Team Plan Document

Jared Cormier jcc4969

Daniel John dcj597

Spring 2017

Jared and Daniel met up to discuss the project and how to implement the project. They discussed different methods of implementing the Adjacency List. Jared completed the Adjacency List and BFS module, along with the other helper functions such as printLadder(), parse(), and initialize(). Daniel initially worked on the DFS module. Both students met up to look over code and debug.

Things we discovered and fixed were:

> An error in the while loop, where the program quit after the search instead of staying running till the "\quit" keyword was entered.
>
> Making the code case-insensitive.

Daniel struggled to get the DFS module working and ran into a stack overflow issue on another attempt. Jared proceeded to save the day by implementing a DFS structure where a secondary helper function was called by the getWordLadderDFS() function and his implementation worked!

Daniel went to Office Hours to inquire about the DFS function, as it had thousands of rungs. He tested to make sure there were no duplicates in the DFS ladder rung by comparing the size of the ArrayList to a HashSet with the DFS ladder array list and making sure they were the same size. Jared, meanwhile, found a brilliant way to reduce the number of rungs by creating a function that prioritized the adjacent neighbors that had more letters in common than others. This significantly reduced the number of ladders.

Jared wrote 5 test cases for BFS. Daniel and Jared worked on DFS test cases and Daniel wrote them. Daniel wrote the Word Ladder Team Plan.

We spent approximately 6 hours together in the same room and close to another 10 hours away from each other. We communicated with each other in person and via Facebook Messenger.

The final testing was done together and the Canvas submissions were also done together.