**Find the number of islands:-**

Given a Matrix consisting of 0s and 1s. Find the number of islands of connected 1s present in the matrix.   
**Note:**A 1 is said to be connected if it has another 1 around it (either of the 8 directions).

**Input:**  
The first line of input will be the number of testcases **T**, then T test cases follow. The first line of each testcase contains two space separated integers N and M. Then in the next line are the NxM inputs of the matrix A separated by space .

**Output:**  
For each testcase in a new line, print the number of islands present.

**Your Task:**  
You don't need to read input or print anything. Your task is to complete the function **findIslands()** which takes the matrix A and its dimensions N and M as inputs and returns the number of islands of connected 1s present in the matrix. A 1 is said to be connected if it has another 1 around it (either of the 8 directions).

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

**Constraints:**  
1 <= T <= 100  
1 <= N, M <= 100  
0 <= A[i][j] <= 1

**Example(To be used only for expected output) :  
Input**  
2  
3 3  
1 1 0 0 0 1 1 0 1  
4 4  
1 1 0 0 0 0 1 0 0 0 0 1 0 1 0 0

**Output**  
2  
2

**Explanation**:  
**Testcase 1:** The graph will look like  
1 1 0  
0 0 1  
1 0 1  
Here, two islands will be formed  
First island will be formed by elements {A[0][0] ,  A[0][1], A[1][2], A[2][2]}  
Second island will be formed by {A[2][0]}**.**

**Testcase 2:**The graph will look like  
1 1 0 0  
0 0 1 0  
0 0 0 1  
0 1 0 0  
Here, two islands will be formed  
First island will be formed by elements {A[0][0] ,  A[0][1], A[1][2], A[2][3]}  
Second island will be formed by {A[3][1]}**.**