Program Description

This program implements the uniform cost search algorithm in finding the optimal path from one location to another on a predefined map that have been already coded in to the program. The map which is the input to the program is defined by seven locations as it appears on the question given. However, there are 18 named routes in the program compared to 9 in the question. This is because a route from location A to B is not the same as route from B to A, therefore a single route in the problem is represented twice in the program. Name of locations is identical to the names on the problem while the naming used for routes is by combining the initials of the locations that the route connects. The appropriate data structure used to represent the map is a graph data structure where the vertices/nodes are locations and edges are routes. In addition, an IDE defined queue is used for the frontier list. But instead of using a closed list for the location already expanded a boolean variable by the name ‘Visited’ is included as an attribute to the location.

There are two major classes used to convey all the information provided on the map given. The first one is the “Location” and its attributes are:

* Name – as mentioned above this attribute holds the name of the location as it appears on the problem.
* Route – one out of the possible multiple routes that emanates from a location is chosen randomly.
* Location – this is the parent node of a location that helps in tracing the path from start to destination.
* Path cost- initially zero, this will maintain the cumulative sum of costs incurred along the way to the goal.
* Visited- a boolean variable that will be true if the location was chosen for expansion and false if not.

The second major class in the program is the “route” class with the attributes:

* Location – routes connect two locations, but this one will bear only the destination location.
* Cost – the cost of using the route in the transition from one location to another.
* Route – the route included in the definition of a location will point to another route and that route to another until all the routes that start from a location are explored.