**Module 4 Critical Thinking Option # 1 – Informed Search Heuristics with SimpleAI**

By Daniel Diaz

In my project I use the informed intelligent search method known as A-Star. When searching for possible paths that one could take through a course in state space, we want to minimize the amount of time and effort in reaching our end goal. Famous applications such as Google Maps make use of informed search heuristic algorithms to find the least time-consuming shortest-distanced route to get someone where they are trying to go. In my project, I randomly generate a 2D maze with ‘obstacles’ that are represented by the number ‘1’. What I’m hoping to create is a very simplistic simulation of google maps. In applications such as Google maps, we need to find the shortest path between two points. For such an endeavor, A-Star is best suited to find this shortest path. A-Star utilizes Uniform Cost Search and Greedy Best First Search to find the shortest path. For A-Star to use both these methods in conjunction, it needs to be told a heuristic it can utilize to calculate the effort cost of each path. After summing costs of neighboring nodes, it will choose the one with the least cost and do this until it’s reached its end point. This is incredibly efficient for finding the shortest path BUT it relies heavily on the heuristic it is told (which is determined by the human programmer). My search is complete and admissible as it never overestimates the cost of the start point to the end point. This is ensured using the Manhattan heuristic method which finds the difference in horizontal and vertical steps taken along the path. It is space efficient as it evaluates every path it can take and finds the shortest of these (by finding the cost value of each node and node path). A big advantage is the combination of looking through possible moves (not just one) and then following the most efficient. It combines the best of UCS and Greedy Best First which is a major advantage of using this search in this case. However, a major disadvantage could come along when an ill-suited heuristic is used for calculating path costs. A-Star is suited for my simple example as we know much about the 2D maze we’ve created and how to travel within it. However, when more dimensions are involved in the maze, Manhattan distance would not work as efficiently as a heuristic.

Using/Testing the program:

The code requires user input for the size of the rows and columns in the maze. Also, the beginning point, and end point will need to be entered in by the user. These are to be entered in as ‘0,0’ (no parentheses or quotes).