**实验五 类与对象**

**【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。

**【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。

5、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。

代码与测试结果

1.

#include<iostream>

using namespace std;

class Time // 定义time类

{

private:

int m\_hour;

int m\_minute;

int m\_sec;

public:

void sethour()

{

int hour;

cin >> hour;

m\_hour = hour;

}

int gethour()

{

return m\_hour;

}

void setminu()

{

int minute;

cin >> minute;

m\_minute = minute;

}

int getminu()

{

return m\_minute;

}

void setsec()

{

int sec;

cin >> sec;

m\_sec = sec;

}

int getsec()

{

return m\_sec;

}

};

int main()

{

Time t1;

cout << "输入小时" << endl;

t1.sethour();

cout << "输入分钟" << endl;

t1.setminu();

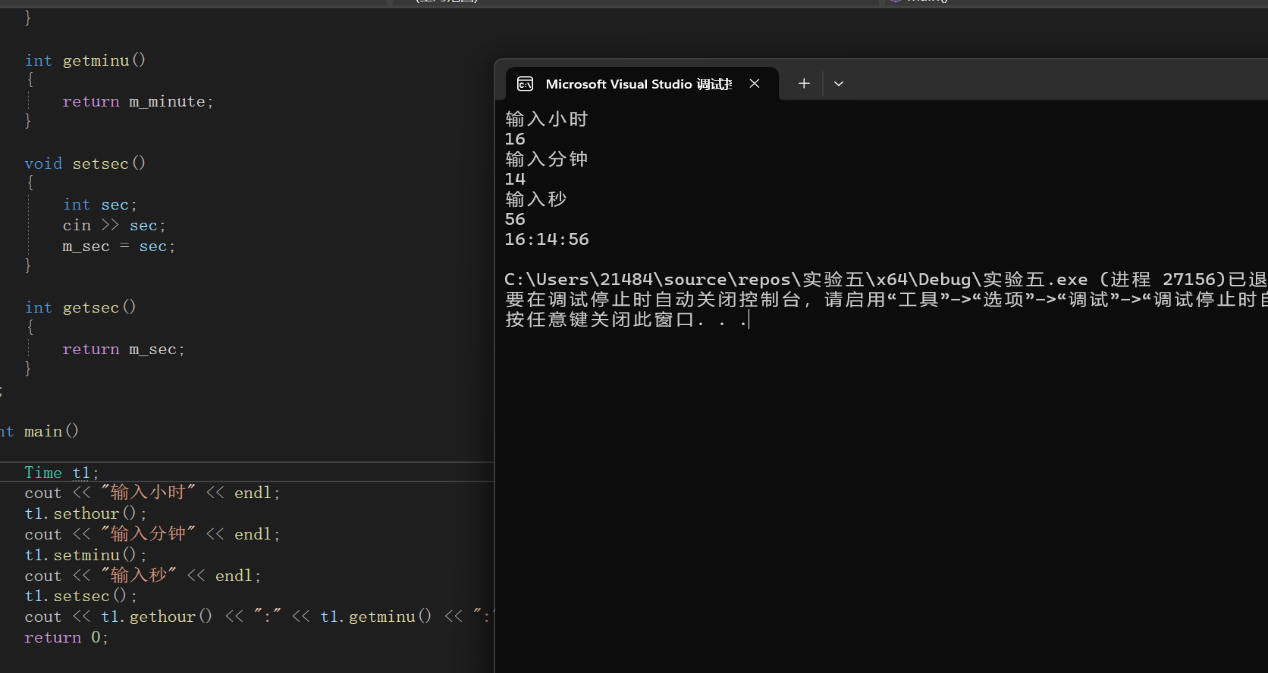
cout << "输入秒" << endl;

t1.setsec();

cout << t1.gethour() << ":" << t1.getminu() << ":" << t1.getsec() << endl;

return 0;

}

2.

#include<iostream>

#include"student.h"

using namespace std;

int main()

{

Student stud;

stud.setvalue();

stud.display();

return 0;

}

#include <iostream>

#include"student.h"

using namespace std;

void Student::display()

{

cout << "num:" << getnum() << endl;

cout << "name:" << getname() << endl;

cout << "sex:" << getsex() << endl;

}

void Student::setvalue()

{

string tnum;

string tname;

string tsex;

cout << "输入学号" << endl;

cin >> tnum;

cout << "输入姓名" << endl;

cin >> tname;

cout << "输入性别" << endl;

cin >> tsex;

num = tnum;

name = tname;

sex = tsex;

}

string Student::getnum()

{

return num;

}

string Student::getname()

{

return name;

}

string Student::getsex()

{

return sex;

}

#pragma once

#include<iostream>

using namespace std;

class Student

{

public:

void display();

void setvalue();

string getnum();

string getname();

string getsex();

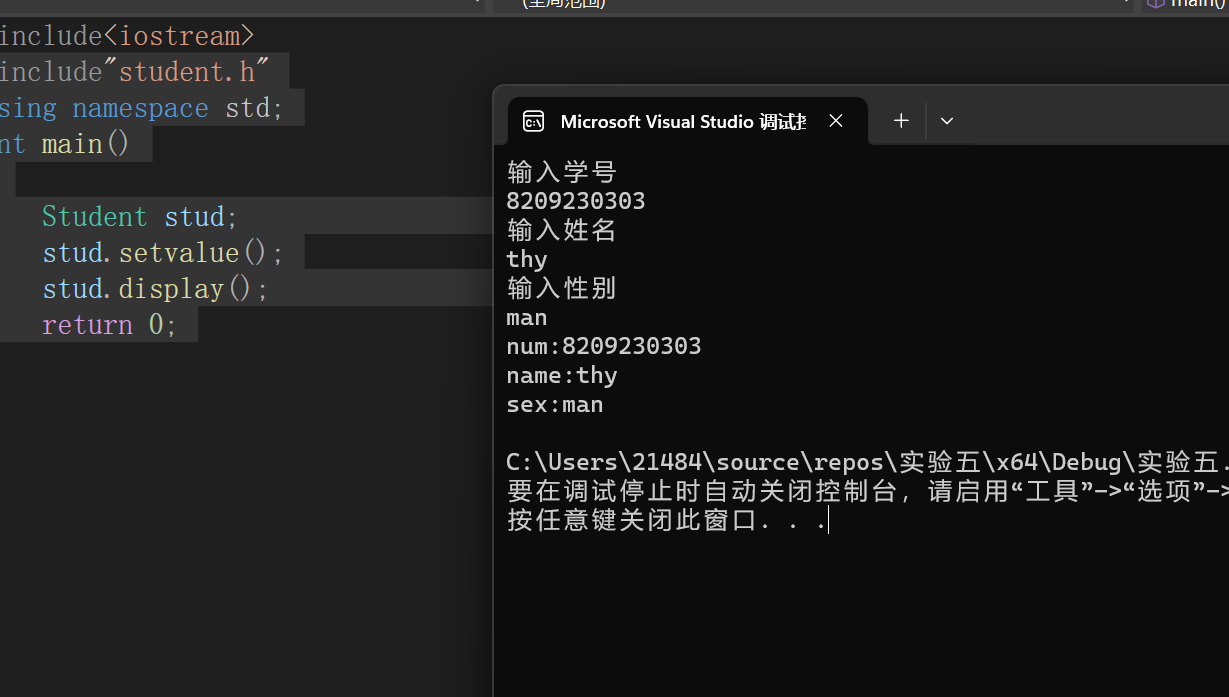
private:

string num;

string name;

string sex;

};

3.

#include<iostream>

using namespace std;

#include<string>

class Cube {

private:

int m\_l;

int m\_w;

int m\_h;

public:

void Setlength()

{

int length;

cout << "输入长" << endl;

cin >> length;

m\_l = length;

}

void Setwidth()

{

int width;

cout << "输入宽" << endl;

cin >> width;

m\_w = width;

}

void Setheight()

{

int height;

cout << "输入高" << endl;

cin >> height;

m\_h = height;

}

int volum()

{

return m\_l \* m\_w \* m\_h;

}

void display()

{

cout << "体积为" << volum() << endl;

}

};

int main()

{

Cube c1;

c1.Setlength();

c1.Setwidth();

c1.Setheight();

c1.display();

