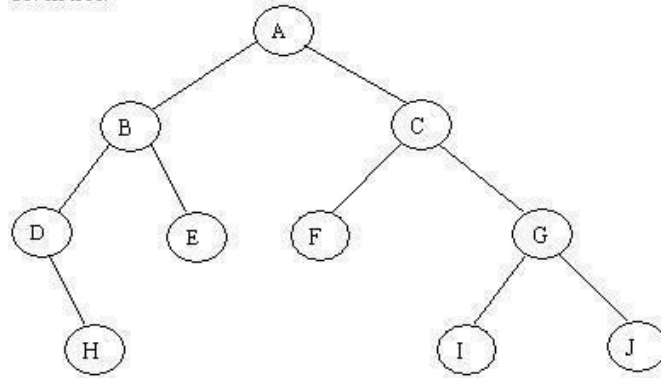


**GUJARAT TECHNOLOGICAL UNIVERSITY****BE- SEMESTER-III EXAMINATION – WINTER 2025****Subject Code:3130702****Date:12-12-2025****Subject Name: Data Structures****Time:10:30 AM TO 01:00 PM****Total Marks:70****Instructions:**

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.

|            |                                                                                                                                                                                                       | <b>Marks</b> |
|------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------|
| <b>Q.1</b> | <b>(a)</b> What is data structure? Differentiate between data types and data structures                                                                                                               | <b>03</b>    |
|            | <b>(b)</b> Define static, dynamic, linear and non-linear data structures.                                                                                                                             | <b>04</b>    |
|            | <b>(c)</b> Explain the concept of recursion and write a recursive C program to calculate factorial of a number.                                                                                       | <b>07</b>    |
| <b>Q.2</b> | <b>(a)</b> Compare and contrast queue, priority queue and dequeue.                                                                                                                                    | <b>03</b>    |
|            | <b>(b)</b> Evaluate the following expression using stack, show each step.<br>$10\ 2 + 8\ 4 / * 6\ 3 - /$                                                                                              | <b>04</b>    |
|            | <b>(c)</b> Write an algorithm to convert an infix expression to postfix.                                                                                                                              | <b>07</b>    |
|            | <b>OR</b>                                                                                                                                                                                             |              |
|            | <b>(c)</b> Write a C functions for insertion and deletion operation in simple queue.                                                                                                                  | <b>07</b>    |
| <b>Q.3</b> | <b>(a)</b> Differentiate Array and Linked lists                                                                                                                                                       | <b>03</b>    |
|            | <b>(b)</b> Convert the following expression to postfix using stack. Show each step clearly<br>$((A-B)+D)/((E+F)*G))$                                                                                  | <b>04</b>    |
|            | <b>(c)</b> Write an algorithm to insert an element in an ordered link list                                                                                                                            | <b>07</b>    |
|            | <b>OR</b>                                                                                                                                                                                             |              |
| <b>Q.3</b> | <b>(a)</b> Define :                                                                                                                                                                                   | <b>03</b>    |
|            | 1. Directed Graph                                                                                                                                                                                     |              |
|            | 2. Acyclic and cyclic graph                                                                                                                                                                           |              |
|            | 3. Weighted and unweighted graph                                                                                                                                                                      |              |
|            | <b>(b)</b> Compare and Contrast DFS and BFS                                                                                                                                                           | <b>04</b>    |
|            | <b>(c)</b> What is minimum cost spanning tree? Discuss Prim's algorithm with an example                                                                                                               | <b>07</b>    |
| <b>Q.4</b> | <b>(a)</b> Explain the properties of binary tree and differentiate between complete binary tree, full binary tree and balanced binary tree                                                            | <b>03</b>    |
|            | <b>(b)</b> Insert the following nodes into an empty BST in this order: 50, 30, 70, 20, 40, 60, 80. Draw the resulting BST after each insertion                                                        | <b>04</b>    |
|            | <b>(c)</b> What is a binary search tree? write an algorithm to search an element in BST and show it's time complexity                                                                                 | <b>07</b>    |
|            | <b>OR</b>                                                                                                                                                                                             |              |
| <b>Q.4</b> | <b>(a)</b> Define following with respect to Tree: i) M-ary tree ii) Out Degree iii) Leaf                                                                                                              | <b>03</b>    |
|            | <b>(b)</b> If the depth of the binary tree is k, the maximum number of nodes in the binary tree is $2^k - 1$ . Justify. Also mention atleast how many nodes a tree should have given it's depth as k. | <b>04</b>    |

Given tree:



