

Dominick White

dw895

CS387

## README

In order to implement A\* in this project I created a Node class. The attributes of my Node class included the x and y positions, g value which is just the distance from the starting cell, and heuristic value. I have getters and setters for each of the attributes. One of the more important function of the Node class is the `getChildrenOf()` function that returns the Node children of the current node. It only returns the up, down, left and right children to keep the branching factor down.

**Heuristic:** My heuristic is just the g value + the manhattan distance

**A\*:** I essentially just followed the pseudo code from the slides with some minor adjustments. In order to pull the lowest heuristic node from OPEN whenever adding a new node to OPEN I inserted it at index 0 because the closer to the target you get the lower the heuristics should be, then performed a linear search to find the lowest heuristic node and pop it.

**computePath:** Compute path takes the last node in CLOSED(goal node) and then backtracks by looking at the parent nodes until it gets to the beginning. Each iteration it adds the coordinates to the path list, then reverses it at the end to put it in correct order. Otherwise, if CLOSED is empty it returns null.

### Runtime:

The run time should be  $O(4^d)$ , 4 because the branching factor is 4(up, down, left, right), and d is the depth.