Pathfinding

The first problem of this practice session is pretty easy and standard – you have to find your way through a maze.

Problem specification

You are given a description of a maze. Produce a short sequence of moves that exits it.

Input specification

The first line of the input contains two positive integers R and C – the number of rows and columns in the maze. Rows are numbered 1 to R, and columns 1 to C.

R lines follow. Each of them contains C integers. The c-th integer in the r-th row is 0 if the cell at (r, c) is free, and it is 1 if the cell at (r, c) is a wall.

In the final line are the coordinates of the cell where you start.

Output specification

The output shall contain any sequence of at most 200 moves. The input will be such that at least one such sequence will exist.

All moves must be valid (i.e., may not crash into a wall), and the last move must be the only move that exits the maze.

Describe the moves using letters `U` for up, `D` for down, `L` for left and `R` for right. (Up is the direction in which row numbers decrease, left is the direction in which column numbers decrease.)

You may separate the moves in the output by whitespace.

Flash version

For your convenience, we provide Flash applications in which you can traverse the given mazes using your keyboard. If you successfully exit the maze, the application will print your sequence of moves, you can simply copy it and submit it.

Example

input

```
7 9
1 1 1 1 1 1 1 1 1
1 0 0 0 0 0 0 0 1
1 0 1 1 1 1 1 0 1
1 0 1 0 0 0 1 0 0
1 0 1 1 1 0 1 0 1
1 0 0 0 0 0 1 0 1
1 1 1 1 1 1 1 1 1
6 2
```

output

```
R L U U U U R R R
R R R D D D U R R
```

Quick search

The second problem in this practice session is based on an actual pen and paper puzzle. The puzzle, as you'll soon discover, is quite tricky – and harder than it might seem.

Problem specification

In this task you will be given a rectangle divided into several polygons. All of these polygons are distinct, except for two that are identical. Your task is to find the two identical ones.

Note: We call two polygons identical iff one of them can be obtained from the other by a combination of translations and rotations. Note that it is **not** allowed to use axis symmetry – i.e., if a polygon is just a mirror image of another one, they are considered distinct for the purpose of this problem.

Input specification

The input is an image of the rectangle, including labels for rows and columns.

Output specification

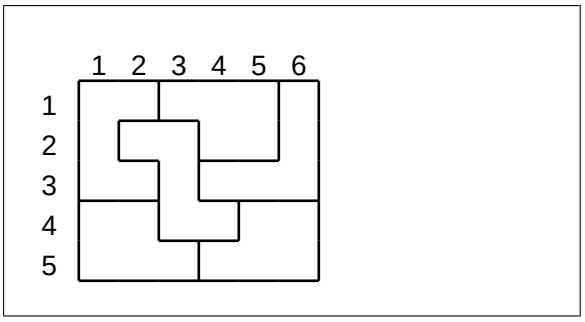
Each polygon in the input is uniquely identified by the coordinates (r, c) of its upper left corner. More precisely, to describe a polygon, we first find the smallest row r in which the polygon contains at least one cell, and then pick the smallest possible column c it contains in this row.

Once you locate the two polygons, output the coordinates (r_1, c_1) and (r_2, c_2) of their upper left corners.

To make the output unique, we require that $r_1 \leq r_2$, and if $r_1 = r_2$, then $c_1 < c_2$. I.e., first output the polygon that starts earlier.

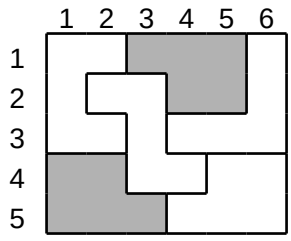
Example

input



output

```
1 3
4 1
```



Note that the polygon at (4,5) is not identical to the highlighted pair.

Reverse quick search

After creating problem Q, we thought it would be entertaining to put you into problemsetters' shoes for a moment. In this task you will find out a way to create test data for problem Q.

Problem specification

You are given integers R, C, and A such that A divides RC.

Your task is to take a $R \times C$ rectangle and divide it into polygons. Each of the polygons must consist of exactly A unit squares. All polygons must be distinct, except for two polygons that must be identical.

Note: We call two polygons identical iff one of them can be obtained from the other by a combination of translations and rotations. Note that it is **not** allowed to use axis symmetry – i.e., if a polygon is just a mirror image of another one, they are considered distinct for the purpose of this problem.

Input specification

The input contains a single line with the three integers R, C, and A.

Output specification

After you divide the rectangle into $N = RC/A$ polygons, number them from 1 to N (in any order you wish).

Output R rows with C integers in each of them. The c-th integer in the r-th row must be the number of the polygon that contains the cell (r, c).

Any valid output will be accepted.

Example

input

```
5 6 5
```

output

```
1 1 2 2 2 3
1 4 4 2 2 3
1 1 4 3 3 3
6 6 4 4 5 5
6 6 6 5 5 5
```