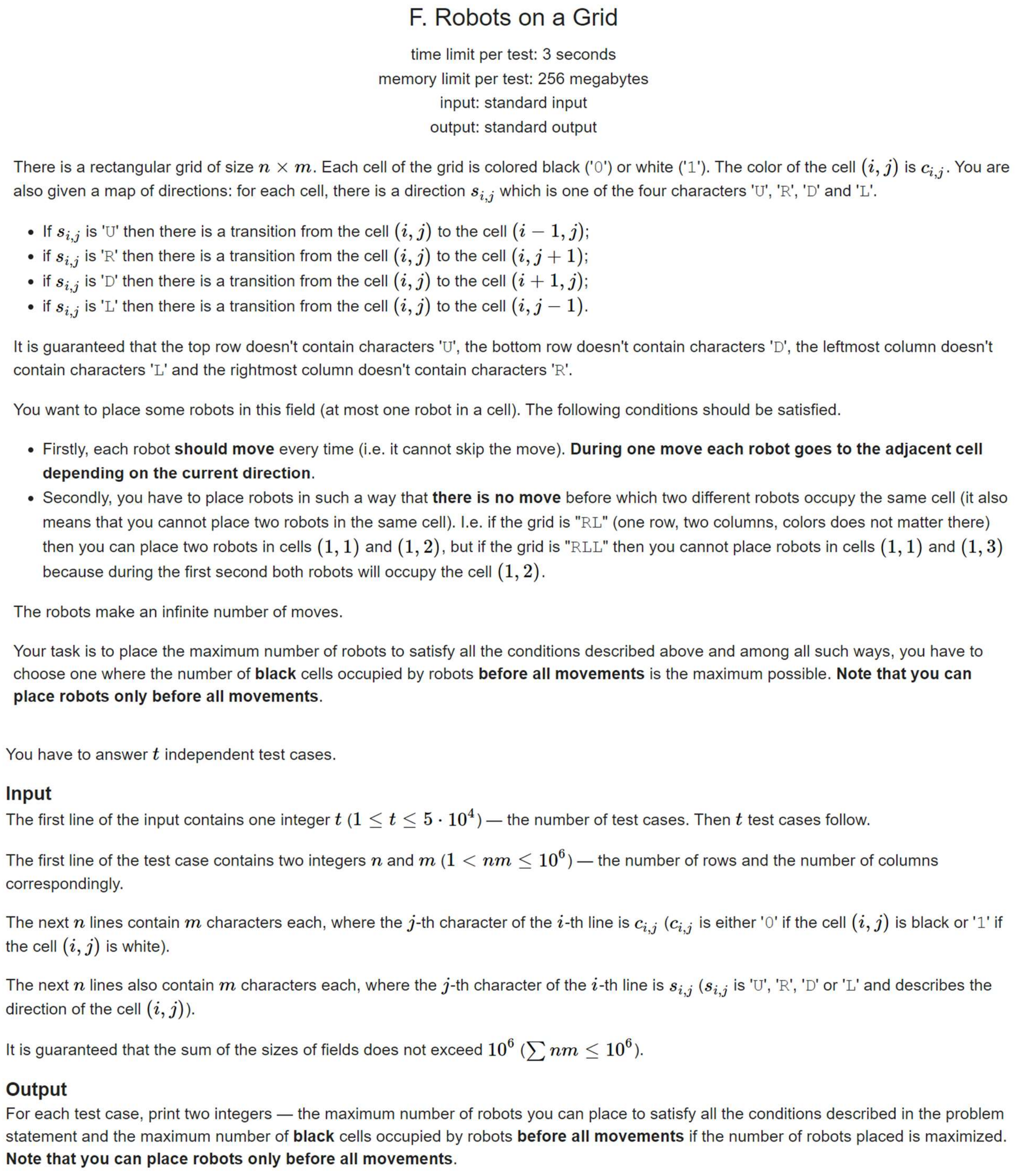
To zadanie pochodzi z platformy Codeforces, link do zadania:
https://codeforces.com/problemset/problem/1335/F



F. Robots on a Grid

time limit per test: 3 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

There is a rectangular grid of size $n \times m$. Each cell of the grid is colored black ('0') or white ('1'). The color of the cell (i, j) is $c_{i,j}$. You are also given a map of directions: for each cell, there is a direction $s_{i,j}$ which is one of the four characters 'U', 'R', 'D' and 'L'.

- If $s_{i,j}$ is 'U' then there is a transition from the cell (i, j) to the cell $(i-1, j)$;
- if $s_{i,j}$ is 'R' then there is a transition from the cell (i, j) to the cell $(i, j+1)$;
- if $s_{i,j}$ is 'D' then there is a transition from the cell (i, j) to the cell $(i+1, j)$;
- if $s_{i,j}$ is 'L' then there is a transition from the cell (i, j) to the cell $(i, j-1)$.

It is guaranteed that the top row doesn't contain characters 'U', the bottom row doesn't contain characters 'D', the leftmost column doesn't contain characters 'L' and the rightmost column doesn't contain characters 'R'.

You want to place some robots in this field (at most one robot in a cell). The following conditions should be satisfied.

- Firstly, each robot **should move** every time (i.e. it cannot skip the move). **During one move each robot goes to the adjacent cell depending on the current direction**.
- Secondly, you have to place robots in such a way that **there is no move** before which two different robots occupy the same cell (it also means that you cannot place two robots in the same cell). I.e. if the grid is "RL" (one row, two columns, colors does not matter there) then you can place two robots in cells $(1, 1)$ and $(1, 2)$, but if the grid is "RLL" then you cannot place robots in cells $(1, 1)$ and $(1, 3)$ because during the first second both robots will occupy the cell $(1, 2)$.

The robots make an infinite number of moves.

Your task is to place the maximum number of robots to satisfy all the conditions described above and among all such ways, you have to choose one where the number of **black** cells occupied by robots **before all movements** is the maximum possible. **Note that you can place robots only before all movements**.

You have to answer t independent test cases.

Input

The first line of the input contains one integer t ($1 \le t \le 5 \cdot 10^4$) — the number of test cases. Then t test cases follow.

The first line of the test case contains two integers n and m ($1 < nm \le 10^6$) — the number of rows and the number of columns correspondingly.

The next n lines contain m characters each, where the j-th character of the i-th line is $c_{i,j}$ ($c_{i,j}$ is either '0' if the cell (i, j) is black or '1' if the cell (i, j) is white).

The next n lines also contain m characters each, where the j-th character of the i-th line is $s_{i,j}$ ($s_{i,j}$ is 'U', 'R', 'D' or 'L' and describes the direction of the cell (i, j)).

It is guaranteed that the sum of the sizes of fields does not exceed 10^6 ($\sum nm \le 10^6$).

Output

For each test case, print two integers — the maximum number of robots you can place to satisfy all the conditions described in the problem statement and the maximum number of **black** cells occupied by robots **before all movements** if the number of robots placed is maximized. **Note that you can place robots only before all movements**.

Example

input Copy

```
3
1 2
01
RL
3 3
001
101
110
RLL
DLD
ULL
3 3
000
000
000
RRD
RLD
ULL
```

output Copy

```
2 1
4 3
2 2
```