ACM ICPC Team

by dheeraj

Problem | Submissions | Leaderboard | Discussions | Editorial | Topics

Problem Statement

You are given a list of N people who are attending ACM-ICPC World Finals. Each of them are either well versed in a topic or they are not. Find out the maximum number of topics a 2-person team can know. And also find out how many teams can know that maximum number of topics.

Note Suppose *a*, *b*, and *c* are three different people, then (*a*,*b*) and (*b*,*c*) are counted as two different teams.

Input Format

The first line contains two integers, N and M, separated by a single space, where N represents the number of people, and M represents the number of topics. N lines follow.
Each line contains a binary string of length M. If the i^{th} line's j^{th} character is 1, then the i^{th} person knows the j^{th} topic; otherwise, he doesn't know the topic.

Constraints
$2 \leq N \leq 500$
$1 \leq M \leq 500$

Output Format

On the first line, print the maximum number of topics a 2-person team can know.
On the second line, print the number of 2-person teams that can know the maximum number of topics.

Sample Input

```
4 5
10101
11100
11010
00101
```

Sample Output

```
5
2
```

Explanation

(*1*, *3*) and (*3*, *4*) know all the *5* topics. So the maximal topics a 2-person team knows is *5*, and only *2* teams can achieve this.

Related Topics

Finding Max Min

Bitwise OR

Submissions: 16175

Max Score: 25

Difficulty: Easy

More

Current Buffer (saved locally, editable) Java 8

```
import java.io.*;
import java.util.*;

class Node{
    int a,b;
    Node(int i,int j){
        a=i;
        b=j;
    }
}
public class Solution {
    static int compare(char c1[],char c2[]){
        int count1=0,count2=0;
        for(int i=0;i<c1.length;i++){
            if(c1[i]=='1')
                count1++;
            if(c2[i]=='1')
                count2++;
        }
        if(count1>count2)
            return 1;
        else if(count1<count2)
            return 2;
        else
            return -1;
    }
    static char[] calOR(char c1[],char c2[]){
        //System.out.println(max_array);
        char c[]=new char[c1.length];
        for(int i=0;i<c1.length;i++){
            if(c1[i]=='0' && c2[i]=='0'){
                c[i]='0';
                //System.out.println("if  "+String.valueOf(max_array));
            }
            else{
                c[i]='1';
                //System.out.println("else "+String.valueOf(max_array));
            }
        }
        //System.out.println(max_array);
        return c;
    }
    public static void main(String[] args) {
        Scanner s=new Scanner(System.in);
        int N,M,comp;
        N=s.nextInt();
        M=s.nextInt();
        char c[][]=new char[N][M];
        char max_array[]=new char[M];
        char tmp[]=new char[M];

        ArrayList<Node> array=new ArrayList<Node>();

        for(int i=0;i<M;i++)
            max_array[i]='0';
```

```
56
57          for(int i=0;i<N;i++)
58              c[i]=(s.next()).toCharArray();
59
60          for(int i=0;i<N;i++){
61              //System.out.println("\nc[i] : "+String.valueOf(c[i])+"////////////////////");
62              for(int j=i+1;j<N;j++){
63                  /*System.out.println("j : "+j);
64                  System.out.println("c[j] : "+String.valueOf(c[j]));
65                  System.out.println("max_array : "+String.valueOf(max_array));
66                  System.out.println("tmp : "+String.valueOf(tmp));*/
67                  tmp=calOR(c[i],c[j]);
68                  /*System.out.println("max_array : "+String.valueOf(max_array));
69                  System.out.println("tmp : "+String.valueOf(tmp));
70                  */
71                  comp=compare(tmp,max_array);
72                  //System.out.println(comp);
73                  if(comp==1){  //>
74                      array.clear();
75                      Node n=new Node(i,j);
76                      array.add(n);
77                      max_array=tmp;
78                  }else if(comp==-1){  //=
79                      Node n=new Node(i,j);
80                      array.add(n);
81                  }
82                  //System.out.println("max_array : "+String.valueOf(max_array));
83              }
84          }
85
86          int count=0;
87          for(int i=0;i<M;i++){
88              if(max_array[i]=='1')
89                  count++;
90          }
91          System.out.println(count);
92          System.out.println(array.size());
93      }
94  }
```

Line: 1 Col: 1

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