4.1. A menu driven program that implements singly linked list for the following operations: - Insert, Display, Count, Delete, Search.

# include <iostream> # include <conio.h> using namespace std;

class node {

private:

public:

};

node \* p; node \* q;

void node :: add(int num)

{

int data;

node \*address;

int count(void); void add(int); void display(void);

void insert(int, int); void remove(int); void search(int);

q=p; if(p==NULL)

{

}

else

{

p=new node; p-> data=num;

p-> address=NULL;

while(q->address!=NULL)

{

q=q->address;

}

q->address=new node; q->address->data=num;

q->address->address=NULL;

}

}

void node :: display(void)

{

q=p;

if( p == NULL)

{

}

else

{

cout<< "No Linkedlist";

while (q != NULL)

{

cout<<q->data<<endl; q = q->address;

}

}

}

int node :: count(void)

{

q=p;

int i = 0;

if(p == NULL)

{

}

else

{

return 0;

while (q!= NULL)

{

}

return i;

}

}

i++;

q = q-> address;

void node :: insert (int pos,int num)

{

node \*q = p; node \*temp; int i;

if(pos == 1)

{

p = new node; p ->data=num; p->address = q; return;

}

if(pos == 1+count())

{

add(num); return;

}

for(i=1; i<=(pos-2);i++)

{

q=q->address;

}

temp=q->address;

q->address = new node; q->address->data = num;

q->address->address=temp;

}

void node :: remove(int pos)

{

node \*temp; node \*q=p; if(pos == 1)

{

p=q->address; delete(q); return;

}

for(int i=1; i<=(pos-2);i++)

{

q=q->address;

}

temp = q->address;

q->address = q->address->address; delete (temp);

}

void node :: search(int num)

{

node \*q=p; int flag=0; int pos=0;

for(q=p;q!=NULL;q=q->address)

{

if(q->data == num)

{

}

pos++;

}

flag=1; break;

if(flag==1)

{

}

int main()

{

}

else

{

}

cout<<"Number Found At Position : "<<pos;

cout<<"Number NA Found";

int num, option,pos; node n;

p=NULL;

char ch='y'; while(ch =='y')

{

cout<<"\n1.Add"; cout<<"\n2.Display"; cout<<"\n3.Count"; cout<<"\n4.Insert"; cout<<"\n5.Delete"; cout<<"\n6.Search"; cout<<"\nEnter an option: "; cin>>option;

switch(option)

{

case 1:

{

}

cout<<"\nEnter the value: "; cin>>num;

n.add(num);

cout<<"\nDo you Want to do again\n"; break;

case 2:

{

n.display();

cout<<"\nDo you Want to do again\n";

break;

}

case 3:

{

}

num=n.count(); cout<<num;

cout<<"\nDo you Want to do again\n"; break;

case 4:

{

}

case 5:

{

}

case 6:

{

}

}

cout<<"Enter Position: "; cin>>pos;

cout<<"Enter Number: "; cin>>num; n.insert(pos,num);

cout<<"\nDo you Want to do again\n"; break;

cout<<"Enter Position: "; cin>>pos; n.remove(pos);

cout<<"\nDo you Want to do again\n"; break;

cout<<"Enter Number: "; cin>>num; n.search(num);

cout<<"\nDo you Want to do again\n"; break;

ch=getch();

}

}