

Coding Challenge 1

The application is a simulation of a robot moving on a square table top, of dimensions 5 units x 5 units. There are no other obstructions on the table surface. The robot is free to roam around the surface of the table, but must be prevented from falling to destruction. Any movement that would result in the robot falling from the table must be prevented, however further valid movement commands must still be allowed.

Create a console application that can read in commands of the following form:

PLACE X,Y,F  
MOVE   
LEFT   
RIGHT   
REPORT

* **PLACE** will put the robot on the table in position X,Y and facing NORTH, SOUTH, EAST or WEST.
* The origin (0,0) can be considered to be the SOUTH WEST most corner. It is required that the first command to the robot is a PLACE command, after that, any sequence of commands may be issued, in any order, including another PLACE command.
* The application should discard all commands in the sequence until a valid PLACE command has been executed.
* MOVE will move the robot one unit forward in the direction it is currently facing.
* LEFT and RIGHT will rotate the robot 90 degrees in the specified direction without changing the position of the robot.
* REPORT will announce the X,Y and F of the robot. This can be in any form, but standard output is sufficient. A robot that is not on the table can choose to ignore the MOVE, LEFT, RIGHT and REPORT commands.
* Input can be from a file, or from standard input, as the developer chooses.

Constraints

The robot must not fall off the table during movement. This also includes the initial placement of the robot. Any move that would cause the robot to fall must be ignored.

Example Input and Output

Input:  
PLACE 0,0,NORTH   
MOVE  
REPORT  
  
Output:   
0,1,NORTH

----------------   
Input:  
PLACE 0,0,NORTH   
LEFT  
REPORT  
  
Output:   
0,0,WEST

----------------   
Input:  
PLACE 1,2,EAST   
MOVE  
MOVE  
LEFT  
MOVE  
REPORT  
Output:   
3,3,NORTH

Submission

* GitHub repository or Zip package of source.
* Documentation that includes instructions on how to run the solution as well as tests.
* Provide test data to exercise the application. It is not required to provide any graphical output showing the movement of the robot.
* Provide an object oriented design solution.
* Include enough unit tests.
* Please take as much time as you need to complete the challenge. We would prefer you take the time to complete a solution you are happy with. This challenge will help us to understand more about your abilities and provide a discussion point should you be successful in progressing to the next stage. It is important for us to understand how you approach the problem in code. A clean code, elegant solutions would be preferred.
* The solution can be provided in any language and any paradigm you feel comfortable with. However, we advise you choose a language relevant to the position you are applying for.