https://course.acciojob.com/idle?question=5e1f2cc6-85c6-4a3e-852f-a9c03fffecbc

- MEDIUM
- Max Score: 40 Points

Search In A Row Wise And Column Wise Sorted Matrix

You are given an `N * N` matrix of integers where each row and each column is sorted in increasing order. You are given a target integer `x`. Find the position of `x` in the matrix.

If it exists, then print the pair `i j` where `i` represents the row and `j` represents the column of the array, otherwise print `-1 -1`

For example: If the given matrix is:

```
1 2 5
3 4 9
6 7 10
```

We have to find the position of 4. We will print {1,1} since A[1][1] = 4.

Input Format

The first line of input contains a single integer `T`, representing the number of test cases or queries to be run.

Then the `T` test cases follow.

The first line of each test case contains two space-separated integers `N` and `X`, representing the size of the matrix and the target element respectively.

Each of the next `N` lines contains `N` space-separated integers representing the elements of the matrix.

Output Format

For each test case, print the position of `X`, if it exists, otherwise print `1 -1`.

Example 1

Input

```
2
3 4
1 2 5
3 4 9
6 7 10
2 5
4 5
8 6
```

Output

```
1 1
0 1
```

Explanation

The element exists in the array

Example 2

Input

```
2
3 16
2 4 8
3 6 9
4 7 16
1 10
4
```

Output

```
2 2
-1 -1
```

Explanation

The element exists in one tc and not in another

Constraints

`1 ≤ T ≤ 10`

`1 ≤ N ≤ 10^3`

`1 ≤ X ≤ 10^6`

`1 ≤ A[i][j] ≤ 10^6`

where 'T' is the number of test cases, 'N' is the number of rows and columns, 'X' is the target value, and A[i][j] is the elements of the matrix.

Topic Tags

- **Arrays**

My code

// n java

```
import java.util.*;
import java.lang.*;
import java.io.*;



//test case wrong




public class Main
{
	public static void main (String[] args) throws
java.lang.Exception
	{
		//your code here
      Scanner s=new Scanner(System.in);
      int n=s.nextInt();
      for(int l=0;l<n;l++)
       {
         int c=-1;
          int m=s.nextInt();
         int arr[][]=new int[m][m];
         int k=s.nextInt();
        for(int i=0;i<m;i++)
          for(int j=0;j<m;j++)
            {
              arr[i][j]=s.nextInt();
              if(arr[i][j]==k) {System.out.println(i+" "+j);c=5;}
```

```
            }
          if(c==-1)
          System.out.println(c+" "+c);
        }

        }
}
```