



# Europa Exploration Mission

## Introduction

NASA is planning a mission to explore Europa, one of Jupiter's moons, which is believed to have a subsurface ocean that could potentially harbor life. We have promised to deliver the navigation module for it. We need your help in writing this module.

## Part 1: Basic Navigation

A squad of robots are to be landed by NASA on Jupiter's satellite Europa's icy surface. The robot's starting position and location are represented by a grid of squares, which cover Europa's surface, and a compass direction (N, E, S, W).

This plateau, which is curiously rectangular, must be navigated by the robots so that their on-board cameras can get a complete view of the surrounding terrain to send back to Earth. A robot's position and location is represented by a combination of x and y coordinates and a letter representing one of the four cardinal compass points. The plateau is divided up into a grid to simplify navigation. An example position might be 0, 0, N, which means the robot is in the bottom left corner and facing North.

In order to control a robot, NASA sends a simple string of letters. The possible letters are 'L', 'R' and 'M'. 'L' and 'R' makes the robot 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  $(x, y+1)$ .

## INPUT

The first line of input is the upper-right coordinates of the plateau, the lower-left coordinates are assumed to be 0,0.

The rest of the input is information pertaining to the robots that have been deployed. Each robot has two lines of input. The first line gives the robot's position, and the second line is a series of instructions telling the robot how to explore the plateau.

The position is made up of two integers and a letter separated by spaces, corresponding to the x and y coordinates and the robot's orientation.

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

## OUTPUT

The output for each robot should be its final coordinates and heading.

## INPUT AND OUTPUT

Test Input:

5 5

1 2 N

LMLMLMLMM

3 3 E

MMRMMRMRRM

Expected Output:

1 3 N

5 1 E

## Submission

1. Push your code to a GitHub repository and share the link with us. Make sure your repo is not private.
2. Provide clear instructions in README on how to run your code locally and to view the desired results.
3. Feel free to make assumptions for information not provided above. List down any such assumptions clearly in the README.
4. This exercise is part 1. If the submission to part 1 checks out, in part 2, we will build on top of the submission in a live session with one of our engineers over a video call.

## Evaluation Criteria



We recommend you submit something that you are proud of and not something that just works.

1. The problem statement itself isn't hard by any means. This is on purpose because we will evaluate on many criteria other than the correct execution of the code.
2. Correctness i.e. your code is able to return the correct output for all above mentioned scenarios, is a basic requirement.
3. We are looking for a well designed and written code.
4. Feel free to choose any (modern) language you are comfortable in (eg: Python, Java, Javascript/Typescript, C#, C++, Go, Rust etc.).

We look forward to reviewing your submission. May the force be with you!