Project Reflection

For this programming challenge, I used breadth first search to find the route from one point to the other. Before that, I created several classes. Among such classes included methods to read the three CSV files. These methods used a buffered reader to read the CSV files. The "," was used to split the elements in the CSV files. When the files are read, they are stored in array lists. I also created a class called CreateObjects that creates an airport object, airline object, and route object of the airports, airline, and route classes.

To use the breadth-first search I implemented, I had to use the file reader to read from the input file and split the input source and destination. The input file is a text file with the first line as the start point and the second line as the destination point. The source represents the start city and country, and the destination represents the destination city and destination country. With the breadth-first search, I used a queue to represent the frontier and HashSet to represent the explored set. First the initial state for the node was set. This was done by using the airportId to find the city and country. It is added to the frontier. Once it is explored, it is added to the explored set. This continues until the solution is found. In this case, the child node and the solution_path(a method created to show the explored. After the breadth-first search is executed, the buffered writer writes the route generated into an output file.

With this programming challenge, I learned that other algorithms could be implemented. For example, A* and Uniform cost search can also be used. I tried implementing the Uniform cost search, but it meant calculating the distance and using the Haversine formula. However, I discarded it since it took a long time to find the route from start to destination compared to the breadth-first search. With the help of some websites, I understood the programming challenge better.

References

GeeksforGeeks. 2022. *Breadth First Search or BFS for a Graph - GeeksforGeeks*. [online] Available at: <https://www.geeksforgeeks.org/breadth-first-search-or-bfs-for-a-graph/> [Accessed 30 September 2022].

www.javatpoint.com. 2022. *Java BufferedReader Class - javatpoint*. [online] Available at: <https://www.javatpoint.com/java-bufferedreader-class> [Accessed 30 September 2022].

W3schools.com. 2022. *Java Getting Started*. [online] Available at: <https://www.w3schools.com/java/java_getstarted.asp> [Accessed 30 September 2022].