Problem 1983: Widest Pair of Indices With Equal Range Sum

Problem Information

Difficulty: Medium
Acceptance Rate: 53.70%
Paid Only: Yes
Tags: Array, Hash Table, Prefix Sum

Problem Description

You are given two **0-indexed** binary arrays `nums1` and `nums2`. Find the **widest** pair of indices `(i, j)` such that `i <= j` and `nums1[i] + nums1[i+1] + ... + nums1[j] == nums2[i] + nums2[i+1] + ... + nums2[j]`.

The **widest** pair of indices is the pair with the **largest** **distance** between `i` and `j`. The **distance** between a pair of indices is defined as `j - i + 1`.

Return _the**distance** of the **widest** pair of indices. If no pair of indices meets the conditions, return _`0`.

Example 1:

Input: nums1 = [1,1,0,1], nums2 = [0,1,1,0] **Output:** 3 **Explanation:** If i = 1 and j = 3: nums1[1] + nums1[2] + nums1[3] = 1 + 0 + 1 = 2. nums2[1] + nums2[2] + nums2[3] = 1 + 1 + 0 = 2. The distance between i and j is j - i + 1 = 3 - 1 + 1 = 3.

Example 2:

Input: nums1 = [0,1], nums2 = [1,1] **Output:** 1 **Explanation:** If i = 1 and j = 1: nums1[1] = 1. nums2[1] = 1. The distance between i and j is j - i + 1 = 1 - 1 + 1 = 1.

Example 3:

Input: nums1 = [0], nums2 = [1] **Output:** 0 **Explanation:** There are no pairs of indices that meet the requirements.

Constraints:

* `n == nums1.length == nums2.length` * `1 <= n <= 105` * `nums1[i]` is either `0` or `1`. * `nums2[i]` is either `0` or `1`.

Code Snippets

C++:

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


    }
};
```

Java:

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


    }
}
```

Python3:

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