2602. Minimum Operations to Make All Array Elements Equal

Description

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

You are also given an integer array `queries` of size `m`. For the `i th` query, you want to make all of the elements of `nums` equal to `queries[i]`. You can perform the following operation on the array **any** number of times:

- **Increase** or **decrease** an element of the array by `1`.

Return *an array* `answer` *of size* `m` *where* `answer[i]` *is the* ***minimum*** *number of operations to make all elements of* `nums` *equal to* `queries[i]`.

Note that after each query the array is reset to its original state.

Example 1:

```
Input: nums = [3,1,6,8], queries = [1,5]
Output: [14,10]
Explanation: For the first query we can do the following operations:
- Decrease nums[0] 2 times, so that nums = [1,1,6,8].
- Decrease nums[2] 5 times, so that nums = [1,1,1,8].
- Decrease nums[3] 7 times, so that nums = [1,1,1,1].
So the total number of operations for the first query is 2 + 5 + 7 = 14.
For the second query we can do the following operations:
- Increase nums[0] 2 times, so that nums = [5,1,6,8].
- Increase nums[1] 4 times, so that nums = [5,5,6,8].
- Decrease nums[2] 1 time, so that nums = [5,5,5,8].
- Decrease nums[3] 3 times, so that nums = [5,5,5,5].
So the total number of operations for the second query is 2 + 4 + 1 + 3 = 10.
```

Example 2:

```
Input: nums = [2,9,6,3], queries = [10]
Output: [20]
Explanation: We can increase each value in the array to 10. The total number of operations will be 8 + 1 + 4 + 7 = 20.
```

Constraints:

- `n == nums.length`
- `m == queries.length`
- `1 <= n, m <= 10^5`
- `1 <= nums[i], queries[i] <= 10^9`