**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?

成员函数应指定为公用的，便于外部调用。数据成员应指定为为私有的，防止外部随意修改造成程序混乱。成员函数最好放在类内定义，通用的函数最好放在类外定义。

#include<iostream>

using namespace std;

class Time // 定义Time类

{

public:

void setTime()//在类体内定义成员函数

{

cout << "请输入设定的小时：";

cin >> hour;

cout << "请输入设定的分钟：";

cin >> minute;

cout << "请输入设定的秒：";

cin >> sec;

}

void showTime()

{

cout << "设定的时间为：" << hour << ":" << minute << ":" << sec << endl;

}

private: // 将数据成员改为私有的

int hour;

int minute;

int sec;

};

int main()

{

Time t1; //定义t1为Time类对象

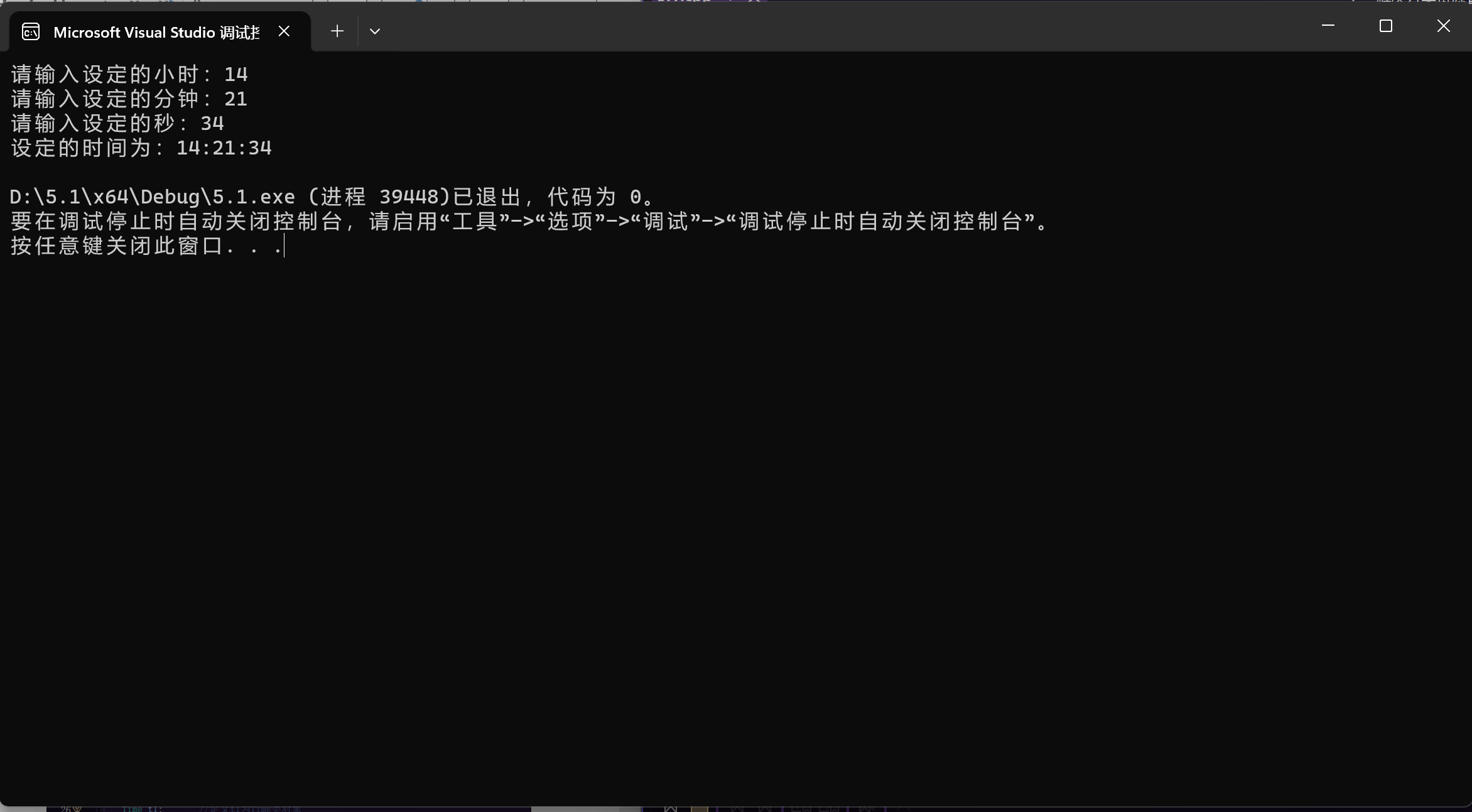
t1.setTime();//输入功能用成员函数实现

t1.showTime();//输出功能用成员函数实现

return 0;

