Project Goals:

Dataset:

We will use the OpenFlights dataset given to us at https://openflights.org/data.html.
Since this dataset has airport data, we will be representing the airports as the nodes in
our graphs and the edges weighted as the minimum distance between each airport.

Traversal:

For the traversal, we will do a breadth first search on our graph in order to output the
result. This will also be useful in finding all airports in the area within a certain radius
relative to one chosen airport.

Covered Algorithm:

For our covered algorithm, we will use Dijkstra's algorithm in order to find the shortest
route from one airport to another. Since not every airport has a direct flight to another
airport, we will have to use Dijkstra's to find the most optimal route of airports to take in
order to get to the destination airport in the minimum distance.

Complex or Uncovered Option:

For our complex/uncovered option, we will be doing strongly connected components.
Specifically, we will be implementing Tarjan's algorithm in order to get the strongly
connected components. In our case, the strongly connected components will represent
all groups of airports that can access one another. For example, if airport A can access
airport B and airport B can access airport A, then we can group together airports A and
B and represent them as one super node. Once we have our strongly connected
components, we can now see which groups of airports do not have access to other
groups of airports, so with that, we can find out which new routes to add to the graph so
that there are definite routes for any arbitrary airport A to any arbitrary airport B.