VIT
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

REG.NO.:

SLOT: E1+TE1

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING
CONTINUOUS ASSESSMENT TEST - I
WINTER SEMESTER 2025-2026

Programme Name & Branch : B.Tech. Computer Science and Engineering
Course Code and Course Name : BACSE105 - Data Structures and Algorithms
Faculty Name (s) : GOVINDA K, SIVAPRASAD DARLA, MOHAN KUMAR P, NALLIAH M SUBRAMANYAM REDDY A, RAJA S P, NITHYA N S, ANURADHA D PRAVEEN KUMAR G, ROHINI S, GANESH SHAMRAO KHEKARE KANAGARAJ R, VETRISELVI T, ARUN PRASATH G, SARAVANAN R SARAVANARAJAN M C, RAVITEJA BUDDALA, RHYMEND UTHARIARAJ V, NIVITHA K
Class Number(s) : VL2025260501352, VL2025260501353, VL2025260501356 VL2025260501362, VL2025260501364, VL2025260501370 VL2025260501372, VL2025260501373, VL2025260501375 VL2025260501378, VL2025260501380, VL2025260501382 VL2025260501384, VL2025260501385, VL2025260501396 VL2025260501398, VL2025260501407, VL2025260501427 VL2025260503721
Date of Exam : 31.01.2026
Exam Duration : 90 minutes
Maximum Marks: 50

General instruction(s):

- Answer All Questions
- M - Max mark; CO – Course Outcome; BL – Blooms Taxonomy Level
- Course Outcomes:
- CO1. Design and analyze efficient algorithms using complexity analysis and formal development techniques
- CO2.Implement and utilize linear and non-linear data structures for effective data management and algorithmic problem-solving

Q.No	Questions	M	CO	BL

1.	Consider the following two algorithms.
	`//Algorithm ReverseNumber(n)`
	`rev = 0`
	`while n > 0 do`
	`  rev = rev × 10 + (n mod 10)`
	`  n = n div 10`
	`end while`
	`return rev`
	`// Algorithm Factorial(n)`
	`fact = 1`
	`for i from 1 to n do`
	`  fact = fact × i`
	`end for`
	`return fact`
	(a). Elucidate step by step analysis to attain Computational Complexity & space complexity. (5 marks)
	(b). For suitable input values, trace both algorithms and show their correctness using loop invariants method. **(5 marks)**
		10	1	BL3

2.	**a).** Consider the following recursive algorithm, which computes the product of an integer x by repeatedly adding it n times: Algorithm Multiply(x, n) Input: Two integers x and n, where $n \geq 1$ Output: Product of x and n if n == 1 then return x			

	else
	return x + Multiply(x, n − 1)
	end if
	
	Formulate the **recurrence relation** representing the **computational time** of this algorithm. Also show the correctness of the algorithm using mathematical induction.
		5	1	BL4

	b). Solve the following recurrence relation using the Master Theorem and determine its time complexity: $T(n) = 2\,T(n/4) + n$.	5		
3.	**Develop** an algorithm that performs the following tasks: ➢ Read **n** integer values from the user and push them into a stack **in the same order they are entered.** ➢ After all elements are pushed, read a **key** value from the user. ➢ Pop elements from the stack **one at a time:** o If the popped element is **greater** than the key, enqueue it into **Queue q1.** o If the popped element is **less than or equal** to the key, enqueue it into **Queue q2.** ➢ After all stack elements are processed, display the contents of **q1** and **q2.** Then, **trace** the algorithm step-by-step for the input set: {45,10,75,25,90,35,65,15,85} with the **key value** = 35.	10	2	BL3
4.	**(i).** Describe the concept of a Singly Linked List and explain its internal structure with the help of a neat, clearly labeled diagram. (2 Marks) **(ii).** Develop step-by-step algorithms for the following operations on a singly linked list. Illustrate each operation using an appropriate diagram to clearly show the modification of links: (6 marks) ➢ Insertion of a node at the beginning ➢ Insertion of a node at the middle (after a specified position) ➢ Deletion of a node from the beginning **(iii).** Compare a singly linked list with an array by identifying any two advantages that justify the use of linked lists in dynamic memory management.(2 Marks)	10	2	BL3
5	In an initially empty **Binary Search Tree (BST),** insert the characters one by one from the word **D A T A S T R U C T U R E** (Insert characters in the given order, ignoring repeated occurrences if any.) After constructing the BST, perform the following tasks: i. Obtain the **postorder traversal** of the constructed BST and state your observation. ii. Delete the key **'T'** from the BST. iii. Insert a new key **'B'.** iv. Attempt to insert the key 'S'. Explain why no insertion is performed. Illustrate the structure of the BST before and after each insertion and deletion. Also, present appropriate algorithms for performing the insertion and deletion operations.	10	2	BL4

****