**Objectives:**

In this lab, students will practice:

1. Doubly Linked List and its operations

**Question 1**

1. Implement a template class ‘Node’ that contains three data members: A template variable ‘data’, a Node pointer ‘next’, and another node pointer ‘prev’. You may define any member functions, if required, for this template class.

https://ssl.gstatic.com/ui/v1/icons/mail/images/cleardot.gif

1. Now using the above class, implement a doubly linked list which has **a dummy head and a dummy tail**, and supports the following operations:
2. Insert at start: void insertAtStart(T const element)
3. Insert at end: void insertAtEnd(T const element)
4. Delete from Start: void DeleteAtStart()
5. Delete from end: void DeleteAtEnd()
6. Print void print() const
7. Reverse all elements of linked list: void reverse()
8. remove all duplicate values: void removeDuplicates()
9. Insert value v1 before value v2: bool insertBefore(T const v1, T const v2 )
10. Delete node after v1: bool DeleteAfter(T const v1)
11. Destructor
12. Now create a main function which has the following instructions:
    1. Define a doubly linked list object of type int.
    2. Insert 7 at start.
    3. Insert 9 at start.
    4. Insert 1 at end.
    5. Insert 2 at end.
    6. Insert 10 at end.
    7. Insert 3 before 2
    8. Print the linked list. (**Sample output: 9->7>1->3->2-10)**
    9. Delete at start.
    10. Delete at start.
    11. Delete at end.
    12. Delete node after 3
    13. Print the linked list. (**Sample output: 1->3)**
    14. Insert 3 at end.
    15. Remove all duplicate values.
    16. Print the linked list. . (**Sample output: 1->3)**
    17. Reverse all elements of linked list.
    18. Now print the linked list. (**Sample output: 3->1)**