I created 6 classes, one for each of the csv data files (Airport, Airlines and Routes), a ReadFile class, RouteFinding class, Node class and two files - a Helpers file, which contains some functions to perform operations like checking if a list contains an element, and a main file. Each of the data file classes create objects of the data files provided, i.e., Route, Airline and Airport objects, with getters for each of the object attributes as defined by the columns in the files. Here is a rundown of how the program works:

The ReadFile class reads and organizes the data from the csv files into unordered maps, in order to prevent duplicates. A RouteFinding object is used to call a run function which implements input validation before calling the breadth first search function. The file name is passed into the object and this call to the RouteFinding constructor also initializes a ReadFile object with the input file name. A node is created out of the source location provided from the input and it is used to carry out the breadth first search implementation with the help of the unordered maps from the ReadFile class. I chose the breadth first search algorithm because it finds the optimal solution based on the path cost which in this case is the number of flights it takes to get to the destination. When a path to the destination is found and returned by the breadth first search function, it is written to the output file.

It was challenging figuring put how to repeat my Java implementation of this problem in C++, and I soon learned that it is not advisable to just transliterate each line of Java code to C++ and hope it will run in the end. This is because even though Java may be similar to C++ in some ways, one thing that worked in Java could definitely not work the way you'd expect it to in C++, even with similar syntax. I have become familiar with some C++ data structures like the unordered map and vectors, and about when and where structures like vectors are more suitable than, say, lists. For instance, I used an unordered map because it allows for insertion – which I was doing for each of the csv files – in constant time. I have also learned about some libraries such as sstream and algorithm.

References

Varun. (2017, November 30). *C++ List – Find | Contains : How to search an element in std::list ?* thisPointer. https://thispointer.com/c-list-find-contains-how-to-search-an-element-in-stdlist/