

# Problem 3049: Earliest Second to Mark Indices II

## Problem Information

**Difficulty:** Hard

**Acceptance Rate:** 21.17%

**Paid Only:** No

**Tags:** Array, Binary Search, Greedy, Heap (Priority Queue)

## Problem Description

You are given two **1-indexed** integer arrays, `nums` and `changeIndices`, having lengths `n` and `m`, respectively.

Initially, all indices in `nums` are unmarked. Your task is to mark **all** indices in `nums`.

In each second, `s`, in order from `1` to `m` (**inclusive**), you can perform **one** of the following operations:

\* Choose an index `i` in the range `[1, n]` and **decrement** `nums[i]` by `1`. \* Set `nums[changeIndices[s]]` to any **non-negative** value. \* Choose an index `i` in the range `[1, n]`, where `nums[i]` is **equal** to `0`, and **mark** index `i`. \* Do nothing.

Return `_` an integer denoting the **earliest second** in the range `[1, m]` when **all** indices in `nums` can be marked by choosing operations optimally, or `-1` if it is impossible.

**Example 1:**

**Input:** `nums = [3,2,3]`, `changeIndices = [1,3,2,2,2,3]` **Output:** `6` **Explanation:** In this example, we have 7 seconds. The following operations can be performed to mark all indices: Second 1: Set `nums[changeIndices[1]]` to 0. `nums` becomes `[0,2,3]`. Second 2: Set `nums[changeIndices[2]]` to 0. `nums` becomes `[0,2,0]`. Second 3: Set `nums[changeIndices[3]]` to 0. `nums` becomes `[0,0,0]`. Second 4: Mark index 1, since `nums[1]` is equal to 0. Second 5: Mark index 2, since `nums[2]` is equal to 0. Second 6: Mark index 3, since `nums[3]` is equal to 0. Now all indices have been marked. It can be shown that it is not possible to mark all indices earlier than the 6th second. Hence, the answer is 6.

**\*\*Example 2:\*\***

**\*\*Input:\*\*** nums = [0,0,1,2], changeIndices = [1,2,1,2,1,2,1,2] **\*\*Output:\*\*** 7 **\*\*Explanation:\*\*** In this example, we have 8 seconds. The following operations can be performed to mark all indices: Second 1: Mark index 1, since nums[1] is equal to 0. Second 2: Mark index 2, since nums[2] is equal to 0. Second 3: Decrement index 4 by one. nums becomes [0,0,1,1]. Second 4: Decrement index 4 by one. nums becomes [0,0,1,0]. Second 5: Decrement index 3 by one. nums becomes [0,0,0,0]. Second 6: Mark index 3, since nums[3] is equal to 0. Second 7: Mark index 4, since nums[4] is equal to 0. Now all indices have been marked. It can be shown that it is not possible to mark all indices earlier than the 7th second. Hence, the answer is 7.

**\*\*Example 3:\*\***

**\*\*Input:\*\*** nums = [1,2,3], changeIndices = [1,2,3] **\*\*Output:\*\*** -1 **\*\*Explanation:\*\*** In this example, it can be shown that it is impossible to mark all indices, as we don't have enough seconds. Hence, the answer is -1.

**\*\*Constraints:\*\***

\*`1 <= n == nums.length <= 5000` \*`0 <= nums[i] <= 109` \*`1 <= m == changeIndices.length <= 5000` \*`1 <= changeIndices[i] <= n`

## Code Snippets

**C++:**

```
class Solution {
public:
    int earliestSecondToMarkIndices(vector<int>& nums, vector<int>&
changeIndices) {

    }
};
```

**Java:**

```
class Solution {
    public int earliestSecondToMarkIndices(int[] nums, int[] changeIndices) {

    }
}
```

```
}
```

### Python3:

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