Problem 1. 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.

Solution. False: a lucky DFS might expand exactly $d$ nodes to reach the goal. $A^*$ largely dominates any graph-search algorithm that is guaranteed to find optimal solutions.

(b) $h(n) = 0$ is an admissible heuristic for the 8-puzzle.

Solution. True: $h(n) = 0$ is always an admissible heuristic, since costs are nonnegative.

(c) $A^*$ is of no use in robotics because percepts, states, and actions are continuous.

Solution. True: $A^*$ search is often used in robotics; the space can be discretized or skeletonized.

(d) Breadth-first search is complete even if zero step costs are allowed.

Solution. True: depth of the solution matters for breadth-first search, not cost.

(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 rock from square A to square B in the smallest number of moves.

Solution. False: a rook can move across the board in move one, although the Manhattan distance from start to finish is 8.

Problem 2. The iterative lengthening search is an iterative analog of uniform cost search. The idea is to use increasing limits on path cost. If a node is generated whose path cost exceeds the current limit, it is immediately discarded. For each new iteration, the limit is set to the lowest path cost of any node discarded in the previous iteration.

(a) Show that this algorithm is optimal for general path cost.
Solution . The algorithm expands nodes in order of increasing path cost; therefore the first goal is encountered will be the goal with the cheapest cost.

(b) Consider a uniform tree with branching factor $b$, solution depth $d$, and unit step costs. How many iterations will iterative lengthening require?

Solution . It will be the same as iterative deepening, $d$ iterations, in which $O(b^d)$ nodes are generated.

(c) Now consider step costs drawn from the continuous range $[\epsilon, 1]$, where $0 < \epsilon < 1$. How many iterations are required in the worst case?

Solution . $d/\epsilon$

Problem 3 (Optional). Describe a state space in which iterative deepening search performs much worse than depth-first search (for example, $O(n^2)$ vs. $O(n)$).

Solution . Consider a domain in which every state has a single successor, and there is a single goal at depth $n$. Then depth-first search will find the goal in $n$ steps, whereas iterative deepening search will take $1 + 2 + 3 + \ldots + n = O(n^2)$ steps.

Problem 4 (Optional). Prove each of the following statements, or give a counterexample:

(a) Breadth-first search is a special case of uniform-cost search.

Solution . When all step costs are equal, $g(n) \propto depth(n)$, so uniform-cost search reproduces breadth-first search.

(b) Depth-first search is a special case of best-first tree search.

Solution . Breadth-first search is best-first search with $f(n) = depth(n)$; depth-first search is best-first search with $f(n) = -depth(n)$; uniform-cost search is best-first search with $f(n) = g(n)$.

(c) Uniform-cost search is a special case of $A^*$ search.

Solution . Uniform-cost search is $A^*$ search with $h(n) = 0$

Problem 5 (Optional). The heuristic path algorithm (Pohl, 1977) is a best-first search in which the evaluation function is $f(n) = (2 - w)g(n) + wh(n)$. For what values of $w$ is this complete? For what values is it optimal, assuming that $h$ is admissible? What kind of search does this perform for $w = 0$, $w = 1$, and $w = 2$?

Solution . It is complete whenever $0 \leq w \leq 2$. $w = 0$ gives $f(n) = 2g(n)$. This behaves exactly like uniform-cost search - the factor of two makes no difference in the ordering of the nodes. $w = 1$ gives $A^*$ search. $w = 2$ gives $f(n) = 2h(n)$, i.e., greedy best-first search. We also have $f(n) = (2 - w)[g(n) + \frac{w}{2-w}h(n)]$ which behaves exactly like $A^*$ search with a heuristic $\frac{w}{2-w}h(n)$. For $w \leq 1$, this is always less than $h(n)$ and hence admissible, provided $h(n)$ is itself admissible.