DSA Lab #3

Question #1:

#include<iostream>

using namespace std;

class Node

{

public:

int data;

Node\* next;

};

class Link\_list

{

private:

Node\* head;

public:

Link\_list()

{

head = NULL;

}

};

int main()

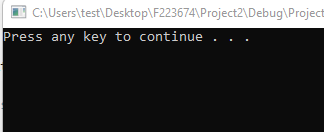
{

Link\_list obj;

system("pause");

}

Output:



Question # 2:

#include<iostream>

using namespace std;

class Node

{

public:

int data;

Node\* next;

};

class Link\_list

{

private:

Node\* head;

public:

Link\_list()

{

head = NULL;

}

void inserAthead(int val)

{

Node\* newnode = new Node();

newnode->data = val;

if (head == NULL)

{

head = newnode;

return;

}

newnode->next = head;

head = newnode;

}

void display()

{

Node\* temp = head;

while (temp != NULL)

{

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

temp = temp->next;

}

cout<<endl;

}

void insertAtend(int val)

{

Node\* n = new Node();

n->data = val;

Node\* temp = head;

while (temp->next!=NULL)

{

temp = temp->next;

}

n->next = temp->next;

temp->next = n;

}

void deleteatstart()

{

head = head->next;

}

void deleteAtend()

{

Node\* n = head;

while (n->next->next!=NULL)

{

n = n->next;

}

n->next = NULL;

delete n->next;

}

};

int main()

{

Link\_list obj;

obj.inserAthead(3);

obj.inserAthead(2);

obj.inserAthead(1);

obj.display();

obj.insertAtend(1);

obj.insertAtend(2);

obj.insertAtend(3);

obj.display();

obj.deleteatstart();

obj.display();

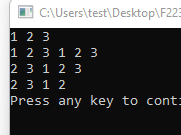
obj.deleteAtend();

obj.display();

system("pause");

}

Output:



Question# 3:

#include<iostream>

using namespace std;

class Node

{

public:

int data;

Node\* next;

Node(int d)

{

this->data = d;

this-> next = nullptr;

}

};

class Link\_list

{

private:

Node\* head,\*tail;

public:

bool insertion(int data)

{

Node\* newNode = new Node(data);

if (head=nullptr)

{

head = newNode;

tail = head;

}

else

{

tail->next = newNode;

tail = newNode;

}

return true;

}

bool search( int key)

{

Node\* temp = head;

while (temp!=NULL)

{

if (temp->data==key)

{

return true;

}

temp = temp->next;

return false;

}

}

void dsiplay()

{

Node\* temp = head;

while(temp!=NULL)

{

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

temp = temp->next;

}

}

};

int main()

{

Link\_list obj;

obj.dsiplay();

system("pause");

}

Question #4:

#include<iostream>

#include<string>

using namespace std;

class Studentrecord //making class to Record student Data

{

public:

Studentrecord\* next;

int semester;

string name;

int rollno;

double GPA;

};

class StudentRec

{

private:

Studentrecord\* Head;

public:

StudentRec() //Making constructor for intializing data

{

Head = 0;

}

void insert(string nam, int roll, double gp, int s)//Taking input data

{

Studentrecord\* newnode = new Studentrecord();

newnode->rollno = roll;

newnode->name = nam;

newnode->GPA = gp;

newnode->semester = s;

newnode->next = Head;

Head = newnode;

}

void search(int roll)//for searching student data

{

Studentrecord\* temp = Head;

bool flag = false;

while (temp!=0)

{

if (temp->rollno==roll)

{

flag = true;

break;

}

else

{

flag = false;

}

temp = temp->next;

}

if (flag)

{

cout << "Name: " << temp->name << endl;

cout << "Roll No: " << temp->rollno << endl;

cout << "Semester :" << temp->semester << endl;

cout << "GPA" << temp->GPA << endl;

}

else

{

cout << "OOP!!..Student Not Exsist" << endl;

}

}

void deletee(int roll)//To delete student data

{

bool flag = false;

int count = 0;

Studentrecord\* temp1 = Head;

while (temp1 !=0)

{

count++;

if (temp1->rollno==roll)

{

flag = true;

break;

}

temp1 = temp1->next;

}

Studentrecord\* temp = Head;

Studentrecord\* prev = Head;

if (flag)

{

if (count==1)

{

Head = Head->next;

delete temp;

}

else

{

for (int i = 0; i < count; i++)

{

prev = temp;

temp = temp->next;

}

prev->next = temp->next;

delete temp;

}

cout << "Student Data is deleted " << endl;

}

else

{

cout << "Sorry!! No Record Found " << endl;

}

}

void dis()//Display function to Show Studenr Data

{

Studentrecord\* temp;

temp = Head;

while (temp!=0)

{

cout << "Name: " << temp->name << endl;

cout << "Roll No: " << temp->rollno << endl;

cout << "Semester :" << temp->semester << endl;

cout << "GPA : " << temp->GPA << endl;

temp = temp->next;

}

cout << endl;

}

};

int main()

{

StudentRec obj;

int roll, seme;

string name;

double GPA;

int choice;

do

{

cout<<"1.Add Student "<<endl;

cout<<"2. Display data "<<endl;

cout<<"3.Search Data "<<endl;

cout<<"4.Delete Student Record "<<endl;

cout<<"5.Exit "<<endl;

cin >> choice;

system("cls");

switch (choice)

{

case 1:

cout << "Enter Name :" << endl;

cin >> name;

cout << "Enter Roll No " << endl;

cin >> roll;

cout << "Enter GPA" << endl;

cin >> GPA;

cout << "Enter semeter " << endl;

cin >> seme;

if (GPA<3)

{

cout << "Sorry!! we can not insert the record " << endl;

}

else

{

cout << "REcord is inserted " << endl;

obj.insert(name, roll, GPA, seme);

}

break;

case 2:

cout << "\*\*\*\*\*\*\*\*\*\*\*\*\*Displaying DATA \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

obj.dis();

break;

case 3:

if (choice==3)

{

cout << "\*\*\*\*\*\*\*\*\*\*\*\*Search Student Record\*\*\*\*\*\*\*\*\*\*\*\* " << endl;

int ro;

cout << "Enter the Roll No to search " << endl;

cin >> ro;

obj.search(ro);

}

break;

case 4:

cout << "\*\*\*\*\*Delete Student Record \*\*\*\*\*\*" << endl;

int r;

cout << "Enter Roll No to delete " << endl;

cin >> r;

obj.deletee(r);

break;

case 5:

cout << "Program has been terminated " << endl;

break;

default:

cout << "Wrong Input !! Try Again " << endl;

break;

}

}while (choice!=5);

system("pause");

return 0;

}

Output:

