

Mars Rover Exercise

Please prepare a console application in C# for the following problem. You can follow any development methodology you wish for this. The interviewer will be interested in the following

- Understanding of the problem and approach
 - Code design and architecture
 - Unit testing
-

A squad of robotic rovers are to be landed by NASA on a plateau on Mars.

This plateau, which is curiously square, must be navigated by the rovers so that their on board cameras can get a complete view of the surrounding terrain to send back to Earth.

A rover's position is represented by a combination of an x and y co-ordinates and a letter representing one of the four compass points e.g. N (North), S (South), E (East) and W (West). The plateau is divided up into a 5 x 5 square grid to simplify navigation. The square in the lower-left of the grid has co-ordinates of 0,0 and square at the top right has co-ordinates of 4,4. An example position might be 0, 0, N, which means the rover is in the bottom left corner and facing North.

In order to control a rover, NASA sends a simple string of letters. The possible letters are 'L', 'R' and 'M'. 'L' and 'R' makes the rover spin 90 degrees left or right respectively, without moving from its current spot.

'M' means move forward one grid point, and maintain the same heading.

Assume that the square directly North from (x, y) is (x, y+1) and the square directly East from (x, y) is (x + 1, y).

The starting position for rover one: 1,2, N

The starting position for rover two: 2,3, E

The following instructions will be sent to each rover:

Rover one: LMLMLMLMM

Rover two: MMRMMRMRRM

Each rovers movement will be executed sequentially, which means that the second rover won't start to move until the first one has finished moving.

Output:

The output for the program should be the final position of each rover.

This is expected to be the following:

Rover one: 1,3, N

Rover two: 4,1, E