Emma Peterson
edp2117
COMS 4701

Homework 1 Report

4) Summary of results, comparison of approaches

Note: All time and space requirements detailed below refer strictly to the algorithms themselves and do not include the extra time/space required to build the graph at the beginning of the program. The graph is stored as m lists of n lists of nodes (sort of a two dimensional array of nodes). The node class itself only contains its value (a string) and its position (a tuple).

Breadth First Search

a) cost of path: **11** for Arena 1, **75** for Arena 2, and **230** for Arena 3. These low costs match those of the optimal A* algorithm, and make sense considering the fact that BFS finds the shallowest solution.
b) memory requirements: **45** nodes were explored for Arena 1, **1132** for Arena 2, and **2480** for Arena 3. This $O(b^d)$ space and and space requirement if reflected in the high rate of growth for both of those properties.
c) runtime: The runtime on Arena 1 was **0.005388 seconds**, on Arena 2 it was **4.264 seconds**, and on Arena 3 it was **22.11 seconds**. The exponential nature of the time complexity of BFS is clear in the rapid growth of these runtimes (and space requirements in part b) as the number of nodes goes up.

Depth First Search

a) cost of path: **19** for Arena 1, **83** for Arena 2, and **664** for Arena 3.
b) memory requirements: **35** nodes were explored for Arena 1, **242** for Arena 2, and **2143** for Arena 3. Although DFS is not an optimal search strategy, in some cases (depending on the state space) it happens to find a path efficiently with respect to space, as is the case for Arena 2. Its O(bm) space complexity causes its memory requirements to be smaller than those of the $O(b^d)$ BFS.
c) runtime: The runtime on Arena 1 was **0.002119 seconds**, on Arena 2 it was **0.07664 seconds**, and on Arena 3 it was **11.72 seconds**.

A* Search

a) cost of path: **11** for Arena 1, **75** for Arena 2, and **230** for Arena 3. This optimal algorithm matches the path cost results of BFS, which means that it also found the shallowest solution.
b) memory requirements: **35** nodes were explored for Arena 1, **676** for Arena 2, and **1855** for Arena 3.

c) runtime: The runtime on Arena 1 was **0.0003421 seconds**, on Arena 2 it was **0.008036 seconds**, and on Arena 3 it was **0.02261 seconds**. This informed search strategy is optimal, and its comparatively low space and time costs reflect that.

5) A* is the only informed search method used in this assignment (besides Iterative Deepening A*), and therefore it is the only method that would be affected if the goal position was not given right at the start of the algorithm. BFS and DFS are uninformed search strategies that operate without “knowledge” of the goal position.

6) Uniform-Cost Search is an example of an uninformed search algorithm that could be applied if the cost of moving from one position to a neighbor was not necessarily equal. It would expand the node n with the lowest path cost g(n). This is similar to A* search, but Uniform-Cost Search would not include heuristic information.