ECS 170 Problem Set #1 Hadi Bagdadi

1) BFS must store all nodes at the current depth, leading to exponential growth with d due to the branching factor b. DFS only stores a single path at a time, which grows linearly with d, as it does not keep all nodes in memory simultaneously.

2) DFS: A Stack b/c DFS explores as deep as possible along each branch before backtracking, and a stack keeps track of the nodes to backtrack to.

BFS: A queue b/c BFS explores all nodes at the current depth before moving to nodes at the next depth level, and a queue ensures nodes are processed in the correct breadth-first order (FIFO)

Dijkstra's: A priority queue b/c Dijkstra's algorithm selects the next node to process based on the smallest cumulative cost, and a priority queue efficiently keeps track of the nodes with their associated costs for optimal selection.

3) a) $O(b^{C^*/\epsilon})$

b) the key assumption is that all edge costs are non-negative

4) a) BFS is a special case of uniform cost search when: all edge costs are equal

b) Uniform cost search is a special case of A^* when: the heuristic function $h(n) = 0$ for all nodes n

5) a) State Space: All possible configurations of the Sodoku board, including partially and fully filled boards

Goal State: A completed Sodoku board where each row, column, and 3x3 subgrid contains all digits from 1-9 with no repetition

Successor Function: Generate new states by filling a blank cell with a number from 1-9 that does not violate the rules of the game

Path Costs: All steps are equal in cost, thus the path cost can be considered the number of moves made.

b) Preferred Method: DFS b/c DFS is memory efficient compared to BFS because it only needs to keep track of the current path, not all nodes at a level. Sodoku also has a deep solution space where valid boards are rare, thus making DFS the better choice.

c) A good heuristic is possible. the number of blank cells could suggest closer proximity to the goal state

6) a) State space: represented as a tuple (Farmer, Fox, Goose, Grain) where each state of each agent could be east or west.

Goal state: (east, east, east, east) where all agents are on the east side of the river

b) at each state, the farmer could either move across the river alone, could move across with the fox, could move across with the goose, or could move across with the grain.

3 c) to avoid invalid states, the fox can not be left alone with the goose and the goose can not be left alone with the grain

d) Count the number of agents (fox, goose, grain) still on the west side of the river and divide by 2 (since the farmer can only take one item at a time but must also return). This heuristic never overestimates the cost because at least $\lceil(\text{remaining agents})/2\rceil$ trips are required to bring the agents to the east side while satisfying constraints.