**Reflective Report**

The challenge involved writing a C++ program whereby given the start city (e.g., Accra, Ghana) and a destination city (e.g., Winnipeg, Canada), outputs a series of flights that take a passenger from the start city to the destination city. Three files were given for the challenge, i.e., airports.csv, airlines.csv, and routes.csv. I started by creating an input file in the main.cpp file that would be used in the search algorithm. This main file contained the start city (city and country) on the first line and the destination city (city and country) on the second line. I then created an Airport class (Airport.h) which contained information about the airport, such as the airport code, city, and country. The Airport class extracted the data from the airports.csv file. I then extracted information such as airline code, source airport code, destination airport code, and the number of stops from the routes.csv file. The data was vital for creating the route map representing the possible flight routes from the start airport to the destination airport.

Afterward, given the start city (city and country) and destination city (city and country), I did a breadth-first search searching for the possible routes from the route map. I used some resources online, for example, checking syntax for using some data structures such as unordered\_map and queue. Some lessons I learned were the difference in syntax between Java and C++. Implementing the challenge using C++ was quite challenging, despite having a general idea of completing the task. I also discovered that the general idea behind solving the problem was similar in Java and C++. The difference occurred when implementing the different methods and functions and working with data structures such as queues and maps. Moreover, I learned the benefits of splitting a program into separate classes, making the program run efficiently and smoothly after integrating all the classes from the different files.

**References**

C++ String to float/double and vice-versa. (n.d.). <https://www.programiz.com/cpp-programming/string-float-conversion>

GeeksforGeeks. (2022, July 7). Traversing a map (or unordered\_map) in C++ STL. <https://www.geeksforgeeks.org/traversing-a-map-or-unordered_map-in-cpp-stl/>

GeeksforGeeks. (2022a, July 6). Queue in C++ Standard Template Library (STL). <https://www.geeksforgeeks.org/queue-cpp-stl/>

GeeksforGeeks. (2022a, May 11). Unordered Sets in C++ Standard Template Library. <https://www.geeksforgeeks.org/unordered_set-in-cpp-stl/>

Iterate through a C++ Vector using a “for” loop. (2012, October 3). Stack Overflow. <https://stackoverflow.com/questions/12702561/iterate-through-a-c-vector-using-a-for-loop>

Mulani, S. (2022, August 3). 3 Ways to Compare Strings in C++. DigitalOcean Community. <https://www.digitalocean.com/community/tutorials/compare-strings-in-c-plus-plus>