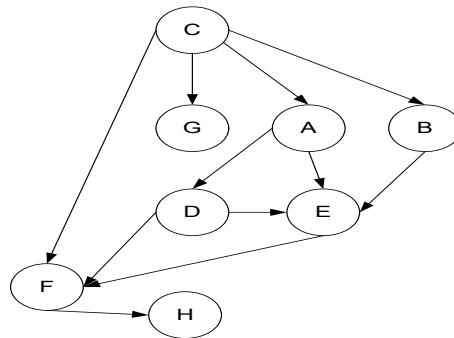


GUJARAT TECHNOLOGICAL UNIVERSITY**BE - SEMESTER-V EXAMINATION – SUMMER 2025****Subject Code:3150703****Date:28-05-2025****Subject Name:Analysis and Design of Algorithms****Time:02:30 PM TO 05:00 PM****Total Marks:70****Instructions:**

1. Attempt all questions.
2. Make suitable assumptions wherever necessary.
3. Figures to the right indicate full marks.
4. Simple and non-programmable scientific calculators are allowed.

		MARKS
Q.1	(a) Find Big-oh (O) notation of following functions $f(n)=3n^6 + 2n \lg n + 6n$, $h(n)= 3n^2+ 6n \lg n+ 2n^{2.5}$	03
	(b) Define Omega and Theta notations with graph.	04
	(c) Write binary search algorithm and analyze it for worst case time complexity. Represent its time complexity using Big-oh (O) notation.	07
Q.2	(a) Prove that $(n + 6)^3 = \Theta(n^3)$.	03
	(b) If $P(n) = a_0 + a_1 n + a_2 n^2 + \dots + a_m n^m$ then prove that $P(n) = \Omega(n^m)$. Here $a_0, a_1, a_2, \dots, a_m$ are constants and $a_m > 0$.	04
	(c) Solve following recurrence relation using suitable method and express your answer using Big-oh (O) notation. $T(n) = T(n/6) + T(5n/6) + \Theta(n)$	07
OR		
	(c) Solve following recurrence relations using suitable method and express your answer using Big-oh (O) notation. 1. $T(n) = 2 T(n/2) + n \log n$ 2. $T(n) = 4 T(n/4) + n$	07
Q.3	(a) Arrange the following growth rates from the lowest to highest asymptotic order: $O(n \log n)$, $O(n^{2n})$, $\Omega(\log n)$, $O(n^{0.5})$, $O(n!)$, $\Omega(2^n)$, $O(n^{0.5} \log n)$	03
	(b) Write the recurrence relation for the quick sort on input instance: 15, 19, 20, 25, 32, 37, 50, 62, 70. Comment on the nature of input i.e. best case, average case or worst case.	04
	(c) Write greedy algorithm for activity selection problem. Give its time complexity. For following intervals, select the activities according to your algorithm. $I_1 (1-3)$, $I_2 (0-2)$, $I_3 (3-6)$, $I_4 (2-5)$, $I_5 (5-8)$, $I_6 (3-10)$, $I_7 (7-9)$.	07
OR		
Q.3	(a) Prove that $\text{Log}(\sqrt{n}) = O(\log n)$.	03
	(b) Write any algorithm (need not to have meaningful) which has a time complexity of $O(1)$ and also includes a loop statement. Also, confirm its time complexity of $O(1)$ using tabular method.	04

- (c) Write DFS traversal and draw DFS tree corresponds to following graph. Also, write time complexity of DFS algorithm. **07**



- Q.4** (a) What is Principle of Optimality in dynamic programming? Explain it with example. **03**
- (b) What is knapsack problem? Using greedy algorithm find an optimal solution for fractional knapsack instance $n=7$, $M=17$, profits $p[] = \{10,5,15,7,6,16,4\}$ and weights $w[] = \{2,3,5,7,1,4,1\}$. **04**
- (c) Find the optimal way of multiplying following matrices using dynamic programming. Also indicate optimal number of multiplications required. **07**
 A: 2×3 , B: 3×5 , C: 5×6 , D: 6×2 , E: 2×3

OR

- Q.4** (a) What is the Travelling Salesman Problem (TSP)? Why TSP is considered an NP-hard problem? **03**
- (b) Illustrate the working of Kruskal's algorithm using greedy technique. Take the suitable graph for illustration. **04**
- (c) Find the longest common subsequence for the following two sequences using dynamic programming. Show the complete process and indicate the time complexity. **07**
 X = 10101010011
 Y = 101001

- Q.5** (a) Define P and NP problems. Also give example of each type of problem. **03**
- (b) Explain the backtracking strategy to solve optimization problem. Explain how it can be applied to find the solution for 4-queen problem. **04**
- (c) Given a text and a pattern, write the naive string matching algorithm to find all occurrences of the pattern in the text. Use the text "ABCABCABC" and the pattern "ABC". Also, give time complexity of naive string matching algorithm. **07**

OR

- Q.5** (a) What is a polynomial reduction? Explain its role in proving NP-completeness. **03**
- (b) Give example of any NP complete problem. Explain how to prove that particular problem is NP complete. **04**
- (c) Show that Hamiltonian cycle is a NP problem. **07**

GUJARAT TECHNOLOGICAL UNIVERSITY

BE - SEMESTER-V (NEW) EXAMINATION – SUMMER 2024

Subject Code:3150703

Date:31-05-2024

Subject Name:Analysis and Design of Algorithms

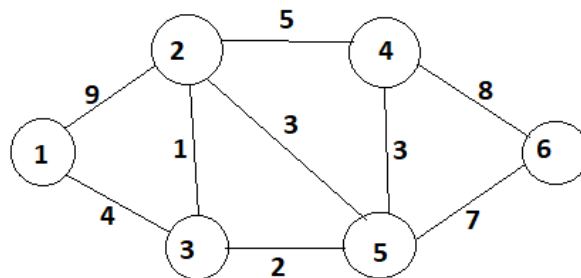
Time:02:30 PM TO 05:00 PM

Total Marks:70

Instructions:

1. Attempt all questions.
2. Make suitable assumptions wherever necessary.
3. Figures to the right indicate full marks.
4. Simple and non-programmable scientific calculators are allowed.

		MARKS
Q.1	(a) Find Omega (Ω) notation of function $f(n)=2n^2 + n * \lg n + 6n$	03
	(b) Define Big-oh and Theta notations with graph	04
	(c) Write and analyze an insertion sort algorithm to sort n items into ascending order.	07
Q.2	(a) Explain: Articulation Point, Graph, Tree	03
	(b) Solve following recurrence using Master Theorem: $T(n) = 3T(n/3) + n^3$.	04
	(c) Illustrate the working of the quick sort on input instance: 25, 29, 30, 35, 42, 47, 50, 52, 60. Comment on the nature of input i.e. best case, average case or worst case.	07
OR		
	(c) What is recurrence? Explain recursion-tree method with suitable example.	07
Q.3	(a) What is Principle of Optimality? Explain its use in Dynamic Programming Method	03
	(b) Explain Binomial Coefficient algorithm using dynamic programming.	04
	(c) Obtain longest common subsequence using dynamic programming. Given A = "acabaca" and B = "bacac".	07
OR		
Q.3	(a) Explain the steps of greedy strategy for solving a problem	03
	(b) Demonstrate Binary Search method to search Key = 14, form the array $A=<2,4,7,8,9,10,12,14,18>$	04
	(c) What is a minimum spanning tree? Draw the minimum spanning tree correspond to following graph using Prim's algorithm.	07

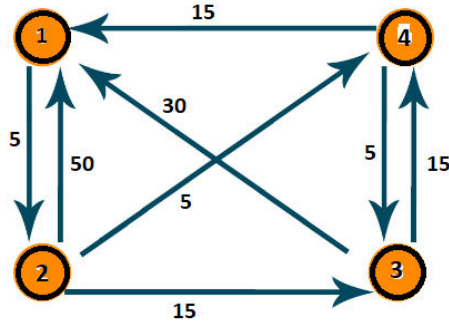


Q.4	(a) Explain Tower of Hanoi Problem, Derive its recursion equation and computer it's time complexity.	03
	(b) Using greedy algorithm find an optimal schedule for following jobs with $n=7$ profits: (P1, P2, P3, P4, P5, P6, P7) = (3, 5, 18, 20, 6, 1, 38) and deadline (d1, d2, d3, d4, d5, d6, d7) = (1, 3, 3, 4, 1, 2, 1)	04

- (c) Solve the following Knapsack Problem using greedy method. Number of items = 5, knapsack capacity $W = 100$, weight vector = $\{50, 40, 30, 20, 10\}$ and profit vector = $\{1, 2, 3, 4, 5\}$. **07**

OR

- Q.4** (a) What are the disadvantages of greedy method over dynamic programming method? **03**
 (b) Find an optimal Huffman code for the following set of frequency. A : 50, b: 20, c: 15, d: 30 **04**
 (c) Consider a Knapsack with the maximum weight capacity M is 7, for the objects with value $[12,10,20,15]$ with weights $[2,1,3,2]$ solve using dynamic programming the maximum value the knapsack can have. **07**
- Q.5** (a) Define NP-Complete and NP-Hard problems. **03**
 (b) Draw the state space tree Diagram for 4 Queens problem. **04**
 (c) Find all pair of shortest path using Floyd's Algorithm for following graph. **07**



