# CS 225 Project Results
Haijian Wang
Alex Che
Sabien Bell
Aasheesh Randeo

The GOALS of the Final Project were :
- To implement the Open flights Dataset as a Graph, where each vertice represented an Airport and the edges represented the routes between multiple airports.
- Use the Dijkstra's Algorithm to find the shortest path between two vertices.
- Use the Eulerian Path Algorithm to make sure that each route between two vertices was unique and no vertex was visited again.
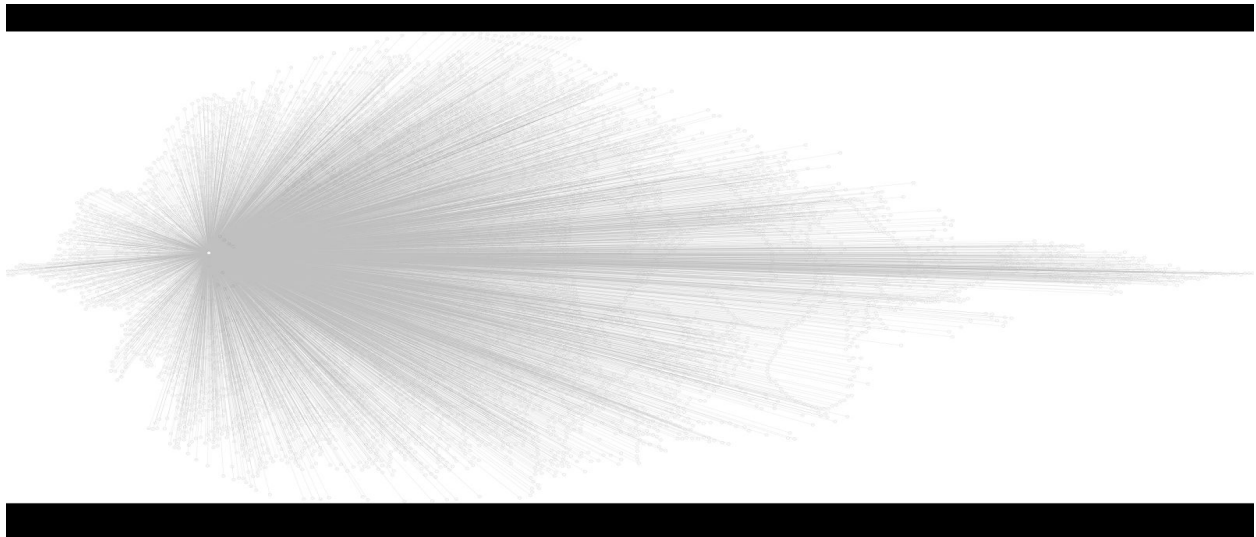
Output From g.savePNG() , for g is a graph as defined by Graph::Graph() constructor

Based on the image output from our main.cpp file, we can conclude that:
- Code works without any fatal compiler errors and within an efficient runtime
- Successfully passed all the data from.txt files into the graph.cpp
- The graph class perfectly defines the airports and routes as we expected it to

We further self-developed test cases under ***final_project/tests/unit_test.cpp*** to check whether the algorithm implementation was correct or not. By running the command *./test* we see that our code passes all our self-developed test cases and hence the algorithm implementation for Eulerian Path and Shortest path is correct, and will work on any graph defined under ***Graph::Graph()***.

```
Shortest Path:

Startpoint:
Fort Smith Regional Airport
Endpoint:
Aksu Airport

Path:

ID: 3437
Name: Fort Smith Regional Airport

ID: 3682
Name: Hartsfield Jackson Atlanta International Airport

ID: 3876
Name: Charlotte Douglas International Airport

ID: 4008
Name: Piedmont Triad International Airport

ID: 3520
Name: Ronald Reagan Washington National Airport

ID: 3448
Name: General Edward Lawrence Logan International Airport

ID: 1638
Name: Humberto Delgado Airport (Lisbon Portela Airport)

ID: 1229
Name: Adolfo Suárez Madrid-Barajas Airport

ID: 1218
Name: Barcelona International Airport

ID: 1353
Name: Marseille Provence Airport

ID: 1324
Name: Ajaccio-Napoléon Bonaparte Airport

ID: 1520
Name: Olbia Costa Smeralda Airport

ID: 1555
Name: Leonardo da Vinci-Fiumicino Airport

ID: 1501
Name: Bari Karol Wojtyła Airport

ID: 1190
Name: Tirana International Airport Mother Teresa

ID: 4317
Name: Sabiha Gökçen International Airport

ID: 2960
Name: Krasnodar Pashkovsky International Airport

ID: 4367
Name: Aktau Airport

ID: 2908
Name: Almaty Airport

ID: 3399
Name: Ürümqi Diwopu International Airport

ID: 6404
Name: Aksu Airport

Total Distance: 181
```

Output from d.print() after running d.shortest path("3437", "6404"), for Dijkstra object d.

Errors/ Failures:
- Starting vertex is reported as "6135". However, this vertice is not a part of the data set. We still haven't been able to find the bug associated with it.
- Choosing the right data structures for Dijkstra's algorithm to allow for both Vertex lookup for distances and sorting by distance

```
PROBLEMS    OUTPUT    DEBUG CONSOLE    TERMINAL
VISITED
VISITED
VISITED
VISITED
VISITED
VISITED
VISITED
VISITED
VISITED
VISITED
VISITED
VISITED
VISITED
VISITED
VISITED
VISITED
===============================================================================
All tests passed (20 assertions in 12 test cases)
```