### Maximum no of 1's row

**Company Tags**

[**Paytm**](https://practice.geeksforgeeks.org/explore/?company%5b%5d=Paytm) [**Microsoft**](https://practice.geeksforgeeks.org/explore/?company%5b%5d=Microsoft) [**24\*7 Innovation Labs**](https://practice.geeksforgeeks.org/explore/?company%5b%5d=24*7%20Innovation%20Labs) [**D-E-Shaw**](https://practice.geeksforgeeks.org/explore/?company%5b%5d=D-E-Shaw)

Given a boolean 2D array, where each row is sorted. Find the row with the maximum number of 1s.

**Example 1:**

**Input:**

N = 3, M = 4

Mat[] = {{0 1 1 1},

{0 0 1 1},

  {0 0 1 1}}

**Output:** 0

**Explanation**: Row 0 has 3 ones whereas

rows 1 and 2 have just 2 ones.

**Example 2:**

**Input**:

N = 2, M = 2

Mat[] = {{0 1},

  {1 1}}

**Output:** 1

**Explanation**: Row 1 has 2 ones whereas

row 0 has just a single one.

### Java code

import java.io.\*;

import java.util.\*;

class CodingMaxima

{

public static void main (String[] args)

{

Scanner sc = new Scanner(System.in);

int t = sc.nextInt();

while(t-- > 0)

{

int n = sc.nextInt();

int m = sc.nextInt();

int [][]arr = new int[n][m];

for (int i = 0; i < n; ++i)

for (int j = 0; j < m; ++j)

arr[i][j] = sc.nextInt ();

System.out.println (new Sol().maxOnes (arr, n, m));

}

}

}

class Sol

{

public static int maxOnes (int Mat[][], int N, int M){

int max1=0, row =-1, l=0, h=M-1, mid=0;

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

l=0;

h=M-1;

while(l<=h){

mid = l + (h-l)/2;

if(Mat[i][mid]==1){

if((M-mid)>max1){

max1 = M-mid;

row=i;

}

h = mid-1;

}

else

l = mid+1;

}

}

return row;

}

}