Problem 2593: Find Score of an Array After Marking All Elements

Problem Information

Difficulty: Medium
Acceptance Rate: 64.49%
Paid Only: No
Tags: Array, Hash Table, Sorting, Heap (Priority Queue), Simulation

Problem Description

You are given an array `nums` consisting of positive integers.

Starting with `score = 0`, apply the following algorithm:

* Choose the smallest integer of the array that is not marked. If there is a tie, choose the one with the smallest index. * Add the value of the chosen integer to `score`. * Mark **the chosen element and its two adjacent elements if they exist**. * Repeat until all the array elements are marked.

Return _the score you get after applying the above algorithm_.

Example 1:

Input: nums = [2,1,3,4,5,2] **Output:** 7 **Explanation:** We mark the elements as follows: - 1 is the smallest unmarked element, so we mark it and its two adjacent elements: [_2_ ,_1_ ,_3_ ,4,5,2]. - 2 is the smallest unmarked element, so we mark it and its left adjacent element: [_2_ ,_1_ ,_3_ ,4,_5_ ,_2_]. - 4 is the only remaining unmarked element, so we mark it: [_2_ ,_1_ ,_3_ ,_4_ ,_5_ ,_2_]. Our score is 1 + 2 + 4 = 7.

Example 2:

Input: nums = [2,3,5,1,3,2] **Output:** 5 **Explanation:** We mark the elements as follows: - 1 is the smallest unmarked element, so we mark it and its two adjacent elements: [2,3,_5_ ,_1_ ,_3_ ,2]. - 2 is the smallest unmarked element, since there are two of them, we choose the left-most one, so we mark the one at index 0 and its right adjacent element: [_2_

,_3_ ,_5_ ,_1_ ,_3_ ,2]. - 2 is the only remaining unmarked element, so we mark it: [_2_ ,_3_ ,_5_ ,_1_ ,_3_ ,_2_]. Our score is 1 + 2 + 2 = 5.

Constraints:

* `1 <= nums.length <= 105` * `1 <= nums[i] <= 106`

Code Snippets

C++:

```
class Solution {
public:
long long findScore(vector<int>& nums) {

}
};
```

Java:

```
class Solution {
public long findScore(int[] nums) {

}
}
```

Python3:

```
class Solution:
def findScore(self, nums: List[int]) -> int:
```