Task1

#include<iostream>

using namespace std;

class node{

public:

int key;

int data;

node\* next;

node(){

key=0;

data=0;

next = NULL;

}

node(int k, int d){

key = k;

data = d;

}

};

class LinkedList{

public:

node\* head;

node\* tail;

LinkedList(){

head = NULL;

tail = head;

}

// Inserting at the tail

void insertAtEnd(int k, int d){

node\* newNode = new node(k,d);

if (head == NULL) {

head = newNode;

tail = newNode;

newNode->next = head;

} else {

tail->next = newNode;

tail = newNode;

tail->next = head;

}

}

// display

void display(){

node\* temp = head;

while(temp!=NULL){

cout<<temp->data<<"->";

if(temp->next == head){

cout<<temp->next->data;

break;

}

temp = temp->next;

}

}

// Inserting at the head

void insertAtHead(int k,int d){

node\* newnode = new node(k,d);

tail->next = newnode;

newnode->next = head;

head = newnode;

}

// Inserting at the position

void insertAtPos(int k,int val,int pos){

if(pos==0){

insertAtHead(k,val);

return;

}

node\* newnode = new node(k,val);

node\* temp = head;

int currpos=0;

while(currpos!= pos-2){

temp = temp->next;

currpos++;

}

newnode->next =temp->next;

temp->next = newnode;

}

// Deleting the node from position

void deleteAtPos(int pos){

if(pos==0){

node\* temp = head;

head= head->next;

temp==NULL;

}

else{

int currpos = 0;

node\* prev = head;

while(currpos!= pos-2){

prev = prev->next;

currpos++;

}

node\* temp = prev->next;

prev->next = prev->next->next;

delete(temp);

}

}

};

int main(){

LinkedList newlist;

newlist.insertAtEnd(3,5);

newlist.insertAtEnd(4,6);

newlist.insertAtEnd(5,7);

newlist.display();

cout<<endl<<endl<<"Inerting 15 10 9 at the head"<<endl;

newlist.insertAtHead(7,9);

newlist.insertAtHead(8,10);

newlist.insertAtHead(9,15);

newlist.display();

cout<<endl<<endl<<"Inserting at the pos 3 val 99"<<endl;

newlist.insertAtPos(11,99,3);

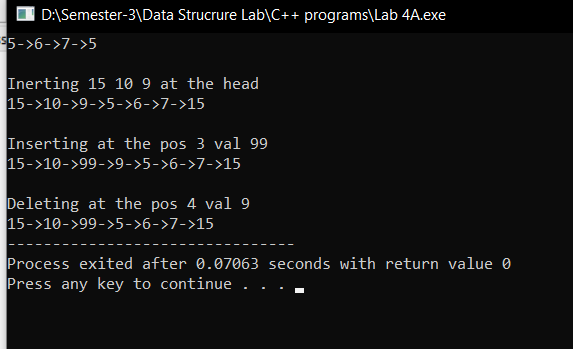
newlist.display();

cout<<endl<<endl<<"Deleting at the pos 4 val 9"<<endl;

newlist.deleteAtPos(4);

newlist.display();

}



Task 2

#include<iostream>

using namespace std;

class node{

public:

int data;

node \*next;

node()

{

next=NULL;

}

node(int a)

{

data=a;

next=NULL;

}

};

class linkedlist

{

public:

node \*head;

linkedlist()

{

head=NULL;

}

void append(int data)

{

node\* newnode = new node(data);

if(head==NULL)

{

head=newnode;

}

else

{

node \*temp=head;

while(temp->next != NULL)

{

temp=temp->next;

}

temp->next=newnode;

}

}

void prepend(int data)

{

node\* newnode = new node(data);

node \*temp;

temp=newnode;

temp->next=head;

head=temp;

}

void function()

{

linkedlist newlist;

node \*temp=head;

while(temp!=NULL)

{

int d=temp->data;

newlist.append(d);

temp=temp->next->next;

}

linkedlist newlist2;

temp=head;

while(temp!=NULL)

{

temp=temp->next;

int d=temp->data;

newlist2.prepend(d);

temp=temp->next;

}

temp=newlist2.head;

while(temp!=NULL)

{

int d=temp->data;

newlist.append(d);

temp=temp->next;

}

newlist.display();

}

void display()

{

node \*temp=head;

while(temp!=NULL)

{

cout<<temp->data<<"->";

temp=temp->next;

}

cout<<" NULL";

}

};

main()

{

linkedlist list;

list.append(10);list.append(4);list.append(9);list.append(1);list.append(3);list.append(5);list.append(9);list.append(4);

list.display();

cout<<endl;

list.function();

}

