Searching in a 2D Array

Write a C Program to read a 2D array. Then, search for a particular given element. If found, display each occurrence of that element with its position(s).

Input Format

- First line specifies the **no: of rows** (m) and and the **no: of columns** (n) in the 2D Array.
- Second line contains m*n integers that signify the **values** for the array.
- Third line specifies the **element to be searched** for.

Constraints

Always make sure that your output has the same spacings as mentioned.

Output Format

- If element found, print "*Element found at (row,column)*".
- If not found, then print "*Element not found*".

Sample Input 0

```
2 3
6 1 4 8 2 1
8
```

Sample Output 0

```
Element found at (1,0)
```

Explanation 0

The given matrix would be,

```
6 1 4
8 2 1
```

Element '8' is at the (1,0) position

Sample Input 1

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

Sample Output 1

```
Element found at (0,0)
Element found at (1,2)
```

Explanation 1

The given matrix would be,

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

Element '1' is present at the (0,0) position and at the position (1,2)

Submissions: 37
Max Score: 10
Difficulty: Medium

Rate This Challenge:

More

Current Buffer (saved locally, editable)

C

```c
#include <stdio.h>
#include <string.h>
#include <math.h>
#include <stdlib.h>

int main()
{
    int m,n,i,j,a[20][20],k,c=0;
    scanf("%d%d",&m,&n);
    for(i=0;i<m;i++)
    {
        for(j=0;j<n;j++)
        {
            scanf("%d",&a[i][j]);
        }
    }
    scanf("%d",&k);

    for(i=0;i<m;i++)
    {
        for(j=0;j<n;j++)
            {
                if(a[i][j]==k)
                {
                        printf("Element found at (%d,%d)\n",i,j);
                        c++;
                }
            }
    }

    if(c == 0)
    {
        printf("Element not found");
        return 0;
    }
    return 0;
}
```

Line: 1 Col: 1

Upload Code as File Test against custom input Run Code Submit Code

Testcase 0 ✔ Testcase 1 ✔

Congratulations, you passed the sample test case.

Click the **Submit Code** button to run your code against all the test cases.

Input (stdin)

```
2 3
6 1 4 8 2 1
8
```

Your Output (stdout)

```
Element found at (1,0)
```

Expected Output

```
Element found at (1,0)
```