Registration No.: | | | | | | | | | | |

Total Number of Pages: 02 **Course: IDD (B.Tech and M.Tech)**
Sub_Code: RCS4C002/RIT4C002

4th Semester Back Examination: 2024-25
SUBJECT: Design and Analysis of Algorithm
BRANCH(S): CSE, CSEAI, CSEAIML, CSEDS, CST, ELECTRICAL & C.E, ELECTRONICS & C.E, IT
Time: 3 Hours
Max Marks: 100
Q.Code: S428

Answer Question No.1 (Part-I) which is compulsory, any eight from Part-II and any two from Part-III.
The figures in the right hand margin indicate marks.

Part-I

Q1 **Answer the following questions:** **(2 x 10)**

a) Which one is bigger n^{100} or 2^n? Justify your answer.
b) What will be the time complexity of merge sort if all the elements are already arranged in increasing order?
c) Differentiate between Dynamic Programming and Greedy methods.
d) Which one is better in terms of space complexity, Quick sort or Merge sort? Justify your answer
e) Differentiate between max heapify and min heapify?
f) Define divide and conquer approach. Mention two examples.
g) If the graph contains 6 nodes, then how many maximum spanning tree is possible?
h) What are the constraints required for a Backtracking method?
i) Define time complexity and space complexity of an algorithm.
j) Determine whether the following problems are P, NP, or NP-Complete, Satisfiability problem, Hamiltonian cycle, TSP problem, Knapsack problem, Clique, Set partitioning.

Part-II

Q2 **Only Focused-Short Answer Type Questions- (Answer Any Eight out of Twelve)** **(6 x 8)**

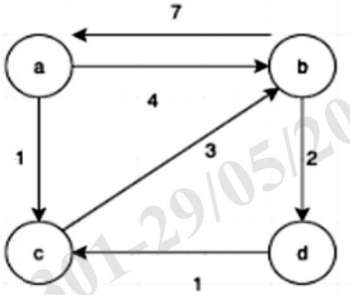
a) Explain how the LIFO Branch and Bound technique works to solve the 8-puzzle problem. Explain by constructing the state space tree.
b) Solve the following recurrence relations by using Master's method.
$T(n)= 2T(n/2) +n^3$
$T(n)=16T(n/4) +n$
c) Write an algorithm for Binary search and analyse its Best, Worst, and Average case time complexity.
d) Explain topological sorting with suitable example.
e) Define recursion. Explain recursion tree method to solve recursion.
f) Solve the knapsack problem for n = 4, W = 40, (p1, p2, p3, p4) = (11, 21, 31, 33)
(w1, w2, w3, w4) = (2, 11, 22, 15)
g) Define Big-Oh and Big-omega notation. Find Big-Oh for the function $f(n) = 4n^2 + 2n + 7$
h) Explain dis-joint set. Write the operation supported by dis-joint set.

i) Differentiated between Prim's and Kruskal algorithm. Let G be the complete undirected graph on 4 vertices having 6 edges weights 1, 2, 3, 4, 5, 6. What is the maximum possible weight that a minimum spanning tree of G can have.

j) Apply the Bottom up Dynamic Approach to the following instance of 0/1 knapsack problem. The maximum capacity of the knapsack is 10.

Item	Weight	Profit
1	7	42
2	3	12
3	4	40
4	5	25

k) Describe Floyd Warshall algorithm for all pairs shortest path problem. Apply the same to the following Graph and discuss its time complexity.



l) Explain MERGE SORT for the given elements: 27, 18, 19, 14, 16, 23, 21, 12

Part-III

Only Long Answer Type Questions (Answer Any Two out of Four) **(16 x 2)**

Q3 **a)** Create max-heap for following elements: 3, 2, 4, 5, 1, 6, 7. Draw diagrams by inserting one element at a time. Find the time complexity if heap is created with n elements by using insertion process. **(8)**

b) Find Huffman code for a - 4, b - 5, c - 7, d - 8, e - 10, f - 12, g – 20. **(8)**

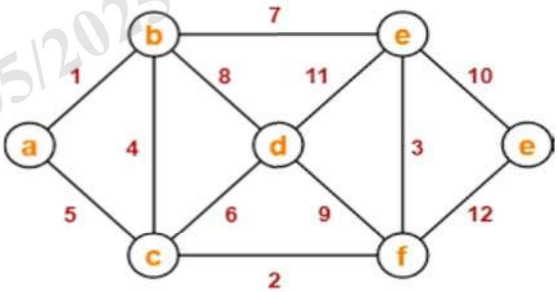
Q4 **a)** Write an algorithm for Activity selection problem. Solve the problem where start times = (1, 3, 0, 5, 3, 5, 6, 8, 8, 2, 12) and Finish times = (4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14). **(8)**

b) Find LCS for X = ABABDC and Y = BCBCABAD. Write an algorithm for LCS. **(8)**

Q5 **a)** Discuss the relation between P, NP, NP-complete and NP-Hard problem with suitable example. **(8)**

b) Discuss the 4 - queen's problem. Draw the portion of the state space tree for n = 4 queens using backtracking algorithm. **(8)**

Q6 **a)** **(8)**



Explain how Kruskal's algorithm can be used to find minimum spanning tree of a graph given above.

b) Find an optimal parenthesization of a matrix chain multiplication whose sequence of dimensions (orders) are < 10, 20, 50, 1, 100 >. **(8)**