Difference Array

Concepts & Qns

Video-4

(M)otivation :-

Everytime you chose hard work over comfort,
you moved closer to greatness.
Because LEGENDS aren't made in ease
– they are forged in sacrifice.

MIK...

3355. Zero Array Transformation I

Medium | Topics | Companies | Hint

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

For each `queries[i]`:

- Select a subset of indices within the range `[l_i, r_i]` in `nums`.
- Decrement the values at the selected indices by 1.

A **Zero Array** is an array where all elements are equal to 0.

Return `true` if it is *possible* to transform `nums` into a **Zero Array** after processing all the queries sequentially, otherwise return `false`.

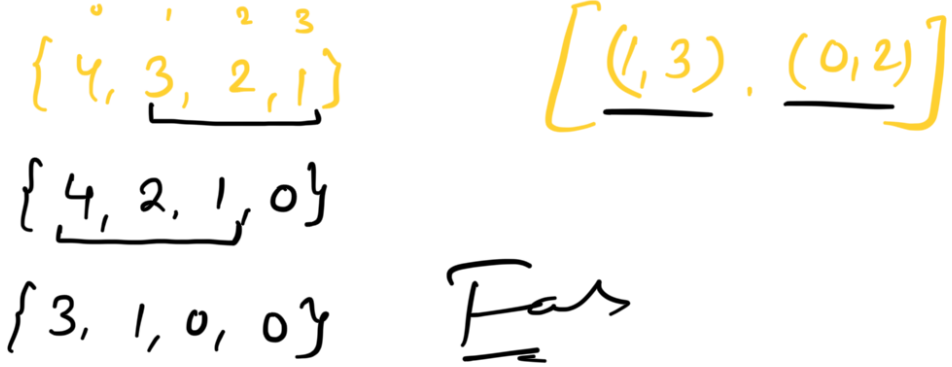
Example 1:

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

Output: `true`

Explanation:

- **For i = 0:**
  - Select the subset of indices as `[0, 2]` and decrement the values at these indices by 1.
  - The array will become `[0, 0, 0]`, which is a Zero Array.



Thought Process

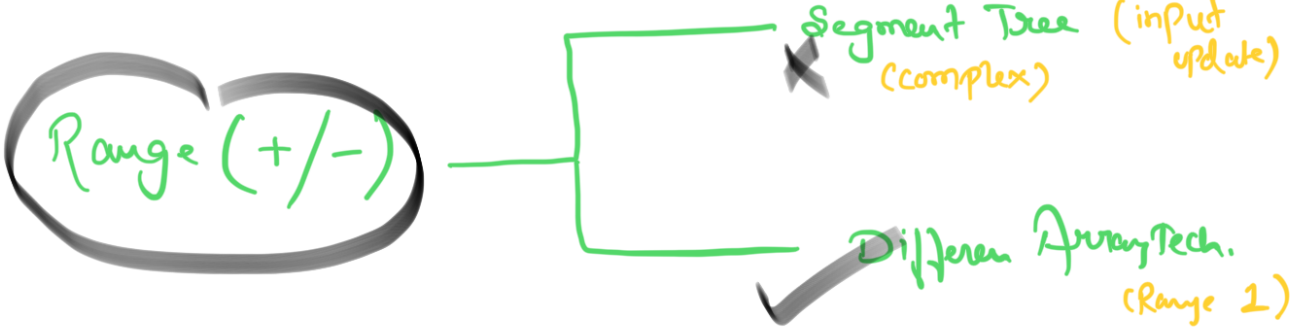
Brute Force

nums = [1, 2, 2, 1] , queries = [(1,3), (0,2)]

I Did what was asked as it is.

T.C = O(Q * n); T.L.E.

Optimal



nums = [1, 2, 2, 1] queries = [(1,3), (0,2)]

Step-1:-

diff =

0	1	2	3
1	1	0	-1

i

start = 0
end = 2

```
diff[start] += 1;
if (end+1 < n)
    diff[end+1] -= 1;
```

Step-2:-

```
vector<int> result (n,0);
```

0	1	2	3
1	1	0	-1

i

cumSum = 1+1+0
-1

result =

0	1	2	3
1	1	2	1

i i i i

input = nums

0	1	2	3
1	2	2	1

0 0 0 0

```
if (result[i] < nums[i]) {
    return false;
}
```

$$T.C = O(Q + n)$$

$$S.C = O(n)$$