- Q.5 (a)** Sort following elements using quick sort. 22 15 26 06 40 31, 92, 45, 78 **03**
- (b)** Compare the below different hashing methods and discuss about each method in a detailed Manner. **04**
- i. Static Hashing. ii. Dynamic Hashing
- (c)** Insert the following keys into a hash table of size 10 using linear probing: 12, 26, 31, 92, 45, 78 Show the steps and the final hash table **07**

**OR**

- Q.5 (a)** Explain AVL tree with suitable example. **03**
- (b)** Construct the 2-3 tree for the following sequence 10, 20, 30, 40, 50, 60, 70, 80, 90. **04**
- (c)** Write an algorithm to sort an array using selection sort method. **07**

\*\*\*\*\*

Enrolment No./Seat No \_\_\_\_\_

## GUJARAT TECHNOLOGICAL UNIVERSITY

BE- SEMESTER-III (NEW) EXAMINATION – WINTER 2024

Subject Code: 3130702

Date: 26-11-2024

Subject Name: Data Structures

Time: 10:30 AM TO 01:00 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     |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------|
| <b>Q.1</b> (a) Differentiate Primitive and Non Primitive Data Structures                                                                                                                                                   | <b>03</b> |
| (b) Derive basic operation of stack and write C function to implement it.                                                                                                                                                  | <b>04</b> |
| (c) Explain Row Major and Column Major with Example.                                                                                                                                                                       | <b>07</b> |
| <b>Q.2</b> (a) What is Sparse Matrix?                                                                                                                                                                                      | <b>03</b> |
| (b) Translate infix expression into its equivalent post fix expression:<br>$A*(B+D)/E-F*(G+H/K)$                                                                                                                           | <b>04</b> |
| (c) Write a program to implement Circular queue and show how it differ from normal queue?                                                                                                                                  | <b>07</b> |
| <b>OR</b>                                                                                                                                                                                                                  |           |
| (c) Explain the concept of Reverse Polish Notation (RPN) and describe the process for evaluating an RPN expression using a stack. Provide a detailed example of evaluating the following RPN expression: 5 1 2 + 4 * + 3 - | <b>07</b> |
| <b>Q.3</b> (a) Write an algorithm for Bubble sort.                                                                                                                                                                         | <b>03</b> |
| (b) Sort 20,35,40,100,3,10,15 using insertion sort. Show all passes.                                                                                                                                                       | <b>04</b> |
| (c) Compare and contrast Depth First Search (DFS) and Breadth First Search (BFS) in terms of their algorithms, uses, and performance.                                                                                      | <b>07</b> |
| <b>OR</b>                                                                                                                                                                                                                  |           |
| <b>Q.3</b> (a) Write an algorithm for Merge Sort.                                                                                                                                                                          | <b>03</b> |
| (b) Sort the given values using Quick Sort? 65, 70, 75, 80, 85, 60, 55, 50, 45. Show all passes.                                                                                                                           | <b>04</b> |
| (c) Describe the concept of hashing and explain different collision resolution techniques, including separate chaining and open addressing. Illustrate how each technique works with examples.                             | <b>07</b> |
| <b>Q.4</b> (a) Define Graph and list any 3 uses of graph.                                                                                                                                                                  | <b>03</b> |
| (b) How to prove array is sequential and contiguous?                                                                                                                                                                       | <b>04</b> |
| (c) Explain the concept of a graph and its representations. Compare adjacency matrix and adjacency list representations in terms of space complexity and use cases.                                                        | <b>07</b> |
| <b>OR</b>                                                                                                                                                                                                                  |           |
| <b>Q.4</b> (a) List any three uses of linked list.                                                                                                                                                                         | <b>03</b> |

- (b) Write an algorithm to delete next element from doubly linked list from given position, **04**
- (c) Write an algorithm to implement singly linked list and its operation such as insert element at front, last and at any position. **07**
- Q.5** (a) What is priority queue? **03**
- (b) Write an algorithm to implement queue using Linked List. **04**
- (c) Explain the concept of a binary search tree (BST) and its properties. Discuss how operations such as insertion, deletion, and searching are performed in a BST. **07**
- OR**
- Q.5** (a) Define (1) Forest (2) Leaf Node (3) Tree **03**
- (b) Construct tree from Following **04**
- **In order Traversal:** D, B, E, A, F, C
  - **Pre order Traversal:** A, B, D, E, C, F
- (c) Discuss the key concepts and techniques of AVL trees and 2-3 trees, focusing on their balance mechanisms and the impact on performance. Provide examples of insertions and deletions to illustrate how these trees maintain balance. **07**

