Final Assessment Test (FAT) - May 2024

Programme	**B.Tech.**	Semester	**WINTER SEMESTER 2023 - 24**
Course Title	**DESIGN AND ANALYSIS OF ALGORITHMS**	Course Code	**BCSE204L**
Faculty Name	**Prof. Jayaram B**	Slot	**A1+TA1**
		Class Nbr	**CH2023240502391**
Time	**3 Hours**	Max. Marks	**100**

General Instructions:

- Write only Register Number in the Question Paper where space is provided (right-side at the top) & do not write any other details.
- If any assumptions are required, assume the same and mention those assumptions in the answer script.
- Use of intelligence is highly appreciated.
- Your answer for all the questions should have both the 'design' component and the 'analysis component'
- The 'Design' component should consist: understanding of the problem, logic to develop the pseudocode, illustration, pseudocode.
- The 'Analysis' component should consist: Computation of T(n), Time-complexity.

Section - I

Answer all questions (4 X 10 Marks = 40 Marks)

01. Let n be a positive integer. Let $\Sigma_1 = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, \ldots n^2\}$. $\Sigma_2 = \{a, b, c, \ldots, z\}$. [10]
Consider a grid of size $n \times n$, with n rows and n columns
Given a grid of size $n \times n$, design a backtracking-based pseudocode to fill the cells of the grid with elements from $\Sigma_1 \cup \Sigma_2$ such that (i) all rows start and end with elements from Σ_2 (ii) all columns start and end with elements from Σ_1 (iii) Condition (i) and Condition (ii) is exempted for the first row, last row, first column and the last column of the grid (iv) No two rows, No two columns, No two diagonals have any repetition of elements from $\Sigma_1 \cup \Sigma_2$.
Your design component should contain all the required steps and the analysis component should contain all the required components.

[**Rubrics:** Logic(2 marks), Illustration (3 marks), Pseudocode (3 marks), Running time & Time-complexity (2 marks)]

02. Given a graph $G = (V, E, s, c, t)$, where V is the set of vertices, E is the set of edges, $s \in V$ is a vertex designated as a source vertex, c is the set of capacities of all the edges of G, t is a vertex designated as the target vertex. Let $|f|$ denote the maximum flow of the network G with s as the source vertex and t as the target vertex. **Given the graph $G = (V, E, s, c, t)$, design a pseudocode to compute the maximum flow from s to t such that the total number of** [10]

augmenting paths computed in the pseudocode to calculate $|f|$, should not be greater than the number of edges in the graph. Your design component should contain all the required steps. Analyze the pseudocode with all the required steps.

[**Rubrics**: Logic(2 marks), Illustration (3 marks), Pseudocode (3 marks), Running time & Time-complexity (2 marks)]

03. a) Consider the problem P: Given a positive integer n, the task is to compute $n^{log_{10}n}$. Compute the class-complexity of the problem P (P/NP/NPC) with justification. [5 marks] **[10]**

[**Rubrics** : Identification of class-complexity (2 marks), Justification (3 marks)

b) Consider the problem, **NAVIGATION** : Consider an n- digit number $a_1a_2a_3 \ldots a_n$, with distinct digits. Assume that a_1, a_2,..., a_n form the nodes of an undirected graph G. There will be an edge connecting a_i and a_j in G iff $i < j$. Task is to navigate from a node in G and reach the same node in such a way that, every node is visited only once except the starting and the ending node. As an example, for the number is12347, one possible NAVIGATION path is:
$1 - 2 - 3 - 4 - 7 - 1$.
Compute the class-complexity of the problem **NAVIGATION** (P/NP/NPC) with justification. [5 marks]

04. Let $\Sigma = \{a, b, c, \ldots x, y, z\}$. Let T and T_1 be any two strings with symbols from Σ. T is said to be a cousin of T_1 if T_1 can be obtained from T by shifting all the symbols(except the first symbol) of T one position to left and the first symbol of T is shifted to the rightmost position, or viceversa. The string $abcdef$ is a cousin of $bcdefa$. **Given two strings T, T_1, design a pseudocode to decide whether T and T_1 are cousins or not. Your pseudocode should use any of the string-matching pseudocodes discussed in the class.** Your design component should contain all the required steps. Analyze the pseudocode with all the required steps. [10 marks] **[10]**

