

# **ACM ICPC Team**



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**

 $2 \le N \le 500$ 

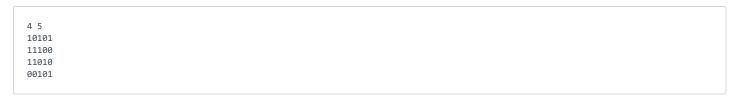
 $1 \leq M \leq 500$ 

## **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**



# 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.

Copyright © 2015 HackerRank. All Rights Reserved

**Related Topics** 

Finding Max Min

Bitwise OR

Submissions: 16175

Max Score: 25

Difficulty: Easy

More

```
Current Buffer (saved locally, editable) & 49
                                                                                            Java 8
                                                                                                                              *
 1 import java.io.*;
   import java.util.*;
 3
 4 ▼ class Node{
 5
        int a,b;
 6 ▼
        Node(int i,int j){
 7
            a=i;
 8
            b=j;
 9
10 }
11 ▼ public class Solution {
        static int compare(char c1[],char c2[]){
12 ▼
13
             int count1=0,count2=0;
            for(int i=0;i<c1.length;i++){</pre>
14 ▼
                 if(c1[i]=='1')
15
16
                     count1++;
17
                 if(c2[i]=='1')
18
                     count2++;
19
            if(count1>count2)
20
21
                return 1;
22
             else if(count1<count2)</pre>
23
                return 2;
24
            else
25
                 return -1;
26
27
        static char[] calOR(char c1[],char c2[]){
28
             //System.out.println(max_array);
29
            char c[]=new char[c1.length];
             for(int i=0;i<c1.length;i++){</pre>
30 ▼
31 ▼
                 if(c1[i]=='0' && c2[i]=='0'){
32
                     c[i]='0';
                     //System.out.println("if "+String.valueOf(max_array));
33
                 }
34
                 else{
35 ▼
36
                     c[i]='1';
37
                     //System.out.println("else "+String.valueOf(max_array));
38
39
            }
40
             //System.out.println(max_array);
41
            return c;
42
        }
        public static void main(String[] args) {
43 ▼
44
            Scanner s=new Scanner(System.in);
45
            int N,M,comp;
46
            N=s.nextInt();
47
            M=s.nextInt();
            char c[][]=new char[N][M];
48
49
            char max_array[]=new char[M];
50
            char tmp[]=new char[M];
51
52
            ArrayList<Node> array=new ArrayList<Node>();
53
             for(int i=0;i<M;i++)</pre>
54
55
                max_array[i]='0';
```

```
56
57
            for(int i=0;i<N;i++)</pre>
58
                c[i]=(s.next()).toCharArray();
59
60 ▼
            for(int i=0;i<N;i++){</pre>
                //System.out.println("\nc[i] : "+String.valueOf(c[i])+"//////////);
61
                for(int j=i+1;j<N;j++){</pre>
62 ₹
                    /*System.out.println("j : "+j);
63 ₹
64
                    System.out.println("c[j] : "+String.valueOf(c[j]));
                    System.out.println("max_array : "+String.valueOf(max_array));
65
66
                    System.out.println("tmp : "+String.valueOf(tmp));*/
                    tmp=calOR(c[i],c[j]);
67
                    /*System.out.println("max_array : "+String.valueOf(max_array));
68 ▼
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 ▼
79
                        Node n=new Node(i,j);
80
                        array.add(n);
81
                    //System.out.println("max_array : "+String.valueOf(max_array));
82
83
                }
84
            }
85
86
            int count=0;
87 ▼
            for(int i=0;i<M;i++){</pre>
88
                if(max_array[i]=='1')
89
                    count++;
90
            System.out.println(count);
91
92
            System.out.println(array.size());
93
        }
94
   }
                                                                                                                  Line: 1 Col: 1
```

■ Test against custom input

Join us on IRC at #hackerrank on freenode for hugs or bugs.

Contest Calendar | Blog | Scoring | Environment | FAQ | About Us | Support | Careers | Privacy Policy | Request a Feature

**1** Upload Code as File

Run Code

Submit Code