Connect 4: Java Mini-Project

Tanner Glantz and Brett Phillips

Group Number: 9

Team Name: null;

1. Approach:

We first made sketches on a whiteboard as to how we wanted the game board to look. After researching many different styles, orientations, and implementations of our game, we decided on the original color schemes and layout for the game. The chips are red and yellow overlaid on a blue board. The board is also the original seven column by six row design. After figuring out the look and feel of the board we began to discuss how each of the items were going to be created. We made image icons for each of the chips and a two dimensional array to store the chips being placed on the board. We began to code as we came up with ideas. We save our work on a git repository, so that we can track changes and revert to older versions if we made a mistake. Ultimately, it saves us from having to meet up every time to work on our project. It also provided the ability to leave comments for each other as we updated the repository. After the board had been implemented and we could get pieces on the board, we began to code more than we were planning. The planning was more focused on the design of the board and which objects were to be used to create certain elements on the board than writing pseudocode. We believe that we should write code in segments, test it, and then together move forward instead of working on many different portions at the same time. We discovered early on that writing code for different parts of the project and then trying to tie them together was more work than if we did one piece at a time.

1. Technical Problems:
   1. We had a lot of difficulty using the paintComponent() method because many of our calls to the repaint() method would not paint a change back to the frame. This did not have an effect on features we planned on implementing, but rather required us to build other new unplanned features into the game. For example, we implemented JOptionPanes into the program to remind the user what they must do in order to proceed. The goal of this approach is to make the user experience as easy as possible. Another problem we faced was working on the same project files from different computers at the same. Because we store our files in one centralized repository, we are both unable to work on the document at the same time. This was nice when we needed to work when the other was not around, however it meant that when we were together we were both working with the same computer. Our approach to this whole project was for us both to be involved in all steps of development. That way if an error occurred, it would be much easier to troubleshoot the problem because we are both aware of what we intended certain blocks of code to do.
2. Group Issues and Interactions:
   1. We met a few times during the week and every weekend on this project. Our meetings were more of a time to review each other's code and discuss where we thought the project was headed. We also took that time to fix errors that each other were having. This proved to be a very effective method for reviewing code. In terms of responsibility and delegation of tasks, we both worked on the Graphical User Interface and the logic portion of the code. However, Tanner worked with the write-ups of the project, while Brett developed the board and pieces for the game. Brett used photoshop to create the red and yellow chips as well as the blue board.
3. Plans for Next Time:
   1. We would recommend student’s taking this class next semester to plan out their meetings or scheduled time for this project in advance. We would warn them that they need to work on it every weekend even though they have other java homework assigned. This project has a lot of unforeseeable errors and troubleshooting involved. We would recommend that the professor demo some of the projects submitted in the past as well as have some dedicated class time before the due date to address problems students are having with their project. We, on future projects, would have spent less time on the design process and started writing the code much sooner.