**CECS 545-50 Exercises 3.10, 3.14, 3.15abc**

**Ryan Schron**

**Exercise 3.10**

State—Describes information for all attributes that are relevant towards reaching a defined goal. In solving an AI problem using search, many states may be visited en route to the solution.

State Space—The collection of all possible states (and sequences of states) who’s traversal may result in reaching the goal.

Search Tree—Generated sequences of states that may branch to represent various traversal options. In a search tree, the nodes represent a given state, and the connections between the nodes are actions taken to move between the states.

Search Node—A representation of a state in a search tree. Multiple nodes can represent the same state given that there may be multiple valid action sequences to reach the state. This means that while a search node may represent a state, there is a distinction between the two.88885555888

Goal—The desired result of a search problem. This is typically defined before starting the search.

Action—An effect that results in a transition between states.

Transition Model—A description of actions that may result in state changes in the system.

Branching Factor—Describes the maximum number of branches possible for any node. That is, how many state transitions are possible from the node.

**Exercise 3.14**

The GRAPH-SEARCH property states that for the search tree generated by the algorithm, any search initiating from the starting state must pass through a frontier node, which separates explored nodes from the unexplored region. This is true for the initial state, as the initial state is a single explored node with frontier nodes (currently unexplored) as potential paths separating the state from the unexplored region. As search commences, a frontier node is explored, and new frontier nodes are identified, regenerating the process and ensuring the GRAPH-SEARCH property is maintained. This property would be violated in the case that the searching algorithm does not identify potential new paths, thus frontier nodes would be able to be identified.

**Exercise 3.15**

1. False. A DFS approach may not always expand the exact number of nodes required to reach the end state, whereas A\* will always find the solution.
2. True. h(n)=0 is an admissible heuristic for the 8-puzzle because it is non-negative with a cost of 0.
3. True. In the case of robotics, A\* is not the best solution.