Rafael Villalobos

CPSC 60000

March 13, 2021

Checkers Final Project

For the final project I decided to create a simple checkers game. The choice was to implement the game with the singleton, command, strategy, and singleton design patterns.

The singleton pattern was used with the piece object. I created the pieces by calling a public method which then called a private method within the piece called createPiece() . This allowed only one instance of each piece to be created.

The command design pattern was used with the button to create a new game. It took the button class provided by javafx to create a new button that would be used to reset the game.

The strategy pattern was used by implementing an enum for MoveType and DirectionType objects. This allowed for switching between blue and green pieces including the direction they would be able to move.

The template pattern was inherently used when creating the Checkers game from an application class provided by javafx. In addition, this pattern was utilized when constructing the layout of the checkers board. I used a rectangle class and modified it to print out the spaces and pieces overlayed on top of the spaces.

I did not create any automated tests, but I did have some unit tests to verify working portions of the code. Initially, I tested if the board and pieces would render. After that was working, I tested if pieces were able to be moved, if they could jump, and if they would be removed after being jumped only by opposing pieces. I verified that they would change color when being kinged and if they could move in any direction after being kinged. I tested if they illegal moves were allowed and if the game may be redrawn.

This project allowed me to experience and learn more about utilizing javafx for GUI implementation, but I was very unprepared for the amount of work it would take to 1) create the window and render it, and 2) merge the rules of the game and present that onto the board. I know this project is very simplistic and I have a host of bugs and features that would need to be resolved. For example, I need to launch the game suing the launcher since it would give me issues otherwise. I was not able to implement the game realizing which player’s turn was in play and it was something I plan on working out. The pieces would be lost and out of bounds errors would occur when moving a piece outside of the scope of the window. I needed to implement borders to rectify this issue.

Overall I’m glad I tried this and I hope I continue to using commonly used patterns in my work, but I will probably try a smaller scoped project next time. I did learn to create a window and utilize some of the classes inherent with javafx package, which I think is great, but I need to continue practicing.