Given two arrays of integers nums1 and nums2, return the number of triplets formed (type 1 and type 2) under the following rules:

* Type 1: Triplet (i, j, k) if nums1[i]2 == nums2[j] \* nums2[k] where 0 <= i < nums1.length and 0 <= j < k < nums2.length.
* Type 2: Triplet (i, j, k) if nums2[i]2 == nums1[j] \* nums1[k] where 0 <= i < nums2.length and 0 <= j < k < nums1.length.

**Example 1:**

**Input:** nums1 = [7,4], nums2 = [5,2,8,9]

**Output:** 1

**Explanation:** Type 1: (1,1,2), nums1[1]^2 = nums2[1] \* nums2[2]. (4^2 = 2 \* 8).

**Example 2:**

**Input:** nums1 = [1,1], nums2 = [1,1,1]

**Output:** 9

**Explanation:** All Triplets are valid, because 1^2 = 1 \* 1.

Type 1: (0,0,1), (0,0,2), (0,1,2), (1,0,1), (1,0,2), (1,1,2). nums1[i]^2 = nums2[j] \* nums2[k].

Type 2: (0,0,1), (1,0,1), (2,0,1). nums2[i]^2 = nums1[j] \* nums1[k].

**Example 3:**

**Input:** nums1 = [7,7,8,3], nums2 = [1,2,9,7]

**Output:** 2

**Explanation:** There are 2 valid triplets.

Type 1: (3,0,2). nums1[3]^2 = nums2[0] \* nums2[2].

Type 2: (3,0,1). nums2[3]^2 = nums1[0] \* nums1[1].

**Example 4:**

**Input:** nums1 = [4,7,9,11,23], nums2 = [3,5,1024,12,18]

**Output:** 0

**Explanation:** There are no valid triplets.

**Constraints:**

* 1 <= nums1.length, nums2.length <= 1000
* 1 <= nums1[i], nums2[i] <= 10^5