|  |  |
| --- | --- |
| Assignment 2 | Zane Rawson  Comp261 |

**Work Completed:**

I managed to finish the core, completion and one of the tasks required for the challenge. The part of the challenge that I managed to complete was taking into account the restrictions file when finding optimal paths. I also atempted to complete the task regarding traffic lights but had difficulty finding the relevant information.

**A\* Search:**

Create new Priority Queue of Search Node for the fringe

Create new set of searched nodes

Create new set of visited nodes

Add the first node to the fringe

While the fringe size > 0: dequeue the highest and add it to the visited set if it hasn’t already been visited

Root CASE: if the node equals the goal, return this node

otherwise: calculate the cost to here add the node to the visited set

get the neighbours of this Node and if are not visited, add them to the fringe

To implement the one way directions I modified my Create Nodes file to add a way to differentiate between the roads going into a node and those going out from a node

**Heuristics:**

A\* almost always guarantees that the lowest cost route will be found in the graph where the costs are positive. One challenge with this was making sure the estimates were not greater than the actual cost of the route. For the way that the program and data is set out I decided that using the cartesian distance between two nodes would be the best way to determine my estimates.

Articulation Points:

To prevent the A\* search visiting every node in the graph the use of articulation points makes it a lot easier to achieve this as it shows the key intersections.in finding the articulations points I had a timer run as the algorithm was working and it record the time taken to find all the articulation points at 68 milliseconds.

**GUI File Modifications:**

To complete the route-finding part of this assignment I had to make a few changes to the GUI file that we were provided. The first of them being the inclusion of a toggle button called “Find Route”. The second modification was to add a parameter to the on load function as well as adding another function to the file. This enabled the program to load and make use of the restrictions file.