\*\*\*\*\*

**GUJARAT TECHNOLOGICAL UNIVERSITY****BE - SEMESTER-III (NEW) EXAMINATION – WINTER 2023****Subject Code:3130702****Date:16-01-2024****Subject Name:Data Structures****Time:10:30 AM TO 01:00 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     |
|------------|-------------------------------------------------------------------------------------------------------------------------------------------------|-----------|
| <b>Q.1</b> | (a) Define data structure. Enlist and explain types of data structure.                                                                          | <b>03</b> |
|            | (b) Differentiate static and dynamic memory allocation.                                                                                         | <b>04</b> |
|            | (c) Convert the following infix expression into postfix expression using stack.<br>(A-B)/C*D^(E/F)^(G+H)                                        | <b>07</b> |
| <b>Q.2</b> | (a) Compare array and linked list.                                                                                                              | <b>03</b> |
|            | (b) Differentiate primitive and non-primitive data structures.                                                                                  | <b>04</b> |
|            | (c) Define stack. Write algorithms for PUSH, POP and PEEP operations of stack.                                                                  | <b>07</b> |
|            | <b>OR</b>                                                                                                                                       |           |
|            | (c) Define queue. Write algorithms for INSERT, DELETE and DISPLAY operations of queue.                                                          | <b>07</b> |
| <b>Q.3</b> | (a) Enlist and explain any 3 applications of linked list.                                                                                       | <b>03</b> |
|            | (b) Design an algorithm to insert an element in doubly linked list.                                                                             | <b>04</b> |
|            | (c) Do as directed:                                                                                                                             |           |
|            | i) Evaluate the postfix expression in tabular forms: 8 2 / 6 7 * +                                                                              | <b>03</b> |
|            | ii) Explain Tower Of Hanoi with example.                                                                                                        | <b>04</b> |
|            | <b>OR</b>                                                                                                                                       |           |
| <b>Q.3</b> | (a) Enlist and explain any 3 applications of stack in computer science.                                                                         | <b>03</b> |
|            | (b) Design an algorithm to delete an element in doubly linked list.                                                                             | <b>04</b> |
|            | (c) Do as directed:                                                                                                                             |           |
|            | i) Evaluate the prefix expression in tabular forms: * / - 8 2 3 4                                                                               | <b>03</b> |
|            | ii) Explain Sparse matrix with example.                                                                                                         | <b>04</b> |
| <b>Q.4</b> | (a) Explain binary search tree with suitable example.                                                                                           | <b>03</b> |
|            | (b) Construct a binary tree from the traversals given below:<br>In-order : D, B, A, E, G, C, H, F, I<br>Pre-order : A, B, D, C, E, G, F, H, I   | <b>04</b> |
|            | (c) Differentiate DFS and BFS in detail.                                                                                                        | <b>07</b> |
|            | <b>OR</b>                                                                                                                                       |           |
| <b>Q.4</b> | (a) Explain AVL tree with suitable example.                                                                                                     | <b>03</b> |
|            | (b) Construct a binary tree from the traversals given below:<br>Pre-order : 1, 2, 4, 8, 9, 5, 3, 6, 7<br>Post-order : 8, 9, 4, 5, 2, 6, 7, 3, 1 | <b>04</b> |
|            | (c) Explain prim's and krushkal's algorithm with suitable example.                                                                              | <b>07</b> |
| <b>Q.5</b> | (a) State the reason why hash function is used. Give one example of hash function.                                                              | <b>03</b> |
|            | (b) Write a C program for bubble sort.                                                                                                          | <b>04</b> |

- (c) Enlist and explain collision resolution techniques in hashing. **07**
- OR**
- Q.5** (a) Differentiate sequential search and binary search. **03**
- (b) Define file organization. Briefly summarize different file organizations. **04**
- (c) Do as directed below:
- i) Sort following elements using quick sort. **05**  
10 15 28 09 40 35 29 12 06 07
- ii) State time and space complexity of quick sort. **02**

\*\*\*\*\*

**GUJARAT TECHNOLOGICAL UNIVERSITY****BE - SEMESTER– III(NEW) EXAMINATION – WINTER 2022****Subject Code:3130702****Date:22-02-2023****Subject Name:Data Structures****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.