Cube c2;

c2.Setlength();

c2.Setwidth();

c2.Setheight();

c2.display();

Cube c3;

c3.Setlength();

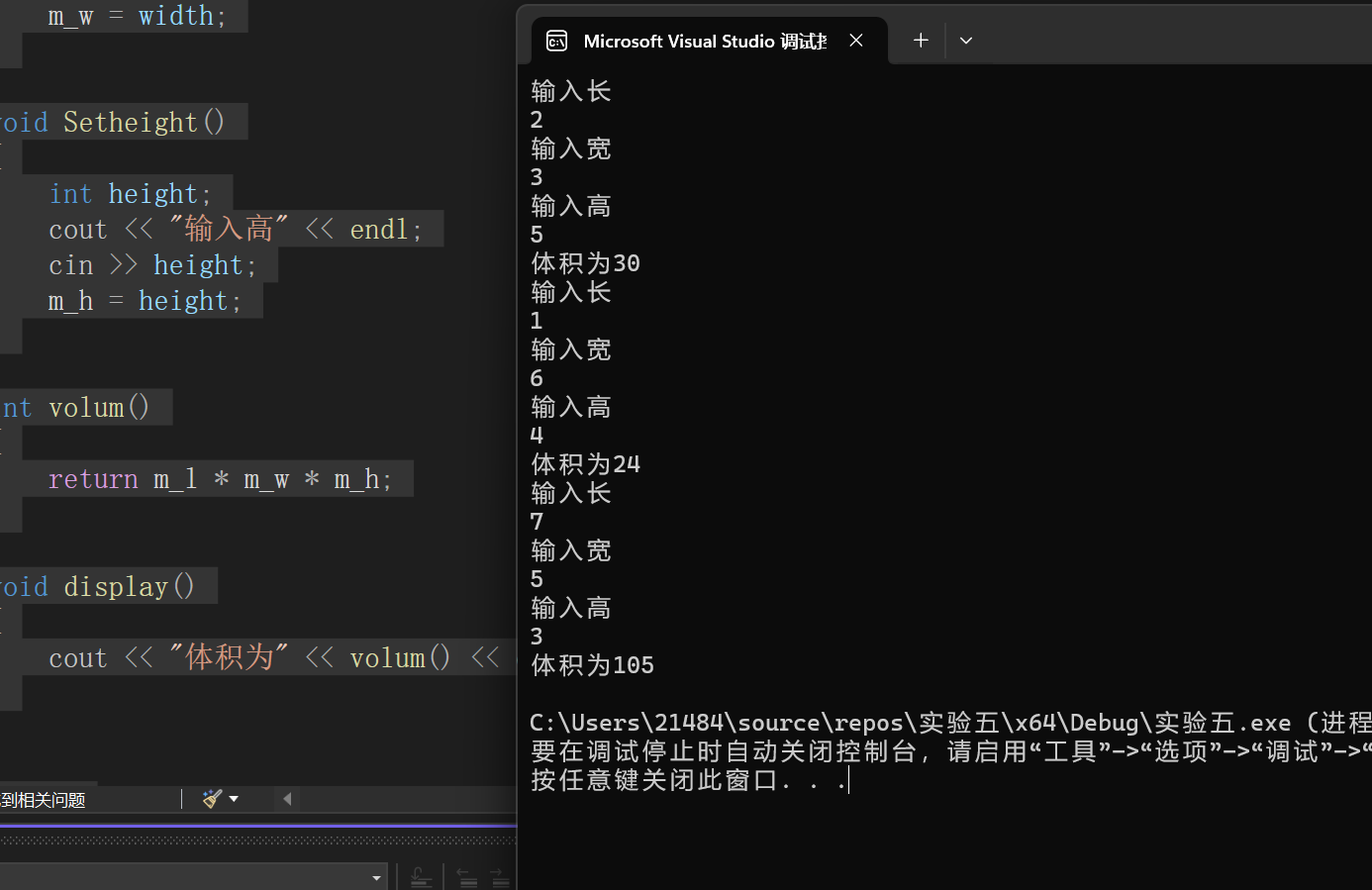
c3.Setwidth();

c3.Setheight();

c3.display();

return 0;

}

4.

