

Week 0 (11/11/2020 - 11/18/2020):

During this preliminary week our group worked on figuring out which direction our project would take. We wrote out our project goals after a great amount of discussion and helped draft and revise our group contracts. Our final decision was to use the open flight data set (more information on this is provided in GOALS.pdf). After creating the basic structure for our group's procedures, workload, and overall project outline, we ensured that each group member individually committed and signed onto the team contract.

Week 1 (11/18/2020 - 11/25/2020):

This week we have moved further towards our goal by creating a basic structure for our graphs in `graph.cpp` and `graph.h`. Another task that was completed was the creation of the `Makefile`. This `Makefile` utilized code from previous assignments and allowed our files to compile successfully. So far, our graph files are present as a means to test if our `makefile` works. We also created the `airport.cpp` and `airport.h` files to keep track of the airports based on their identification number, and we added files for directed edges which will serve the purpose of implementing a directed graph. We held a team meeting over Zoom between all four members during the middle of the week to ascertain which aspects of the final project we had accomplished as well as detailing future goals.

Week 2 (11/25/2020 - 12/2/20):

We deleted the files `airport.cpp` and `airport.h` and replaced them with `Airport.cpp` and `Airport.h` respectively due to a dependency error caused by the previous files. We also deleted `directed_edge.cpp` and `directed_edge.h` because their use was deemed unnecessary for our goals. Additionally, we tested and fully developed our graph implementation this week. As a result our graph can parse data as well as create vertices and edges for the corresponding data. Our team fixed an issue from previous weeks, so now every time a vertex or edge is made our graph will programmatically add an edge wherever needed. During our Zoom meeting this week we were able to resolve the issue of multiple people working on the codebase and committing at the same time by utilizing our avenues of communication at an increased rate. This ensured that each team member would be informed when the other was working on certain elements of the project, and they would subsequently wait until a working git commit had been pushed before they themselves modified the codebase.