CS260: GUI Assignment

Due: Sun, 12 Dec at 5pm.

In this assignment, we are going to solve two problems by using the knowledge of stack, set/file, and Exception. Study the following problem descriptions and prepare your solutions for the two questions. Submit the solutions on the Blackboard website by Sunday, 12 December 2021.

For each program add sufficient java doc comments to explain the logic of the codes throughout the program. Use exception handler whenever appropriate. In addition, add screenshots that will show the output of each of the programs.

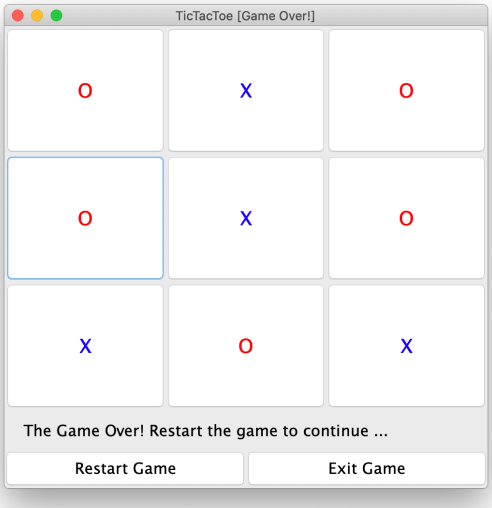
Question : TicTacToe GUI [Board.java, Player.java, CompPlayer.java, and TicTacToeApp.java]

You are familiar with the program requirement of this game. You can use as much content as you prefer from your first lab submission if needed. Let us review some of them in the following:

The TicTacToeApp.java program will run the game, using a Board object and a computer player object. This java file will also be responsible for drawing the TicTacToe GUI so that the user can play the game by using GUI.

The students are given a template file. The template file creates a GUI with various drawing objects and have basic input processing and input validation in place. Below, we see the output generated when the template file is run. The player can input their selection by using the buttons as shown below. The selected button can't be reselected, therefore, it provides basic input validation. When 9 moves are complete the Game Over message is displayed. The Restart Game button resets the game and the button texts will be given a default labelling.

As you remember, the Board class will hold the board positions in the game. The board will consist of a 3x3 two dimensional array grid or you can incorporate other data structure. The constructor of the Board class will create and initialize the data structure.



- Once run, the will display game board (as shown above). The player can only choose the button that is not colored. The number values indicate that these positions are empty and can be selected by the player. H stands for the human player move and C stands for the computer player move.

The Player class is comprised of one integer variable. The integer variable denotes the cell number that has been selected.

TicTacToe class will allow the user to select the available cell number when it is the human player's turn.

The CompPlayer will extend the Player class and automatically generate the next move. The next move will be a cell number. The class will examine the content of the Board and determine the cell number that will increase the winning chance of the CompPlayer. The game then will use the returned cell number to display C on the selected cell on the GUI window.

CompPlayer should use the Board object in order to access the Board data structure and calculate the next position for the computer move.

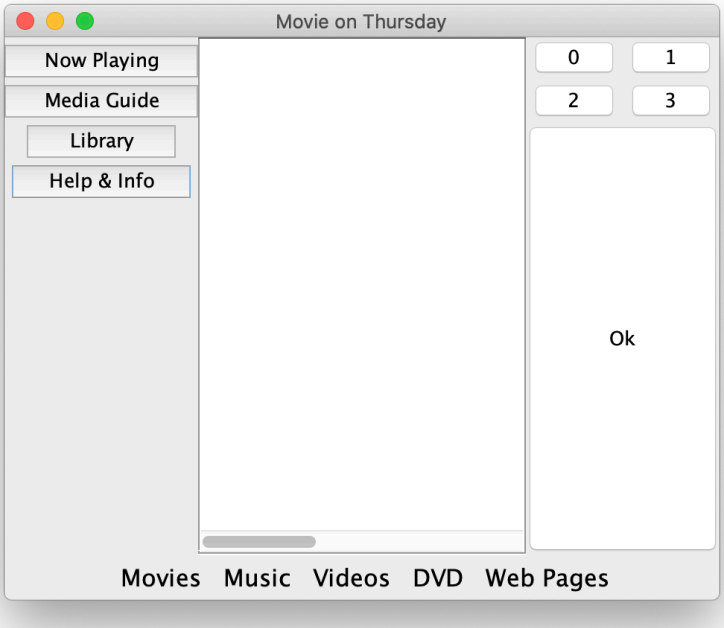
The initial move can be randomized (when there are only one or no move on the Board). The CompPlayer will return the first position that increases the winning chance of the CompPlayer. Similarly, the CompPlayer will select a position and ensure that it is not loosing the game. Likewise, the CompPlayer will prioritize winning when selecting the next move. When there is a winning chance, it should select the winning move instead of defending the game.