}





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。

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();

void set\_value();

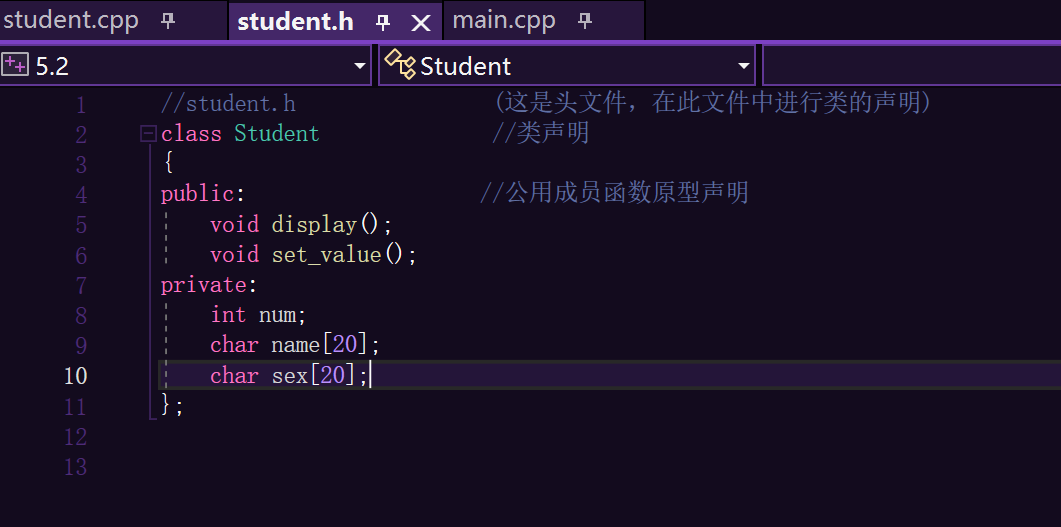
private:

int num;

char name[20];

char sex[20];

};



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

using namespace std;

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

{

cout << "num：" << num << endl;

cout << "name：" << name << endl;

cout << "sex：" << sex << endl;

}

void Student::set\_value()//在类外定义set\_value函数

{

cout << "请赋值：" << endl;

cout << "num：";

cin >> num;

cout << "name：";

cin >> name;

cout << "sex：";

cin >> sex;

}



包含主函数的源文件main.cpp

//main.cpp 主函数模块

#include <iostream> //将类声明头文件包含进来

#include"student.h"

using namespace std;

int main()

