

Problem: ACM ICPC Team

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

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.

Solution:

```

int topicsKnow(string str1, string str2,int length)
{
    int topics=0;

    for(int i=0; i<length; i++)
        {if(str1[i]=='1' || str2[i]=='1')
            {topics+=1;}}
    // cout<<topics<<endl;
    return topics;
}

```

```

int main()
{
    int n, m;
    cin>>n >>m;
    string topics[n];
    for(int i=0; i<n; i++)
        { cin>> topics[i]; }
}

```

```

int maxKnown=0;
int teams=0;
/*procesisng the data*/
for(int i=0; i<n-1; i++)
    {for(int j=i+1; j<n; j++)
        { int topic=topicsKnow(topics[i],topics[j], m);
            if(topic>maxKnown)
                {teams=1; maxKnown=topic;}
            else if(topic==maxKnown)
                {teams+=1;}}
        }
    }
cout<<maxKnown<<endl;
cout<<teams<<endl;
return 0;
}

```

- Anshul Aggarwal