



## CL-2001

### Data Structures

### Lab # 2 and Lab 3 Task (6 wtg)

#### Objectives:

- Linked List(insertion)
- Linked List(deletion)
- Linked List(searching)
- Linked List(Traversal)

**Note: Carefully read the following instructions (*Each instruction contains a weightage*)**

1. There must be a block of comments at start of every question's code by students; the block should contain brief description about functionality of code.
2. Comment on every function and about its functionality.
3. Mention comments where necessary such as comments with variables, loop, classes etc to increase code understandability.
4. Use understandable name of variables.
5. Proper indentation of code is essential.
6. Write a code in C++ language.
7. Make a Microsoft Word file and paste all of your C++ code with all possible screenshots of every task **outputs in Microsoft Word and submit word file. Submit all .cpp file.**
8. First think about statement problems and then write/draw your logic on copy.
9. After copy pencil work, code the problem statement on MS Studio C++ compiler.
10. At the end when you done your tasks, attached C++ created files in MS word file and make your submission on Google Classroom. (Make sure your submission is completed).
11. Please submit your file in this format **19F1234\_L4**.
12. Do not submit your assignment after deadline. Late and email submission is not accepted.
13. Do not copy code from any source otherwise you will be penalized with negative marks.

## Single Linked List (Lab 2 – Task 3wtg)

### Problem: 1 | Median and Mode [1] 3.3 marks

Write a program that inserts 15 random integers from 0 to 100 in order in a linked list object. The program should calculate the median and mode of the elements.

### Problem: 2 | Copying a List in Reverse Order [1] 3.3 marks

Write a program that creates a linked list object of 10 characters and creates a second list object containing a copy of the first list, but in reverse order.

### Problem: 3 | Linked List the Palindrome [1] 3.3 marks

Write a function to check whether the given Singly Linked List is Palindrome or not.

## Doubly Linked List (Lab 3 – Task 3wtg)

### Problem: 4 | | Doubly Linked List [1] 3 marks

Write a C++ program to check sort doubly linked list is sorted or not.

### Problem: 5 | Split Doubly Linked List [1.5] 5 marks

Write a code to split a doubly link list into sub lists — one for the front half, and one for the back half. If the number of elements is odd, the extra element should go in the front list. So FrontBackSplit() on the list {2, 3, 5, 7, 11} should yield the two lists {2, 3, 5} and {7,11}. You should check your solution against a few cases (length = 2, length = 3, length=4) to make sure that the list gets split correctly near the short-list boundary conditions. You will probably need special case code to deal with the (length<2) cases.

### Problem: 6 | Circular Link List [0.5] 2 marks

Write a menu driven C++ program for following functions of a Circular Linked list.

1. InsertAtBegin()
2. DeleteAtEnd()
3. Print()

Best of luck 😊