You are given N integers in sorted order not necessarily distinct. Also, you are given Q queries. In each query, you will be given an integer X and you have to tell whether that integer is present in the array. If so, you have to tell at which index (leftmost) it is present and if it is not present, you have to tell the rightmost index at which the current integer can be inserted such that the array remains sorted.

NOTE: You are not required to insert the value in the array. Array is not changed at any point.

Input Format

First line contains an integer N.
Second line contains N space-separarted integer in sorted order.
Third line contains an integer Q, the number of queries.
Each of the next Q lines contain a single integer X.

Output Format

For each query, print the index as described.

Sample Input

```
6
5 15 21 21 22 22
5
10
21
23
3
22
```

Sample Output

```
1
2
6
0
4
```

Code:

```
#include <iostream>

#include <vector>

#include <algorithm>
```

```
using namespace std;

int main() {
    int N;
    cin >> N;
    vector<int> arr(N);

    for (int i = 0; i < N; i++) {
        cin >> arr[i];
    }

    int Q;
    cin >> Q;  // Read the number of queries

    for (int i = 0; i < Q; i++) {
        int X;
        cin >> X;

        auto it = lower_bound(arr.begin(), arr.end(), X);
        int index = it - arr.begin();

        if (index < N && arr[index] == X) {
            cout << index << endl;
        } else {
            cout << index << endl;
        }
    }

    return 0;
}
```