

LAB Manual

PART A

(PART A : TO BE REFERRED BY STUDENTS)

Experiment No.05

A.1 Aim:

To implement application of doubly Linked List data structure.

A.2 Prerequisite:

Prior knowledge of introduction to data structure.

A.3 Outcome:

After successful completion of this experiment students will be able to:

After successful completion of this experiment students will be able to:

1. Understand the concept of data structures
2. Know the use and implementation of different types of linked lists

A.4 Theory:

Data structure is logical or mathematical organization of data; it describes how to store the data and access data from memory. Actually in our programming data stored in main memory(RAM) and To develop efficient software or firmware we need to care about memory. To efficiently manage we required data structure.

There are two different types of data structure:

Linear Data Structure: In linear data structure data elements stored in sequential manner. Stack, Queue and Linked List are the types of linear data structure.

Non Linear Data Structure: In Non-Linear data structure data elements are not stored in the sequence manner. Tree and Graph are the type of non-linear data structure.

Linked Lists:

A Linked List is a linear data structure. Every linked list has two parts, the data section and the address section that holds the address of the next element in the list, which is called a node.

The size of the linked list is not fixed, and data items can be added at any locations in the list. The disadvantage is that to get to a node, we must traverse to all the way from the first node to the node that we require. The Linked List is like an array but unlike an array, it is not stored sequentially in the memory.

The most popular types of a linked list are:

1. Singly linked list
2. Doubly linked list
3. Circular linked list

Circular Linked List is a variation of Linked list in which the first element points to the last element and the last element points to the first element. Both Singly Linked List and Doubly Linked List can be made into a circular linked list.

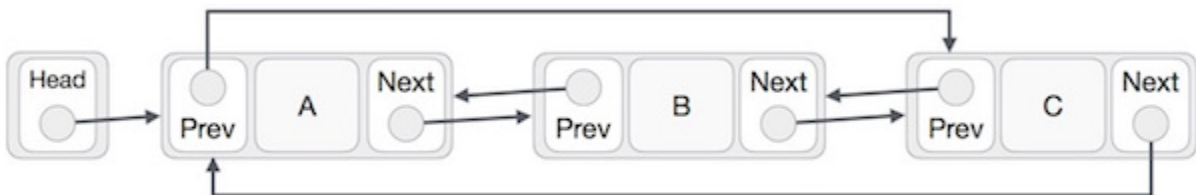
Singly Linked List as Circular

In singly linked list, the next pointer of the last node points to the first node.



Doubly Linked List as Circular

In doubly linked list, the next pointer of the last node points to the first node and the previous pointer of the first node points to the last node making the circular in both directions.



A.5 Procedure/Algorithm:

A.5.1 TASK 1: Write programs for the following questions:

Q1. Write a program in C/C++ to **insert a node at the beginning** of a doubly linked list. Take the element to be inserted from User.

Q2. Write a program in C/C++ to **insert a node at any position** in a doubly linked list. Take the element to be inserted from User.

Q3. Write a program in C/C++ to **delete a node from the end** of a doubly linked list. Take the element to be deleted from User.

Q4. Write a program in C/C++ to **search an element** in a circular linked list. Take the element to be searched from User.

A.5.1 TASK 2:

Save and close the file and name it as **DSA_Experiment05_YourName**.

PART B

(PART B : TO BE COMPLETED BY STUDENTS)

(Students must submit the soft copy as per following segments within two hours of the practical. The soft copy must be uploaded on the Blackboard or emailed to the concerned lab in charge faculties at the end of the practical in case there is no Black board access available)

Roll No.	Name:
Class :	Batch :
Date of Experiment:	Date of Submission
Grade :	

B.1 Answers of Task to be written by student:

B.1.1 Code

B.1.2 Output

B.2 Observations and learning:

(Students are expected to comment on the output obtained with clear observations and learning for each task/ sub part assigned)

B.3 Conclusion:

(Students must write the conclusion as per the attainment of individual outcome listed above and learning/observation noted in section B.2)
