1. (Problem 7.3 Nillson) Three missionaries and three cannibals come to a river. There is a boat on their side of the river that can be used by either one or two persons. How should they use this boat to cross the river in such a way that cannibals never outnumber the missionaries on either side of the river.

   (a) Specify the form of state description, the initial state and the goal state for this problem. Describe the state space using variables (as if you are using an array in a program). Determine how many states are in state space.

   (b) Describe the set of operators using if-then rules.

   (c) Draw the entire state space graph (include only legal states, that is, states in which cannibals do not outnumber missionaries on either side of the river)

   (d) Describe a depth-first search algorithm and show a trace leading to a solution.

2. (Problem 7.4 Nillson) Refer to the three-disc Tower-of-Hanoi puzzle defined in Exercise 5.3. Let the operators that describe actions be given by the schema move($x$, $y$, $z$), where $x$ can be any of the three discs $D_1$, $D_2$, or $D_3$, and $y$ and $z$ can be any pair of distinct pegs $A$, $B$, or $C$. Define state description for this puzzle, identify the start state and a goal state, and draw the complete search space containing all possible states of the puzzle. Label the arcs by the appropriate operator. (Each move is reversible; you need label only one of each pair of reversible moves.)

3. (Problem 8.2 Nillson) List the order in which nodes are visited in the tree in Figure 1 for each of the following three search strategies (choosing leftmost branches first in all cases):

   (a) Depth-first search
   (b) Depth-first iterative-deepening search (increasing the depth by 1 each iteration)
   (c) Breadth-first search

4. (Problem 8.3 Nillson) Consider a finite tree of depth $d$ and branching factor $b$. (A tree consisting of only a root node has depth zero; a tree consisting of a root node and its $b$ successors has depth 1; etc.) Suppose the shallowest goal node is at depth $g \leq d$.

   (a) What is the minimum and maximum number of nodes that might be generated by a depth-first search with depth bound equal to $d$?

   (b) What is the minimum and maximum number of nodes that might be generated by a breadth-first search?

   (c) What is the minimum and maximum number of nodes that might be generated by a depth-first iterative-deepening search? (Assume that you start with an initial depth limit of 1 and increment the depth limit by 1 each time no goal is found within the current limit.)
5. (Problem 8.5 Nilsson) Assume we are searching a tree with branching factor $b$. However, we do not know that we are really searching a tree, so we are considering checking each state description generated to see if it matches a previously generated state description. How many such checks would have to be made in a search of the tree to depth $d$?

6. Consider a sliding block puzzle with the following initial configuration:

| B | B | W | W | W | E |

There are three black tiles (B), three white tiles (W), and an empty cell (E). The puzzle has the following moves:

- A tile may move to an adjacent empty cell with unit cost.
- A tile may hop over at most two other tiles into an empty cell with a cost equal to the number of tiles hoped over.

The goal of the puzzle is to have all of the white tiles to the left of all of the black tiles (without regard to the position of the blank cell).

(a) Is the search graph a tree? (Does it have cycles of length 3 or more)? If so, give an example.
(b) Specify a heuristic function, $h$, for this problem.
(c) Describe the first 10 nodes expanded by BFS in the order they are expanded.