CBCS SCHEME

USN | | | | | | | | | | |

BCS304

# Third Semester B.E./B.Tech. Degree Examination, June/July 2025
## Data Structures and Applications

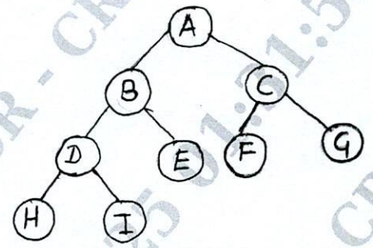
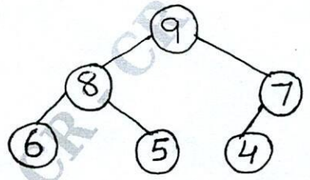
Time: 3 hrs. Max. Marks: 100

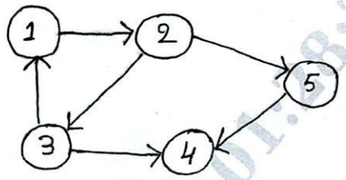
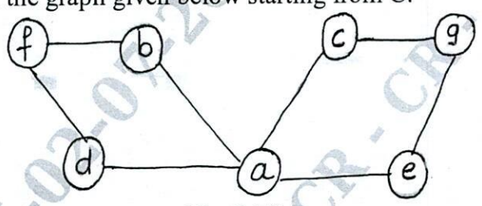
***Note: 1. Answer any FIVE full questions, choosing ONE full question from each module.***
***2. M : Marks , L: Bloom's level , C: Course outcomes.***

| | | Module – 1 | M | L | C |
|---|---|---|---|---|---|
| Q.1 | a. | Define data structure. With a neat diagram, explain the classification of data structure. | 5 | L1 | CO1 |
| | b. | Explain dynamic memory allocation functions with suitable examples. | 5 | L2 | CO1 |
| | c. | For the given sparse matrix draw the triplet representation and also draw the transpose of resultant triplet. $A = \begin{bmatrix} 15 & 0 & 0 & 22 & 0 & -15 \\ 0 & 11 & 3 & 0 & 0 & 0 \\ 0 & 0 & 0 & -6 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 91 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 28 & 0 & 0 & 0 \end{bmatrix}$ | 10 | L3 | CO1 |
| | | **OR** | | | |
| Q.2 | a. | Define stack. With suitable example write the array representation of stack. | 5 | L1 | CO2 |
| | b. | Write a C functions to implement push( ), pop( ) and display( ) operations for stack using array. | 5 | L2 | CO2 |
| | c. | Translate the following infix expressions to postfix form using stack:<br>i) ((A * B) + C) / D ii) A * B * C + D | 10 | L3 | CO2 |
| | | **Module – 2** | | | |
| Q.3 | a. | What are the disadvantages of linear queue? | 5 | L1 | CO2 |
| | b. | With suitable example discuss the representation of linear queue with array. | 5 | L2 | CO2 |
| | c. | Develop functions in C to implement insertion, deletion and display operations on circular queue of integers. | 10 | L3 | CO2 |
| | | **OR** | | | |
| Q.4 | a. | What is linked list? With suitable examples explain different types of linked lists. | 5 | L1 | CO3 |
| | b. | Write a C functions to implement a stack of integers using a Singly Linked List (SLL). | 5 | L2 | CO3 |





| | | | | | |
|---|---|---|---|---|---|
| | c. | Develop a functions in C for the following operations on singly linked list of integers:<br>i) Insert a element at end of SLL<br>ii) Delete a element at end of SLL<br>iii) Concatenation of two SLL | 10 | L3 | CO3 |
| | | **Module – 3** | | | |
| Q.5 | a. | Write a structure definition for Doubly Linked List (DLL) of integers. What are the advantages of DLL over SLL? | 4 | L1 | CO3 |
| | b. | Develop a C functions for the following operations on DLL of integers:<br>i) Insert a node at front of DLL<br>ii) Delete a node at end of DLL | 10 | L3 | CO3 |
| | c. | For the given sparse matrix design the linked list representation. $A = \begin{bmatrix} 2 & 0 & 0 & 0 \\ 4 & 0 & 0 & 3 \\ 0 & 0 & 0 & 0 \\ 8 & 0 & 0 & 1 \\ 0 & 0 & 6 & 0 \end{bmatrix}$ | 6 | L4 | CO3 |
| | | **OR** | | | |
| Q.6 | a. | Define binary tree. Write array and linked list representation for given binary tree.<br>  <br><br>Fig.Q.6(a) | 4 | L1 | CO4 |
| | b. | Develop recursive C functions for inorder, preorder and postorder traversal of a binary tree. Find inorder, preorder and postorder traversals for the given binary tree.<br>  <br><br>Fig.Q.6 (b) | 10 | L3 | CO4 |
| | c. | Design threaded binary tree for the given elements<br>10, 20, 30, 40, 50 | 6 | L4 | CO4 |

| | | | | | |
|---|---|---|---|---|---|
| | | **Module – 4** | | | |
| **Q.7** | **a.** | Write a adjacency matrix and adjacency linked list representation for following given graph.<br>  <br><br>Fig.Q.7(a) | **6** | **L1** | **CO4** |
| | **b.** | Develop a C function to traverse a graph using Depth First Search (DFS). Apply DFS for the graph given below starting from C.<br>  <br><br>Fig.Q.7(b) | **8** | **L4** | **CO4** |
| | **c.** | List and explain different types of selection trees with suitable examples. | **6** | **L4** | **CO4** |
| | | **OR** | | | |
| **Q.8** | **a.** | Define forest data structure. With a suitable example write a procedure to transform forest into binary tree. | **6** | **L1** | **CO4** |
| | **b.** | For a given data design a binary search tree. Apply inorder, preorder and postorder traversals on resultant binary search tree.<br>6, 9, 5, 2, 8, 15, 24, 14, 7, 8, 5, 2 | **8** | **L4** | **CO4** |
| | **c.** | Develop a C functions to perform the following operations on Binary Search Tree (BST):<br>i) Inserting an element into BST<br>ii) Recursive search of given key element on BST. | **6** | **L4** | **CO4** |
| | | **Module – 5** | | | |
| **Q.9** | **a.** | Explain hashing with suitable example. Explain different types of hashing functions in details. | **10** | **L2** | **CO5** |
| | **b.** | Explain static hashing and dynamic hashing in detail. | **10** | **L2** | **CO5** |
| | | **OR** | | | |
| **Q.10** | | Write a note on: | | | |
| | **a.** | Leftist trees | **6** | **L2** | **CO5** |
| | **b.** | Optimal binary search trees | **7** | **L2** | **CO5** |
| | **c.** | Priority queues. | **7** | **L2** | **CO5** |

* * * * *