	Practical List: 03
1.	Write a menu driven program to implement following functionality with Singly Linked List: A. Insert node at first B. Insert node at end of list (append) C. Insert node at user choice position D. Remove node from first E. Remove node from last F. Remove node at user choice position G. Display list
2.	Design a menu driven program in C++ to implement the following operations on the SinglyLinked List: 1. Insert data a. By inserting a node at front b. By inserting a node at end c. By inserting a node at any other position 2. Delete data a. By deleting a node from the front b. By deleting a node from the end c. By deleting a node from any position 3. Copy all the data into another list 4. Merge the linked list with another one to make a larger list 5. Searching for an element in the list 6. Display the list.
3.	Write a menu driven program to implement following functionality with Doubly Linked List: A. Insert node at first B. Insert node at end of list (append) C. Insert node at user choice position D. Remove node from first E. Remove node from last F. Remove node at user choice position G. Display list
4.	Considering the music player which has next and previous buttons. Design a menu drivenprogram in C++ to implement the following operations on the Doubly Linked List: 1. Insert data (Upon your choice… By inserting a node at front or at end or at any other position) 2. Delete data (Upon your choice… By deleting from the front or from the end or from any position) 3. Merge the linked list with another one to make a larger list 4. Searching for an element in the list 5. Traversal the list

5.	Design a menu driven program to demonstrate the following operations on Stack usingLinked List: A. Insert node (Push) B. Remove node (Pop) C. Display stack
6.	Design a menu driven program in C++ to demonstrate the following operations on theQueue using Linked List: A. Insert node (Enqueue) B. Remove node (Dequeue) C. Display Queue

Objective(s)	To vibrant the concept of Linked List.
Pre-requisites	Basic of Object-Oriented Concepts.
Duration for Completion	10 Hours
PSO(s) to be achieved	PSO3: Student will be able to devise and conduct experiments and provide well informed conclusions using recent tools, technologies and industrial trends.
PO(s) to be achieved	PO2: Problem Analysis and Solution: Identify, Analyse and provide the solution for emerging real-world problems with the help of theoretical and practical understanding of tools and technologies.
CO(s) to be achieved	CO2: Recognize problem properties where Arrays, stacks, queues, and deque are appropriate data structures. CO3: Implement Linked Data Structure such as Linked List and Tree.
Solution must contain	Source Code with comments and Output Screen Shot
Nature of submission	Handwritten on A4 size blank papers
References for solving the problem	Textbook: Classic Data Structures, Debasis Samanta, PHI.
Post Laboratory questions	1. How to create a node in linked list? 2. Demonstrate the following operations steps-by-step for representing dynamic STACK: a. PUSH(STACKLL, 1) d. POP(STACKLL, ITEM) b. PUSH(STACKLL, 2) e. PUSH(STACKLL, 4) c. PUSH(STACKLL, 3) f. DISPLAY() [Display records in vertical form] 3. Define the C++ functions StackPushLL() to insert nodes and StackPopLL() to delete nodes, for a dynamic stack implementation, having the following structure for each node: struct student { int enro;

	char name[100]; student *link; }*top=NULL; 4. Demonstrate the following operations steps-by-step for representing dynamic QUEUE: a. ENQUEUE(QUEUE, Plum) d. ENQUEUE(QUEUE, Grape) b. ENQUEUE(QUEUE, Kiwi) e. DEQUEUE(QUEUE, ITEM) c. DEQUEUE(QUEUE, ITEM) f. DISPLAY() [Display records in vertical form] 5. Write a function QueueDequeueLL() in C++ to display and delete the data from a dynamically allocated Queue containing nodes of the following given structure:
	struct product { int code; char name[20]; product *link; }*front=NULL, *rear=NULL;