VIT

Reg. No. :

Final Assessment Test (FAT) - APRIL/MAY 2023

Programme	B.Tech	Semester	Winter Semester 2022-23
Course Title	DESIGN AND ANALYSIS OF ALGORITHMS	Course Code	BCSE204L
Faculty Name	Prof. K Muthukumaran	Slot	B1+TB1
		Class Nbr	CH2022235000283
Time	3 Hours	Max. Marks	100

1. If any assumptions are required, assume the same and mention those assumptions in the answer script.
2. Use of intelligence is highly appreciated.
3. Your answer for all the questions should have both the 'design' component and the 'analysis component'
4. The 'Design' component should consist: logic to develop the pseudocode, illustration and pseudocode.
5. The 'Analysis' component should consist: Computation of T(n) and Time-complexity.

Section A (4 X 10 Marks)
Answer All questions

01. Let $\Sigma = \{0, 1, 2, 3, \ldots 9\}$. We call 1 is the successor of 0. 2 is the successor of 1, ... and 9 is the successor of 8. A number in which all the digits from the left are in successive order, is called as an LR-Successive number (LRS-number). 789 is an LR-Successive number where as 798 is not an LR-Successive number. Similarly, a number in which all the digits from the right are in successive order, is called as an RL-Successive number (RLS- number). Two numbers are said to be digit-distinct number (dd-number) if there are no digits common between the numbers. 123 and 789 are dd-numbers. [10]

Given a set S that consists of either RLS-numbers or LRS-numbers, design a greedy-based pseudocode to compute $S' \subset S$, with the maximum number of dd-numbers. If $S = \{123, 234, 789, 4567, 98765, 89\}$. $\{123, 4567, 89\}$ shall be one of the subset of S, with maximum number of dd-numbers. Your design component should contain all the required steps. Analyse the algorithm with all the required steps.

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

02. Consider the 2-dimensional plane where the points are represented by a pair of integers. A line with the end-points p_i and p_j is represented as $l(p_i, p_j)$. A polygon $P = \{p_1, p_2, \ldots, p_m\}$ is a set of points that forms a closed figure with the edges $\overline{p_1p_2}, \overline{p_2p_3}, \overline{p_3p_4}, \ldots, \overline{p_{(m-1)}p_m}, \overline{p_mp_1}$. Here $p_1, p_2, \ldots, p_m$ are called the nodes of the polygon. Edges of the polygon can also be called as the line connecting the points. For example, the edge $\overline{p_1p_2}$ can be called as a line l whose end-points are p_1 and p_2. A line connecting the non-adjacent nodes of the polygon is called as the diagonal of the polygon. There will be more than one diagonal for a polygon. [10]

Given a polygon $P = \{p_1, p_2, \ldots, p_m\}$, design a pseudocode to identify the pair of diagonals of the P which intersect. 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. Understand the Algorithm and answer the following questions .

- Algorithm f(A, l, h, s) , A is an array such that A[i] ≥ A[i+1], ∀ i=1 to n [10]

```
- m = (l+h)/2
- If m==A.length
-     If s < A[A.length]:
-         return A.length+1
-     else:
-         return A.length
- If m==1:
-     if s > A[1]:
-         return 1
-     else:
-         return 2
- If s < A[m-1] and s ≥ A[m]:
-         return m
- elseif s <A[m] and s ≥A[m+1]:
-     return m+1
- elseif s < A[m-1] and s < A[m]:
-     return f(A, m+1, h, s)
- elseif s > A[m+1] and s > A[m]:
-     return f(A, l, m-1, s)
```

a. Given an ordered array A = [31, 27, 24, 18, 14, 12], whose position-index starts with 1. Compute the output of the Algorithm f when s= 25, s=6, s=30, s= 33. You are required to give four outputs, one each for a value of s. [4 marks]

b. Describe the functionality of the above algorithm [2 marks]

c. For the array A (given in 4(a)), what shall be the value of s if the output is 4? [2 marks]

d. Compute the time complexity of the algorithm f. [2 marks]

04. a. Consider a problem Q: Given two positive integers N_1 and N_2, the task is to compute all the common digit(s) that occur(s) in N_1, N_2 and $(N_1 * N_2)$ (Here $*$ is the usual multiplication operation) . For example, given two numbers 21 and 12, your algorithm should return 2, since 2 is the digit that occurs in 21, 12 and (252 = 21*12). Identify the complexity-class (P/NP/NPC) of the problem Q with justification. [10]

b. Consider a problem P_1 : Given a set S of integers and an integer t, does there exist a subset S' of S such that the sum of the elements of S' equals t? For example, for the input $S = \{3, 13, 24, 45, 102\}$ and $t = 72$, the solution to the problem is 'yes' since there exists the subset $S' = \{3, 24, 45\}$ whose sum of the elements is 72. Identify the complexity-class (P/NP/NPC) of the problem P_1 with justification.