{

Student stud; //定义对象

stud.set\_value(); //执行stud对象的set\_value属性

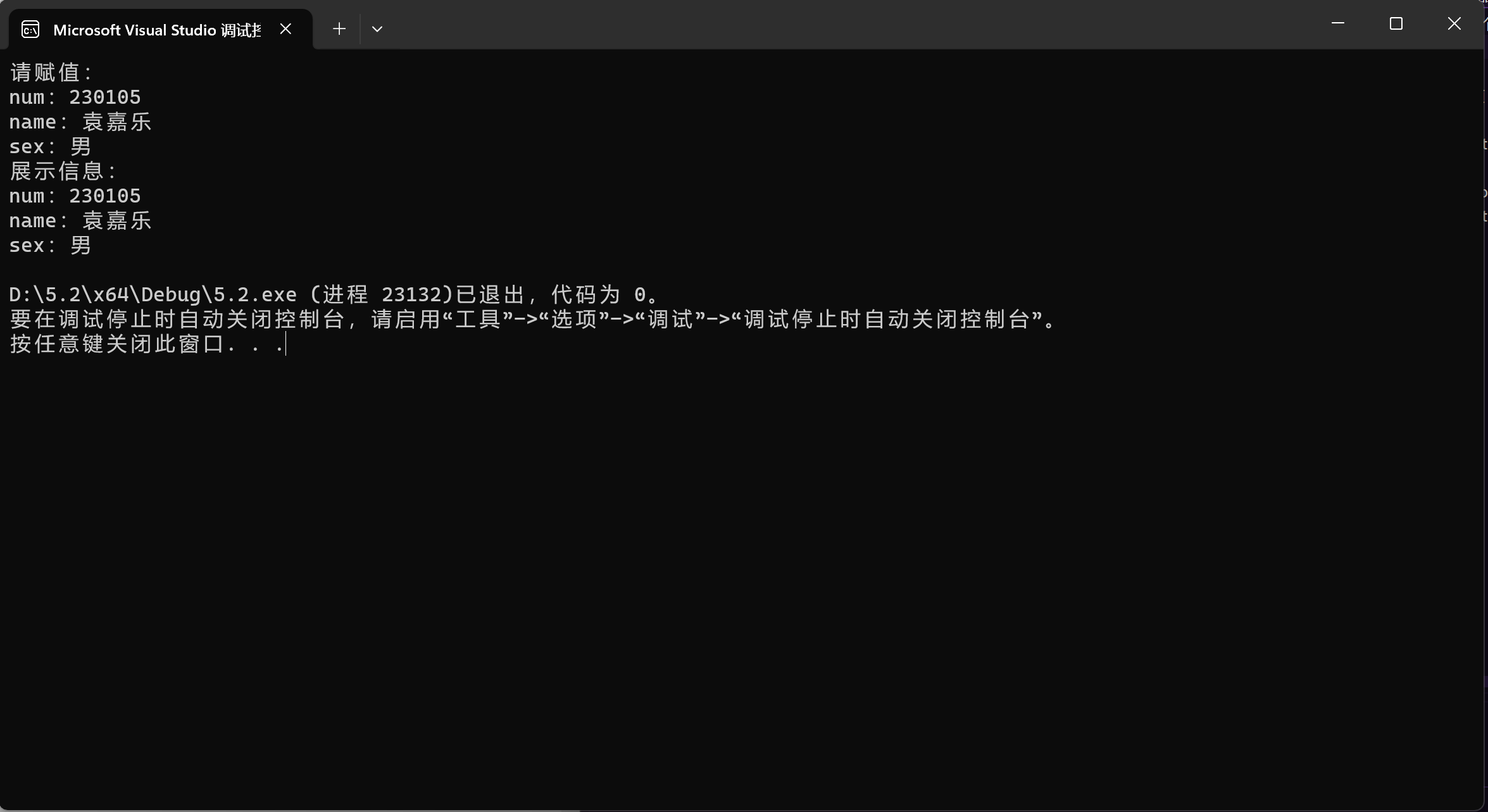
cout << "展示信息：" << endl;

stud.display(); //执行stud对象的display函数

return 0;

}





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。

#include<iostream>

using namespace std;

class cuboid

{

private:

float length, width, height,V;

public:

void setCuboid()//用成员函数输入长方柱的长宽高

{

cout << "请输入长方柱的长：";

cin >> length;

cout << "请输入长方柱的宽：";

cin >> width;

cout << "请输入长方柱的高：";

cin >> height;

}

void volume() //用成员函数计算长方柱的体积

{

V = length \* width \* height;

}

void showVolume()//用成员函数输出3个长方柱的体积

{

cout <<V<< endl;

}

};

int main()

{

cuboid c1, c2, c3;

cout << "请分别输入三个长方柱的长，宽，高：" << endl;//调用成员函数分别输入长宽高

c1.setCuboid();

c2.setCuboid();

c3.setCuboid();

cout << "第一个长方柱体积为：" ;

c1.volume();

c1.showVolume();

cout << "第二个长方柱体积为：" ;

c2.volume();

c2.showVolume();

