University of Computer and Emerging Sciences

Laboratory Manual

for

Data Structures Lab

Course Instructor	Ma’am Abeeda Akram
Lab Instructors	Mr. Sohaib Ahmad Ms. Ammarah Nasir
Section	BCS-3A
Semester	Fall 2022

Department of Computer Science

FAST-NU, Lahore, Pakistan

NOTE

Zero Tolerance for Plagiarism. Penalty will be given in accordance with the severity of plagiarism. This also includes forwarding the case to the DC Committee.

Objective of this lab:
Linked List

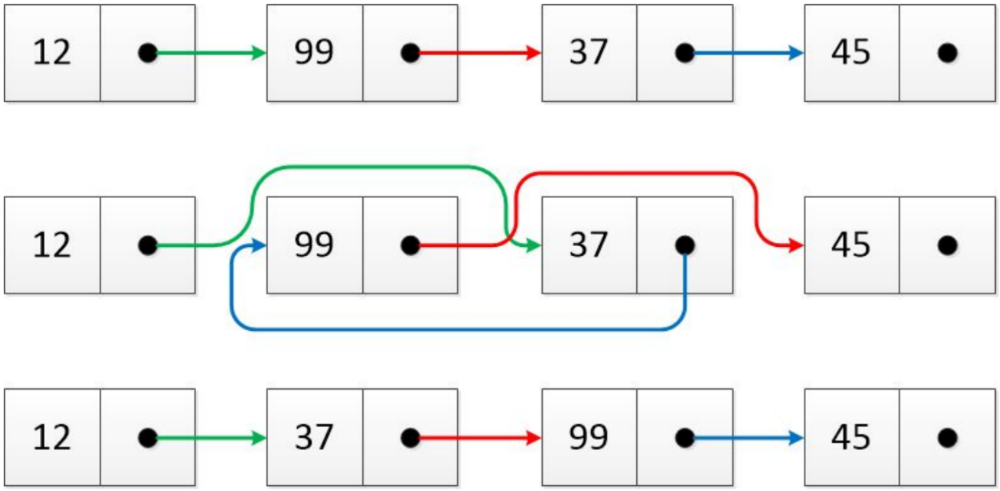
Instructions:
• Make a separate project for each task.
• Indent your code properly.
• Use meaningful variable and function names. Follow the naming conventions.
• Use meaningful prompt lines and labels for all input/output.
• Make sure that there are NO dangling pointers or memory leaks in your program.

YOUR PROGRAM MUST HAVE A MAIN FUNCTION THAT TEST ALL THE IMPLEMENTED FUNCTIONS

1. Implement a class **‘Node’** that contains two data members: An int variable ‘data’, a Node pointer ‘next’. You may define any member functions, if required, for this template class.

2. Now using the above class, implement a singly integer linked list (having two pointer head and tail) which supports the following operations:

A. void insertAtHead(int const element);
B. void insertAtTail(int const element);
C. void printForward() const; This function should print values from head to tail.
D. void DeleteFromHead();
E. void DeleteFromTail();
F. int getMiddle() const; return the middle element of the link list.
G. int size() const; This function should return the size of Linked List.
H. bool IsEmpty(); Return true if FRONT/TAIL is pointing to NULL otherwise false.
I. void InsertAfter(val, key); It should enter the new Node with the value key, after the first occurance of value val. If not found insert at Tail
J. void InsertBefore(val, key); It should enter the new Node with the value key, before the first occurance of value val. If not found insert at Tail
K. int getMax() const;
L. int getMin() const;
M. int getAverage() const;
N. bool Swap(LeftIndex,RightIndex): Swap the Node on Left index with Node on Right index, you are not allowed to swap the data, you have to swap the addresses of these nodes to apply the Swap. take care of the edge cases like swapping the first and last value. Example: Swap between the Nodes on index 1 and 2 is shown in this image. Maintain the previous pointers as well. (BONUS)



O. Desctructor(): It will delete all the Nodes Independently using a loop, calling each node’s destructor.

3. using the Node class, implement a singly character linked list (that have one pointer Head), which supports the following operations:

A. void insertAtHead(char const element);
B. void insertAtTail(char const element);
C. void print() const;
D. void DeleteFromHead();
E. void DeleteFromTail();
F. int totalCount() const; This function should return the total count of all characters in the list.
G. void split(singlyLinkedList &firstHalf, singlyLinkedList &secondHalf);Split the list in two separate lists from middle.
H. void merge(singlyLinkedList &firstHalf, singlyLinkedList &secondHalf);Merge two link lists in one list

GOOD LUCK