ECS 170 Project 1: Part 4

Otto P Chen 912260569, Stephan Zharkov 999706363

February 2017

1 Modified A* Algorithm

1.1 Speed

To reduce spacial and time complexity, we decided to create a new class of points that would hold relevant information such as the cost, the parent, and coordinates we are dealing with for the given node. For fast access to these points, we decided to use Java's Hash Map data structure for O(1) retrieval and easy look up based on a string. We use to separate hash maps to differentiate between opened and closed nodes. The priority queue has an O(log(n)) insertion and deletion complexity, making it an incredibly fast data structure for accessing visited nodes. Otherwise, we did not use any special modifications to alter our algorithm.

1.2 Heuristic

We used the same exact exponential heuristic that we spoke about in part 1 of our solution, due to it's speed and coverage of a large variety of test cases. We use the number of moves away from the goal state to the current state, using diagonal movement as our basis.