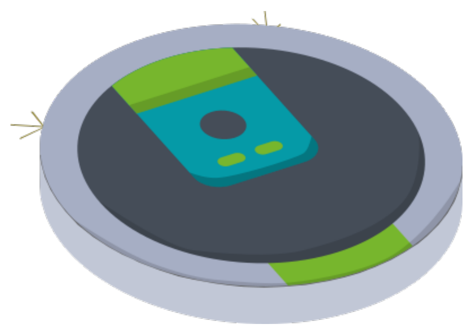
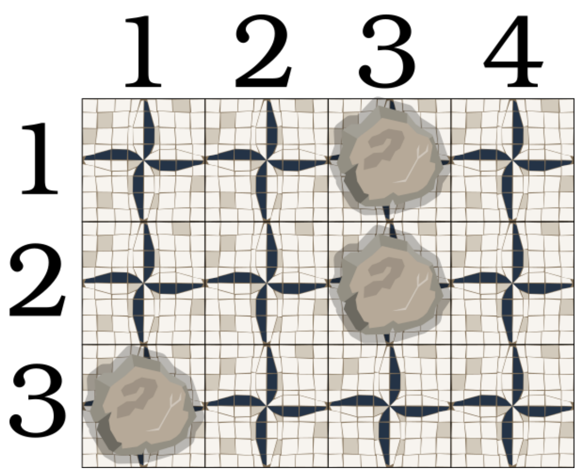
Problem A - Aspirar a Sala

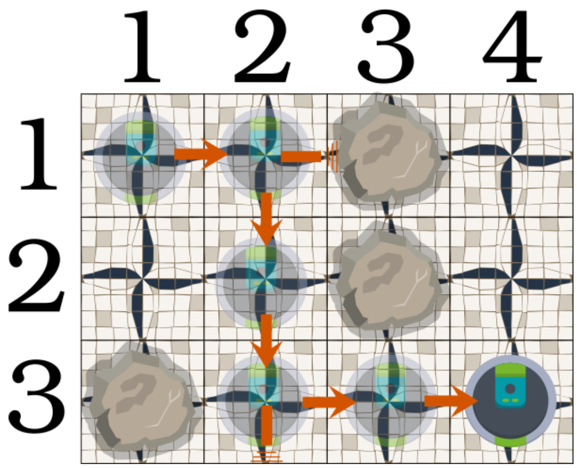
If this is your first time competing, check our instruction page for detailed information on the format of this problem.

The new version of ONI, the latest robot vacuum cleaner, is in the final stage of development. This robot models a room by a grid of $\boldsymbol{N}$ rows by $\boldsymbol{M}$ columns, where each cell of the grid can be occupied (representing an obstacle) or unoccupied, and there are walls around the room. It is guaranteed that the upper left corner, that is, the position $(1, 1)$, is always unoccupied. The following image illustrates an example for $\boldsymbol{N} = 3$ and $\boldsymbol{M} = 4$, where the rocks represent occupied cells and the remaining cells are unoccupied.





The ONI is initially placed in the upper left corner and can move around the grid in 4 directions: right, left, down or up, and can only move to a cell if it is unoccupied. Thanks to the best Portuguese artificial intelligence technology, the ONI can create a path to follow to clean the room efficiently. This path consists of $\boldsymbol{K}$ instructions that tell an ONI robot to move in one of the 4 directions: right, left, down or up. If the robot receives an instruction that results in a movement to outside the grid or to an occupied cell, then the robot ignores that instruction. For the example above, let's assume that an ONI robot starts in the upper left corner and receives the following $\boldsymbol{K} = 7$ instructions: right, right, down, down, down, right, right. The following image illustrates the behavior of the robot for these instructions:



The behavior of the robot is detailed as follows:

- The robot starts by receiving an instruction to move 'right' and moves to the right, ending up at the position $(1, 2)$.
- Next, it receives another instruction to move 'right', but it is ignored as the cell $(1, 3)$ is occupied.
- Two instructions to move 'down' follow, resulting in the robot occupying the position $(2, 2)$ and then $(3, 2)$.
- Another instruction to move 'down' follows, but it is ignored as it would result in the robot moving outside the grid.
- Finally, the last two instructions to move 'right' result in the final position, $(4, 3)$.

Part I

Given the configuration of an $\boldsymbol{N}$ by $\boldsymbol{M}$ room where the upper left corner is unoccupied, as well as $\boldsymbol{T}$ sets of movement instructions, where the i-th set has $\boldsymbol{K_i}$ instructions. Assuming that an ONI robot starts in the upper left corner of the room, for each set of instructions, determine the coordinates of the robot's final position after completing all the instructions. (See the Input Format section for more information on how these values will be provided.)

