

วิชา COS2161 Lab 11 เรื่อง pointer

ชื่อ.....รหัส.....เลขที่.....

ข้อที่ 1 จงอธิบายคำสั่งโปรแกรมต่อไปนี้

```
#include<iostream>

#include<string>

using namespace std;

struct node //1.....
{
    int score; //2.....
    node* link; //2.....
};

void print(node* first)
{
    while (first != NULL) //3.....
    {
        cout<<first->score<<endl; //4.....
        first=first->link; //5.....
    }
}

int main()
{
    char ch;
    node *p ; node *first; node *q; //6.....
    first = NULL; //7.....
    do
    {
        p = new node; //8.....
        cout<<"score="; cin>>p->score; //9.....
        p->link = NULL; //10.....
        if (first==NULL) //11.....
            first=p; //12.....
        else
        {
            q = first; //13.....
            while (q->link != NULL) //14.....
```

```

        q = q->link;
        q->link = p;                                //15.....
    }
    cout<<"run again y/n?";cin>>ch;
} while (ch=='y');                                //16.....
print(first);                                    //17.....
return 0;
}

```

โปรแกรมนี้ทำอะไร?

.....

.....

.....

ข้อที่ 2 โปรแกรมต่อไปนี้เป็นการหาค่าเฉลี่ยของ Linked queued จงเติมคำให้สมบูรณ์

```
#include<iostream>
```

```
#include<string>
```

```
using namespace std;
```

```
struct node
```

```
{
```

```
    int score;
```

```
    ..... link;
```

```
};
```

```
void enqueue(.....)
```

```
{
```

```
    char ch;  node *p;
```

```
    do
```

```
    {
```

```
        p = .....;
```

```
        cout<<"score="; cin>>p->score;
```

```
        p->link = NULL;
```

```
        if (rear==NULL)
```

```
        {
```

```
            front = p;
```

```
            rear = .....;
```

```
        }
```

```
    else
```

```
    {
```

```

        rear->link = p;
        rear      = p;
    }
    cout<<"run again y/n?";cin>>ch;
} while (ch=='y');
}

void dequeue(.....)
{
    node* p;
    double sum=0.0 , size=0;
    double average;
    while (front != NULL)
    {
        cout<<front->score<<" ";
        sum = .....;
        size++;
        p = .....;
        front=front->link;
        delete(p);
    }
    average = .....;
    cout<<endl<<"Average= "<<average<<endl;
    cout<<endl;
}

int main()
{
    node *front,*rear;
    front = NULL;
    rear  = NULL;
    enqueue(front,rear);
    dequeue(front);
    return 0;
}

```

เมื่อ run โปรแกรมโดยป้อนข้อมูลดังนี้

```

score=50
run again y/n?y
score=90
run again y/n?y
score=20
run again y/n?y
score=45
run again y/n?n
50 90 20 45
Average= 51.25
Press any key to continue_

```

ข้อที่ 3 จงเขียนโปรแกรมเรียงลำดับข้อมูล โดยป้อนข้อมูลคะแนนเก็บไว้ใน linked list ที่ตัวแปรพอยเตอร์ first ซึ่งอยู่จากนั้นนำข้อมูลในลิสต์นี้มาเรียงลำดับข้อมูลจากน้อยไปมากเก็บไว้ใน linked list ใหม่ที่ตัวแปรพอยเตอร์ newlist ซึ่งอยู่ที่ไหนคแรก จงเขียนส่วนที่ขาดของโปรแกรมให้สมบูรณ์

```

#include<iostream>

#include<string>

using namespace std;

struct node {

    int score;

    node* link;

};

void input(.....)

{

    char ch; node *p , *q;

    do

    {

        p = .....;

        <<"score="; cin>>p->score; p->link = NULL;

        if (first==NULL)

            first=.....;

        else {

            q = first;

            while (q->link != NULL)

                q = q->link;

            q->link = .....;

        }

        cout<<"run again y/n?">>cin>>ch;

    }while (ch=='y');

}

void print(node* first)

{

```


ข้อที่ 4 จงเขียนโปรแกรมนำกลุ่มของหนังสือเก็บในโครงสร้างของคิว ข้อมูลหนังสือประกอบด้วยชื่อหนังสือ ชื่อ และผู้แต่ง ต่อจากนั้นให้ป้อนชื่อผู้แต่งโดยนำชื่อผู้แต่งไปตรวจสอบกับข้อมูลทั้งหมดที่เก็บในคิวเพื่อลบข้อมูล หนังสือที่ผู้แต่งคนนั้นเขียนขึ้นออกทั้งหมด พิมพ์หนังสือที่เหลือออกทางจอภาพ จงเติมคำสั่งที่ขาดหายไปให้ ถูกต้องสมบูรณ์

```
#include<iostream>

#include<string>

using namespace std;

struct book_node{

    string name_book;

    string author_book;

    book_node *link;

};

void EnQueue(.....)

{

    ..... *p; string name;

    cout<<"book name=?"; cin>>name;

    while(name != "####") {

        p = .....;

        p->name_book = .....;

        cout<<"author book=?"; cin>>.....;

        p->link = .....;

        if (rear==.....) {

            front = p;

            rear = p;

        }

        else{

            rear->link = .....;

            rear=.....;

        }

        cout<<"book name=?";

        cin>>.....;

    }

}

void Print(.....) {

    if (front!=NULL) {

        cout<<front->name_book<<" "<<front->author_book<<endl;
```

```

        Print(.....);
    }
}

bool DelData(book_node *&front, string name_del){
    book_node *prev,*current,*p;
    prev = .....;
    current = .....;
    bool flag=false;
    while(current != .....){
        if(current->author_book==name_del){
            p=.....;
            prev->link = .....;
            current=.....;
            delete(p);
            flag=true;
        }
        else{
            prev=.....;
            current=.....;
        }
    }
    return flag;
}

int main(){
    book_node *front,*rear;    string name_del;
    front=NULL; rear=NULL;
    EnQueue(front,rear);      Print(front);
    cout<<"name delete=?"; cin>>name_del;
    if (DelData(front,name_del))
        Print(front);
    else
        cout<<"no data delete";
    return 0;
}

```