Group Project
Shortest (cheapest) Path - Project Documentation
Data Structures and Algorithms (CS20230)

Thomas Anderson
17202065

April 30, 2018

Functional Specification

My program was written to find the cheapest option(weighted shortest path) for a given number of inputs.

The project requirements were to find the cheapest option when given:

- A maximum route size of 5 IATA airport codes
- An specific airplane
- the latitude and longitude coordinates of the airports
- Airport, Aircraft, Country-currency and currency rate data in separate CSV files

My program calculates the shortest path using the "brute force" method. This means that I calculate the number of permutations of airports for a given input, calculate the great circle distance for each permutation and then take choose the cheapest option.

The great circle distance is the shortest distance between two points on a sphere. It was used because it is needed to accurately measure the distances between airports.

The program is written in an object oriented style in Python. Classes were created to for the various functions my program has to perform. Emphasis was given to cleaning the data prior to loading it into memory.

Design

1. Data structures for holding the information provided, i.e. implement Classes to load into mem- ory Airport, Currency Rate and Country-Currency informations

   Dingle

2. Code to calculate the distance between airports
3. Code to price the cost of a single itinerary leg
4. A data structure to store an itinerary (a multi-leg trip) that can be either a set for inputs or a sequence for the best route
5. Calculate the distance of each leg of a given route (undirected graph with weighted edges)
6. Calculate the cost of each leg of a given route (directed graph with weighted edges
7. Code to permute all of the possible itineraries
8. Code to price an given itinerary
9. Calculate and display the cost of each leg of a given route
10. Calculate the best round trip route (cheapest cost) for an itinerary returning to the home airport using each airport once (5 airport solution)

11. Read in a testroutes.csv with airports and equipment as input and return a bestroutes.csv with the optimal route and cost of the route.

12. Basic unit tests

Sprint Three

Testing