Example

Using the previous example with $\boldsymbol{N} = 3$, $\boldsymbol{M} = 4$, and $\boldsymbol{K} = 7$, the final position of the robot is $(3, 4)$.

Constraints

The following limits are guaranteed for all test cases of this part that will be given to the program:

$1 \le \boldsymbol{N}, \boldsymbol{M} \le 100$	Size of the room
$1 \le \boldsymbol{T} \le 10$	Number of instruction sets
$1 \le \boldsymbol{K_i} \le 100$	Number of instructions

The test cases of this part of the problem are organized into two groups:

Subtask	Points	Additional Constraints
1	15	$\boldsymbol{M} = 1$
2	25	No further restrictions

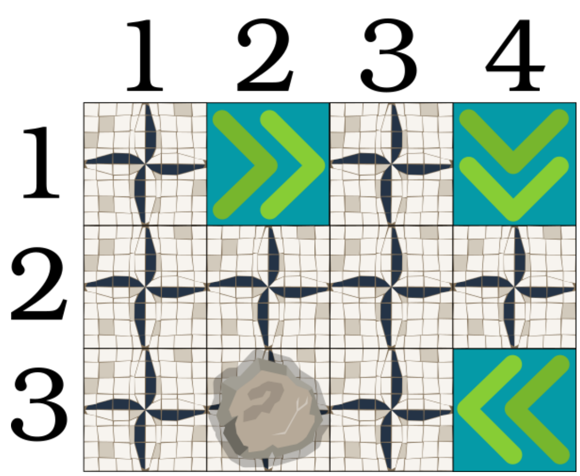
Part II

There is also a set of special cells that have a predetermined direction. When an ONI robot steps on one of these cells, it enters in "automatic" mode, meaning that it continues to follow the direction given by the special cell until: the next movement causes the robot to be off the grid or in an occupied cell, in which case it stops walking and regains control over its movement, that is, it exits "automatic" mode; or the robot moves to a special cell, in which case it starts following the direction of the new special cell it just stepped on. Notice that it is possible for the robot to enter a cycle, that is, the robot may never exit "automatic" mode after stepping on a special cell.

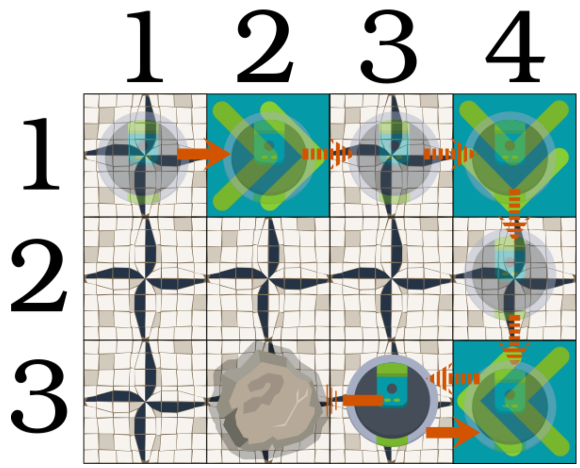
Given the configuration of an $\boldsymbol{N}$ by $\boldsymbol{M}$ room where the upper left corner is unoccupied, as well as $\boldsymbol{T}$ sets of movement instructions, where the i-th set has $\boldsymbol{K_i}$ instructions, assuming an ONI robot starts at the upper left corner of the room, for each set of instructions determine whether the robot enters a cycle or, otherwise, the coordinates of the robot's final position after completing all instructions.

Example

Consider the following room with $\boldsymbol{N} = 3$, $\boldsymbol{M} = 4$.



If we have the following $\boldsymbol{K} = 3$ instructions: right, right, right, then the robot follows the following path, ending in the cell $(3, 3)$.



The behavior of the robot is detailed as follows:

- The robot starts by receiving an instruction to go 'right' and moves to the right, ending up in position $(1, 2)$. This cell is special, so the robot enters automatic mode and begins moving to the right.
- During the automatic movement, the robot moves to the right twice, first to cell $(1, 3)$ and then to $(1, 4)$. This cell is special, so the robot now starts moving downwards.
- During the automatic movement, the robot moves downwards twice, first to cell $(2, 4)$ and then to $(3, 4)$. This cell is special, so the robot now starts moving to the left.
- During the automatic movement, the robot moves to the left, ending up in cell $(3, 3)$. The next movement would result in the robot being in an occupied cell, so the robot exits automatic mode and continues to follow the rest of the instructions.
- The robot receives an instruction to go 'right' and moves to the right, ending up in position $(3, 4)$. This cell is special, so the robot enters automatic mode and starts moving to the left.
- During the automatic movement, the robot moves to the left, ending up in cell $(3, 3)$. The next movement would result in the robot being in an occupied cell, so the robot exits automatic mode and continues to follow the rest of the instructions.
- The two previous lines repeat and the robot finishes its movement in cell $(3, 3)$.

