Question - 1

SCORE: 50 points

Psychometric Testing

Binary Search | Data Structures | Easy | Algorithms | Arrays | Problem Solving

Psychometric testing is designed to find job-relevant information about an applicant that the traditional interview process would not uncover. It typically includes a combination of online aptitude and personality tests that measure cognitive ability and personality traits.

A company has psychometric scores for *n* candidates, and it will only extend job offers to candidates with scores in the inclusive range given by *[lowerLimit, upperLimit]*. Given the list of scores and a sequence of score ranges, determine how many candidates the company will extend offers to for each range of scores. For example, if the scores are *scores = [1,2,2,3,4]* and the limits are *2* and *4*, there are *4* candidates, i.e. those with scores *[2,2,3,4]* that match the conditions.

Function Description

Complete the *jobOffers* function in the editor below. The function must return an array of *q* integers where the value at each index *i* denotes the number of candidates in the inclusive range *[lowerLimits[i], upperLimits[]]* that the company will extend offers to.

jobOffers has the following parameter(s):

scores[scores[0],...scores[n-1]]: an array of integers

lowerLimit[lowerLimit[0],...lowerLimit[q-1]]: an array of integers

upperLimit[upperLimit[0],...upperLimit[q-1]]: an array of integers

Constraints

- $1 \le n \le 10^5$
- $1 \le scores[j] \le 10^9$
- $1 \le q \le 10^5$
- $1 \le lowerLimits[i] \le upperLimits[i] \le 10^9$

▼ Input Format For Custom Testing

Input from stdin will be processed as follows and passed to the function.

The first line contains an integer, *n*, that denotes the number of elements in *scores*.

Each line *j* of the *n* subsequent lines (where $0 \le j < n$) contains an integer that describes *scores[j]*.

The next line contains an integer, *q*, that denotes the number of elements in *lowerLimits*.

Each line *i* of the *q* subsequent lines (where $0 \le i < q$) contains an integer that describes *lowerLimits[i]*.

The next line contains an integer, *q*, that denotes the number of elements in *lowerLimits*.

Help

Each line *i* of the *q* subsequent lines (where *0 ≤ i < q*) contains an integer that describes *upperLimits[i]*.

▼ Sample Case 0

Sample Input For Custom Testing

Sample Input 0

```
5
1
3
5
6
8
1
2
1
6
```

Sample Output 0

```
3
```

Explanation 0

Given *scores = [1, 3, 5, 6, 8]*, *lowerLimits = [2]*, and *upperLimits = [6]*, perform the following *q = 1* query:

0. Find all the scores in the inclusive range *[2, 6]*. There are three such candidates (i.e., scores *3*, *5*, and *6*), so store *3* in index *0* of the return array.

The function then returns the array *[3]*.

▼ Sample Case 1

Sample Input For Custom Testing

Sample Input 1

```
3
4
8
7
2
2
4
2
8
4
```

Sample Output 1

```
3
1
```

Explanation 1

Given *scores = [4, 8, 7]*, *lowerLimits = [2, 4]*, and *upperLimits = [8, 4]*, perform the following *q = 2* queries:

0. Find all the scores in the inclusive range *[2, 8]*: there are three such candidates (i.e., scores *4*, *7*, and *8*), so store *3* in index *0* of the return array.