- Q.1** (a) Differentiate primitive and non-primitive data structures **03**  
 (b) Define time complexity? Explain worst case and best case complexity with examples. **04**  
 (c) Convert following infix expression into postfix expression using stack. **07**  
 $(A-B)/C*D^{(E/F)^{(G+H)}}$

- Q.2** (a) What is Sparse matrix? Write efficient vector representation of following Sparse matrix. **03**

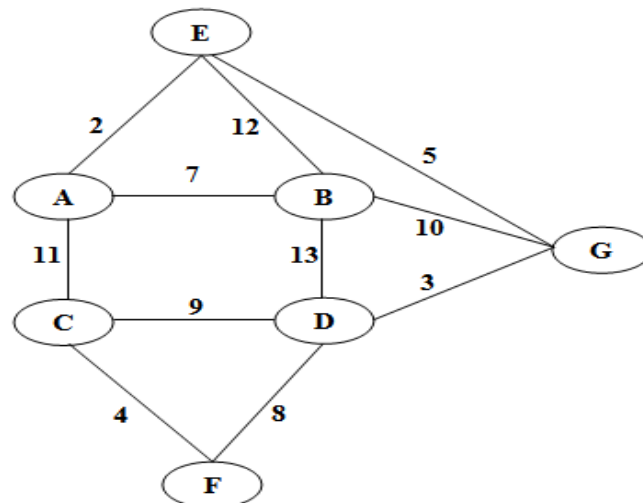
|   |   |   |
|---|---|---|
| 1 | 0 | 0 |
| 0 | 2 | 0 |
| 0 | 0 | 3 |

- (b) What is the worst case complexity of binary search? Write an algorithm for binary search. **04**  
 (c) Create Binary Search Tree for following Data and write pre-order traversal, in-order traversal and post-order traversal of the constructed tree. **07**  
 10 15 28 09 39 31 30 14 07 08

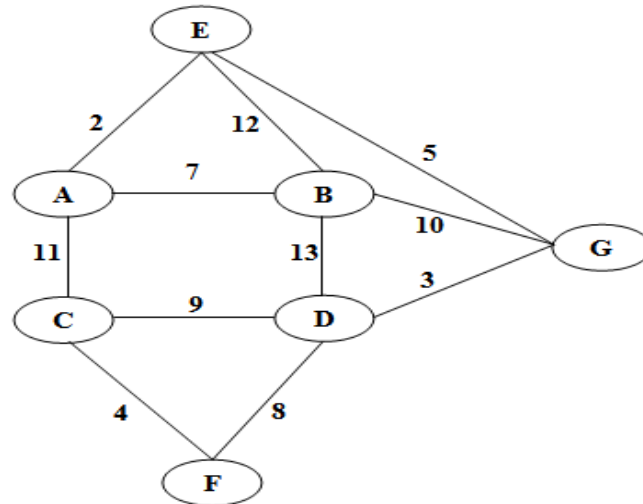
**OR**

- (c) Create AVL tree for following Data and write pre-order traversal of the constructed tree. **07**  
 10 15 28 09 39 31 30 14 07 08

- Q.3** (a) Write c program for bubble sort. **03**  
 (b) Write algorithm to insert into simple queue and mention the limitation of simple queue? **04**  
 (c) Find the minimum spanning tree of following graph using Kruskal's method. **07**

**OR**

- Q.3** (a) Write c program for selection sort. **03**  
 (b) Write algorithm to delete from circular queue and mention the advantage of circular queue over simple queue? **04**  
 (c) Find the minimum spanning tree of the following graph using prim's method. **07**



- Q.4** (a) What is priority queue? Is simple queue is anyhow priority queue? Explain your answer. **03**  
 (b) Write C program to find the Fibonacci sequence of n terms using recursion. **04**  
 (c) Explain DFS and BFS with appropriate example. **07**

**OR**

- Q.4** (a) What is linked list? States the advantages of linked list over array. Also list various types of the linked list. **03**  
 (b) Write recursive solution for tower of Hanoi. How many moves require for transferring three discs? **04**  
 (c) Explain Dijkstra's shortest path algorithm with appropriate example **07**

- Q.5** (a) Define following terms **03**  
     1) Balance factor  
     2) Hash function  
     3) Cyclic graph  
 (b) Sort following data using merge sort. **04**  
     10 15 28 09 39 31 30 14 07 08  
 (c) Define hash collision? Explain collision various resolution techniques. **07**

**OR**

- Q.5** (a) Define following terms **03**  
     1) Hash table  
     2) Graph  
     3) Complete binary tree  
 (b) Sort following data using quick sort. **04**  
     10 15 28 09 39 31 30 14 07 08  
 (c) What is file? Explain various types of file organization. **07**