CSCD 300 Homework 3
Solve Maze Puzzle with Recursion

Total 100 points

Turn in: On EWU Canvas, CSCD300→ Assignments→Hw3→Submit.
Please put all your source code(.java) files and output files together into a zip file. Name the zip file with your last name followed by first initial plus hw3.zip. For example, smithjhw3.zip is for John Smith.

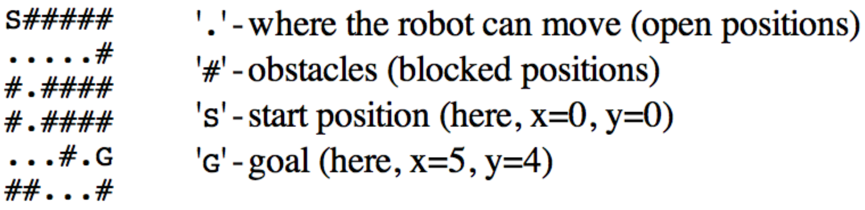
If you forget to include your source code in the zip file, you get a zero credit for this homework. If your code shows a compile-time error, you get a zero credit.

Problems Description

Based on the materials we discussed in classroom, you are required to implement and solve a given maze puzzle using **recursion**. You have to implement **two algorithms**. 1) Only print out the first path from the Start to Goal your program finds. Please write this path along with the puzzle representation into a text file, named **path1.txt**. 2) Print out all possible paths from the Start point to Goal in the puzzle. Please write all paths along with its puzzle representation into another text file, named **pathAll.txt. The format of the output text files are provided in section below.**

Rules:

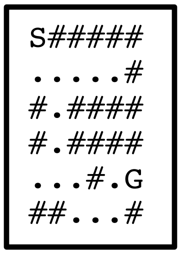
1) You have to use the required input format. That is, the maze puzzle is represented as a matrix of characters as following.



```
S#####
.....#
#.####
#.####
...#.G
##...#
```

'.' - where the robot can move (open positions)
'#' - obstacles (blocked positions)
'S' - start position (here, x=0, y=0)
'G' - goal (here, x=5, y=4)

Your program should read in the maze puzzle from a text file, like this



```
S#####
.....#
#.####
#.####
...#.G
##...#
```

2) When output the first path that your program finds (subproblem one), the output file is formatted like the following. A '+' is used to mark a path from Start to Goal, 'X' shows your program went that cell, but no path is found through that cell.

```
+#####
++XXX#
#+####
#+####
.++#+G
##+++#
```

3) You have to use recursion and implement the pseudo code provided on the website. And you have to extend that idea and print out all paths in the puzzle. If your input text file is the following,

```
S#####
.....#
#.##.#
#.##.#
...#.G
##...#
```

The output file for subproblem 2 would look like below, (output all paths into a file). Different solutions are separated by dashed line(one line or two lines).

```
---------------------------------------
+#####
+++++#
#.##+#
#.##+#
...#+G
##...#
---------------------------------------
---------------------------------------
+#####
++...#
#+##.#
#+##.#
.++#+G
##+++#
---------------------------------------
```

4) Please do not copy other’s code from the Internet. (Please check the syllabus about academic integrity) You have to implement the algorithm idea in the given materials. This is the time that tests your own programming skills.

5) **Note:** the discussed pseudo code just shows general ideas about how this algorithm works. You have to design your own java class and appropriate method interface for your solution. **It is worth mentioning that** your java method interface (input parameters & return types) might be quite different from the pseudo code.

What is provided?

Two input text files are provided for test purpose, **inputPuzzle1** and **inputPuzzle2**.