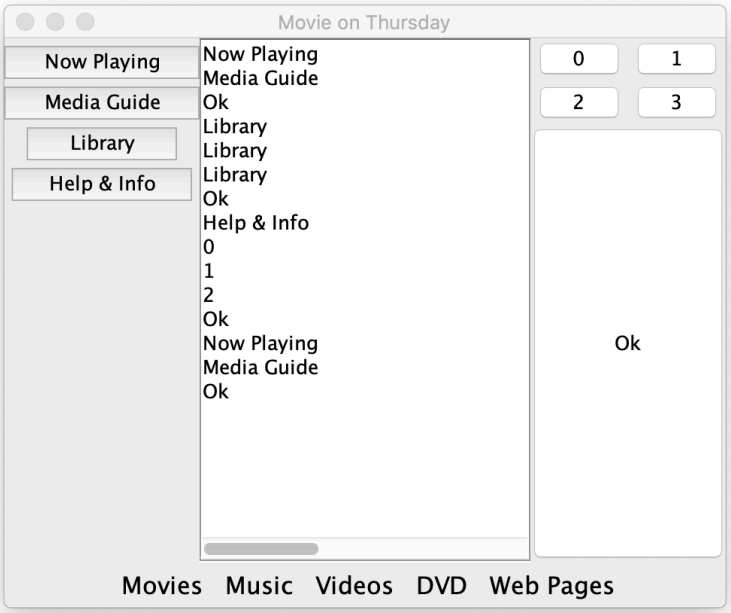
After processing each move, the game should check the Board and determine win, loss, or tie status and update the GUI accordingly. When the game reaches win, loss, or tie condition, the user will no longer be able to select any cell anymore until the game is restarted. The players will take turns.

TickTackToe GUI | Total points: 30
[5]: The player/computer moves are stored in the board object
[10] The computer generates move (win first, defend position second)
[5] The moves are displayed on the GUI
[10] The game can determine win, loss, or draw status of the game
[5] When game win/loss/draw happens, the user can't select a cell until the game restarts

Question : Designing a GUI [GuiProgram.java]

In this program the student is tasked to draw the following GUI by using various layout manager. Ensure that when your program is run it produces the following GUI. Please note, you can use preferredSize method to customize the size of the JPanel, or JButtons. Lastly, create ActionListener for the command buttons, so that when buttons are pressed, the text of the buttons will be appended in the JTextArea. The output of such interaction is shown in the following figure:





Designing UI | Total points: 5
[3] The program was able to draw the given GUI correctly. [2] When the buttons are clicked the text of the button was appended in the JTextArea object as shown in the figure

Criterion	Details	Deductions
Classes	At minimum, the students need to submit the following java files Board.java, TicTacToeApp.java, Player.java, CompPlayer.java, GuiProgram.java	-10 x: The required Java files not submitted -5 x: The program does not provide the desired output -3 x: screenshot of the program run was not attached
Code quality	Identifier names, class names, proper use of public/private, ample comments in main, etc.	-15: incoherent, inconsistent coding style -10: comments were not used throughout the code - 5 x: exceptions were not handled throughout the code
Test	Note, whether the program compile and run	-30: The portion of the code of the assignment is a copy -10 x: The program does not compile