1. Give the initial state, goal state, successor function, and cost function for each of the following. Choose a formulation that is precise enough to be implemented. [Exercise 3.7] (20 pts)

(a) You have to color a planar map using only four colors, in such a way that no two adjacent regions have the same color.

(b) A 3-foot-tall monkey is in a room where some bananas are suspended from the 8-foot ceiling. He would like to get the bananas. The room contains two stackable, movable, climbable 3-foot-high crates.

(c) You have a program that outputs the message “illegal input record” when fed a certain file of input records. You know that processing of each record is independent of the other records. You want to discover what record is illegal.

(d) You have three jugs, measuring 12 gallons, 8 gallons, and 3 gallons, and a water faucet. You can fill the jugs up or empty them out from one to another or onto the ground. You need to measure out exactly one gallon.

2. Consider a lawn mower agent. (10 points)

(a) Write a PEAS description for the agent.

(b) Characterize the lawn mower agent environment according to the following properties, and briefly justify your answer.

- Accessible
- Episodic
- Static
- Discrete

3. Consider the Tower of Hanoi puzzle (see Figure 1), with \( n \) disks and \( k \) pegs. The disks are of different sizes. A disk can only be moved if there are no other disks on it. A disk can only be moved to a peg that
is empty or to a peg whose top disk is larger than it is. Figure 1 shows a puzzle with 3 pegs and 3 disks.

Give a problem formulation, including description of initial state, goal state, successor function and path cost.

Show the state space through the first two valid moves. Do not show repeated states. (10 points)

4. Consider the tree in Figure 2. The tree is labelled as in the class example. Values on the edges are path costs, values in the nodes are heuristic estimates. Goal nodes are darkened.

For each search strategy below, show the complete open list at each stage, where the first element in the open list is the next to be removed and explored. Note: the heuristic is not admissible, but perform the trace anyway.

(a) Iterative Deepening (5 points)
(b) Uniform Cost (5 points)
(c) Greedy-Best-First (10 points)
(d) A* (10 points)

5. Consider the following sliding-tile puzzle. The puzzle consists of three black tiles (marked B), three white tiles (marked W), and an empty space (marked *).

**Initial state:** BBB*WWW

**Goal:** To have all the white tiles to the left of the black tiles. Position
of the blank is unimportant.

**Operators:** A tile may move into an adjacent empty location, or it may “hop over” one or two other tiles into the empty position.

**Path cost:** One to move to the adjacent empty position, one to hop over a single tile, or two to hop over two tiles.

Let the *positions* of the puzzle be numbered 1 through 7, left to right. Consider the following heuristic, h:

Let p be the position of the tile

If tile is white and p is 5, 6 or 7 then score is p-4
Else if tile is black and p is 1, 2 or 3 then score is 4-p
Else score is 0

For example:

\[
\begin{align*}
    h(*BBWWBW) &= 0 + 2 + 1 + 0 + 1 + 0 + 3 = 7 \\
    h(BW*BBWW) &= 3 + 0 + 0 + 0 + 0 + 2 + 3 = 8 \\
    h(WW*WBBB) &= 0 + 0 + 0 + 0 + 0 + 0 + 0 = 0
\end{align*}
\]

- Draw the A* search tree for the first three expansions. That is, apply each valid operation to the start state, and compute the \( f \) values. Select the “best” node according to the A* algorithm, and expand it. Show the \( f \) values for these nodes too, select the next and expand it. Finally, indicate which would be the next one expanded (but do not expand anymore yourself!) You do not have to show repeated states. (15 pts)

- Is this heuristic admissible? Explain your reasoning. (5 points)

6. **Graduate students only** (10 points)

   (a) What are two advantages of breadth-first search (BFS) over hill climbing?

   (b) What is the major advantage of hill-climbing over greedy search?

   (c) Sometimes there is no good evaluation function for a problem, but there is a good comparison method: a way to tell if one node is better than another, without assigning numerical values to either. Show that this is enough to do a best-first search. Is there an analog of A*? [Exercise 4.12]