CS 225 Project Goals
Haijian Wang
Alex Che
Sabien Bell
Aasheesh Randeo

For the basic data structure implementation, we are using the map/ linked list/ adjoint matrix that's covered in class. The OpenFlights data has routes and airports which can basically be interpreted as edges and vertices of a cyclic graph. Each airport has a unique ID which we can use for the index as the "key" to search.

By using the Dijkstra's Algorithm, the program will give out the most efficient and shortest routes between two vertices. It will output the graph of the shortest route (edge traversal) with the rest of the route options.

In graph theory, an Eulerian trail (or Eulerian path) is a trail in a finite graph that visits every edge exactly once (allowing for revisiting vertices). We will also create an Eulerian Path implementation which will help us determine that each route is unique and no edges overlap within the graph that is created.