forfront

Technical Test

Overview

A squad of robotic spiders are to be sent to explore micro fractures on the wall of a building!

This wall, which is rectangular, must be navigated by the Spiders so that their on-board diagnostics can get a complete view of the wall from close up before sending the data back to the control deck. The spiders are highly autonomous and will follow the instructions sent to them in the start command.

A spider's position is represented by a combination of x and y coordinates and a current orientation. The wall is divided up into a grid to simplify navigation.

An example position might be "0 0 Up" which means the spider is in the bottom left corner and facing up the wall.

In order to control a spider, Control sends a simple string of letters:

- The possible letters are 'L', 'R' and 'F'.
- 'L' and 'R' makes the spider spin 90 degrees left or right respectively, without moving from its current spot.
- 'F' means move forward one grid point, and maintain the same direction.

Assume that the square directly Up from (x, y) is (x, y+1).

INPUT

Three lines of input are to be received:

1) The first line of input is information pertaining to the size of the wall. 0 0 (bottom left) to x y (Top right)
2) The second line of input is information about the spider's current location and orientation. This is made up of two integers and a word separated by spaces, corresponding to the x and y coordinates and the spider's orientation. E.g. 4 10 Left
3) The last line of input received is a series of instructions telling the spider how to explore the wall. E.g. FLFLFRFFLF

OUTPUT

The output for the spider should be its final co-ordinates and heading. E.g. 5 8 Right

Example

Test Input:

7 15
4 10 Left
FLFLFRFFLF

Expected Output:

5 7 Right

Requirement

Please write code that accepts input in the form specified and returns the final co-ordinates and heading. A unit test is to be created to verify the final result for a given set of inputs.

Please pay particular attention to making the code you write robust, readable and clean.

Candidate Notes: Whilst it should be written C#, you are free to approach the problem however you wish. We expect for the test to be completed in somewhere between 1 - 3 hours. Please submit zipped VS solution to your recruiter or provide a link to publicly accessible source control (i.e. GitHub etc.)