#include<iostream>

using namespace std;

#include<string>

class Student {

public:

int m\_id;

int m\_score;

Student(int id, int score) :m\_id(id), m\_score(score)

{

}

};

int max(Student stu[])

{

Student s(0,0);

Student\* p = stu;

for (int i = 0; i < 5; i++)

{

if (s.m\_score < p[i].m\_score)

{

s.m\_score = p[i].m\_score;

s.m\_id = p[i].m\_id;

}

}

return s.m\_id;

}

int main()

{

Student stu[5] = {

Student(315,91),

Student(1351,56),

Student(153,89),

Student(34,88),

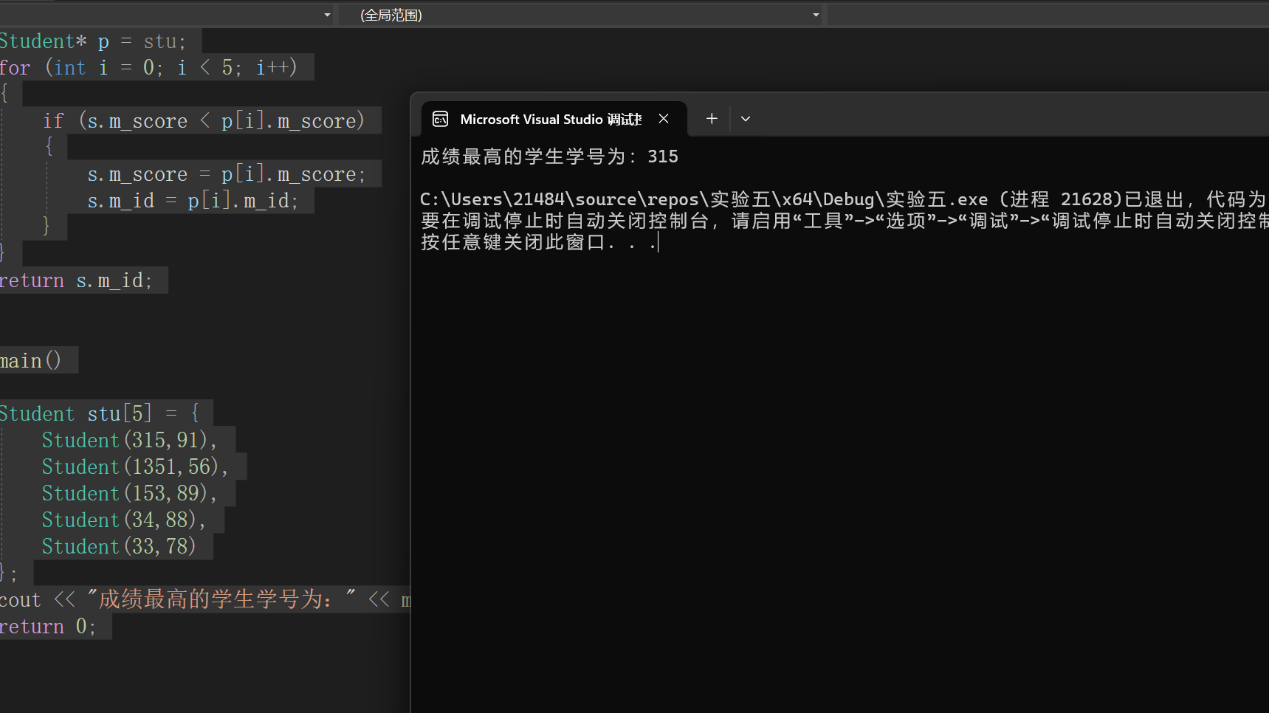
Student(33,78)

};

cout << "成绩最高的学生学号为：" << max(stu) << endl;

return 0;

}

5.

#include<iostream>

using namespace std;

class Point {

public:

int m\_x;

int m\_y;

Point() :m\_x(60), m\_y(80) {

}

void setpoint(int i,int j)

{

m\_x = m\_x + i;

m\_y = m\_y + j;

}

void display()

{

cout << "坐标为：" << "(" << m\_x << "," << m\_y << ")" << endl;

}

};

int main()

{

Point p;

p.setpoint(2, 3);

p.display();

return 0;

}

