

Competitive Programming and Contests

06/07/2018

A grid and a domino

We are given rectangular grid with h rows and w columns. Each cell is a square, either empty (denoted by '.') or forbidden (denoted by '#'). Rows are numbered 1 through h from top to bottom. Columns are numbered 1 through w from left to right.

We have a single domino that we could place somewhere in the grid. A domino will occupy exactly two adjacent cells, located either in one row or in one column. Both adjacent cells must be empty (i.e., '.') and must be inside a grid.

Then, we are given a sequence of q queries. Each query specifies a rectangle in the grid and the goal is to report in how many ways we can place a single domino inside of the chosen rectangle.

Problem from Codeforces (<http://codeforces.com/problemset/problem/611/C?locale=en>).

We point out that there exist

1. A $\Theta(wh + A)$ time solution, where A is the overall area covered by the queries' rectangles. If you find and implement this solution, your grade will be 25;
2. A $\Theta(wh + q)$ time solution. If you find and implement this solution, your grade will be 30.

Input. The first line contains values h and w , separated by a space.
The next h lines describe the grid.

The next line contains the value q .

The next h lines describe the queries. A query is a quadruple r_1, c_1, r_2, c_2 , where r_1 and c_1 denote the row and the column (respectively) of the upper left cell of the rectangle and numbers r_2 and c_2 denote the row and the column (respectively) of the bottom right cell of the rectangle.

Output. The number of length of the longest subarray.

Example

Input

```
5 8
....#..#
.#.....
##.#....
##..#.#
.....
4
1 1 2 3
4 1 4 1
1 2 4 5
2 5 5 8
```

Output

```
4
0
10
15
```