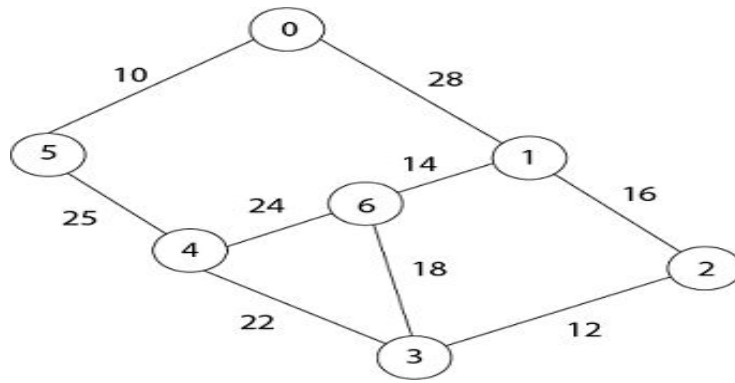
OR

- Q.5** (a) Define BFS. How it is differ from DFS. **03**
 (b) Explain the naive string matching algorithm. **04**
 (c) Explain spurious hits in Rabin-Karp string matching algorithm with example. Working modulo $q=13$, how many spurious hits does the Rabin Karp matcher encounter in the text $T = 2359023141526739921$ when looking for the pattern $P = 31415$? **07**

GUJARAT TECHNOLOGICAL UNIVERSITY**BE – SEMESTER- V EXAMINATION-SUMMER 2023****Subject Code: 3150703****Date: 03/07/2023****Subject Name: Analysis and Design of Algorithms****Time: 02:30 PM TO 05:00 PM****Total Marks: 70****Instructions:**

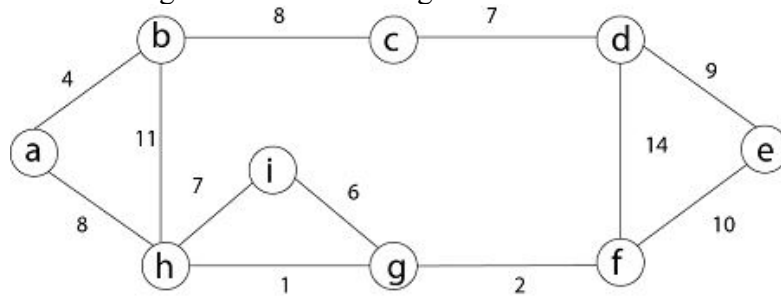
1. Attempt all questions.
2. Make suitable assumptions wherever necessary.
3. Figures to the right indicate full marks.
4. Simple and non-programmable scientific calculators are allowed.

		MARKS
Q.1	(a) Define following terms: (i) Big O Notation, (ii) Big Theta Notation, (iii) Big Omega Notation.	03
	(b) Perform Bucket sort for following sequence: 30, 12, 22, 66, 48, 27, 35, 43, 47, 41.	04
	(c) Explain the bubble sort algorithm and derive its best case, worst case, and average case time complexity.	07
Q.2	(a) Define Algorithms and characteristics of algorithms.	03
	(b) What is a recurrence? Solve recurrence equation for $T(n) = T(n-1) + 1$ using substitution method.	04
	(c) Discuss Binary search algorithm, also write and solve its recurrence relation.	07
OR		
	(c) Explain Merge Sort algorithm with suitable example.	07
Q.3	(a) Explain principle of optimality with suitable example.	03
	(b) Explain advantages and disadvantages of dynamic programming.	04
	(c) Given the denominations: $d_1=1, d_2=4, d_3=6$. Calculate for making change of Rs. 8 using dynamic programming.	07
OR		
Q.3	(a) Explain Weighted Graph, Undirected Graph, Directed Graph.	03
	(b) Discuss advantages and disadvantages of greedy algorithm.	04
	(c) Consider weights $w=(3,4,6,5)$ and profit $v=(2,3,1,4)$ and Knapsack capacity $W=8$. Find the maximum profit using dynamic approach.	07
Q.4	(a) Find an optimal Huffman code for the following set of frequency. a : 40, b: 20, c: 15, d: 30, e: 10.	03
	(b) Explain depth first traversal using suitable example.	04
	(c) Draw the minimum spanning tree correspond to following graph using Prim's algorithm and find the MST weight:	07



OR

- Q.4** (a) Differentiate between Kruskal's algorithm and Prim's algorithm for finding MST. **03**
 (b) Explain the need of topological Sort with example. **04**
 (c) Draw the minimum spanning tree correspond to following graph using Kruskal's algorithm and find weight of MST: **07**



