Fred Kelemen 2715308

Connect 4

Spring Semester 2020

Course CIS 434

Cleveland State University

**Abstract**

I proposed a game of connect 4. Connect 4 is a simple zero-sum game like tic-tac-toe or chess, though connect 4 is considerably more difficult than tic-tac-toe and much easier than chess. My connect 4 game will implement the game algorithm and show the board state in a simple java display.

**Introduction & Background**

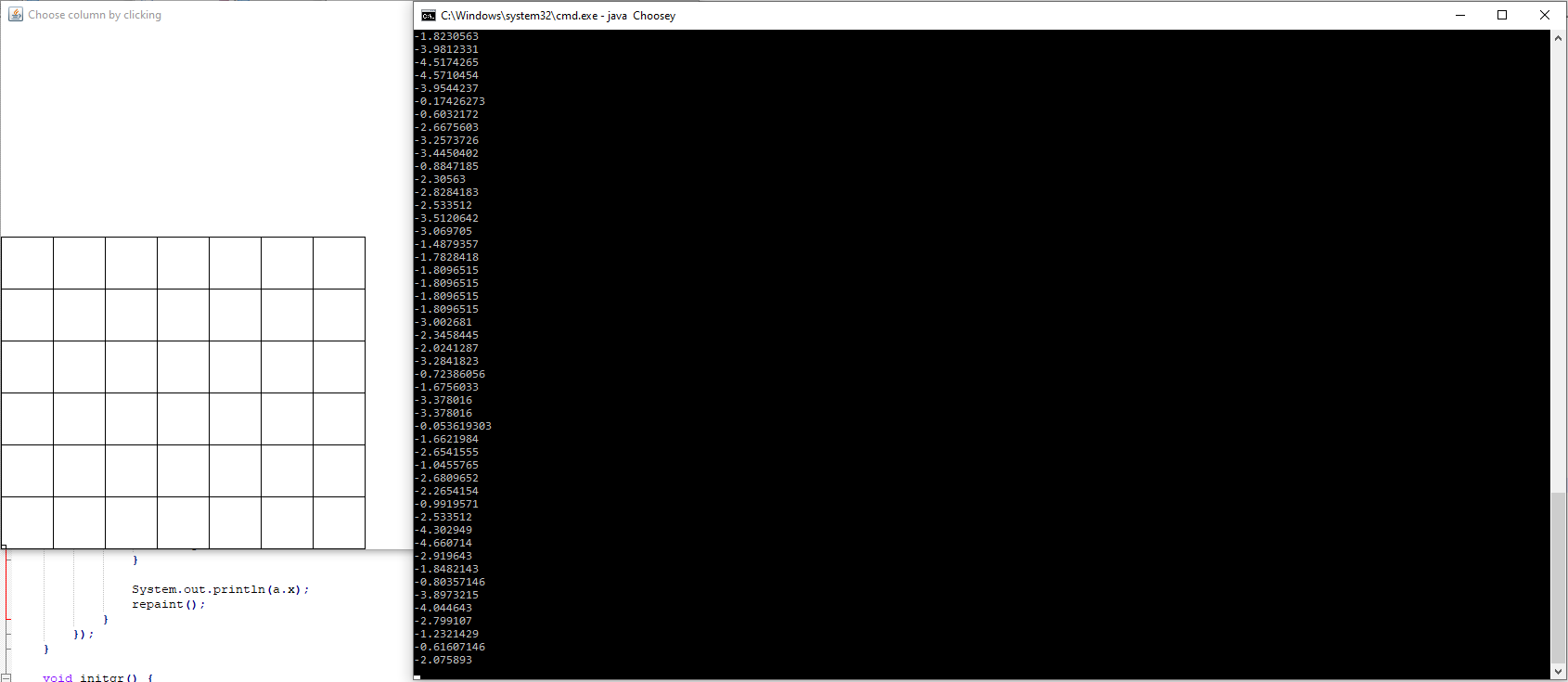
I am a Computer Science student. I have taken several coding courses which gives me the experience necessary to create the game. The game itself was a simple matter, I needed to write win conditions and a way to play a piece. Writing the code for the user interface was a bit more difficult, I wrote the code using squares.java and DefPoly.java to create the interface and take input, while this combination of codes worked, I was unable to then combine these with the code I wrote to run the game.

