Dsa test

1. #include<iostream>

using namespace std;

class Node{

int data;

Node\* next;

public:

Node(int);

void setData(int);

void setNext(Node\*);

int getData();

Node\* getNext();

};

#include"node.h"

Node::Node(int data){

this->data=data;

this->next=NULL;

}

void Node::setData(int data){

this->data=data;

}

void Node::setNext(Node\* n){

this->next=n;

}

int Node::getData(){

return this->data;

}

Node\* Node::getNext(){

return this->next;

}

#include"node.h"

class linkedList{

Node\* start;

public:

linkedList();

void insertBySort(int);

void display();

Node\* searchForDelete(int);

void deleteBYdata(int);

};

#include"linkedlist.h"

linkedList::linkedList(){

this->start=NULL;

}

void linkedList::insertBySort(int data){

Node\* temp=new Node(data);

Node\* p=start;

if(start==NULL){

start=temp;

return;

}

if(p->getNext()==NULL){

if(temp->getData()<p->getData()){

temp->setNext(p);

start=temp;

return;

}

else{

p->setNext(temp);

return;

}

}

while(p->getNext()->getNext()!=NULL){

if(temp->getData()<p->getNext()->getData()){

temp->setNext(p->getNext());

p->setNext(temp);

return;

}

else{

p=p->getNext();

}

}

//for last element

if(temp->getData()<p->getNext()->getData()){

temp->setNext(p->getNext());

p->setNext(temp);

return;

}

}

void linkedList::display(){

Node\* p=start;

while(p!=NULL){

p=p->getNext();

}

}

Node\* linkedList::searchForDelete(int data){

Node\* p=start;

if(p->getNext()==NULL){

if(p->getData()==data){

return p;

}

}

//for 1st element

if(p->getData()==data){

cout<<p->getData();

return p;

}

while(p->getNext()->getNext()!=NULL){

if(p->getNext()->getData()==data){

return p;

}

else{

p=p->getNext();

}

}

//for last element;

if(p->getNext()->getData()==data){

return p;

}

return NULL;

}

void linkedList::deleteBYdata(int data){

Node\* preAdd=searchForDelete(data);

Node\* q=NULL;

if(preAdd->getNext()==NULL){

q=preAdd;

start=q->getNext();

}

if(preAdd!=NULL){

q=preAdd->getNext();

cout<<q->getData()<<"is delete";

preAdd->setNext(q);

delete q;

return;

}

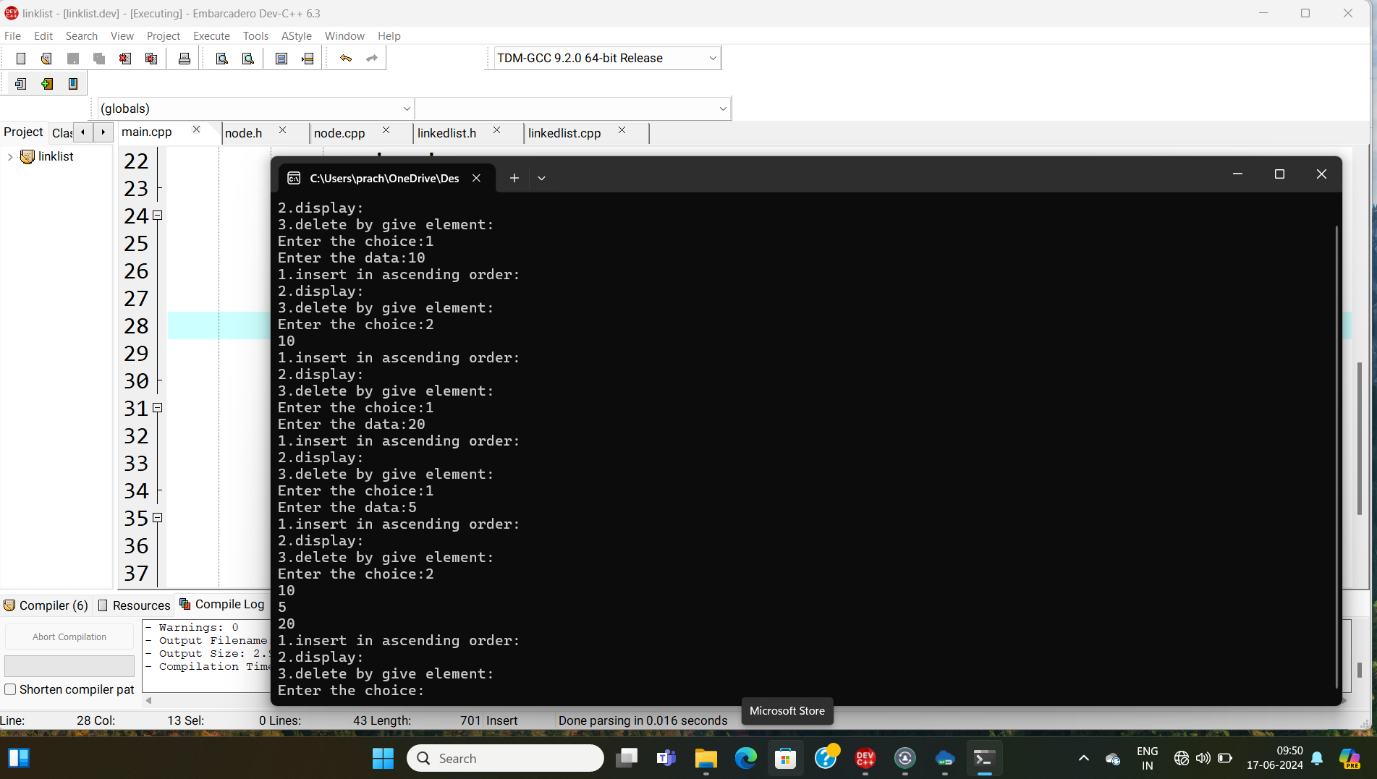
else{

cout<<"element not found";

return;

}

}



2.

#include<iostream>

using namespace std;

int main(){

int k=0;

char ch='A';

for(int i=1;i<=5;i++){

char alpha=ch;

for(int j=1;j<=5-k;j++){

cout<<alpha;

alpha++;

}

cout<<"\n";

ch++;

k++;

}

ch='D';

for(int i=1;i<=4;i++){

char beta=ch;

for(int j=1;j<=i+1;j++){

cout<<beta;

beta++;

}

cout<<"\n";

ch--;

}

return 0;

}

