BACHELOR OF TECHNOLOGY (CBCS) (2021-COURSE)

B. Tech. Sem - V COMPUTER SCIENCE & BUSINESS SYSTEMS : SUMMER : 2024

SUBJECT: DESIGN & ANALYSIS OF ALGORITHMS

Day : Wednesday
Date : 08/05/2024

S-24166-2024

Time : 10:00 AM-01:00 PM
Max. Marks : 60

N.B.:

1) All questions are **COMPULSORY.**
2) Figures to the right indicate FULL marks.
3) Use of **CALCULATOR** is allowed.

Q.1 Explain Asymptotic Notation with suitable diagram. **(10)**

1) Theta (Θ)
2) Big-Oh (O)
3) Big-Omega (Ω)

OR

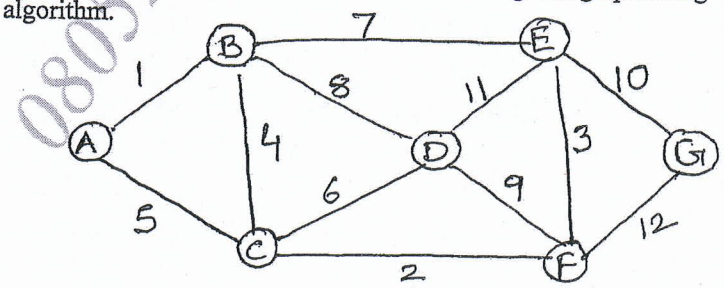
Q.1 Explain in detail what an algorithm is with suitable example. Also explain the characteristics of an algorithm. **(10)**

Q.2 What is a Fractional Knapsack problem and how to solve it using Greedy Strategy? Solve the numerical to find the maximum profit where the capacity of the knapsack M/W is 6. **(10)**

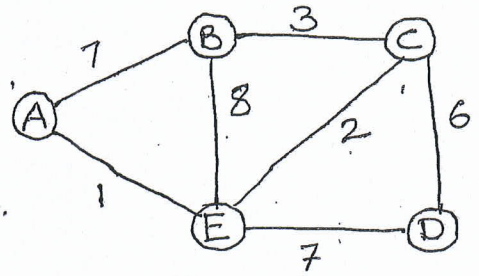
Item Number	1	2	3	4
Profit (P_i)	15	20	30	40
Weight (W_i)	3	2	5	2

OR

Q.2 Differentiate between Prim's and Kruskal's Minimum Spanning Tree (MST) algorithms. Find out the cost of the MST for the given graph using Kruskal's algorithm. **(10)**



Q.3 Write an algorithm to find shortest path in a graph from a single source that uses Dynamic Programming. Discuss the running time of that algorithm. Find out the shortest path in the given graph considering A as the source vertex. **(10)**

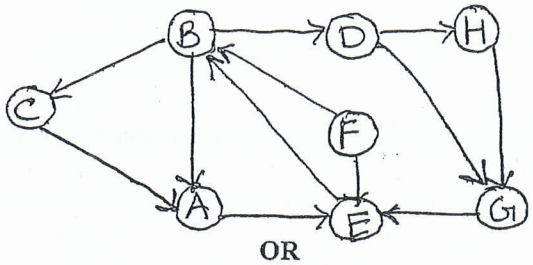


OR

Q.3 Explain Backtracking strategy using 4 Queen's problems **(10)**

P.T.O.

Q.4 Write an algorithm for Depth First Search Graph Traversal. Discuss running time. Apply on the given graph. **(10)**



OR

Q.4 a) Explain with suitable example: Topological Sort. **(05)**

b) Explain with suitable example: Transitive Closure of a graph. **(05)**

Q.5 Explain Polynomial Time Verification and Polynomial Time Reduction with suitable example. **(10)**

OR

Q.5 Explain the P class, NP class, NP-Hard and NP-Complete problems. Prove that Clique is NP-Complete. **(10)**

Q.6 Explain Randomized algorithms in detail using a suitable example. **(10)**

OR

Q.6 How do the complexity classes NP and P-SPACE relate to each other in terms of computational complexity? Some problems are known to be in NP but not known to be in P-SPACE and vice versa. Justify. **(10)**

* * * * *