- Q.5** (a) Explain Spurious hits with an example. **03**
 (b) Write the pseudocode for Naïve String-Matching Algorithm. **04**
 (c) What is state space tree. How do you solve the Eight queens problem using backtracking with the help of state space tree. **07**

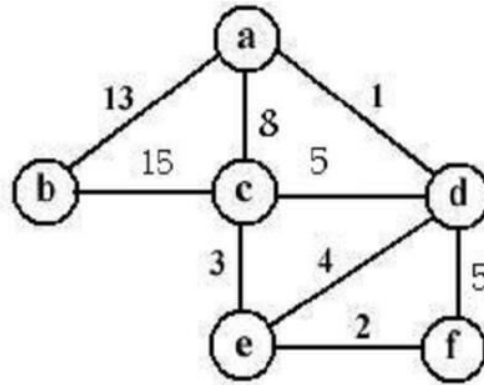
OR

- Q.5** (a) Explain polynomial time reduction. **03**
 (b) Differentiate between Backtracking and Branch-and-Bound algorithms. **04**
 (c) Define P, NP, NP complete and NP-Hard problems. Give examples of each. **07**

GUJARAT TECHNOLOGICAL UNIVERSITY**BE - SEMESTER-V(NEW) EXAMINATION – SUMMER 2022****Subject Code:3150703****Date:07/06/2022****Subject Name:Analysis and Design of Algorithms****Time:02:30 PM TO 05:00 PM****Total Marks: 70****Instructions:**

1. Attempt all questions.
2. Make suitable assumptions wherever necessary.
3. Figures to the right indicate full marks.
4. Simple and non-programmable scientific calculators are allowed.

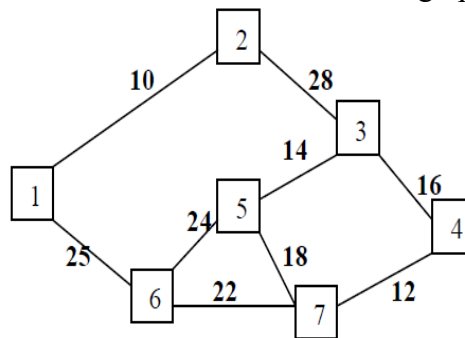
		Marks
Q.1	(a) Define Algorithm, Time Complexity and Space Complexity	03
	(b) Explain: Worst Case, Best Case and Average Case Complexity with suitable example.	04
	(c) Sort the following list using quick sort algorithm:< 5, 3 ,8 ,1 ,4 ,6 ,2 ,7 > Also write Worst and Best case and Average case of quick sort algorithm.	07
Q.2	(a) Write an algorithm of Selection Sort Method.	03
	(b) Demonstrate Binary Search method to search Key = 14, form the array A=<2,4,7,8,10,13,14,60>	04
	(c) Write the Master theorem. Solve following recurrence using it. (i) $T(n) = T(n/2) + 1$ (ii) $T(n) = 2T(n/2) + n \log n$	07
	OR	
	(c) Solve following recurrence relation using iterative method $T(n) = T(n - 1) + 1$ with $T(0) = 0$ as initial condition. Also find big oh notation	07
Q.3	(a) What is Principle of Optimality? Explain its use in Dynamic Programming Method	03
	(b) Find out LCS of $A = \{K, A, N, D, L, A, P\}$ and $B = \{A, N, D, L\}$	04
	(c) Discuss Assembly Line Scheduling problem using dynamic programming with example.	07
	OR	
Q.3	(a) Give the characteristics of Greedy Algorithms	03
	(b) Give difference between greedy approach and dynamic programming.	04
	(c) Consider Knapsack capacity $W=15$, $w = (4, 5, 6, 3)$ and $v=(10, 15, 12, 8)$ find the maximum profit using greedy method.	07
Q.4	(a) Explain: Articulation Point, Graph, Tree	03
	(b) Find Minimum Spanning Tree for the given graph using Prim's Algorithm.	04



- (c) Explain Breath First Traversal Method for Graph with algorithm with example. 07

OR

- Q.4** (a) Explain Huffman code with Example. 03
 (b) Write the Kruskal's Algorithm to find out Minimum Spanning Tree. Apply the same and find MST for the graph given below 04



- (c) Explain fractional knapsack problem with example. 07

- Q.5** (a) What is string-matching problem? Define valid shift and invalid shift. 03
 (b) Define P, NP, NP-Hard and NP-Complete Problem 04
 (c) Explain Backtracking Method. What is N-Queens Problem? Give solution of 4- Queens Problem using Backtracking Method. 07

OR

- Q.5** (a) Explain "P = NP ?" problem. 03
 (b) Explain Minimax principal. 04
 (c) What is Finite Automata? Explain use of finite automata for string matching with suitable example. 07