Connect Four Game Final Report

Inchara Nagaraju

Pawan Kumar

Department of Computer Science, Stevens Institute of Technology

CS-501, Introduction to Java Programming

Dr. M Peter Jurkat

December 13th, 2021

1.Introduction

The goal of this project was to make a Connect Four computer game using Java Programming Language. We have tried our best to make sure our Java version of Connect Four looks as realistic as to the actual board game. We wanted all the same rules to apply. This project is interesting because the code involves some complicated nested loops and if else statements. This project took a lot of problem solving, but with lots of trial and error we were successful with creating Connect Four using Java.

2. ABOUT OUR GAME CONNECT FOUR

We created code that set up a 6 by 7 board. The rules for the game are exactly the same as any Connect Four game. Two players play against each other: red versus yellow. Here, by red means letter ‘R’ and yellow means letter ‘Y’. The player can chooses any number between 0 to 6. These number represent the 7 different columns; the letter will fall in the selected number column. If there are no letter in that column, it will fall to the bottom, but if there are already letter in that column, it will fall to the row directly above it. The code detects if you try and place a letter in an already taken spot, and it does not allow you to do that. We also made code that determines when there is four in a row and someone has won the game.

3.CONNECT FOUR DESIGN

A. METHODS

For creating our Java version of the game connect four, we have developed many methods in our program other than main method. These methods are mentioned below: -

1. `public static boolean enteredPositionOfTheDisk(char[][] board, int column, char player)`

   Using this method, we are placing the disc in the position entered by the user.

2. `public static void displayingTheBoard(char[][] board)`

   This method is to display the connect four board to the user.

3. `public static boolean isWin(char[][] board)`

   Using this method, we are determining whether a player win the game.

4. `public static boolean isDraw(char[][] board)`

   Using this method, we are checking, whether the game ends in draw or not.

5. `public static boolean isStraightFour(char[][] values)`

   Here, we are checking for the four consecutive letter occurrences either horizontally, vertically or diagonally.

6. `public static boolean isConnectFour(char[] enteredvalues)`

   This method checks whether there are four consecutive drops.

B. EXCEPTION HANDLING

We have also used exception handling to handle unexpected values entered by user, which is not correct. We have used try and catch for exception handling. We have used one built-in exception, which is **ArrayIndexOutOfBoundsException**, to prevent program from terminating if user enter integer a value other than integer value from (0 to 6).

The next exception handling we used is **InputMismatchException**, to prevent program from terminating if user entered a value other than integer values.

C. REPEAT LOOP

We have used while loop in our main method, so that after one round of game ends, user will have the choice to play again. After the game ends user will have the choice to either play the game again or exit.

4. CONCLUSION

We succeeded in achieving the goals we set out for ourselves, namely: to make playable connect four games. Java is well suited to support software for simple games such as Connect Four, and the project was manageable given the concept we've learnt this semester.