CIS 343 – Structure of Programming Languages
Winter 2016, Section 1, 1/28/2016

Programming Assignment #1
Minesweeper in C
Due Date: Monday, February 15, 2016

Objectives

- Basic input and output
- Two-dimensional arrays
- Structures
- Modularizing program with functions

Problem Specification

Write a C program to play a game of minesweeper. You can play this game online here: http://minesweeperonline.com/

Design Requirements

For this project, you are supplied with a C source file named `minesweeper.c` that contains the functions listed below. The `main()` and `displayMenu()` functions are already completed for you and should NOT be changed. Your task is to implement the remaining functions.

- `int main()`
- `void displayMenu()`
- `void initBoard(int size, Cell board[][size])`
- `void placeMinesOnBoard(int size, Cell board[][size], int nbrMines)`
- `void fillInMineCountForNonMineCells(int size, Cell board[][size])`
- `int nbrOfMines(int size, Cell board[][size])`
- `int getNbrNeighborMines(int row, int col, int size, Cell board[][size])`
- `void displayBoard(int size, Cell board[][size], bool displayMines)`
- `int getBoardSize()`
- `int getPercentMines()`
- `Status selectCell(int row, int col, int size, Cell board[][size])`
- `int nbrVisibleCells(int size, Cell board[][size])`
- `void setImmediateNeighborCellsVisible(int row, int col, int size, Cell board[][size])`
- `void setAllNeighborCellsVisible(int row, int col, int size, Cell board[][size])`

You can choose to implement either a **simpler** or a **more realistic** implementation of minesweeper.

In a simpler implementation of minesweeper, you reveal only cells that are immediate neighbors (up to eight possible cells surrounding a selected cell) when a user selects a cell that has a zero mine count. If you choose to do this, then implement the function `setImmediateNeighborCellsVisible()` only and ignore the function `setAllNeighborCellsVisible()`.

To implement a more realistic version of minesweeper, you must implement the function `setAllNeighborCellsVisible()` and ignore the function `setImmediateNeighborCellsVisible()`.

If you implement the simpler version of minesweeper, the maximum possible score is only 90 out of total 100 points for the project.

Sample Program Execution

```
$ ./minesweeper
!!!!!WELCOME TO THE MINESWEEPER GAME!!!!!

Enter the board size (5 .. 15): 10
Enter the percentage of mines on the board (10 .. 70): 25

     1  2  3  4  5  6  7  8  9 10
  1  ?  ?  ?  ?  ?  ?  ?  ?  ?  ?
  2  ?  ?  ?  ?  ?  ?  ?  ?  ?  ?
  3  ?  ?  ?  ?  ?  ?  ?  ?  ?  ?
  4  ?  ?  ?  ?  ?  ?  ?  ?  ?  ?
  5  ?  ?  ?  ?  ?  ?  ?  ?  ?  ?
  6  ?  ?  ?  ?  ?  ?  ?  ?  ?  ?
  7  ?  ?  ?  ?  ?  ?  ?  ?  ?  ?
  8  ?  ?  ?  ?  ?  ?  ?  ?  ?  ?
  9  ?  ?  ?  ?  ?  ?  ?  ?  ?  ?
 10  ?  ?  ?  ?  ?  ?  ?  ?  ?  ?

Enter command (m/M for command menu): m
List of available commands:
   Show Mines: s/S
   Hide Mines: h/H
   Select Cell: c/C
   Display Board: b/B
   Display Menu: m/M
   Quit: q/Q

Enter command (m/M for command menu): s

     1  2  3  4  5  6  7  8  9 10
  1  *  ?  ?  ?  ?  ?  ?  ?  *  ?
  2  ?  ?  ?  ?  ?  ?  ?  *  ?  ?
  3  *  ?  ?  ?  ?  ?  *  *  ?  ?
  4  ?  *  ?  ?  ?  ?  ?  ?  *  ?
  5  ?  *  ?  ?  ?  *  *  *  ?  ?
  6  ?  ?  *  ?  ?  *  *  ?  ?  *
  7  ?  ?  ?  *  *  *  ?  *  *  ?
  8  ?  ?  ?  ?  *  ?  ?  ?  ?  *
  9  ?  ?  ?  ?  ?  ?  ?  ?  ?  ?
 10  ?  *  ?  ?  ?  ?  *  ?  ?  ?

Enter command (m/M for command menu): c
Enter row and column of cell: 1 5

     1  2  3  4  5  6  7  8  9 10
  1  *  1  0  0  0  0  1  ?  *  ?
  2  ?  2  0  0  0  1  3  *  ?  ?
  3  *  2  1  0  0  1  *  *  ?  ?
  4  ?  *  2  0  1  3  ?  ?  *  ?
  5  ?  *  3  1  2  *  *  *  ?  ?
  6  ?  ?  *  ?  ?  *  *  ?  ?  *
  7  ?  ?  ?  *  *  *  ?  *  *  ?
  8  ?  ?  ?  ?  *  ?  ?  ?  ?  *
  9  ?  ?  ?  ?  ?  ?  ?  ?  ?  ?
 10  ?  *  ?  ?  ?  ?  *  ?  ?  ?

Enter command (m/M for command menu): h

     1  2  3  4  5  6  7  8  9 10
  1  ?  1  0  0  0  0  1  ?  ?  ?
  2  ?  2  0  0  0  1  3  ?  ?  ?
  3  ?  2  1  0  0  1  ?  ?  ?  ?
  4  ?  ?  2  0  1  3  ?  ?  ?  ?
  5  ?  ?  3  1  2  ?  ?  ?  ?  ?
  6  ?  ?  ?  ?  ?  ?  ?  ?  ?  ?
  7  ?  ?  ?  ?  ?  ?  ?  ?  ?  ?
  8  ?  ?  ?  ?  ?  ?  ?  ?  ?  ?
  9  ?  ?  ?  ?  ?  ?  ?  ?  ?  ?
 10  ?  ?  ?  ?  ?  ?  ?  ?  ?  ?

Enter command (m/M for command menu): c
Enter row and column of cell: 10 7

     1  2  3  4  5  6  7  8  9 10
  1  ?  1  0  0  0  0  1  ?  ?  ?
  2  ?  2  0  0  0  1  3  ?  ?  ?
  3  ?  2  1  0  0  1  ?  ?  ?  ?
  4  ?  ?  2  0  1  3  ?  ?  ?  ?
  5  ?  ?  3  1  2  ?  ?  ?  ?  ?
  6  ?  ?  ?  ?  ?  ?  ?  ?  ?  ?
  7  ?  ?  ?  ?  ?  ?  ?  ?  ?  ?
  8  ?  ?  ?  ?  ?  ?  ?  ?  ?  ?
  9  ?  ?  ?  ?  ?  ?  ?  ?  ?  ?
 10  ?  ?  ?  ?  ?  ?  *  ?  ?  ?

Oops. Sorry, you landed on a mine. Bye
```

Deliverables

1. Upload `minesweeper.c` file on Blackboard by midnight on due date.
2. I will use the submission date/time on Blackboard as your official submission date/time.
3. It is your responsibility to make sure the submission on Blackboard went through successfully.
4. Because of possible portability issues, make sure your program compiles and runs on EOS machines before submitting any source file(s) on Blackboard. I will compile, run, and test your program on EOS when grading.
5. Use the following commands on EOS to compile and run your program:
   ```
   $ gcc –Wall –std=c99 –o minesweeper minesweeper.c
   $ ./minesweeper
   ```
6. Late penalty (10% per day) applies after Monday, February 15th.