Problem 1855: Maximum Distance Between a Pair of Values

Problem Information

Difficulty: Medium
Acceptance Rate: 54.12%
Paid Only: No
Tags: Array, Two Pointers, Binary Search

Problem Description

You are given two **non-increasing 0-indexed** integer arrays `nums1` and `nums2`■■■■■■.

A pair of indices `(i, j)`, where `0 <= i < nums1.length` and `0 <= j < nums2.length`, is **valid** if both `i <= j` and `nums1[i] <= nums2[j]`. The **distance** of the pair is `j - i`■■■■.

Return _the**maximum distance** of any **valid** pair _`(i, j)`_. If there are no valid pairs, return_`0`.

An array `arr` is **non-increasing** if `arr[i-1] >= arr[i]` for every `1 <= i < arr.length`.

Example 1:

Input: nums1 = [55,30,5,4,2], nums2 = [100,20,10,10,5] **Output:** 2 **Explanation:** The valid pairs are (0,0), (2,2), (2,3), (2,4), (3,3), (3,4), and (4,4). The maximum distance is 2 with pair (2,4).

Example 2:

Input: nums1 = [2,2,2], nums2 = [10,10,1] **Output:** 1 **Explanation:** The valid pairs are (0,0), (0,1), and (1,1). The maximum distance is 1 with pair (0,1).

Example 3:

Input: nums1 = [30,29,19,5], nums2 = [25,25,25,25,25] **Output:** 2 **Explanation:** The valid pairs are (2,2), (2,3), (2,4), (3,3), and (3,4). The maximum distance is 2 with pair (2,4).

Constraints:

* `1 <= nums1.length, nums2.length <= 105` * `1 <= nums1[i], nums2[j] <= 105` * Both `nums1` and `nums2` are **non-increasing**.

Code Snippets

C++:

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

}
};
```

Java:

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

}
}
```

Python3:

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