Course Code:23CS2T01

BONAM VENKATA CHALAMAYYA INSTITUTE OF TECHNOLOGY & SCIENCE

(AUTONOMOUS)

I - B. Tech II-Semester Supplementary Examinations (BR23), Sep/Oct - 2024

DATA STRUCTURES (common to CSE, CSE(AI&DS), AI&ML)

Time: 3 hours Max. Marks: 70

Question Paper consists of Part-A and Part-B
*Answer **ALL** the question in **Part-A and Part-B***

PART–A (10X2 = 20M)

		Marks	CO	BL
1. a)	What are linear data structures?	(2M)	1	1
b)	What is Binary search?	(2M)	1	1
c)	Explain the basic structure of a singly linked list.	(2M)	2	2
d)	What is the primary difference between arrays and linked lists in terms of memory allocation?	(2M)	2	1
e)	Explain the LIFO (Last In, First Out) principle in stacks.	(2M)	3	1
f)	Explain how a stack can be used to reverse a list.	(2M)	3	2
g)	Describe what happens during a queue overflow.	(2M)	4	2
h)	Describe the role of the front and rear pointers in a deque.	(2M)	4	2
i)	Illustrate how a binary tree is different from a binary search tree.	(2M)	5	2
j)	Illustrate how chaining works to resolve collisions in a hash table.	(2M)	5	2

PART–B (5X10 = 50M)

2.a)	Design a linear data structure that combines the advantages of arrays and linked lists.	5(M)	CO 1	BL6
b)	Describe the process of insertion sort and how it differs from selection sort.	5(M)		BL2
	(OR)			
3.a)	Explain the concept of space complexity and its significance.	5(M)	CO 1	BL2
b)	Illustrate merge sort with an example	5(M)		BL3
4.a)	Compare the operations (insertion, deletion) of a doubly linked list with those of a singly linked list in terms of time and space required.	5(M)	CO 2	BL5
b)	Compare the time complexities of searching for an element in an array versus in a linked list.	5(M)		BL4
	(OR)			
5.a)	Explain Insertion operation in Singly Linked List with an algorithm.	5(M)	CO 2	BL2
b)	Explain how linked lists can be used to implement a queue.	5(M)		BL3
6.a)	Analyse the advantages of using stacks in recursive algorithms.	5(M)	CO 3	BL4

b)	Assess the use of stacks in depth-first search (DFS) algorithms for graph traversal.	5(M)		BL5
	(OR)			
7.a)	Demonstrate stack operations in detail.	5(M)		BL2
b)	Write an algorithm to convert an infix expression to a postfix expression using a stack.	5(M)	CO 3	BL3
8.a)	Show how to handle queue overflow in an array-based queue.	5(M)	CO 4	BL3
b)	Evaluate the limitations of using queues for backtracking in large-scale problems.	5(M)		BL4
	(OR)			
9.a)	Compare the performance of queue operations in linear arrays and circular arrays.	5(M)	CO 4	BL4
b)	Write an algorithm to implement the add to rear operation in a deque.	5(M)		BL3
10.a)	Describe in detail how tree traversal methods (in-order, pre-order, post-order) can be used to solve different types of problems. Provide specific examples for each traversal method.	5(M)	CO 5	BL2
b)	Explain the importance of selecting a good hash function. What are the characteristics of a good hash function, and how do they reduce collisions? Provide examples to illustrate your points.	5(M)		BL2
	(OR)			
11.a)	Develop an algorithm to find the height of a binary tree. Implement this algorithm and explain the steps involved. Provide an example to demonstrate the output.	5(M)	CO 5	BL3
b)	Discuss the applications of hashing in database indexing. How do hash tables improve the efficiency of data retrieval in large databases? Use an example to support your explanation.	5(M)		BL2

****