Seat No. : _____

DH-107

December-2025

Integrated M.Sc. IT, Sem.-III

MDC-SMD-234T : Fundamentals of Data Structure

[Software Development (Web & Mobile)]

Time : 1:00 Hour] **[Max. Marks : 25**

Instructions :

(1) Attempt **all** questions.

(2) Make suitable assumptions wherever necessary.

(3) Figures to the right indicate full marks.

1. MCQs : **10**

(1) Which of the following is the time complexity to access an element in an array using its index ?

(a) O(1) (b) O(n)

(c) O(log n) (d) $O(n^2)$

(2) Which of the following sorting algorithms works best on nearly sorted data ?

(a) Bubble Sort (b) Selection Sort

(c) Insertion Sort (d) Quick Sort

(3) What is the maximum number of elements that can be stored in a two-dimensional array arr[m][n]?

(a) m (b) n

(c) m + n (d) m × n

(4) What does a "NULL" pointer represent in a linked list?

(a) Beginning of the list (b) Middle node

(c) End of the list (d) None of these

(5) Which data structure uses LIFO (Last In First Out) ?

(a) Queue (b) Stack

(c) Tree (d) Heap

(6) Which of the following operations is not possible in a circular queue ?

(a) Enqueue (b) Dequeue

(c) Traversal (d) None of the above

(7) BFS (Breadth First Search) uses which data structure ?

(a) Stack (b) Queue

(c) Array (d) Linked List

(8) What is the worst-case time complexity of searching an element in a hash table ?

(a) O(1) (b) O(n)

(c) O(log n) (d) O(n log n)

(9) Divide and Conquer strategy is used in :

(a) Bubble Sort (b) Merge Sort

(c) Insertion Sort (d) Selection Sort

(10) In which traversal of a binary tree is the root node visited first ?

(a) Inorder (b) Postorder

(c) Preorder (d) Level order

2. (a) Perform Insertion Sort on [12, 11, 13, 5, 6] and show each iteration. **5**

OR

(b) Write short notes on different types of linked lists (Singly, Doubly, and Circular) with suitable diagrams.

3. Explain the representation of arrays in memory and write any two applications of arrays. **5**

4. (a) Insert the following keys into a hash table of size 7 using hashing with chaining : **5**

i. h(k) = k mod 7

ii. Keys: 14, 22, 3, 7, 11, 26, 18, 9. Calculate the hash index for each key.

iii. Handle collisions using separate chaining. Show the final table with linked lists at the appropriate indices.

OR

(b) Explain the concept of a circular queue. How is it different from a simple linear queue ?