**实验五 类与对象**

**一、实验目的与要求**

1、掌握声明类的方法，类和类的成员的概念以及定义对象的方法。

2、初步掌握用类和对象编制基于对象的程序。

3、学习检查和调试基于对象的程序。

**二、实验内容**

1、有以下程序：

#include<iostream>

using namespace std；

class Time // 定义Time类

{

public： // 数据成员为公用的

int hour;

int minute;

int sec ;

};

int main()

{

Time tl; //定义t1为Time类对象

cin>>t1.hour; //输入设定的时间

cin>>t1.minute;

cin>>t1.sec;

cout<<t1.hourl<<”：”<<t1.minute<<”：”<<t1.sec<<endl ;

return 0;

}

改写程序，要求：

(1)将数据成员改为私有的；

(2)将输入和输出的功能改为由成员函数实现；

(3)在类体内定义成员函数。

然后编译和运行程序。请分析什么成员应指定为公用的?什么成员应指定为私有的? 什么函数最好放在类中定义? 什么函数最好在类外定义?

2、分别给出如下的3个文件：

(1)含类定义的头文件student.h，

//student.h (这是头文件，在此文件中进行类的声明)

class Student //类声明

{

public: //公用成员函数原型声明

void display();

private:

int num;

char name[20];

char sex;

};

(2)包含成员函数定义的源文件student.cpp

//student.cpp 在此文件中进行函数的定义

#include <iostream>

#include”student.h” //不要漏写此行，否则编译通不过

void Student::display() //在类外定义display类函数

{

cout<<”num：”<<num<<endl;

cout<<”name：”<<name<<endl;

cout<<”sex：”<<sex<<endl;

}

(3)包含主函数的源文件main.cpp。

为了组成一个完整的源程序，应当有包括主函数的源文件：

//main.cpp 主函数模块

#include <iostream> //将类声明头文件包含进来

#include “student.h”

int main()

{

Student stud; //定义对象

Student stud1(007,”tcg”,’m’);

stud.display(); //执行stud对象的display函数

return 0;

}

请完善该程序，在类中增加一个对数据成员赋初值的成员函数set\_value。上机调试并运行。

3、需要求3个长方柱的体积，请编一个基于对象的程序。数据成员包括length(长)、width(宽)、height(高)。要求用成员函数实现以下功能：

(1)由键盘分别输入3个长方柱的长、宽、高；

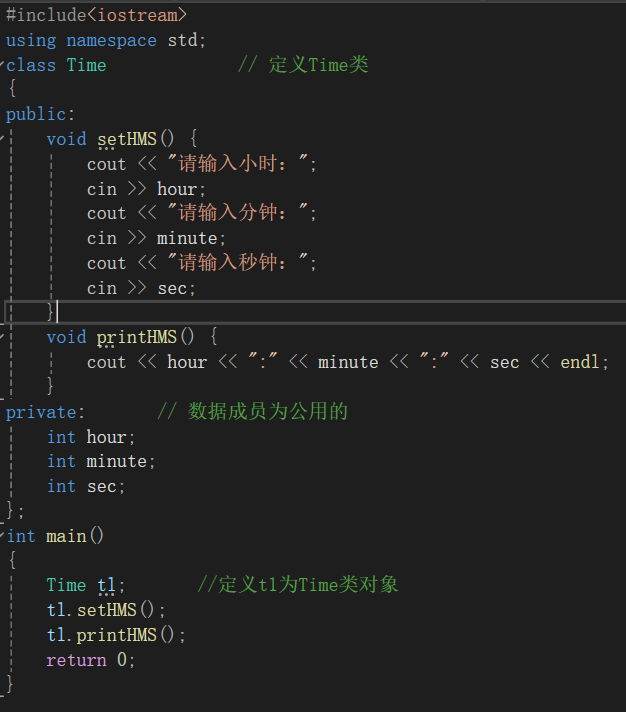
(2)计算长方柱的体积；

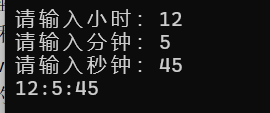
(3)输出3个长方柱的体积。

请编程序，上机调试并运行。

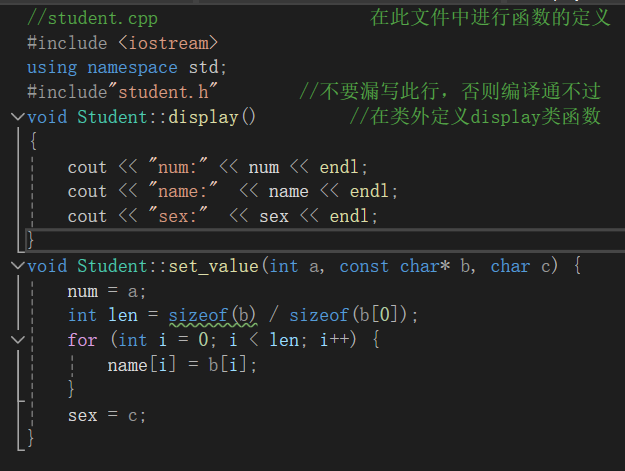
4、建立一个对象数组，内放5个学生的（学号，成绩），设立一个函数max，用指向对象的指针作函数参数，在max函数中找出5个学生的最高成绩者，并输出其学号。