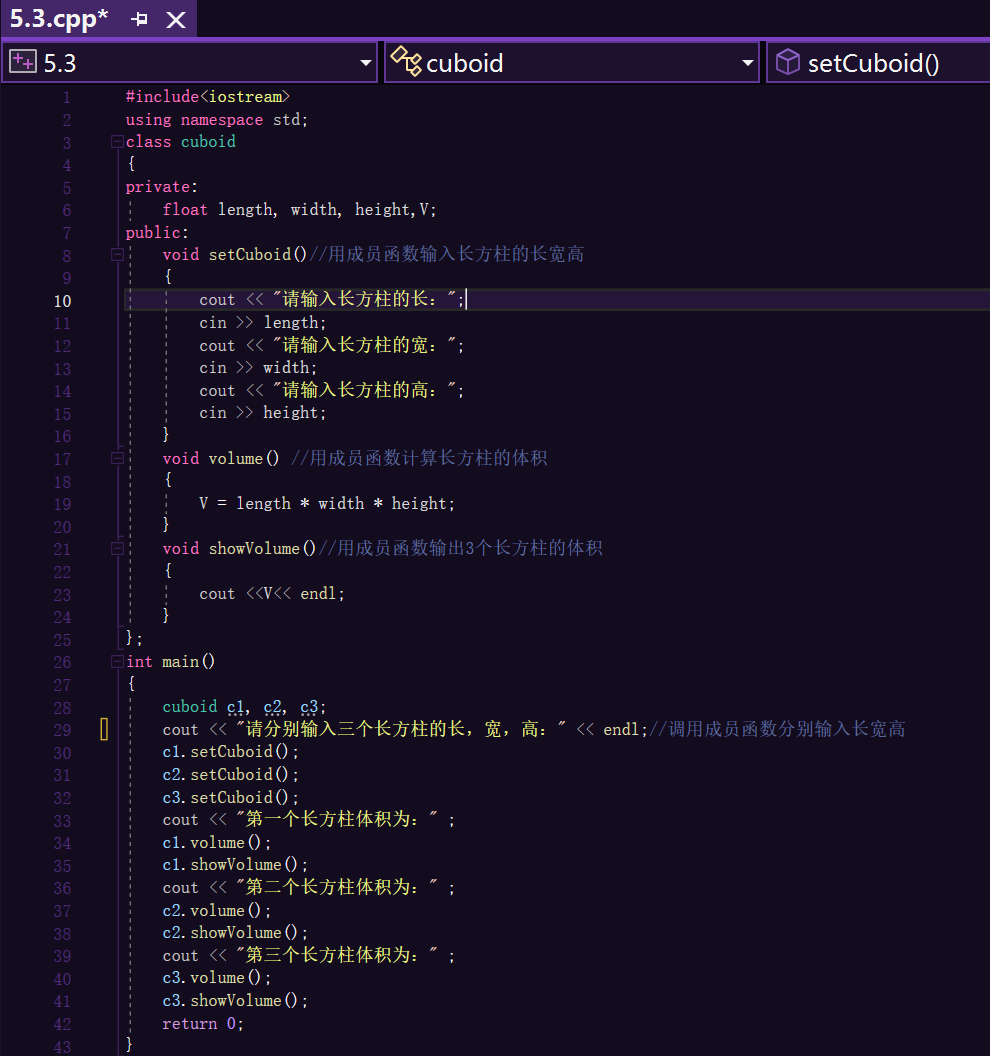
cout << "第三个长方柱体积为：" ;

