Problem 2321: Maximum Score Of Spliced Array

Problem Information

Difficulty: Hard
Acceptance Rate: 58.04%
Paid Only: No
Tags: Array, Dynamic Programming

Problem Description

You are given two **0-indexed** integer arrays `nums1` and `nums2`, both of length `n`.

You can choose two integers `left` and `right` where `0 <= left <= right < n` and **swap** the subarray `nums1[left...right]` with the subarray `nums2[left...right]`.

* For example, if `nums1 = [1,2,3,4,5]` and `nums2 = [11,12,13,14,15]` and you choose `left = 1` and `right = 2`, `nums1` becomes `[1,**_12,13_** ,4,5]` and `nums2` becomes `[11,**_2,3_** ,14,15]`.

You may choose to apply the mentioned operation **once** or not do anything.

The **score** of the arrays is the **maximum** of `sum(nums1)` and `sum(nums2)`, where `sum(arr)` is the sum of all the elements in the array `arr`.

Return _the**maximum possible score**_.

A **subarray** is a contiguous sequence of elements within an array. `arr[left...right]` denotes the subarray that contains the elements of `nums` between indices `left` and `right` (**inclusive**).

Example 1:

Input: nums1 = [60,60,60], nums2 = [10,90,10] **Output:** 210 **Explanation:** Choosing left = 1 and right = 1, we have nums1 = [60,_**90**_ ,60] and nums2 = [10,_**60**_ ,10]. The score is max(sum(nums1), sum(nums2)) = max(210, 80) = 210.

Example 2:

Input: nums1 = [20,40,20,70,30], nums2 = [50,20,50,40,20] **Output:** 220 **Explanation:** Choosing left = 3, right = 4, we have nums1 = [20,40,20,_**40,20**_] and nums2 = [50,20,50,_**70,30**_]. The score is max(sum(nums1), sum(nums2)) = max(140, 220) = 220.

Example 3:

Input: nums1 = [7,11,13], nums2 = [1,1,1] **Output:** 31 **Explanation:** We choose not to swap any subarray. The score is max(sum(nums1), sum(nums2)) = max(31, 3) = 31.

Constraints:

* `n == nums1.length == nums2.length` * `1 <= n <= 105` * `1 <= nums1[i], nums2[i] <= 104`

Code Snippets

C++:

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

}
};
```

Java:

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

}
}
```

Python3:

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