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

Section B (4 X 15 Marks)

Answer All questions

05. $A_1 * A_2 * A_3 * \ldots * A_n$ is called as a multiplicative chain of length n. where $A_1, A_2, \ldots, A_n$ are matrices. For the three matrices A_1, A_2, A_3,, multiplicative chains are $A_1 * A_2 * A_3$, $A_1 * A_3 * A_2$, $A_2 * A_3 * A_1$, $A_2 * A_1 * A_3$, $A_3 * A_1 * A_2$, $A_3 * A_2 * A_1$. If there is a matrix A such that $A = A_1 * A_2 * A_3 * \ldots * A_n$, then A is called the value of the multiplicative chain. A multiplicative chain of length n is called a Feasible Multiplicative Chain (FMC) if there is a value for that chain. Given three matrices A_1, A_2, A_3 with the respective sizes $(2, 3), (4, 2), (3, 5)$, then the multiplicative chain $A_2 * A_1 * A_3$ is an FMC . $A_1 * A_2 * A_3$ is not an FMC since there is no value for $A_1 * A_2 * A_3$. [15]
Given n matrices $A_1, A_2, A_3, \ldots, A_n$ with the respective sizes $(r_1, c_1), (r_2, c_2), \ldots, (r_n, c_n)$, design a dynamic programming based pseudocode to compute all the FMCs and identify the FMC that requires minimum number of scalar multiplication among all the FMCs. If there are two FMCs with same number of minimum multiplications, your pseudocode should return both. For the given matrices, if no FMC is possible, your pseudocode should return 0. Your design component should contain all the required steps. Analyse the algorithm with all the required steps.
Rubrics: Logic(3 marks), Illustration (5 marks), Pseudocode (5 marks), Running time \& Time-complexity (2 marks)

06. A string Y is said to be an 1-order cyclic rotation of a string X if Y is obtained from X by the process of shifting the last character of X to the first position and shifting all the other characters of X one position to the right. Similarly we define the cyclic rotation of 2-order, 3-order and so on. For example, the 1-order cyclic rotation of the string *abcd* is *dabc*, 2-order cyclic rotation of the string *abcd* is *cdab*, 3-order cyclic rotation of the string *abcd* is *bcda*. [15]
Given two strings S_1 and $S2$, design a brute-force algorithm A to check if $S2$ is a 1-order cyclic rotation of S_1 or vice versa. Also design a string-matching algorithm B (other than Brute-Force) to compute whether S_2 is a 1-order cyclic rotation of S_1 or vice-versa. Your design component should contain all the required steps. Analyze the pseudocode with all the required steps.
Rubrics: Logic(3 marks), Illustration (5 marks), Pseudocode (5 marks), Running time \& Time-complexity (2 marks)

07. 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 identify the edge(s) (u, v) which contributes more to $|f|$. For the purpose of the illustration, you have to take a graph with a minimum of six vertices and a minimum of eight edges. Your design component should contain all the required steps. Analyze the pseudocode with all the required steps. [15]
Rubrics: Logic(3 marks), Illustration (5 marks), Pseudocode (5 marks), Running time \& Time-complexity (2 marks)

08. Consider a grid of size $m \times n$, with m rows and n columns. Every cell is referred with a pair $[i, j]$ which conveys that the cell is at the intersection of the i^{th} row and the j^{th} column. Column-adjacent cells of the cell $[i, j]$ are the cells $[i - 1, j]$, $[i + 1, j]$. Row-adjacent cells of the cell $[i, j]$ are the cells $[i, j - 1]$, $[i, j + 1]$. Diagonal-adjacent cells of $[i, j]$ are $[i - 1, j - 1]$, $[i - 1, j + 1]$, $[i + 1, j - 1]$, $[i + 1, j + 1]$. Two cells are said to be adjacent if the $k \geq 5$ cells are either row-adjacent or column-adjacent or diagonal-adjacent. Given k colours [15]

and an $m \times n$ grid, design a backtracking pseudocode to colour the cells of $,C_1, C_2, C_3, \ldots C_k$ the grid such that no two adjacent cells are coloured same. Your pseudocode should return $m \times n$ array, R such that $R[i, j] = C_j$ if the cell $[i, j]$ is coloured with C_j. Your design component should contain all the required steps. Analyze the pseudocode with all the required steps

Rubrics : Logic for A and B)4 marks(, Illustration for A and B)4 marks(, Pseudocodes : A and B)4 marks(, Running time \& Time-complexity of A and B)3 mark.(