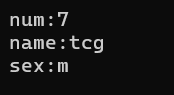
1. 设有一描述坐标点的类Point，其私有变量x和y代表一个点的(x,y)坐标值。请编写程序实现以下功能：利用构造函数传递参数，在定义对象时将x、y坐标值初始化为（60,80）；利用公有成员函数void setPoint(int i, int j)将坐标值修改为(60+i,80+j)；利用公有成员函数display()输出修改后的坐标值。主函数中通过定义对象，验证各个函数。
2. **算法分析，程序结果**
3. 要将数据成员私用，只需要在程序段中加上private关键字即可。再通过get和set函数来对数据成员进行访问和修改。在程序设计中，公用的成员一般应设置为成员函数，对外提供访问该类数据的方法；私有的成员一般设置为数据成员，进行保护，防止类外随意的修改。构造函数和析构函数以及简单的函数一般在类中定义；实现较为复杂的函数和友元函数以及静态成员函数一般放在类外定义，实现较为复杂的函数在类外定义可以减小程序的大小，提高性能；而静态成员函数通常在类外定义，因为它们不依赖于类的实例。

程序：

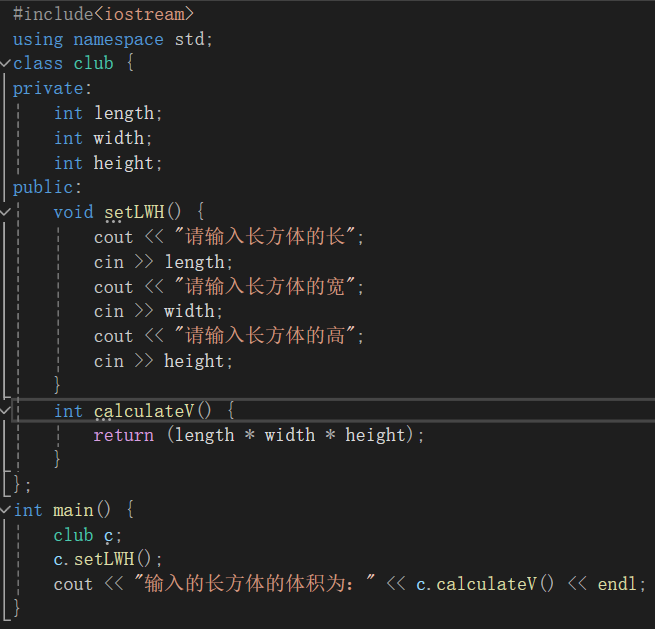
结果：

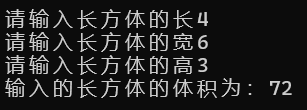
1. 增加成员函数set\_value，需要在student.h文件中增加该函数的声明，然后在student.cpp中增加其定义。

程序：

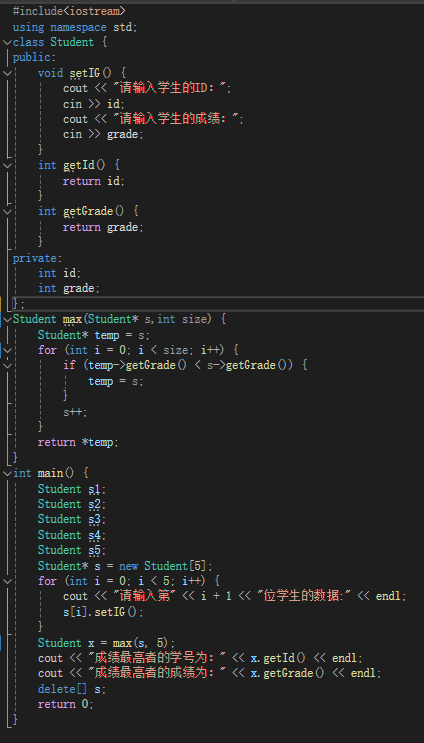
结果：

1. 定义长方形类club，将长宽高属性私有，提供setLWH函数来输入长宽高以及calculateV来计算体积。

程序：

结果：

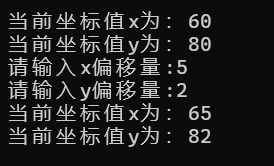
1. 定义类Student，私有化学号和成绩，通过函数setIG来输入学号和成绩，以及相应的get函数来获取学号和成绩。定义全局函数max，参数为指向Student数组的首个元素的指针和数组的长度，然后利用指针循环遍历每个学生，比较找出成绩最高的学生并返回，然后在主函数中输出。

程序：

结果：

1. 定义类Point，私有化其数据成员x,y然后提供有参构造函数来输入x,y，以及display函数来输入x,y的偏移量。在主函数中初始化对象p然后调用display来输入偏移量并输出改变后的x,y。

程序：

结果:

1. **遇到的问题与解决方法**

在问题2中set\_value函数的设计存在问题，即在函数中用什么类型来接收字符串”tcg”，了解到字符串实际上被定义为const char数组，因此用const char\*类型来接收，然后通过for循环对name数组赋值；在问题4中问题出在max函数的设计，首先是形参，应该传入Student数组，即数组首个元素的指针，但是应该还需要传入数组的长度，因为我在主函数中定义的是动态数组，此时无法通过sizeof(s)/sizeof(s[0])来获取数组的长度，因为对于动态分配的数组，sizeof(s)将返回该指针的大小，因此发生错误，所以需要传入数组的长度以通过指针遍历数组，并将成绩最大的学生的指针赋值给temp，然后返回该学生对象\*temp。

1. **体会**

类中定义的函数均为内联函数，特点是在编译过程中，编译器会将内联函数的函数体复制到内联函数的调用点处，相当于不用执行进入函数的步骤，直接执行函数体，可以提高程序的运行速度，不过需要合理使用，程序体过大的内联函数会消耗大量的内存空间，因此要合理的设计类中成员函数的定义位置。在函数中传入的类的数组实际上是传入指向数组首个元素的指针，不管该数组是在堆中创建的还是栈中创建的，因此通过sizeof关键字计算得到的是指针的大小，而非整个数组的大小。要通过指针来遍历数组每个索引的数据，对于存储的类数组而言，获取到指向其的指针后可以通过箭头运算符(->)来直接访问类中成员。