2113. Elements in Array After Removing and Replacing Elements Premium

Medium | Topics | Hint

You are given a **0-indexed** integer array `nums`. Initially on minute `0`, the array is unchanged. Every minute, the **leftmost** element in `nums` is removed until no elements remain. Then, every minute, one element is appended to the **end** of `nums`, in the order they were removed in, until the original array is restored. This process repeats indefinitely.

- For example, the array `[0,1,2]` would change as follows: `[0,1,2] → [1,2] → [2] → [] → [0] → [0,1] → [0,1,2] → [1,2] → [2] → [] → [0] → [0,1] → [0,1,2] → ...`

You are also given a 2D integer array `queries` of size `n` where `queries[j] = [time_j, index_j]`. The answer to the `j^th` query is:

- `nums[index_j]` if `index_j < nums.length` at minute `time_j`
- `-1` if `index_j >= nums.length` at minute `time_j`

Return *an integer array `ans` of size `n` where `ans[j]` is the answer to the `j^th` query.*

Example 1:

```
Input: nums = [0,1,2], queries = [[0,2],[2,0],[3,2],[5,0]]
Output: [2,2,-1,0]
Explanation:
Minute 0: [0,1,2] - All elements are in the nums.
Minute 1: [1,2]   - The leftmost element, 0, is removed.
Minute 2: [2]     - The leftmost element, 1, is removed.
Minute 3: []      - The leftmost element, 2, is removed.
Minute 4: [0]     - 0 is added to the end of nums.
Minute 5: [0,1]   - 1 is added to the end of nums.

At minute 0, nums[2] is 2.
At minute 2, nums[0] is 2.
At minute 3, nums[2] does not exist.
At minute 5, nums[0] is 0.
```

Example 2:

```
Input: nums = [2], queries = [[0,0],[1,0],[2,0],[3,0]]
Output: [2,-1,2,-1]
Minute 0: [2] - All elements are in the nums.
Minute 1: []  - The leftmost element, 2, is removed.
Minute 2: [2] - 2 is added to the end of nums.
Minute 3: []  - The leftmost element, 2, is removed.

At minute 0, nums[0] is 2.
At minute 1, nums[0] does not exist.
At minute 2, nums[0] is 2.
At minute 3, nums[0] does not exist.
```

Constraints:

- `1 <= nums.length <= 100`
- `0 <= nums[i] <= 100`
- `n == queries.length`
- `1 <= n <= 10^5`
- `queries[j].length == 2`
- `0 <= time_j <= 10^5`
- `0 <= index_j < nums.length`

Seen this question in a real interview before? 1/5

Yes No

Accepted 2.1K | Submissions 2.8K | Acceptance Rate 73.5%

Topics

Array

Hint 1

Try to find a pattern in how nums changes.

Hint 2

Let m be the original length of nums. If time_i / m (integer division) is even, then nums is at its original size or decreasing in size. If it is odd, then it is empty, or increasing in size.

Hint 3

time_i % m can be used to find how many elements are in nums at minute time_i.

Discussion (1)