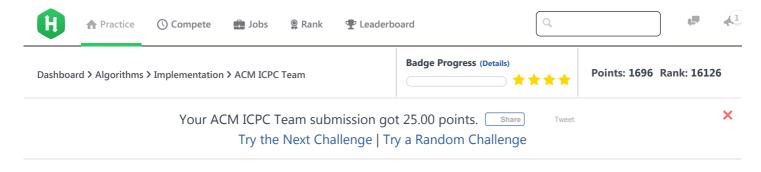
3/31/2018 HackerRank





| Problem | Submissions | Leaderboard | Discussions | Editorial 🔒 | Topics | |
|---------|-------------|-------------|-------------|-------------|--------|--|
|---------|-------------|-------------|-------------|-------------|--------|--|

There are a number of people who will be attending ACM-ICPC World Finals. Each of them may be well versed in a number of topics. Given a list of topics known by each attendee, you must determine the maximum number of topics a 2-person team can know. Also find out how many ways a team can be formed to know that many topics. Lists will be in the form of bit strings, where each string represents an attendee and each position in that string represents a field of knowledge, 1 if its a known field or 0 if not.

For example, given three attendees' data as follows:

10101 11110 00010

These are all possible teams that can be formed:

Members Subjects
(1,2) [1,2,3,4,5]
(1,3) [1,3,4,5]
(2,3) [1,2,3,4]

In this case, the first team will know all 5 subjects. They are the only team that can be created knowing that many subjects.

Input Format

The first line contains two space-separated integers n and m, where n represents the number of attendees and m represents the number of topics.

Each of the next n lines contains a binary string of length m. If the ith line's jth character is 1, then the ith person knows the jth topic.

Constraints

 $2 \le n \le 500$ $1 \le m \le 500$

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 ways to form a 2-person team that knows the maximum number of topics.

Sample Input

Sample Output

3/31/2018 HackerRank

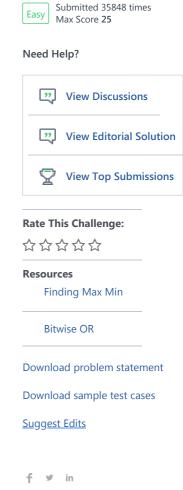
5 2

Explanation

Calculating topics known for all permutations of 2 attendees we get:

```
(1,2) 
ightarrow 4
(1,3) \rightarrow 5
(1,4) \rightarrow 3
(2,3) 	o 4
```





```
Current Buffer (saved locally, editable) \ \mathscr{V} \ \mathfrak{O}
                                                                                        C++14
                                                                                                                          Ö
 1 ▼ #include <iostream>
 2 #include <vector>
 3 #include <map>
   #include <algorithm>
 5
 6
    int main()
 7 ▼ {
         int n; std::cin >> n; //the number of attendees
 8
 9
         int m; std::cin >> m; //represents the number of topics(str length)
10
11
         std::map<int,int> answer;
         std::vector<std::string> vec(n);
12
13
         for(auto &it: vec) std::cin >> it;
14
15
         int maxTopics = std::numeric_limits<int>::min();
16
         for(int i = 0; i < n-1; ++i)
17
```

3/31/2018 HackerRank

```
18 ▼
               for(int j = i+1; j < n; ++j)
  19
  20 ₹
  21 ▼
                   std::string A = vec[i];
  22 ₹
                   std::string B = vec[j];
  23
                   int count = 0;
  24
  25
                   for(int s = 0; s < m; ++s)
                       if( (A[s] == '1') || (B[s] == '1'))
  26 ₹
  27
                           ++count;
  28
  29
                   maxTopics = std::max(maxTopics, count);
  30 ▼
                   answer[count]++;
               }
  31
  32
          }
          std::cout << maxTopics << std::endl << answer[maxTopics] << std::endl;</pre>
  33 ▼
  34
          return 0;
  35 }
  36
                                                                                                               Line: 36 Col: 1
                                                                                                   Run Code
Upload Code as File
                      Test against custom input
                                                                                                                Submit Code
```

 $Contest\ Calendar |Blog|Scoring|Environment|FAQ|About\ Us|Support|Careers|Terms\ Of\ Service|Privacy\ Policy|Request\ a\ Feature$