**Row with max 1s**

[array](http://www.practice.geeksforgeeks.org/tag-page.php?tag=array&isCmp=0)[Amazon](http://www.practice.geeksforgeeks.org/tag-page.php?tag=Amazon&isCmp=1)

Given a boolean 2D array. Find the row with the maximum number of 1s.

**Input:**

The first line of input contains an integer T denoting the number of test cases.  
The first line of each test case is n and m,n is the number of rows and m is the number of columns.  
The second line of each test case contains C[n][m].  
  
**Output:**

Print the row with the maximum number of 1s.  
  
**Constraints:**

1 ≤ T ≤ 50  
1 ≤ n,m ≤ 50  
0 ≤ C[] ≤ 1  
  
**Example:**

**Input:**  
1  
4 4  
0 1 1 1 0 0 1 1 1 1 1 1 0 0 0 0

**Output:**  
2

\*\*For More Examples Use Expected Output\*\*

<http://www.practice.geeksforgeeks.org/problem-page.php?pid=476>

#include <iostream>

#include <stdio.h>

using namespace std;

int main() {

  int t;

  scanf("%d", &t);

  while(t--) {

       int n, m;

       scanf("%d %d", &n, &m);

       int C[n][m];

       for(int i =0; i<n; i++) {

          for(int j =0; j<m; j++) {

              scanf("%d", &C[i][j]);

            }

        }

       int max\_unos=0;

       int max\_row=0;

       for(int i =0; i<n; i++) {

           int unos =0;

           for(int j =0; j<m; j++) {

                if(C[i][j] == 1)

                    unos++;

           }

           if(unos > max\_unos) {

              max\_unos=unos;

              max\_row = i;

           }

        }

        printf("%d**\n**", max\_row);

    }

  return 0;

}