TUTORIAL QUESTIONS

Subject: Design and Analysis of Algorithms **Regulation: R20**

UNIT I

1) a) Implement iterative function for sum of array elements and find itsspace complexity
   b) Discuss the Pseudo code conventions for expressing algorithms.
2) a) Show that the following equalities are incorrect with suitable notations
   i) $10n^2+9 = O(n)$ ii) $n^2\log n=\Theta(n^2)$
   b) Implement an algorithm to generate Fibonacci number sequence and determine the time complexity of the algorithm using the frequency method.
   c) Write a recursive algorithm to solve Towers of Hanoi problem with an example.
3) a) Describe the asymptotic notations used for algorithm analysis with at least three examples for each.
b) Write a recursive algorithm to find the sum of first n integers and Derive its time complexity.

4) a) Describe the Pseudo code conventions for specifying algorithms of recursive and an iterative algorithm to compute n!
b) Write recursive binary search algorithm with example.
5 a) Show the result of running Quick sorting technique on the sequence
38, 27, 43, 3, 9, 82, 10
b) Derive the Best, Worst and Average time complexities of Quick sortingtechnique.
7 a) Determine the number of passes required to search the element 44 in the following list of elements
5, 12, 17, 23, 38, 44, 77, 84, 90
b) Write the Binary search algorithm and analyze for its best, worst andaverage case time complexity.
8) a) Write algorithm for finding maximum and minimum values in a list and derive its time complexity.
b) Write iterative binary search algorithm with example.

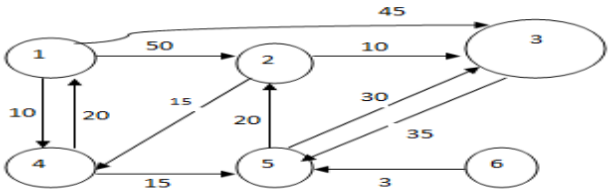
UNIT II

1. Use an algorithm for greedy strategies for the knapsack to find an optimal solution to the knapsack instance n=7,m=15,$(p1,p2\ldots.,p7)=($10,5,15,7,6,18,3),and $(w1,w2,\ldots w7)=$(2,3,5,7,1,4,1).

2. Write greedy algorithm for finding optimal solution of fractional knapsack problem.

3. Apply greedy algorithm to generate single-source shortest path with an examplegraph. Mention its time complexity.
4. Write algorithm for Optimal Merge Patterns and explain with an example

6. Find optimal merge pattern for a given set of 6 files and their records 5,3,2,7,9,13 respectively.
7. Find shortest paths from source vertex 1 in the following graph using Dijkstra's algorithm?



UNIT III

1. Describe the Dynamic 0/1 Knapsack Problem. Find an optimal solution for the dynamic programming 0/1 knapsack instance for n=3, m=6, profits are (p1, p2, p3) = (1,2,5), weights are (w1,w2,w3)=(2,3,4).
2. Describe All Pairs Shortest Path problem. Apply dynamic programming to find shortest paths between all pairs of nodes in the following graph

∞	4	5	∞	3
5	∞	6	7	∞
6	∞	∞	3	8
10	8	∞	∞	∞
∞	6	∞	9	∞

3. Describe Single Source Shortest Path problem. Apply dynamic programming to find shortest paths in the following graph

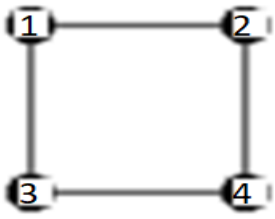
∞	4	5	∞	-3
5	∞	6	-2	∞
-3	7	∞	3	8
10	8	∞	5	6
∞	6	∞	9	∞

4. Describe String Editing problem. Find minimum cost sequence to transform string X= aababb into string Y= babaa. Costs of Insert, Delete and Change operations are 2, 1 and 2 respectively.

5. Compare and contrast Greedy approach and Dynamic Programming approach.
6. Explain traveling sales person problem and solve the problem using dynamic programming method using any example.

UNIT IV

1. State n-queens problem and Explain 8-queens problem using backtracking.
2. Discuss the 4 – queen's problem. Draw the portion of the state space tree for n = 4 queens using backtracking algorithm.
3. Write control abstraction for backtracking.
4. Solve the following instance for the subset problem s=(1,3,4,5) and d=11using backtracking technique.
5. Explain the solution to the graph coloring problem using backtracking.
6. Explain the Graph – coloring problem and Construct the state space tree for m= 3 colors n=4 vertices graph. Discuss the time and space complexity.



7. Write an algorithm to determine the Hamiltonian cycle using a graph by using backtracking.

8. State the sum –of subsets problem. Find all sum of subsets for n=4, (w1, w2, w3, w4) = (11, 13, 24, 7) and M=31. Construct the portion of the state space tree using fixed – tuple sized approach.

UNIT V

1. State the concept of branch and bound method and also list its applications.
2. Solve the Travelling Salesman problem using branch and bound algorithms.
3. Distinguish between backtracking and branch – and bound techniques.
4. Distinguish between FIFO and LC branch and bound solutions.
5. Explain the Travelling sales person problem using LCBB procedure with the following instance and draw the portion of the state space tree and find an optimal tour.

∞	11	10	9	6
8	∞	7	3	4
8	4	∞	4	8
11	10	5	∞	5
6	9	5	5	∞

6. Explain how branch and bound technique is used to solve 0/1 knapsack problem.
7. State the 0/1 Knapsack LCBB algorithm. Explain how to find optimal solution.

8. Construct the portion of state space tree generated by LCBB for the 0/1 Knapsack instance: n = 5, (p1,p2,…,p5) = (10,15,6,8,4), (w1,w2,..,w5) = (4,6,3,4,2) and m=12. Find an optimal solution using fixed – tuple sized approach.
9. Explain the FIFO BB 0/1 Knapsack problem procedure with the knapsack instance for n=4.m=15,(p1,p2,p3,p4)=(10,10,12,18) (w1,w2,w3,w4) =(2, 4, 6,9). Draw the portion of the state space tree and find optimal solution.
10. Compare and contrast about NP Hard and NP Complete Problems.