



THE UNIVERSITY
OF LAHORE
**ISLAMABAD
CAMPUS**

Data Structures and algorithm (CS09203)

Lab Report

Name: Muhammad Talha Khalid
Registration #: CSU-S16-135
Lab Report #: 4
Dated: 21-04-2018
Submitted To: Mr. Usman Ahmed

The University of Lahore, Islamabad Campus
Department of Computer Science & Information Technology

Experiment # 4

Data Handling by Link List

Objective

To understand How to Insert and Delete Data using Link list .

Software Tool

1. Linux Ubuntu
2. G++
3. Miktext editor

1 Theory

In this experiment we learn how to handle our data using link listing using Link list , Create new node and move address of 1st node to second location When we enter our data it create a new node the 1st node has our data and second has its address and the last one is NULL and so on

It has 4 rules:

1. Nodes.
2. *Head .
3. *Tail.
4. *New Node.

2 Task

2.1 Procedure: Task 1

In this We eter our data into link list the link list create node and enter our data and moves its address to next node and the other node is now null to delete it it search for the data you enter into the nodes and if data is not found it will go to next node and so on

```
noobster@noobster-VirtualBox: ~/Desktop/Labtasks/LAB4
Press 1 to input data
Press 2 to Display data
Press 3 to Delete data
Press 4 to Exit
Choose from above: █
```

Figure 1: Main menu of my program

```
noobster@noobster-VirtualBox: ~/Desktop/Labtasks/LAB4
44
33
66
90
12
43
76
wana Continue? y/n █
```

Figure 2: enter 7 number into link list

2.2 Procedure: Task 2

```
#include<iostream>
#include<stdio.h>
#include <unistd.h>
#include<cstdlib>
using namespace std;
class Node {
public:
    int Data;
    Node *Next;
};
class linklist {
public:
    Node *Head;
    Node *Tail;
    Node *NewNode;
    linklist() {
        Head=Tail=NULL;
        NewNode=NULL;
    }
    void AddNode(int num) {
        NewNode=new Node;
        NewNode->Data=num;
        cout<<" Sucesfully entered " <<NewNode->Data<<" \n" ;
        if (Head==NULL){
            Head=NewNode;
            Tail=NewNode;
        }
        else {
            Tail->Next=NewNode;
            Tail=Tail->Next;
        }
    }
    void Display(){
        Node *temp;
        temp=Head;
        while(temp!=NULL) {
            cout<<temp->Data<<endl;
            temp=temp->Next;
        }
    }
};
```

```

    }
}

void Delete(int num) {
    Node *temp;
    Node *prev;
    //Previous=Head;
    if (Head==NULL) {
        cerr << "Node is Empty!! \n";
        return;
    } else {
        if (Head->Data==num) {
            Head=Head->Next;

        } else {
            temp=Head;

            while (temp!=NULL && temp->Data!=num){
                prev=temp;
                temp=temp->Next;
            } //while

            prev->Next=temp->Next;
            delete temp;

        } //if
    } //wada if
}

};

void menu() {
    cout << "Press 1 to input data \n";
    cout << "Press 2 to Display data \n";
    cout << "Press 3 to Delete data \n";
    cout << "Press 4 to Exit \n";
}

int Choice;
string Schoice="y";
int main()
{
    int num;
    linklist LI;

```

```

do {
    system("clear");
    menu();
    cout<<"Choose from above: ";
    cin>>Choice;
    switch (Choice) {
        case 1:
            system("clear");
            do {
                cout<<"Enter a number: ";
                cin>>num;
                LI.AddNode(num);
                cout<<"\nwana Continue? y/n ";
                cin>>Schoice;
            }while( Schoice!="n");
            break;

        case 2:
            system("clear");
            do {
                LI.Display();
                cout<<"\nwana Continue? y/n ";
                cin>>Schoice;
            }while( Schoice!="n");
            break;
        case 3:
            system("clear");
            do {
                LI.Display();
                cout<<"\nEnter a number you wana delete: ";
                cin>>num;
                LI.Delete(num);
                cout<<"\nwana Continue? y/n ";
                cin>>Schoice;
            }while( Schoice!="n");
            break;
        }
    }while(Choice!=4);
    return 0;
}

```

3 Conclusion:

So we come to a conclusion that how to use link list basic concept of nodes and Pointers and how nodes work and basic concept about data structures

4 Output:

```
noobster@noobster-VirtualBox: ~/Desktop/Labtasks/LAB4
44
33
66
90
12
43
76

Enter a number you wana delete: 44
wana Continue? y/n
```

Figure 3: Display output of my Stored Data

```
noobster@noobster-VirtualBox: ~/Desktop/Labtasks/LAB4
12
43
76

Enter a number you wana delete: 44
wana Continue? y/ny
33
66
90
12
43
76

Enter a number you wana delete: 76
wana Continue? y/ny
33
66
90
12
43

Enter a number you wana delete:
```

Figure 4: Remove 44 and 76 from my link list