

Lab 4: Loops and Arrays

Setup

1. Merge your lab repository as described in Lab2.pdf.
2. You should see the lab4 folder and TicTacToe.java inside this folder after the merge operation.
3. TicTacToe.java is a partial implementation of Tic-tac-toe¹ game. We will be implementing the command line version of this two players turn based game incrementally in the following exercises.
4. First compile and run the existing code.

Exercise 1 : Use a loop to allow players to fill the board

1. Set loop condition so that each player can play while there is still empty place on the board.
2. At every iteration
 - a) request coordinates from the current player.
 - b) If the given location is already occupied request another coordinate from the same user.
 - c) If the location is out of range such as row = 4 col = 5 request another coordinate from the same user.
 - d) If the given location is empty (valid move)
 - i. put the player's symbol to the location and
 - ii. Print the board
 - iii. Set the current player to the other player
 - iv. If all the coordinates are occupied, stop the loop

Exercise 2 : Check win after each move

1. Write a method named checkboard which accept board as the parameter. If the board contains a winning state method will return true otherwise method will return false.
2. Call this method after each valid move from the main
3. If the method returns true, stop the loop and print the name of the winner
4. If game ends without a winner print that games ended with a draw

¹ <https://en.wikipedia.org/wiki/Tic-tac-toe>

NOTE: Your lab will **not be graded** if

- Your account name does not have the format described in lab1.pdf
- Your repository name is not lab
- Your files have compilation errors
- You haven't complete the steps described in exercises
- Your added/modified files are not submitted to Bitbucket.
 - You have to add commit and push files as described in lab1.pdf