[**Rubrics**: Logic(2 marks), Illustration (3 marks), Pseudocode (3 marks), Running time & Time-complexity (2 marks)]

Section - II

Answer all questions (4 X 15 Marks = 60 Marks)

05. Consider the 2-dimensional plane where the points are represented by a pair of integers. An edge with the end-points p_i and p_j is a line represented as $l(p_i, p_j)$, connecting the points p_i and p_j. A polygon $P = \{p_1, p_2, \ldots, p_m\}$ is a set of points that form a closed figure with the edges $p_1 - p_2, p_2 - p_3, p_3 - p_4, \ldots, p_{(m-1)} - p_m, p_m - p_1$. The points $p_1, p_2, \ldots p_m$ are the vertices of the polygon. A polygon is said to be simple if none of the edges of the polygon cross itself. A convex polygon is a simple polygon where, any line segment connecting two points within the polygon will lie entirely inside the polygon or on its boundary. Convex Hull of a set of points is the smallest simple convex polygon that contains all the points in the set. Circumcircle of a convex hull of the points $p_1, p_2, \ldots, p_n$, is the circle that passes through all the vertices of the convex hull. **[15]**

Given n points $\{p_1, p_2, \ldots, p_n\}$ in the 2-dimensional plane, design a pseudocode to compute the radius of the circumcircle of the convex hull of the given n points. Your 'design' component should contain all the required steps and the 'analysis' component should contain all the required steps. [15 marks]

[**Rubrics**: Logic(4 marks), Illustration (4 marks), Pseudocode (4 marks), Running time & Time-complexity (3 marks)]

06. You are organizing a function for which you have invited n guests. Every guest will be picked up from their house and dropped at the venue of the function. [15]

Every car will start from the venue of the function to pick up the guest and drop the guest at the venue. Every guest g_i will have a pair (s_i, d_i), where s_i is the start-time of the car from the venue to pick the guest and the d_i is the drop-time of the guest g_i at the venue. **Given the details (s_i, d_i) of the guests $g_i, i = 1, 2 \ldots, n$, design an algorithm to compute the minimum number of cars to be booked for the purpose.**

For example, If

$g_1 : (8:15, 9:05), g_2 : (8:40, 9:25), g_3 : (9:10, 9:45), g_4 : (9:47, 10:50), g_5 : (9:30, 10:20)$

then minimum of two cars are required. Your `design' component should contain all the required steps and the `analysis' component should contain all the required steps. [15 marks]

[**Rubrics**: Logic(4 marks), Illustration (4 marks), Pseudocode (4 marks), Running time & Time-complexity (3 marks)]

07. Given a chain of n matrices $< A_1, A_2, \ldots, A_n >$ where each matrix A_i is of size $p_{(i-1)} \times p_i, i = 1, 2, 3, \ldots n$. Product of the chain of matrices $< A_1, A_2, \ldots, A_n >$ is the value of the expression $A_1 \times A_2 \times A_3 \ldots \times A_{n-1} \times A_n$. Given an array $[p_0, p_1, p_2, \ldots, p_n]$, **Design a divide-conquer-combine pseudocode to parenthesize the chain $< A_1, A_2, \ldots, A_n >$ such that minimum number of scalar multiplications are involved in the computation of the product of the chain.** Your `design' component should contain all the required steps and the `analysis' component should contain all the required components. [15 marks] [15]

[**Rubrics**: Logic(4 marks), Illustration (4 marks), Pseudocode (4 marks), Running time & Time-complexity (3 marks)]

08. Given an unlimited supply of coins of denominations $x_1, x_2, \ldots, x_n$ and a value v, the task is to make change for the value v. That is, for the given v, we have to find a set of coins whose total value is v. If the denominations are 5 and 10, we can make the change for 15 as 10+5 and we can not make the change for 12. **Given the denominations $x_1, x_2, \ldots, x_n$ and a value v, design an dynamic programming based algorithm to express v using denominations $x_1, x_2, \ldots, x_n$ such that minimum number of coins are required .** In case, if v can not be expressed in terms of the given denominations, your algorithm should return -1. Your `design' component should contain all the required steps and the `analysis' component should contain all the required components. [15 marks] [15]

[**Rubrics**: Logic(4 marks), Illustration (4 marks), Pseudocode (4 marks), Running time & Time-complexity (3 marks)]