**Queue Operations :-**

Easy Accuracy: 77.76% Submissions: 18K+ Points: 2

Given **N** integers, the task is to insert those elements in the queue. Also, given **M** integers, the task is to find the frequency of each number in the Queue.  
**Note:**

* insert() will be called N times by main().
* findFrequency() will be called M times by main();
* Where k is each element passing through respective function calls.

**Example 1:**

**Input:**

N = 8

1 2 3 4 5 2 3 1

M = 5

1 3 2 9 10

**Output:**

2

2

2

-1

-1

**Explanation:**

After inserting 1, 2, 3, 4, 5, 2, 3 and 1 into the queue, frequency of 1 is 2, 3 is 2 and 2 is 2. Since 9 and 10 are not there in the queue we output -1 for them.

**Example 2:**

**Input:**

N = 6

1 2 1 1 1 4

M = 4

1 5 4 3

**Output:**

4

-1

1

-1

**Explanation:**

After inserting 1, 2, 1, 1, 1 and 4 into the queue, frequency of 1 is 4 and that of 4 is 1. Since 5 and 3 are not there in the queue we output -1 for them.

**Your Task:**  
Your task is to complete the functions **insert()** and **findFrequency()** which inserts the element into the queue and find the count of occurences of element in the queue respectively.

**Expected Time Complexity:** O(N\*M)  
**Expected Space Complexity:** O(N)

**Constraints:**  
1 <= N <= 103  
1 <= M <= 103  
1 <= Elements of Queue <= 106

**Code :-**

//{ Driver Code Starts

//Initial Template for C++

#include <bits/stdc++.h>

using namespace std;

// } Driver Code Ends

//User function Template for C++

// Helper class Solution to implement

// insert() and findFrequency()

class Solution{

public:

// Function to insert element into the queue

void insert(queue<int> &q, int k){

// Your code here

q.push(k);

return;

}

// Function to find frequency of an element

// return the frequency of k

int findFrequency(queue<int> &q, int k){

// Your code here

int n = q.size(), count = 0;

while(n--){

int front = q.front();

q.pop();

q.push(front);

if(front==k)

++count;

}

return count;

}

};

//{ Driver Code Starts.

int main() {

int testcase;

cin>>testcase;

while(testcase-- > 0){

// Declaring Queue

queue<int> q;

int n, k;

cin>>n;

// Invoking object of Geeks class

Solution obj;

for(int i = 0;i<n;i++){

cin >> k;

obj.insert(q, k);

}

int m;

cin >> m;

for(int i = 0;i<m;i++){

cin >> k;

int f = obj.findFrequency(q, k);

if(f != 0){

cout<<f<<endl;

}

else{

cout<<"-1\n";

}

}

}

return 0;

}

// } Driver Code Ends

**T.C :- O(n)**

**S.C :- O(1) except for the given passed queue**