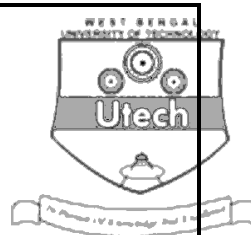


[illegible]

36014 (07 / 07)



DO NOT WRITE ON THIS PAGE



CS/M.Tech(MSS)/SEM-2/MSS-206/09
DESIGN & ANALYSIS OF ALGORITHMS
SEMESTER - 2



Time : 2 Hours]

[Full Marks : 70

The figures in the margin indicate full marks.

Candidates are required to give their answers in their own words as far as practicable.

Answer any *five* questions.

5 × 14 = 70

1. Define NP-hard and NP-complete problems. How do you prove the NP-hardness of a given problem ? Prove that the clique decision problem is NP-hard. 4 + 3 + 7

2. What is an AND-OR graph ? How to construct an AND-OR graph for a given problem ? Construct an AND-OR graph to decide whether the Boolean expression $p.(q + r'.s') + r'.(r + s'.(s + p))$ is satisfiable or not. 2 + 4 + 8

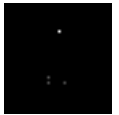
3. a) Explain the terms *a priori* complexity analysis and *a posteriori* complexity analysis. Differentiate, with a proper example, among best case, average case and worst case time complexity of an algorithm. 2 + 6

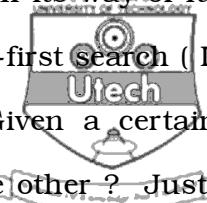
 b) Define the big-oh (O), omega (Ω) and the theta (Θ) notations. Consider the function $f(n) = 18n + 10$. Prove that $f(n) = \Theta(n)$. 6

4. What do you mean by backtracking ? Present a backtracking algorithm to solve the n -queen problem and apply it to solve the 4-queen problem. Show the intermediate steps. 2 + 6 + 6

5. Do you agree with the statement 'The concept of an algorithm is formalized as a Turing Machine (TM)' ? Justify your answer. 14

6. Design a Turing Machine to copy a given string w of consecutive a 's to ww , i.e., to a string of twice as many consecutive a 's. Explain its way of working. 9 + 5



7. Present a generalized graph search algorithm and explain its way of functioning. How do you obtain the breadth-first search (BFS) and depth-first search (DFS) as special cases of the generalized graph search algorithm ? Given a certain graph search problem, which of BFS or DFS is more efficient than the other ? Justify your answer with suitable examples.  4 + 4 + 6

8. Explain the basic strategy of 'A' algorithm. How does it differ from A^* algorithm ? Present an A^* algorithm to solve the Travelling Salesperson Problem (TSP). 3 + 2 + 9

END