The goal of assignment 6 was to create four different classes, a class of vertexes, a class of edges, a class which creates a graph and performs operations on it, and a class which manages a graph given user input. While doing this project, I learned a lot about coding a graph and being able to do things to it, and also keeping track of data and variables which ultimately were used between methods, so if one method indirectly affects a data point, it was important to adjust it based on the operation.

During this project, what I struggled most with was being able to code the shortest path algorithm. This algorithm was the most challenging thing I have ever attempted to code in my life, and I spent 2+ days alone just on making it the best that it could be, and it still fails some test cases. We did not really talk about it much in class, so I was left to search up visualizations of what it actually does to understand what data needs to be stored, and how the shortest path is constructed. Everything else besides this algorithm was fairly straightforward.

What would you do differently in the future?

In the future, I would make sure to make note of any algorithms in a given project that looks unfamiliar and make sure I have the basic idea down, so I don’t get stuck like how I did. In particular, for this project I would search up Dijkstra’s algorithm and make sure that I understood it or review the few slides where it was discussed. Furthermore, I would also seek additional resources or ask for clarification from the instructor if needed. It became clear during this project that a solid understanding of complex algorithms is crucial for success, and I would prioritize gaining that understanding early on in future assignments.