**Boolean Matrix Problem**

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

Given a boolean matrix mat[M][N] of size M X N, modify it such that if a matrix cell mat[i][j] is 1 (or true) then make all the cells of ith row and jth column as 1.

**Input:**

The first line of input contains an integer T denoting the number of test cases.  
The first line of each test case is r and c,r is the number of rows and c is the number of columns.  
The second line of each test case contains all the elements of the matrix in a single line separated by a single space.  
  
**Output:**

Print the modified array.  
  
**Constraints:**

1 ≤ T ≤ 50  
1 ≤ r,c ≤ 20  
0 ≤ Elements of matrix ≤ 1  
  
**Example:**

**Input:**  
3  
2 2  
1 0 0 0  
2 3  
0 0 0 0 0 1  
3 4  
1 0 0 1 0 0 1 0 0 0 0 0

**Output:**  
1 1 1 0  
0 0 1 1 1 1  
1 1 1 1 1 1 1 1 1 0 1 1

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

<http://www.practice.geeksforgeeks.org/problem-page.php?pid=478>

#include <iostream>

#include <stdio.h>

#include <math.h>

#include <vector>

using namespace std;

int main() {

    int T;

    scanf("%d", &T);

    while(T--) {

        int R, C;

        scanf("%d %d", &R, &C);

        bool matriz[R][C];

        for(int i =0; i<R; i++) {

            for(int j =0; j<C; j++) {

              //scanf("%d", &matriz[i][j]);

              cin >> matriz[i][j];

            }

        }

        std::vector<int> filas;

        std::vector<int> columns;

        for (int i = 0; i < R; i++) {

            for (int j = 0; j < C; j++) {

                if (matriz[i][j] == true)

                {

                    filas.push\_back(i);

                    columns.push\_back(j);

                }

            }

        }

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

            for (int j = 0; j < C; j++) {

                if(std::find(filas.begin(), filas.end(), i) != filas.end()

                 ||std::find(columns.begin(), columns.end(), j) != columns.end() ) {

*/\* v contains x \*/*

                    matriz[i][j] = true;

                    }

              }

            }

        for (int i = 0; i < R; i++) {

            for (int j = 0; j < C; j++) {

                printf("%d ", matriz[i][j]);

            }

        }

        printf("**\n**");

    }

  system("pause");

 return 0;

}