24                          count++;
25                      }
26                      k++;
27                  }
28              }
29              j++;
30          }
31      }
32  return count;
```

Line: 108 Col: 1

Upload Code as File Test against custom input Run Code Submit Code