VIT
Vellore Institute of Technology

Final Assessment Test – November 2025

Course: **BITE201L - Data Structures and Algorithms**

Class NBR(s): **3028/3031/3039/3042** Slot: **B2+TB2**

Time: **Three Hours** Max. Marks: **100**

- **KEEPING MOBILE PHONE/ANY ELECTRONIC GADGETS, EVEN IN 'OFF' POSITION IS TREATED AS EXAM MALPRACTICE**
- **DON'T WRITE ANYTHING ON THE QUESTION PAPER**

COs	CO Statements
CO1	Identify user defined data types, linear data structures for solving real world problems.
CO2	Analyse and derive time and space complexity for algorithms designed.
CO3	Illustrate various techniques for searching, sorting and hashing.
CO4	Write modular programs on nonlinear data structures and algorithms for solving engineering problems efficiently.
CO5	Design new algorithms or modify existing algorithms for new applications and reason about the efficiency of the result.

BL – Blooms Taxonomy Level (1 – Remember, 2 – Understand, 3 – Apply, 4 – Analyse, 5 – Evaluate, 6 – Create)

Answer ALL Questions
(10 X 10 = 100 Marks)

1. a) The following expression is given in postfix (Reverse Polish) notation: **[5] CO1 BL2**

   4 2 ^ 3 * 3 – 8 4 / 1 1 + / +

   Evaluate the expression step by step using a stack-based approach. Count and report the total number of PUSH and POP operations performed during the evaluation. Assume all operands are single-digit integers and ^ denotes exponentiation

   b) Why is the head pointer important in a singly linked list? What will happen if the head pointer is lost? Explain the differences between a singly linked list and an array in terms of memory usage. **[5]**

2. A call center uses a circular queue to keep track of customer calls waiting in a line. Assume the queue size = 6. During peak hours, the system must reuse freed slots immediately when calls are answered. Justify why a linear queue would fail in this situation. Write pseudocode to implement insertion and deletion in this scenario using a circular queue, and illustrate with an example. **CO1 BL3**

3. A hospital's emergency room manages patient records using a stack implemented with a linked list. Each patient's record (name, age, emergency_level) is pushed onto the stack when they arrive. If a patient is treated, their record is popped from the stack. Due to emergencies, the most recently arrived patient should always be treated first. CO1 BL3

a) Justify why a stack using an array might fail in this situation, while a stack using a linked list is better.

b) Write pseudocode to implement: PUSH(patient) and POP() using a linked list.

c) Demonstrate the process with an example: Push 3 patients → Pop 1 patient → Show final stack. [5] CO2 BL4

4. a) Find the time complexity for the following code:

(i)
```
for (int i = 0; i < n; i++)
{
for (int j = 0; j < n; j = j + 2)
{
// constant work
}
}
```

(ii)
```
int fib (int n)
{
if (n <= 1)
return n;
return fib(n-1) + fib(n-2);
}
```

b) Consider the following recurrence relation used to analyze the time complexity of the algorithm: [5]

$T(n) = 2\,T(n/2)+n^2$

Use master's theorem to solve the recurrence relation and prove its time complexity.

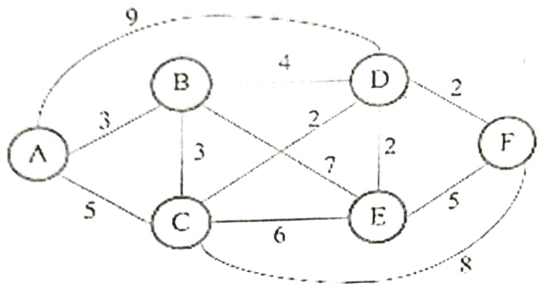
5. Why is stability important in sorting algorithms even if they have the same time complexity? Suppose the array is already sorted. What will be the time complexity of Merge Sort? Can it be optimized to detect this? Provide an algorithm for Merge Sort and demonstrate it with a step-by-step example. CO3 BL2

6. Can two different infix expressions correspond to the same expression tree? Justify your answer. Construct the expression tree for the infix expression using a stack-based method: **CO4 BL3**

4 + 6 * 3 - 8 / 2 + 5 * 2 ^ 3 - 7

Also, derive the Preorder and Postorder traversals of the constructed binary expression tree.

7. Consider the following weighted, undirected graph: **CO4 BL3**



Using Dijkstra's algorithm, find the shortest path from vertex A to all other vertices (B, C, D, E, F). Show all steps, including the distance array and the set of visited vertices at each iteration.

8. A networking company uses a compression technique to encode the message before transmitting over the network. The message contains the following characters with their frequencies: **CO5 BL3**

Character	Frequency
A	50
B	30
C	10
D	5
E	5

Assume each character is stored using 1 byte (8 bits) originally.

a) *Construct the Huffman Tree.* **[2.5]**

b) *Find the average code length.* **[2.5]**

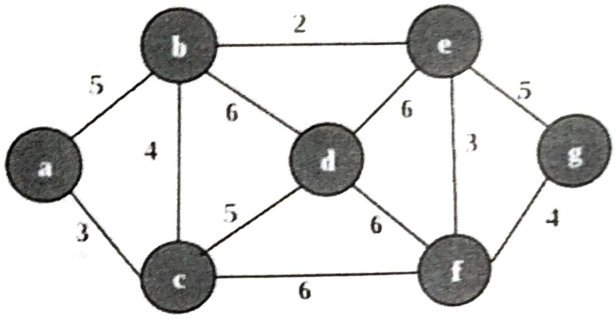
c) *Compute the total number of bits required after compression.* **[2.5]**

d) *Calculate how many bits are saved compared to the original message.* **[2.5]**

9.a) *Analyze the best-case and worst-case time complexities for the searching, insertion, and deletion operations in AVL tree and B-Tree. Construct an AVL Tree by inserting the numbers 1 to 8 sequentially (in order). Show the tree after each insertion and necessary rotations.* **CO4 BL3**

OR

9.b) Explain how a spanning tree differs from an ordinary tree. For a given undirected weighted graph, determine the Minimum Spanning Tree (MST) using Kruskal's algorithm and Prim's algorithm, providing a detailed, step-by-step demonstration of both methods. CO4 BL3



10.a) In real-world applications such as network routing, resource allocation, and scheduling, explain under which circumstances you would apply the following algorithmic paradigms: CO5 BL4

- Greedy Algorithm
- Dynamic Programming (DP)
- Divide and Conquer (D&C)

Provide appropriate justification for each choice.

OR

10.b) The following table shows the distance (in kilometers) between four cities A, B, C, and D. A salesman starting from city A must visit every other city exactly once and return to the starting city. CO5 BL4

From / To	A	B	C	D
A	0	10	15	20
B	10	0	35	25
C	15	35	0	30
D	20	25	30	0

i. Formulate the Travelling Salesman Problem using the Dynamic Programming (DP) approach.

ii. Compute step-by-step the minimum cost tour starting and ending at city A using the DP method.

⇔⇔⇔ T/K/TY⇔⇔⇔