**ECCS 1611 – Programming 1**

**Fall Semester 2020**

**MP3 – Tic Tac D’Oh! – Due by Thursday 15 October 2020 @ 11:59 pm**

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 telephone keypad layout, as follows:

1 | 2 | 3

---+---+---

4 | 5 | 6

---+---+---

7 | 8 | 9

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 | 2 | 3

---+---+---

O | O | 6

---+---+---

X | 8 | 9

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 two-dimensional arrays. You are allowed to use global variables only in the case of symbolic constants (e.g., a variable for the number of rows and columns). You **cannot** have cout statements in your functions unless the entire function is designed to display the board or a “chunk” of information (for example: displayBoard and displayGameStats functions) or you are getting information from the user (for example: getPlayerInput). 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 2D array (rows and columns is both 3) or the total number of locations within the 2D array (i.e., 3 \* 3 = 9).

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. However, you do need to verify the input is within the correct range (i.e., it is a number on the gameboard shown above). You do not need to verify if the location is a legal move yet.

* **bool isLegalMove(char board[SIZE][SIZE], int location)**

Returns true if the indicated location on the board yet not yet been played, false otherwise.

* **void placeMarkOnBoard(char board[SIZE][SIZE], 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[SIZE][SIZE])**

Restores the tic-tac-toe board to its original (i.e., with no crosses or circles being marked) configuration.

* **bool hasThreeInRow(char board[SIZE][SIZE], 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 locationToBoardLocation(int location, int &row, int &column)**

Translates from the location entered by the user to the row and column format. (This will allow you to index into the 2D array correct). Remember user enters a value from 1-9.

User View of the Board with Numbers:

1 | 2 | 3

---+---+---

4 | 5 | 6

---+---+---

7 | 8 | 9

Locations within the 2D array:

|  |  |  |
| --- | --- | --- |
| (0, 0) | (0, 1) | (0, 2) |
| (1, 0) | (1, 1) | (1, 2) |
| (2, 0) | (2, 1) | (2, 2) |

So if the location being passed in as a parameter is 6, your function should set the variable row equal to 1 and column equal to 2. Please see the following chart for the complete list:

|  |  |  |
| --- | --- | --- |
| Location | Row | Column |
| 1 | 0 | 0 |
| 2 | 0 | 1 |
| 3 | 0 | 2 |
| 4 | 1 | 0 |
| 5 | 1 | 1 |
| 6 | 1 | 2 |
| 7 | 2 | 0 |
| 8 | 2 | 1 |
| 9 | 2 | 2 |

* **void displayBoard(char board[SIZE][SIZE])**

Prints out the current board; the board display must be as shown above. (This function is allowed to have cout statements).

* **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 function is allowed to have cout statements). [This function can be modified to have two additional string parameters for the player names. Therefore it would be: **void displayGameStats(int ties, int player1Score, int player2Score, string player1Name, string player2Name)**].

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:



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