

Problem 1

3.14 Which of the following are true and which are false? Explain your answers.

- a Depth-first search always expands at least as many nodes as A^* search with an admissible heuristic.
- b $h(n) = 0$ is an admissible heuristic for the 8-puzzle.
- c A^* is of no use in robotics because precepts, states, and actions are continuous.
- d Breadth-first search is complete even if zero step costs are allowed.
- e Assume that a rook can move on a chessboard any number of squares in a straight line, vertically or horizontally, but cannot jump over other pieces. Manhattan distance is an admissible heuristic for the problem of moving the rook from square A to square B in the smallest number of moves.

Problem 2

3.15 Consider a state space where the start state is number 1 and each state k has two successors: numbers $2k$ and $2k + 1$.

- a Draw the portion of the state space for states 1 to 15.
- b Suppose the goal state is 11. List the order in which nodes will be visited for breadth-first search, depth-limited search with limit 3, and iterative deepening search.
- c How well would bidirectional search work on this problem? What is the branching factor in each direction of the bidirectional search?
- d Does the answer to (c) suggest a reformulation of the problem what would allow you to solve the problem of getting from state I to a given goal state with almost no search?
- e Call the action going from k to $2k$ left, and the action going to $2k + 1$ right. Can you find an algorithm that outputs the solution to this problem without any search at all?

Problem 3

3.18 Describe a state space in which iterative deepening search performs much worse than depth-first search (for example, $\mathcal{O}(n)$ vs. $\mathcal{O}(n)$).

Problem 4

3.21 Prove each of the following statements, or give a counterexample:

- a Breadth-first search is a special case of uniform-cost search.
- b Depth-first search is a special case of best-first tree search.
- c Uniform-cost search is a special case of A^* search.

Problem 5

3.23 Trace the operation of A^* search applied to the problem of getting to Bucharest from Lugoj using the straight-line heuristic. That is, show the sequence of nodes that the algorithm will consider and the f, g , and Li score for each node.