CS-141 Introduction to Programming
TicTacToe

Goals:

1. Practice using 2D arrays.
2. Practice working with code composed of multiple methods.

You are to write a program that plays TicTacToe where the board is stored as a 2D array of Strings. The code should be well structured using methods. Each method should be fairly short (e.g. fit easily on one screen).

Suggestion: Begin by implementing initBoard() and printBoard() methods. You can test that they work by calling them from main (first, comment out playGame()) . Then implement userTurn(), which prompts the user for the row and column of their move, and sets the corresponding board position to an "X". Test it by calling it from main() and printing out the resulting board.

Sample Structure

You may start with the code below. However, feel free to change it if you want.

```
/**
 * A TicTacToe Program.
 * @author
 */

import java.util.Scanner;
public class TicTacToe {

  public static String[][] board;
  public static String currentPlayer= "X";
  public static Scanner reader = new Scanner(System.in);

  public static void main(String[] args) {
    playGame();
  }

  /** Plays the game of TicTacToe.  Stops when either 1) a player
    * wins or 2) no spaces are left on the board.
    */
  public static void playGame() {
    initPlayer();  // pick starting player
    initBoard();   // initialize the board
```

```
  int moves = 0;
  boolean hasWon = false;
  while (moves < 9  && !hasWon ) {
    if (currentPlayer.equals("X")) userTurn();
    else computerTurn();
    printBoard();
    hasWon = checkWin();   // has a player won?
    swapPlayer();          // swap the currentPlayer
    moves++;
  }
  if (!hasWon) System.out.println("No one won.");
}

/* Create a board and initialize the values to all blanks.
 * Print the board.
 * Note, each board position is either:
 *   "X" - this is the user
 *   "O" - this is the computer
 *   " " - a space indicates the position is empty
 */
public static void initBoard() {
  // your code goes here
}

/** Swap the current player.  That is, if the current
  * player is "X", it is swapped to be "O", and vice versa.
  */
public static void swapPlayer() {
  if (currentPlayer.equals("X")) currentPlayer = "O";
  else currentPlayer = "X";
}

/** Randomly pick which player (user or computer) goes first.
  * Set the currentPlayer to the first player.
  * It is assumed that the user is always "X" and the computer
  * is always "O".
  */
public static void initPlayer() {
  System.out.println("You are X and the computer is O.");
  if (Math.random() < .5) {
    currentPlayer = "X";
    System.out.println("You are the first player.");
  }
  else {
    currentPlayer = "O";
```

```
      System.out.println("The computer is the first player.");
    }
  }

  /** Choose which position the computer will play.
    * Set that position of the board to "O"
    */
  public static void computerTurn() {
    // your code goes here
  }

  /** Have the user choose which position to play.
    * Set that position of the board to "X"
    */
  public static void userTurn() {
    // your code goes here
  }

  /** Checks to see if the currentPlayer has won.
    * @return true if the currentPlayer has won.
    */
  public static boolean checkWin() {
    // your code goes here
    return false;  // this is a placeholder so code compiles.
  }

  /** Print the board.
    */
  public static void printBoard() {
    // your code goes here
  }
}
```

Sample Output

The output of your code should be look similar to the following:

```
> run TicTacToe
You are X and the computer is O.
You are the first player.
   1   2   3
 1   |   |
   ---------
 2   |   |
   ---------
```

```
 3   |   |
It is your turn.
Enter the row (1-3) and column (1-3) of your move:
 [DrJava Input Box]
   1   2   3
 1   |   |
   ---------
 2   | X |
   ---------
 3   |   |
It is the computer's turn:
   1   2   3
 1 O |   |
   ---------
 2   | X |
   ---------
 3   |   |
It is your turn.
Enter the row (1-3) and column (1-3) of your move:
 [DrJava Input Box]
   1   2   3
 1 O |   | X
   ---------
 2   | X |
   ---------
 3   |   |
It is the computer's turn:
   1   2   3
 1 O | O | X
   ---------
 2   | X |
   ---------
 3   |   |
It is your turn.
Enter the row (1-3) and column (1-3) of your move:
 [DrJava Input Box]
   1   2   3
 1 O | O | X
   ---------
 2   | X |
   ---------
 3 X |   |
Congratulations - you have won!
```