Indian Institute of Technology, Jodhpur
B.Tech.(CSE) 3rd Year, II Semester 2014-15
CS324: Artificial Intelligence Assignment # 4

1. Suggest a heuristic function for the 8-puzzle that sometimes overestimates, and show how it can lead to a suboptimal solution on a particular case.

2. Give the name of the algorithms that results from each of the following special cases:
   (a) Local beam search with $k = 1$.
   (b) Local beam search with $k = \infty$.
   (c) Simulated annealing with $T = 0$ at all times.
   (d) Genetic algorithm with population size $N = 1$.

3. In the Traveling Salesperson Problem (TSP) one is given a fully connected, weighted, undirected graph and is required to find the Hamiltonian cycle (a cycle visiting all of the nodes in the graph exactly once) that has the least total weight.
   (a) Outline how hill-climbing search could be used to solve TSP.
   (b) How good results would you expect hill-climbing to attain?
   (c) Can other local search algorithms be used to solve TSP?

4. Is there a danger of Local maximum in GA? How does the algorithm tries to avoid it?

5. What are the different data structures used to implement the *open* list in BFS, DFS, Best first search?

6. If there is no solution, will A^* explore the whole graph?

7. Define the and describe the following terms related to heuristics:

   Admissibility, monotonicity, informedness.

8. Show that:
   (a) The A^* will terminate ultimately.
   (b) During the execution of the A^* algorithm, there is always a node in the open list, that lies on the path to the goal.
   (c) If there exits a path to goal, the algorithm A^* will terminate by finding the path of to goal.

9. Discuss the ways using which h function in $f(n) = g(n) + h(n)$ can be improved, during the search.

10. Why must the A^* algorithm work properly on a graph search, with graph having cycles?

11. Find a appropriate state space representation for the following problems. Also, suggest suitable heuristics for each.
    (a) cryptarithmetic problems (e.g., TWO+TWO = FOUR)
    (b) Towers of Hanoi
    (c) Farmer, Fox, Goose, and Grain.

12. Suggest appropriate heuristics for each of the following problems:
    (a) Theorem proving using resolution method
    (b) Blocks world

13. If $P =$"heuristic is consistent", and $Q =$"heuristic is admissible". Then show that $P \Rightarrow Q$. Demonstrate by counter example that $Q \nRightarrow Q$.

14. Consider the magic-puzzle shown in figure 1. Suggest the formalism for searching the goal state, when started from the start state.(Note that in the goal state all the rows, columns, and diagonals some equal).

1	2	3
4	5	6
7	8	9

Start state

6	7	2
1	5	9
8	3	4

Final state

Figure 1: Magic Puzzle.

15. *Sudoku* can be viewed as a binary constraint satisfaction problem.
    (a) What are the variables of this CSP?
    (b) What are their domains?
    (c) How would you translate the requirement that no two of the same digit may occur in the same row, column, or block into binary constraints?
    (d) Does the requirement that each digit occur at least once in each row, column, or block have to be directly specified? Why or why not?

16. Solve the following CSP problems:
    (a) $TWO + TWO = FOUR$
    (b) $ABC + DEF = GHIJ$

17. Consider the problem of coloring a complete three-vertex graph with three colors, where the nodes X_1, X_2, X_3 are all the set $\{r\ g\ b\}$; X_{12}, X_{13}, and X_{23} all equal $\{rg\ rb\ gr\ gb\ br\ bg\}$ and $X_{123} = \{rgb\ rbg\ brg\ bgr\ grb\ gbr\}$, the six possible colorings. Make use of extended synthesizing algorithm to color this graph.

18. Demonstrate the min-max search for tic-tac-toe puzzle (10 moves).

19. Demonstrate the alpha-beta search for tic-tac-toe puzzle.

20. A game *nim* is played as follows: there are two players who remove one, two, or three coins, alternately from a stack of five coins. Represent this game playing as a search tree. Suggest any suitable strategy for winning this game?

21. Given the tree in figure 2, explore this tree using the alpha-beta procedure. Indicate all parts of this tree that are cut-off. Also, indicate the winning path(s), and strike out all the values that are not required to be computed.

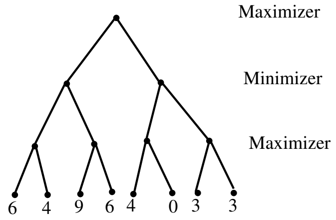

Figure 2: Alpha-beta search.

22. Consider the standard Towers of Hanoi problem with 3-pegs, and 4 number of disks (d_1, d_2, d_3, d_4, with d_1 at top). The disks are to be transferred from start-peg to end-peg, using intermediate peg, one at a time such that at no time larger disk comes over the smaller. The disk d_1 is smallest and d_4 is largest. Make use of only 3-predicates: unary predicate: *clear*, and binary predicates: *on* and *smaller*, and only one action: $puton(x, y)$ needs to be used.

    Write the domain of the problem, and make use of *forward planning* to plan the solution to move all the 4 disks from start-peg to end-peg.

23. Given the 3-SAT problem:

$$(\neg p_1 \vee p_2 \vee p_3) \wedge (p_1 \vee \neg p_2 \vee p_3) \wedge (p_1 \vee p_2 \vee \neg p_3)$$

    Solve it using *forward planning.* (Hint: you need to assume some *operators* (i.e., actions) to assign the values to variables $p_1 \dots p_3$.)

24. Given the Table T, and blocks A, B, C, D, apply the STRIPS to plan the solution of following problem:

    Initial state **I** :

$$clear(A), clear(B), clear(C), clear(D), on(A, T),$$
$$on(B, T), on(C, T), on(D, T)$$

    i.e., the blocks $A \dots D$ are on table, and their tops are clear.

    Final State **F** :
    $on(A, T), on(A, B), on(B, C), on(C, D), on(D, T), clear(A)$.

    Use the action $puton(X, Y)$, where X is a block $A \dots D$ and Y is either table T or block $A \dots D$. Give the forward planning to reach state **G** starting from state **I**.

The assignment is for practice only.