Notice that if we had a special cell with an upward direction instead of the obstacle, then the robot would enter a cycle, indefinitely moving between the 4 special cells.

Constraints

The following limits are guaranteed for all test cases of this part that will be given to the program:

$1 \leq \boldsymbol{N}, \boldsymbol{M} \leq 100$	Size of the room
$1 \leq \boldsymbol{T} \leq 10$	Number of instruction sets
$1 \leq \boldsymbol{K_i} \leq 100$	Number of instructions

The test cases of this part of the problem are organized into two groups:

Subtask	Points	Additional Constraints
3	25	$M = 1$
4	35	No further restrictions

Summary of Subtasks

The test cases for the problem are organized into four groups with different additional restrictions:

Subtask	Points	Part	Additional Constraints
1	15	Part I	$M = 1$
2	25	Part I	No further restrictions
3	25	Part II	$M = 1$
4	35	Part II	No further restrictions

Input Format

The first line contains an integer P, which represents the part that the test case represents. If it is 1, then the test case refers to Part I, if it is 2 then it refers to Part II.

This is followed by a line with two integers separated by spaces, first N, representing the number of rows of the grid, followed by M, the number of columns.

Next, N lines follow, each with M characters, representing the grid. A '`.`' is used to represent an unobstructed cell and a '`#`' is used to represent a cell with an obstacle. For Part II, we can also have the characters '`D`' (right), '`E`' (left), '`B`' (down), '`C`' (up), representing special cells with the indicated direction.

Next is a line containing an integer T, representing the number of instruction sets.

T sets of lines follow, each following the following format:

A line with an integer K, the number of instructions.

A line with a sequence of K characters without spaces between them, representing instructions. The characters can be '`D`' (right), '`E`' (left), '`B`' (down), '`C`' (up), representing an instruction in the indicated direction.

It is guaranteed that the upper left corner of all grids will always be an unobstructed cell.

Output Format

Part I

The output should contain T lines with two integers separated by spaces, representing the row and column of the cell where the robot ends up after executing all instructions.

Part II

The output should contain T lines containing either the word 'ciclo', if the robot enters a cycle, or two integers separated by spaces, representing the row and column of the cell where the robot ends up executing all instructions.

Notice: there must be exactly one single space between each integer and there should be no space at the end of the line (that is, after the last integer there should only be a line break). If this format is not respected, the result of a submission will be `Presentation Error` (see the instructions for more information).

Example 1 Input

```
1
3 4
..#.
..#.
#...
1
7
DDBBBDD
```

Example 1 Output

```
3 4
```

Example 1 Description

This example corresponds to the example mentioned in Part I of the problem statement.

Example 2 Input

```
2
3 4
.D.B
....
.#.E
1
3
DDD
```

Example 2 Output

```
3 3
```

Example 2 Description

This example corresponds to the Part II example mentioned in the statement.

Example 3 Input

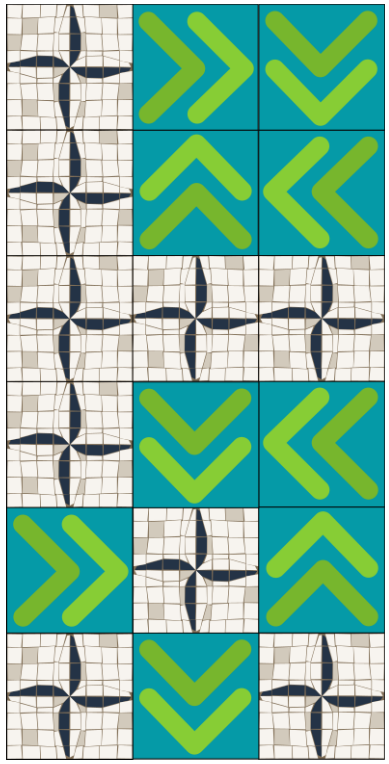
```
2
6 3
.DB
.CE
...
.BE
D.C
.B.
2
4
DEBC
5
BBBBD
```

Example 3 Output

```
ciclo
6 3
```

Example 3 Description

The grid of the previous example corresponds to the following:



Organizers

Sponsors

Supported by