**Individual Reflection**

**Intermediate Computer Programming (C++) Final Project**



Obed Yiadom Boakye

(\_64572024\_)

Ashesi University College

Dr. Ebo Agyepong

Dr. Robert Sampah

Nov 28th, 2022

It would be an understatement to state that the C++ Project was easy. My initial impression of the project was that it would be simpler to implement than the Java Project we did since its execution was very similar to the Java Project. However, I realized that some of the libraries and methods that were more intuitive and easily accessible in Java were not accessible in C++. Implementing this project was not a straight - forward transference from one language to another. This, therefore, meant that I had to alter my approach to solving the Project Problem.

My first approach, which was thankfully very similar to the Java implementation, was to read the data files we were given. In order to do so, classes were created to provide an interface which allowed us to manipulate and retrieve information from the data in the files. The difference, however, between this C++ implementation and the previous Java implementation was that the C++ provided header files which allowed me to employ abstraction for each of the classes I created. And I actually came to appreciate the necessary readability it allowed me and helped efficiently define the member functions in the source file. Despite the lessons in class about the implementation of access modifiers and the proper application of private variables to a class in enhancing the security of data, I decided to make all my class member variables public to reduce the stress of creating and operating with mutators so that I could access the member variables directly.

I created classes so that I could properly and efficiently access and manipulate the data in the provided csv files. Therefore, each class represented each csv file, and the member variables of the class represented the attributes or columns of the csv files. I implemented this by reading each line in the file, and basically creating different objects per each line read in the file.

In order to store the entire entries of each file, I used maps and vectors, (which were the Java equivalent of Hashmaps), and array lists respectively. These data structures allowed me to store the various instances of the file classes.

The vectors in the program were used to store each of the instances of the file classes, while the maps were used to implement an indexing procedure for accessing particular object details(the values of the hashmap) based on specific information about the object(key). These data structures allowed me to work with the data files such that I could retrieve and use specific pieces of information to implement a path taken from one airport location to an airport destination. Therefore, if any user should enter a specific airport start location and airport destination location, the program would be able to extract and use the user input as keys to generate paths from the airport location to airport destination. These paths were generated based on the information from the files stored as object values in the Vector data structure.

In order to make use of the file manipulation abilities the data structures gave me, I proceeded to use an algorithm called the breadth-first search algorithm, which allowed me to find the subsequent paths from a particular airport location to a particular airport destination. The breadth-first search algorithm allowed me to track subsequent routes and airports that were reached before arriving at a destination airport location from a start airport location. After the destination airport location has been found, the program then returns the entire route from source to destination.

Just like the Java Project, the C++ final project really forced me to solicit a lot of help from peers, internet research articles and youtube videos in order to complete it. Although this experience was more arduous than insightful, I still managed to learn other various features of the C++ language.

One lesson I learnt from this task was that in order to be a happy programmer, one would have to be very cognizant of the available tools and the rules of operations of the tools of a given language.