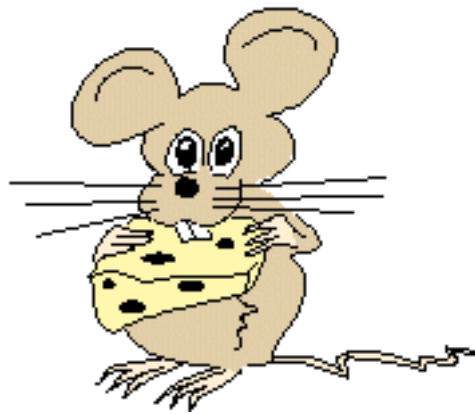CS 211
Data Structures
Due Sept 16

Assignment 2

Each program should be in a separate file.

You can download the code for a stack from the class website (Other Stuff). Use either the ArrayList or Linked implementation so the size of the stack is not fixed.

1.
Write a program to get Algernon and out of the maze

Input to your program, should be a representation of a maze, like the one pictured below. The two numbers above the maze represent   the number of rows and columns. Also, the maze has a wall of 1's surrounding it (as we discussed in class). Input should come from a file called maze.txt. (You may have to review how to read from a text file.)

Notice the staring position is marked by s and the exit by e.

```
10 10
1111111111
1s00100001
1101101101
1000101101
1100001101
1111110001
1111000111
1111010001
11110001e1
1111111111
```

Output is the direct path through the maze, marked by X's.

```
1111111111
1sX01XXXX1
11X11X11X1
10XX1X11X1
110XXX11X1
111111XXX1
111100X111
111101XXX1
11110001e1
1111111111
done
```

Notes:

- You should read your input from a file maze.txt. The input file will contain a single maze. Output is to the screen. Use the above maze.
- You will have to determine the start and end position from the input
- If no path exists, you should print the maze and the message: "Trapped -- No Way out!"
- You may assume all input is correct
- Use a stack and the algorithm from class.

THE SAGA OF JACK STACK AND SUE QUEUE
IS THEIR MARRIAGE DOOMED?

Call it Jack.java

Jack Stack and Sue Queue plan to spend their honeymoon visiting each US state (except Alaska and Hawaii) as well as Washington DC. They plan to start in Maine. And, of course they will use the private jet that they received for a wedding gift form Sue's dad, Lou Queue. After studying their Rand McNally Atlas, Jack and Sue made a table showing which state borders another. For example, if there are just four states A,B,C,D :
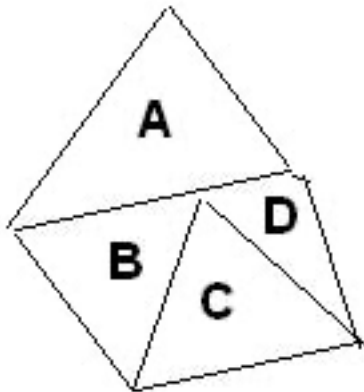
The table or matrix would be:

	A	B	C	D
A	0	1	0	1
B	1	0	1	0
C	0	1	0	1
D	1	0	1	0

This matrix can be interpreted as "A borders on B and D (look for the 1's); B borders on A and C; C borders on B and D; D borders on A and C."

This array is called the Adjacency Matrix for the map. And Jack has a plan for visiting each state:
Jack's plan is:

Start in Maine, visit a state **adjacent** to Maine, say New Hampshire ; now visit a state adjacent to New Hampshire, say Vermont , now visit a state adjacent to Vermont, perhaps NY, etc. If at any time there is no adjacent state to visit, Jack backtracks to the last visited state S and visits an unvisited state adjacent to S.

For example suppose that, you have the following "map."

Using Jack's plan and starting at A, you would proceed as follows:

Visit A, visit B, visit C, visit H, visit D, *backtrack to H, backtrack to C, backtrack to B, backtrack to A*, visit E, visit F, visit G,

Thus the traversal is : ABCHDEFG

You can use a stack to keep track of the states that Jack visits. This will facilitate backtracking.

Here is a more formal description of Jack's plan:

<u>Jack's plan is given by the following algorithm:</u>

```
Visit Maine
Mark Maine as visited
Push Maine onto a stack
While the stack is not empty do
 {
        While there is an unvisited state W adjacent to top of stack
           {
               visit w,
               mark w as visited,
                push w onto the stack
           }
        Pop the stack
 }
```

Typical output:

The trip is as follows:
  Maine
  Vermont,
  New York etc

You can download the adjacency matrix for the states and DC [here](http://web.stonehill.edu/compsci/CS211/Assignments/usmap.txt).

Or if the link does not work

http://web.stonehill.edu/compsci/CS211/Assignments/usmap.txt