Semester: 3rd			
Paper code: AIDS251/AIML251/IOT251	L	T/P	Credits
Subject: **Data Structures Lab**	0	2	1
Marking Scheme			

1. Teachers Continuous Evaluation: As per university examination norms from time to time
2. End term Examination: As per university examination norms from time to time

INSTRUCTIONS TO EVALUATORS: Maximum Marks: As per university norms
1. This is the practical component of the corresponding theory paper.
2. The practical list shall be notified by the teacher in the first week of the class commencement under the intimation to the office of the HOD/ Institution in which they appear is being offered from the list of practicals below.
3. Instructors can add any other additional experiments over and above the mentioned in the experiment list which they think is important.
4. At least 8 experiments must be performed by the students.

Course Objectives:	
1.	To teach students how to analyse different types of data structures.
2.	To design applications based on different types of data structures.

Course Outcomes:	
CO1	Design programs using a variety of data structures such as stacks, queues, hash tables, binary trees, search trees, heaps, graphs, B-trees, list, set, tuples, dictionary.
CO2	Implement and analyse abstract data types such as lists, graphs, search trees to solve real world problems efficiently.

CO/PO	PO01	PO02	PO03	PO04	PO05	PO06	PO07	PO08	PO09	PO10	PO11	PO12
CO1	2	2	2	2	1	-	-	-	-	-	-	1
CO2	2	2	2	2	1	1	1	1	1	1	1	2

LIST OF EXPERIMENTS:

1. Perform Linear Search and Binary Search on an array.

2. Create a stack and perform Pop, Push, and Traverse operations on the stack using array.

3. Create a stack and perform Pop, Push, and Traverse operations on the stack using linked list.

4. Create a Linear Queue using Linked List and implement different operations such as insert, delete, and display the queue elements.

5. Implement the following sorting techniques:

   a. Insertion sort

   b. Merge sort

   c. Bubble sort

   d. Selection sort

6. Create a linked list with nodes having information about a student. Insert a new node at the specified position.

7. Create a doubly linked list with nodes having information about an employee and perform Insertion at front of doubly linked list and perform deletion at end of that doubly linked list.

8. Create a circular linked list having information about a college and perform Insertion at the front end and perform deletion at the end.

9. Create a Binary Tree and perform Tree Traversals (Preorder, Postorder, Inorder) using the concept of recursion.

10. Implement insertion, deletion, and display (Inorder, Preorder, Postorder) on binary search tree with the information in the tree about the details of an automobile (type, company, year of make).