UCF "Practice" Local Contest — Aug 24, 2013

Dr. Sukthankar's Robot

filename: `robot`

You are taking Dr. Sukthankar's Introduction to Robotics course and are working on your first assignment, to program your new robot to bring Dr. S a cup of coffee. Unfortunately, a fellow student, who decided that he wants your coffee instead, has mis-programmed your robot. Luckily, your robot sends messages to your computer for every move it makes. Using this information, you can deduce where your robot is and direct it to the coffee station and then to Dr. S.

The Problem:

You will be given a listing of the robot's movements on the Cartesian plane. Initially, the robot is located at position (0,0), facing in the direction of the positive x-axis. You will also be given coordinates of both the coffee machine and Dr. S. Your goal will be to first follow the moves the robot has already made, and then give the robot instructions to travel from its current location to the coffee machine and then to Dr. S. The robot understands two types of instructions:

1) Turning (LEFT, RIGHT, UTURN)
2) Moving a specified length in the current direction the robot is facing.

There are further restrictions with respect to what the robot can do:

1) After making the moves following the instructions given in the initial listing (by your mischievous classmate), when moving from the robot's location to the coffee maker, the robot MUST first move in the direction of the x-axis (if necessary), THEN move in the direction of the y-axis (if necessary), to reach the coffee maker. This also applies when moving from the coffee maker to Dr. S.

2) When moving from one location to another, the robot MUST minimize the distance it travels AND the number of commands it takes, given that restriction #1 is satisfied.

The syntax for the instructions given to the robot is as follows:

1) Each command should be on a line by itself.
2) Commands for turning are LEFT, RIGHT and UTURN.
3) The command for moving is MOVE *m*, where *m* represents the number of units to move (in the direction the robot is facing). Note: *m* must be a positive integer.

Note: The left and right turns are always 90 degree turns and the u-turn is 180 degrees. Also, assume that nothing blocks the robot, e.g., the robot can go through the location where Dr. S is to get to the coffee machine (i.e., the robot does not have to go around a block).

The Input:

There will be multiple test cases. The first line of input file contains a positive integer, *n*, representing the number of test cases in the input. The test cases follow. The first line of each test case contains a single positive integer, *k*, representing the number of commands the robot has already executed from its starting position (the origin, facing in the direction of the positive x-axis). The next *k* lines contain the commands the robot has executed. These commands start in column one and there is exactly one space separating the different parts on a line. The list of commands is followed by two ordered pairs on a line separated by spaces, representing the x-y coordinates of the coffee machine and Dr. S, respectively. These four numbers are integers. Assume that the locations of the coffee machine and Dr. S are distinct.

The Output:

At the beginning of each test case, output "`Robot Program #p:`", where *p* is the test case number (starting from 1). For each test case, follow all the commands the robot has already processed and print its location. Then, print (give) the commands necessary to take the robot to the coffee machine and then print (give) the commands necessary to take the robot from the coffee machine to Dr. S, complying to the restrictions in the problem statement. Leave a blank line after the output for each test case. Follow the format illustrated in Sample Output.

(Sample Input/Output on the next page)

Sample Input:

```
3
3
MOVE 100
LEFT
MOVE 50
100 175 200 200
4
UTURN
MOVE 50
LEFT
MOVE 100
-25 50 0 0
1
MOVE 1
2 0 3 0
```

Sample Output:

```
Robot Program #1:
The robot is at (100,50)
MOVE 125
RIGHT
MOVE 100
LEFT
MOVE 25

Robot Program #2:
The robot is at (-50,-100)
LEFT
MOVE 25
LEFT
MOVE 150
RIGHT
MOVE 25
RIGHT
MOVE 50

Robot Program #3:
The robot is at (1,0)
MOVE 1
MOVE 1
```

Notes:
For Sample Input test case 1, a sequence of moves with fewer commands (instructions) will be:
MOVE 125, MOVE 25, RIGHT, MOVE 100. But this sequence does not comply to restriction
(1), thus is not the desired output.