Assignment 1: Graph Search Problem - Grid World

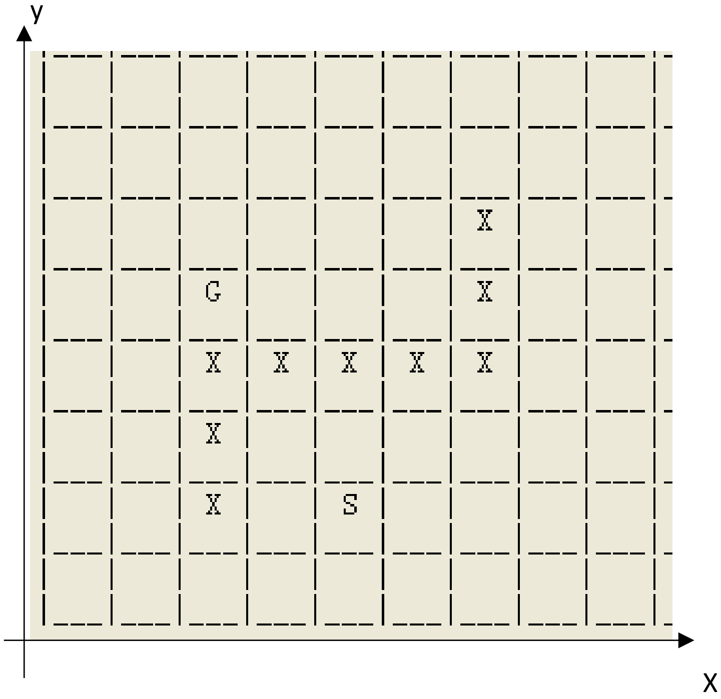
Issue: August 28, Monday
Due: September 10, Sunday (11:59 PM)

1. Objectives:

To gain an in-depth understanding of the 2 graph search algorithms: Breadth-first, Depth-first search. To learn how to develop solutions to tackle the uninformed search agent.

2. Application and Requirements:

1) Suppose we have a grid map, where each cell has its coordinate (x, y), and has at most 4 neighbors, North (x, y+1), West (x-1, y), South (x, y-1), and East (x+1, y). S and G are the starting and goal locations that can be specified through user input. Cells with mark ***X*** are the barriers cells that cannot be accessed.



						X		
		G				X		
		X	X	X	X	X		
		X						
		X		S				

The task is to use breath first search (BFS) and depth first search (DFS) to find a path from **S** to **G**. This path consists of a list of coordinates, each of which is one step moving from the current cell to one of its neighbors (except boundary or a forbidden cell)

2) In order to make the system interactive, your system needs to do:
   a. Allow user to specify the the size of the grid, eg, 8x8, or 8x9, 10x10, etc. (can be specified in a configuration file, *.ini)
   b. Allow user to specify the barrier cells. (can be specified in a configuration file, *.ini)
   c. Allow user to specify the location of S and G, eg. use coordinate (5,2)
   d. Allow use to choose the search methods (BFS or DFS)
   e. Print out the visiting nodes
   f. Print out the final path

3. Other Instructions:

In this assignment the README.txt file will serve as a report which includes the following parts:

1) Overall structure of the code.
2) Instructions on how to compile and run the program.

4. Marking scheme:

- The correctness of the result: 50%
- The user interface: 18%
- The modularity and readability of the code: 12%
- The README report: 20%

5. Have fun!

6. Sample command interface (suppose grid size and barriers have been specified):

```
Please enter the column for the start node: (0-9)
4

Please enter the row for the start node: (0-9)
1

Please enter the column for the goal node: (0-9)
2

Please enter the row for the goal node: (0-9)
4

      0   1   2   3   4   5   6   7   8   9
    _________________________________________
 9  |   |   |   |   |   |   |   |   |   |   |
    _________________________________________
 8  |   |   |   |   |   |   |   |   |   |   |
    _________________________________________
 7  |   |   |   |   |   |   |   |   |   |   |
    _________________________________________
 6  |   |   |   |   |   |   |   |   |   |   |
    _________________________________________
 5  |   |   |   |   |   |   | X |   |   |   |
    _________________________________________
 4  |   |   | G |   |   |   | X |   |   |   |
    _________________________________________
 3  |   |   | X | X | X | X | X |   |   |   |
    _________________________________________
 2  |   |   | X |   |   |   |   |   |   |   |
    _________________________________________
 1  |   |   | X |   | S |   |   |   |   |   |
    _________________________________________
 0  |   |   |   |   |   |   |   |   |   |   |
    _________________________________________
```

```
Breadth-first search:
* Number of nodes searched: 47
* The solution path:
(4,1),(3,1),(3,0),(2,0),(1,0),(1,1),(1,2),(1,3),(1,4)(2,4)

Depth-first search:
* Number of nodes searched: 44
* The solution path:
(4,1),(4,2),(5,2),(6,2),(6,1),(6,0),(7,0),(8,0),(8,1),(8,2),(8,3),(8,4),(8,5),(8,6),(8,7),(8,8),(8,9),(7,9),(6,9),(5,9),
(4,9),(3,9),(2,9),(1,9),(0,9),(0,8),(0,7),(0,6),(0,5),(0,4),(1,4)(2,4)

Would you like to continue? (Y/N)
N
Goodbye!
```