Problem 1874: Minimize Product Sum of Two Arrays

Problem Information

Difficulty: Medium
Acceptance Rate: 89.11%
Paid Only: Yes
Tags: Array, Greedy, Sorting

Problem Description

The **product sum** of two equal-length arrays `a` and `b` is equal to the sum of `a[i] * b[i]` for all `0 <= i < a.length` (**0-indexed**).

* For example, if `a = [1,2,3,4]` and `b = [5,2,3,1]`, the **product sum** would be `1*5 + 2*2 + 3*3 + 4*1 = 22`.

Given two arrays `nums1` and `nums2` of length `n`, return _the**minimum product sum** if you are allowed to **rearrange** the **order** of the elements in _`nums1`.

Example 1:

Input: nums1 = [5,3,4,2], nums2 = [4,2,2,5] **Output:** 40 **Explanation:** We can rearrange nums1 to become [3,5,4,2]. The product sum of [3,5,4,2] and [4,2,2,5] is 3*4 + 5*2 + 4*2 + 2*5 = 40.

Example 2:

Input: nums1 = [2,1,4,5,7], nums2 = [3,2,4,8,6] **Output:** 65 **Explanation:** We can rearrange nums1 to become [5,7,4,1,2]. The product sum of [5,7,4,1,2] and [3,2,4,8,6] is 5*3 + 7*2 + 4*4 + 1*8 + 2*6 = 65.

Constraints:

* `n == nums1.length == nums2.length` * `1 <= n <= 105` * `1 <= nums1[i], nums2[i] <= 100`

Code Snippets

C++:

```
class Solution {
public:
int minProductSum(vector<int>& nums1, vector<int>& nums2) {

}
};
```

Java:

```
class Solution {
public int minProductSum(int[] nums1, int[] nums2) {

}
}
```

Python3:

```
class Solution:
def minProductSum(self, nums1: List[int], nums2: List[int]) -> int:
```