ECCS 1611 – Programming 1 – Fall Semester 2022
MP3 – Tic Tac D'Oh! – Due Thursday 10 November 2022

Updated 3 November 2022

Write a program that will allow two people to play the game of tic-tac-toe. The program must ask for moves alternately from each player, one of whom **always** plays the "crosses" ('x') while the other **always** plays the "circles" ('o') regardless of the number of games played. The program displays the gameboard positions, which are similar to the numeric keypad layout, as follows:

```
 7 | 8 | 9
---+---+---
 4 | 5 | 6
---+---+---
 1 | 2 | 3
```

The players enter their moves by entering the position number they wish to mark. After each move, the program displays the changed board. A sample board configuration is as follows:

```
 X | 8 | 9
---+---+---
 O | O | 6
---+---+---
 X | 2 | 3
```

At the end of each game, your program must indicate who won (by user name) or that it was a tied game, after which the players are to be allowed the choice to play again or quit. The players alternate who starts first in each game; additionally, the program is to keep track of the number of wins for each player, plus the number of ties, printing this information out at the end of each game.

Your program is required to use functions and one-dimensional arrays. You are allowed to use global variables only in the case of symbolic constants. You cannot have `cout` statements in your functions unless:

- the purpose of the function is to display output
- the purpose of the function is to obtain input, for which a prompt is somehow used
- the purpose of the function is to inform the user (e.g., error message, whose turn it is, etc.)

At the start, your program must ask for the names of the two players; these names are to be used when providing information to the players (e.g., whose turn it is, who won, etc.).

IMPORTANT NOTE: As this is a game with a fixed board size, you may use a global symbolic constant to represent the size of the array containing the gameboard and thereby dispense with passing the size of the array with your function calls. Note that this is an exception to the general rule for passing arrays to functions.

The following are the functions that you are required to properly implement for this assignment (you may implement others as well):

- **`int getPlayerInput(string playerName)`**
  Returns a value between 1 and 9, inclusive, indicating the square where the current player wants to place his/her mark on the board. The input routine is to use the name of the player whose turn it is and is to check for correct input. You may assume that the input is an integer, but the value entered might be where a mark on the board already exists or is out of the expected range of legal values.

- **`bool isLegalMove(char board[], int location)`**
  Returns true if the indicated location on the board yet not yet been played, false otherwise.

- **`void placeMarkOnBoard(char board[], char playerMark, int location)`**
  Places the indicated mark at the specified board location; routine assumes that this is a legal placement.

(function list continues on next page)

- **`void clearBoard(char board[])`**
  Restores the tic-tac-toe board to its original (i.e., with no crosses or circles being marked) configuration.

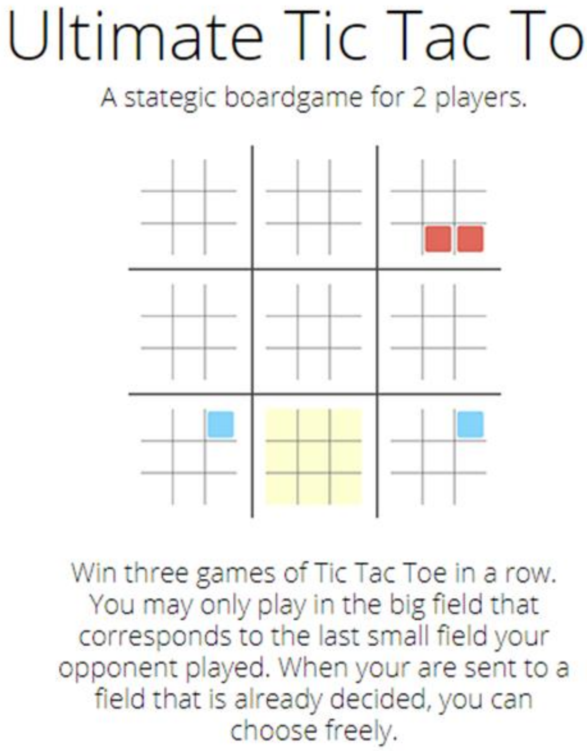
- **`bool hasThreeInRow(char board[], char playerMark)`**
  Returns true if, for the specified mark, the board has the equivalent of three of those marks in a row, either vertically, horizontally, or diagonally.

- **`void displayBoard(char board[])`**
  Prints out the current board; the board display must be as shown above.

- **`void displayGameStats(int ties, int player1Score, int player2Score)`**
  Prints out the “stats” for all the games: number of games tied, number of games player 1 won, and number of games player 2 won.

This assignment will be evaluated only during lab on 10 November.

In case you’re wondering why the board is not being represented as a global variable, consider the following “next step” for making this a more interesting program:



Ultimate Tic Tac Toe

A stategic boardgame for 2 players.

Win three games of Tic Tac Toe in a row. You may only play in the big field that corresponds to the last small field your opponent played. When your are sent to a field that is already decided, you can choose freely.

Please note that we are **NOT** implementing Ultimate Tic Tac Toe at this time!