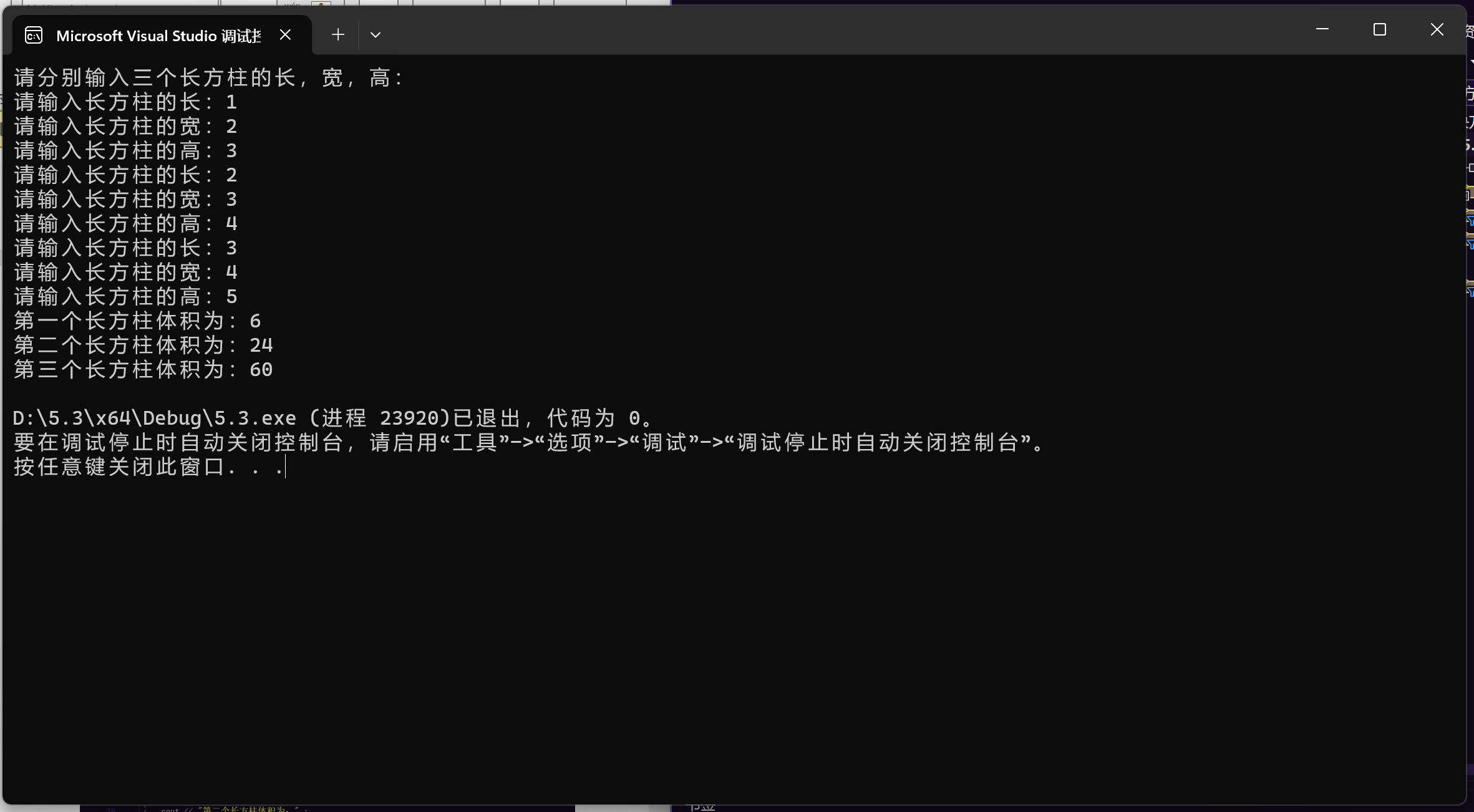
c3.volume();

c3.showVolume();

return 0;

}





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。

#include<iostream>

using namespace std;

class stu

{

public:

int num;

int score;

stu(int Num, int Score)

{

num = Num;

score = Score;

}

};

int max(stu\* a)//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

{

for (int i = 4; i >= 0; i--)

{

if (i == 0)

{

return a[i].num;

}

if (a[i].score > a[i - 1].score) {

a[i - 1].num = a[i].num;

a[i - 1].score = a[i].score;

}

}

}

int main()

