Lab 7: Tic-Tac-Toe

You are going to use a 2D array to create a program that allows two users to play tic-tac-toe. DO NOT attempt to write the whole program in one shot. Steps 1-3 are specifically focused on making sure that you understand how to use 2D arrays and **required** for the lab; steps 5-7 are **optional**.

1. Create a ***2D character array*** in your main function and use ***nested for loops*** to fill the array with the letter *‘e’* to represent empty spaces.
2. Create a function to print the board on the screen using a nested for loop. The function header is:   
   void printBoard (char board [][3])
3. Create a function that checks whether a particular space has already been filled. If the space is filled it returns a boolean value of *true*, otherwise *false*. (***No loops required***.) Here is the function header:   
   bool isFilled (char board [][3], int row, int col)
4. Write a function to prompt the user for their next move. The function should validate that the move is legal and then update the board to reflect the move (hint: arrays are always passed by reference). A move is valid if the values of row and col are both numbers between 0 and 2 and the space is not already filled (hint: you have a function to do part of this). The parameter *player* will be ‘X’ if the function is to get X’s move and ‘O’ if it is to get O’s move. The function header is:   
   void move (char board [][3], char player)
5. **Demo your program for me … you’ll need to have main() modify the board and call your functions for testing purposes. Don’t move on until you get credit for the required part of the lab!**
6. Create a function to detect whether it is game over: either someone has won or a stalemate has been reached. If you’re short on time, you could just ask the user. If you have enough time, test to see whether any of the rows, columns or diagonals contains all Xs or all Os and print out a congratulatory message. Return *true* if the function finds a winner or if the board is completely full and *false* otherwise. The function header is:  
   bool isGameOver(char board[][3])
7. Put all the function calls together in main ( ) and alternate getting a move from player X and player O until someone wins.

