## Dataset

Title: OpenFlights

Source Location: https://openflights.org/data.html

Format: One .dat file with airports and one .dat file with routes.

## Algorithms

Traversals: Breadth First Search

Covered Algorithms: Dijkstra's algorithm

Complex or Uncovered Options: Graphic output of graph - we will map out the airports and graphically show the shortest path from a to b

Alternative or Additional Algorithms:

- Landmark path: we can alternatively determine the optimal paths that one might take should they have to travel to an intermediate point(node). For instance, many people prefer to stop at a certain location for a certain reason. Hence, it would help if they knew the ideal path to go from point A to point B through a certain point C. Moreover, we can also extend this idea by taking into account multiple intermediate points. How would the optimal path change if we were to add 2 or 3 intermediate points? These cases need to be accounted for and we also have to determine a certain maximum so that the algorithm doesn't become too complex
- For determining the busiest airports we can use either the PageRank algorithm or an algorithm for betweenness centrality. This can be used as a supplement to the landmark path so people can find the path that avoids a busy airport.

## Rationale:

In our day and age, optimizing travel times is essential, as it can lead to increased productivity, reduced consumption of fuel and natural resources, and reduced emissions of harmful pollutants and greenhouse gases. Using this data will help people make informed decisions, based on historical data, and optimize travel time and expenditure. This will also help people visualize their potential travel options, which will aid in the decision of visual learners, and give them several options if they don’t like a specific route or plane for whatever reason.