Our project: Scrabble

We came up with the idea by brainstorming for a few days. At first my partner suggested doing some sort of rhythm game, where the player needs to click buttons to a beat, but we didn't really know how to implement music along with timing. Afterwards, I suggested to her Scrabble, as I'm pretty good at it and very familiar with the interface. She agreed and we started thinking about ideas on how to implement it. We determined that we needed separate objects for the board, the tiles, the players and the GUI. I personally thought that the tile slots and the tiles themselves should have been different objects, but we decided otherwise. We tried to divide work so that one person worked on scoring, while another worked on coding the GUI and other smaller tasks. I began work on scoring, but we switched because my partner was having some issues with GUIs. We didn't really meet in person, as our code wasn't really dependent on each other.

The project came together nicely, after quite a few bugs. The scoring mechanism is very well done (honestly I have no idea how she got that to code correctly) and the GUI is alright except for one major issue. Since I couldn't paint buttons, and Java layouts either require you to submit to a rigid design or write all the layout code yourself, I did everything in one canvas. This lead to some issues with loading everything quickly, flickering can occur when pressing a button. Furthermore, I didn't get to make a scorebox, instead using the terminal to output score. If I had more time, I would probably gut most of the GUI code (ugh) and rewrite it with multiple canvases (ugh) or maybe multithreading (ugh). I'd then make a scorebox and maybe support for more players. However, I'm pretty proud of my code. I tried to not repeat myself too much and I felt like the code reflected that. Also using absolute value and modular arithmetic for the score bonuses was pretty cool. Getting all the letters and point values to work was also a great job on my partner's end.

In the end I learned a little bit about how GUIs work, specifically canvases and painting. I also came to the conclusion that GUIs take a lot of time and patience to research, not necessarily because they're very hard to code (okay, yeah they're a pain), but because the documentation/tutorials really aren't written well. I'm defintely more satisfied with this project versus the Stuyablo one, as this one gave me the freedom to worry about how to code existing rules and not worry about how to design my own game in the process. Furthermore, the satisfaction of seeing your work respond in a GUI is great. Especially after a few hours of wondering why your damn MouseListener doesn't work. I think I could have coded this in September, but it would have taken a while longer. The examples in class helped a lot with understanding how GUIs function.