PROBLEM

Count the elements

Easy Accuracy: 25.32% Submissions: 38K+ Points: 2

Given two arrays **a** and **b** both of size **n**. Given **q** queries in an arrray **query** each having a positive integer **x** denoting an index of the array **a**. For each query, your task is to find all the elements **less than or equal** to **a[x]** in the array **b**.

Example 1:

Input:
n = 3
a[] = {4,1,2}
b[] = {1,7,3}
q = 2
query[] = {0,1}
Output :
2
1
Explanation:
For 1st query, the given index is 0, a[0] = 4. There are 2 elements(1 and 3) which are less than or equal to 4.
For 2nd query, the given index is 1, a[1] = 1. There exists only 1 element(1) which is less than or equal to 1.

Example 2:

Input:
n = 4
a[] = {1,1,5,5}
b[] = {0,1,2,3}
q = 4
query[] = {0,1,2,3}
Output :
2
2
4
4
Explanation:
For 1st query and 2nd query, the given index is 0 and 1 respectively, a[0] = a[1] = 1. There are 2 elements(0 and 1) which are less than or equal to 1.
For 3rd query and 4th query, the given index is 2 and 3 respectively, a[2] = a[3] = 5. All the 4 elements are less than or equal to 5.

Your Task:
You don't need to take any input, as it is already accomplished by the driver code. You just need to complete the function **countElements()** that takes array **a** and **b** of size **n,** and array **query** of size **q** as parameters and returns an array that contains the answer to the corresponding queries.

Expected Time Complexity: O(n+q+maximum of **a** and **b**).

Expected Auxiliary Space: O(maximum of **a** and **b**).

Constraints:
$1 \leq q \leq n \leq 10^5$
$1 \leq a[i], b[i] \leq 10^5$
$0 \leq query[i] < n$

CODE

```
#User function Template for python3

class Solution:

    def countElements(self, a, b, n, query, q):

        # Preprocess b to create a sorted version

        sorted_b = sorted(b)


        result = []

        for x in query:

            # Perform binary search to find the index of a[x] in sorted_b

            left, right = 0, n - 1

            while left <= right:

                mid = left + (right - left) // 2

                if sorted_b[mid] <= a[x]:

                    left = mid + 1

                else:

                    right = mid - 1


            # Index + 1 represents the count of elements less than or equal to a[x]

            result.append(left)
```

```
return result
```