Problem 3362: Zero Array Transformation III

Problem Information

Difficulty: Medium
Acceptance Rate: 54.94%
Paid Only: No
Tags: Array, Greedy, Sorting, Heap (Priority Queue), Prefix Sum

Problem Description

You are given an integer array `nums` of length `n` and a 2D array `queries` where `queries[i] = [li, ri]`.

Each `queries[i]` represents the following action on `nums`:

* Decrement the value at each index in the range `[li, ri]` in `nums` by **at most****** 1. * The amount by which the value is decremented can be chosen **independently** for each index.

A **Zero Array** is an array with all its elements equal to 0.

Return the **maximum** number of elements that can be removed from `queries`, such that `nums` can still be converted to a **zero array** using the _remaining_ queries. If it is not possible to convert `nums` to a **zero array** , return -1.

Example 1:

Input: nums = [2,0,2], queries = [[0,2],[0,2],[1,1]]

Output: 1

Explanation:

After removing `queries[2]`, `nums` can still be converted to a zero array.

* Using `queries[0]`, decrement `nums[0]` and `nums[2]` by 1 and `nums[1]` by 0. * Using `queries[1]`, decrement `nums[0]` and `nums[2]` by 1 and `nums[1]` by 0.

Example 2:

Input: nums = [1,1,1,1], queries = [[1,3],[0,2],[1,3],[1,2]]

Output: 2

Explanation:

We can remove `queries[2]` and `queries[3]`.

Example 3:

Input: nums = [1,2,3,4], queries = [[0,3]]

Output: -1

Explanation:

`nums` cannot be converted to a zero array even after using all the queries.

Constraints:

* `1 <= nums.length <= 105` * `0 <= nums[i] <= 105` * `1 <= queries.length <= 105` * `queries[i].length == 2` * `0 <= li <= ri < nums.length`

Code Snippets

C++:

```
class Solution {
public:
    int maxRemoval(vector<int>& nums, vector<vector<int>>& queries) {

    }
};
```

Java:

```
class Solution {
public int maxRemoval(int[] nums, int[][] queries) {


}
}
```

Python3:

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