{

stu five[5] = {//建立一个对象数组

stu(230101,83),

stu(230102,86),

stu(230103,91),

stu(230104,89),

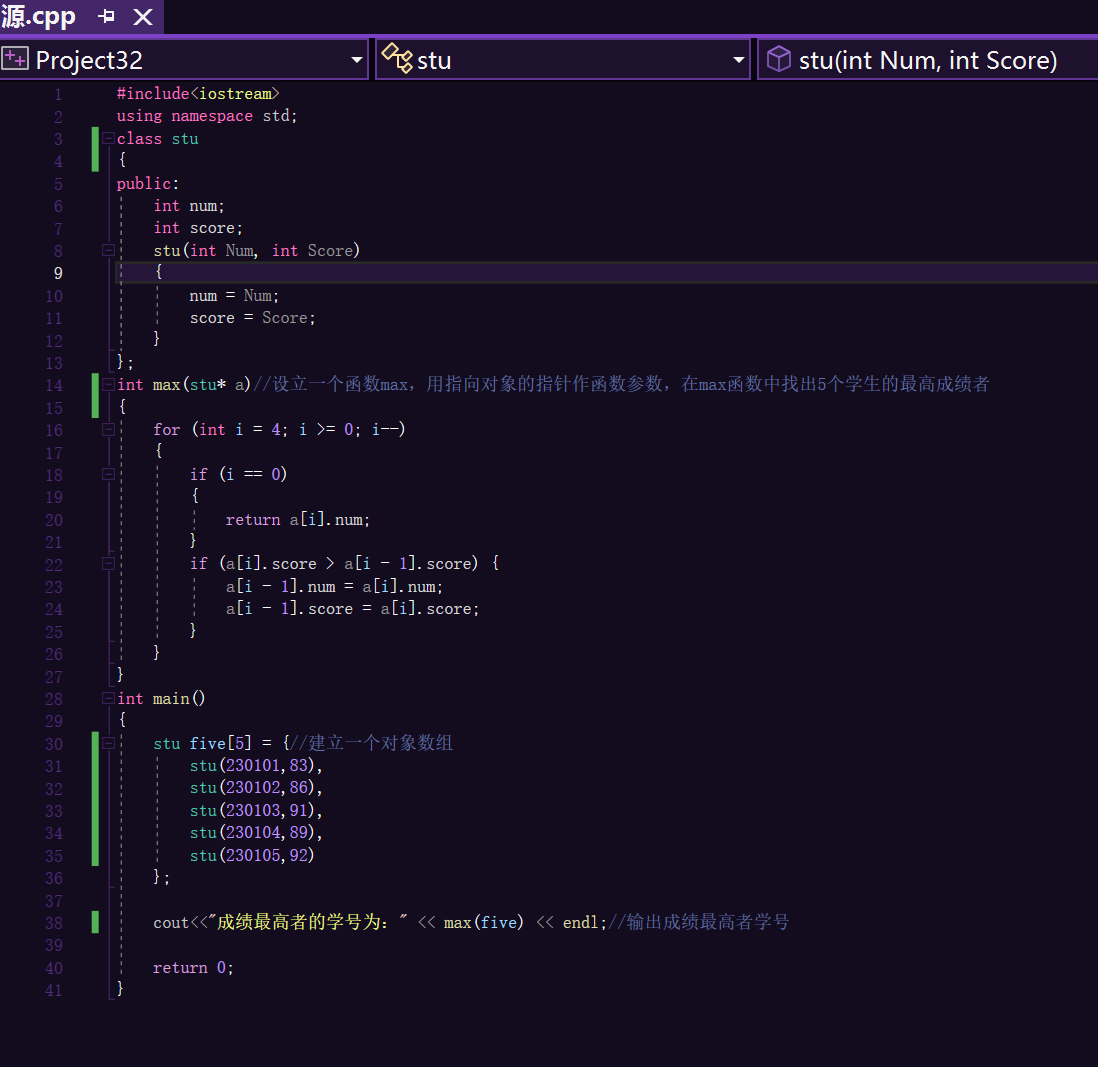
stu(230105,92)

