Problem 3724: Minimum Operations to Transform Array

Problem Information

Difficulty: Medium
Acceptance Rate: 38.13%
Paid Only: No
Tags: Array, Greedy

Problem Description

You are given two integer arrays `nums1` of length `n` and `nums2` of length `n + 1`.

You want to transform `nums1` into `nums2` using the **minimum** number of operations.

You may perform the following operations **any** number of times, each time choosing an index `i`:

* **Increase** `nums1[i]` by 1. * **Decrease** `nums1[i]` by 1. * **Append** `nums1[i]` to the **end** of the array.

Return the **minimum** number of operations required to transform `nums1` into `nums2`.

Example 1:

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

Output: 4

Explanation:

Step | `i` | Operation | `nums1[i]` | Updated `nums1` ---|---|---|---|--- 1 | 0 | Append | - | [2, 8, 2] 2 | 0 | Decrement | Decreases to 1 | [1, 8, 2] 3 | 1 | Decrement | Decreases to 7 | [1, 7, 2] 4 | 2 | Increment | Increases to 3 | [1, 7, 3] Thus, after 4 operations `nums1` is transformed into `nums2`.

Example 2:

Input: nums1 = [1,3,6], nums2 = [2,4,5,3]

Output: 4

Explanation:

Step | `i` | Operation | `nums1[i]` | Updated `nums1` ---|---|---|---|--- 1 | 1 | Append | - | [1, 3, 6, 3] 2 | 0 | Increment | Increases to 2 | [2, 3, 6, 3] 3 | 1 | Increment | Increases to 4 | [2, 4, 6, 3] 4 | 2 | Decrement | Decreases to 5 | [2, 4, 5, 3] Thus, after 4 operations `nums1` is transformed into `nums2`.

Example 3:

Input: nums1 = [2], nums2 = [3,4]

Output: 3

Explanation:

Step | `i` | Operation | `nums1[i]` | Updated `nums1` ---|---|---|---|--- 1 | 0 | Increment | Increases to 3 | [3] 2 | 0 | Append | - | [3, 3] 3 | 1 | Increment | Increases to 4 | [3, 4] Thus, after 3 operations `nums1` is transformed into `nums2`.

Constraints:

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

Code Snippets

C++:

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

```
}
};
```

Java:

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

}
}
```

Python3:

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