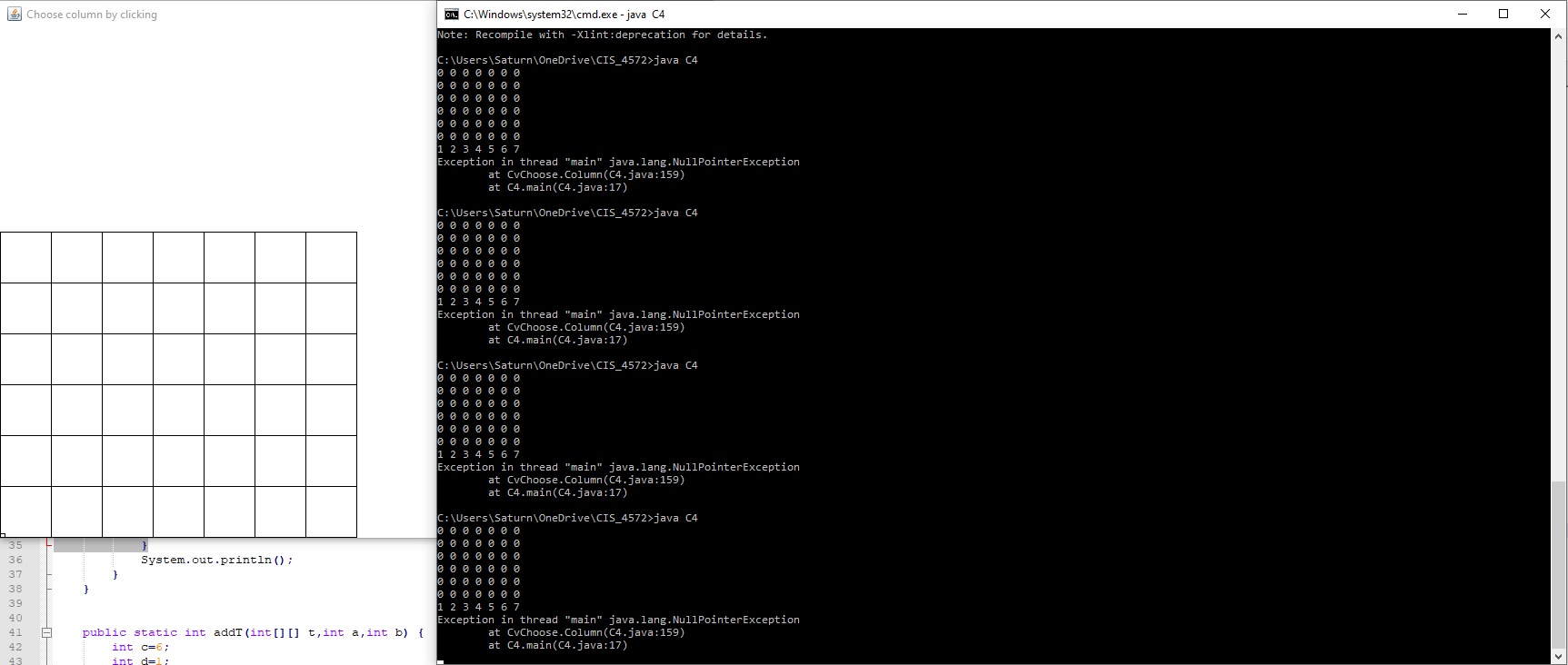
**Project Objectives**

1. Let the user easily add pieces.
2. Let the user easily determine game state.
3. Create a visually pleasing UI.

**Methodology**

The first thing I needed to do was write java code that runs the game. This code had to set win conditions, in connect 4 the win conditions are playing four contiguous pieces in a row, column or diagonal. The game should allow for any of the win conditions and should declare a winner and end when any one of them is met. It took me only a few hours to get the game working, I wrote the code to check every spot, from that spot it would check each of the three possible win conditions to see if a player wins. This method is highly inefficient but as the board only has 42 spots, I determined it would be unnoticeable to the player. To test the game code, I set it up to take typed inputs, after I was confident the game could handle any board state I deleted the code that takes typed input.

The next task was to create the user interface. In my game of connect four everything was to take place in a java window. To do this the board needed to be drawn, ideally the board would look nice and not just be a matrix of squares. After the board was drawn, the board had to take inputs based on the users click, placing the new pieces in the desired row without redrawing the entire board. To create the UI, I used squares.java to create a 6\*7 matrix of squares which would be the board. I then used parts of DefPoly.java, specifically the mouse listener part, I used this to take input. I mixed this with the UI I made, and the codes didn’t interfere with each other so I had code that would take input and draw a matrix of squares.

When the code for the game and the UI were done, I needed the two to be put together. This was a rather straight forward process I had the connect 4 code call to the UI code. I set it up so that when input was given in the UI code the input would be set to the corresponding column. The UI code would then pass back the column value to the connect 4 code, in which I had a matrix to define the board state. This is where I was unable to get the code to work, the UI was taking inputs and the connect 4 was advancing through it’s while loop but the two weren’t working together like they should have been. Had I gotten this to work I would have set it up so that the connect 4 code would pass the board matrix to the UI code where the UI code would fill in the square taken with the current player’s color.

**Project Timeline**

The first week will be used to setup the game, writing code to allow inputs and set win conditions. The following three weeks, I will work on the UI, making it so that it takes input and doesn’t redraw the board. After that I will spend any remaining time on beautifying and debugging my code, if there’s time after this I would also like to make a simple opponent to play against.

**Conclusion**

I intended to make a connect 4 game. I wanted to make the game so that it follows the rules of connect 4. I meant to have a visually pleasing user interface on which the game would be played. This did not happen as I meant it to. I managed to write the code to play connect 4, this was quite easy. After I got the game working, I used squares.java and DefPoly.java in an attempt to make a matrix of squares which would take mouse input for user’s play. This combination of codes appeared to work well, but I couldn’t get them to work with my connect 4 code. It appears that the input is taken but the connect 4 code doesn’t accept it and goes into an infinite loop. Because I spent so much time attempting to fix this I failed to beautify the UI.