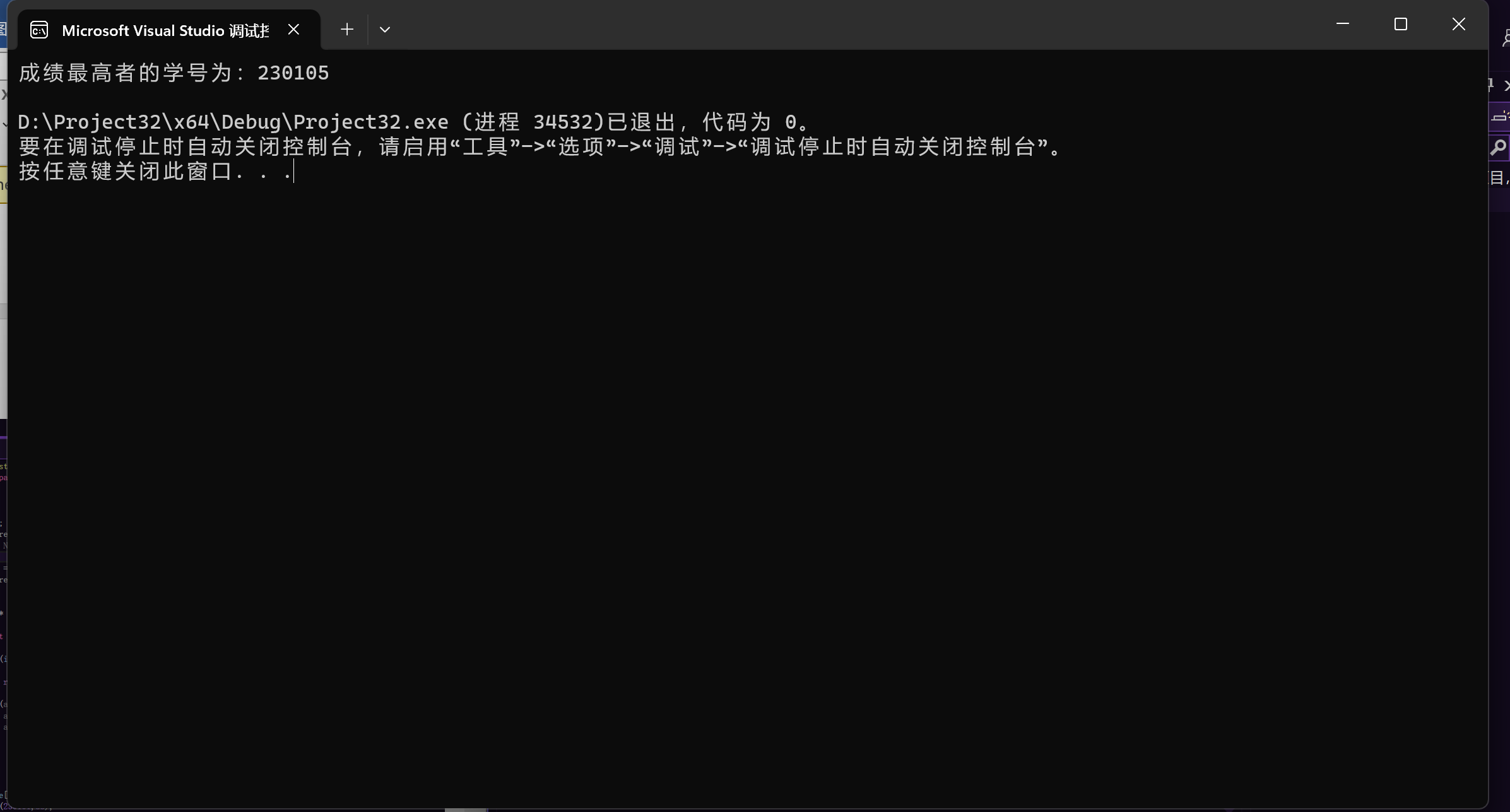
};

cout<<"成绩最高者的学号为：" << max(five) << endl;//输出成绩最高者学号

return 0;

}





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。

#include<iostream>

using namespace std;

class Point//设有一描述坐标点的类Point

{

private:

int x;//私有变量x和y代表一个点的(x,y)坐标值

int y;

public:

Point() {//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）

x = 60;

y = 80;

}

void setPoint(int i, int j)//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)

{

x += i;

y += j;

}

void display()//利用公有成员函数display()输出修改后的坐标值

{

cout << "(" << x << "," << y << ")" << endl;

}

};

int main()

{

Point p;

p.display();//验证构造函数是否发挥作用

p.setPoint(5, 6);//验证公有成员函数void setPoint是否发挥作用

p.display();//验证公有成员函数display是否发挥作用

return 0;

}



