

Czech ACM Student Chapter

Charles University in Prague
Slovak University of Technology
University of Žilina
Matej Bel University in Banská Bystrica

Czech Technical University in Prague

Technical University of Ostrava
Pavol Jozef Šafárik University in Košice
Masaryk University
University of West Bohemia



CTU Open Contest 2014

Karel the Robot

`karel.c`, `karel.cpp`, `Karel.java`

Karel wants to register his robot Karel for a robot contest. The aim of the contest is to escape from a maze. The maze is a square grid W units wide and H units high where every unit square is either a wall or free terrain. One of the squares with free terrain is designated as an exit.

The design of the robot was inspired by the ancient (it is even much older than CTU Open) educational programming language called Karel. The robot understands three commands: Step, Right, and Left. When executing the Step command, the robot moves to the neighboring square in the forward direction if possible. If the square one unit ahead is a wall or it lies outside the maze, the robot does not move. The Right command turns the robot 90° clockwise and the Left command turns it 90° counterclockwise. The robot has a memory for a program consisting of up to 10 commands. After the robot executes the last command, it continues executing the program from the beginning.

The robot is initially positioned at some free square. We do not know its position but we know it is facing up (towards the neighboring square one row above). Karel wants to write a universal program that will make the robot escape from *any* such initial position. That means, the robot will eventually reach the exit during the program execution.

This sounds like a nice contest problem, doesn't it? We want to give you an idea what it is like to organize a programming contest. Therefore, your task is to write a validator for the problem described above. (You may read more about validators in the *validate* problem.)

Input Specification

The input contains several test cases. The first line of each input contains two space-separated integers: H and W satisfying $1 \leq H, W \leq 100$. Each of the following H lines contains exactly W characters describing the maze. The character "`X`" means a wall, "`.`" is free terrain, and "`E`" is free terrain with the exit.

The next line contains a single integer L, $1 \leq L \leq 10$, the length of the program. The last line of the test case contains L characters describing the commands of the program. The character "`S`" stands for Step, "`L`" for Left, and "`R`" for Right.

Output Specification

Print a single line for each test case. The line must contain "`OK`" if the robot escapes for every initial position using the given program. Otherwise, the line contains the number of initial positions (including the exit), from which the robot escapes successfully.

Sample Input

```
3 4
E...
X.X.
....
3
SSL
3 3
E..
...
...
3
RSS
```

Output for Sample Input

```
5
OK
```