Weighted A* Algorithm

Peter Conant — 012155624

October 7, 2024

1 Requirements

My implementation of Weighted A* Algorithm was developed in VSCode on a windows machine using Python (v3.12.4), Matplotlib, and FFMPegWriter.

2 Operating the code

The script can be run using command `python 012155624.py`.

There are two global boolean values on lines 8-9: `plot_graph` and `compile_video`. If `plot_graph` is TRUE, the graph will be displayed and updated for every iteration of the Algorithms. If `compile_video` boolean is `True` every frame of the graph will be saved and compiled into an mp4 named 012155624.mp4. These boolean work independently of each other, however when both of them are true the script slows greatly.

If you wish to change the start and end values on the graph from those retrieved from input.txt, you can do so by changing the `start` and `end` variables on lines 295-296.

3 Implementation

My implementation of A* is based on my implementation of Dijkstra. The code is updated to run 6 algorithms at a time each with a different weight. The heuristic value for each node is calculated when the node is created.

The `Edge` class keeps track of the start, end, and weight of each edge. *Node* tracks the coordinates (for plotting), a list of attached edge classes, the numeric identifier, the distance from the start node, and the parent node of it's shortest path.

Classes are generated when the input and coordinate files are read. The Edges are created first holding the weight, starting, end ending node and are stored in a list. Nodes are then created using the coordinates file and Edges list. Nodes hold the x,y coordinate for updating the plot, a empty parent node to be filled later, and the heuristic value which is calculated based on the distance from the node to the end node. The nodes are stored in a list and kept there for later use.

The plot is created by repeatedly plotting the same edges and nodes over 6 separate sub graphs and placing them all on a single figure.

4 Iterations

Dikstra (A* 0)	A* 1	A* 2	A* 3	A* 4	A* 5
30.14	30.14	30.63	31.8	31.8	31.8
98	62	30	26	27	26

Table 1: Number of Iterations for all weights of A*