5.1

#include<iostream>

using namespace std;

class Time // 定义Time类

{

private: // 数据成员为公用的

int hour = 0;

int minute = 0;

int sec = 0;

public:

int set\_time(int h,int m,int s)

{

cin >> h >> m >> s;

hour = h;

minute = m;

sec = s;

return 0;

}

int showtime()

{

cout << hour << ":" << minute << ":" << sec << endl;

return 0;

}

};

int main()

{

Time t1; //定义t1为Time类对象

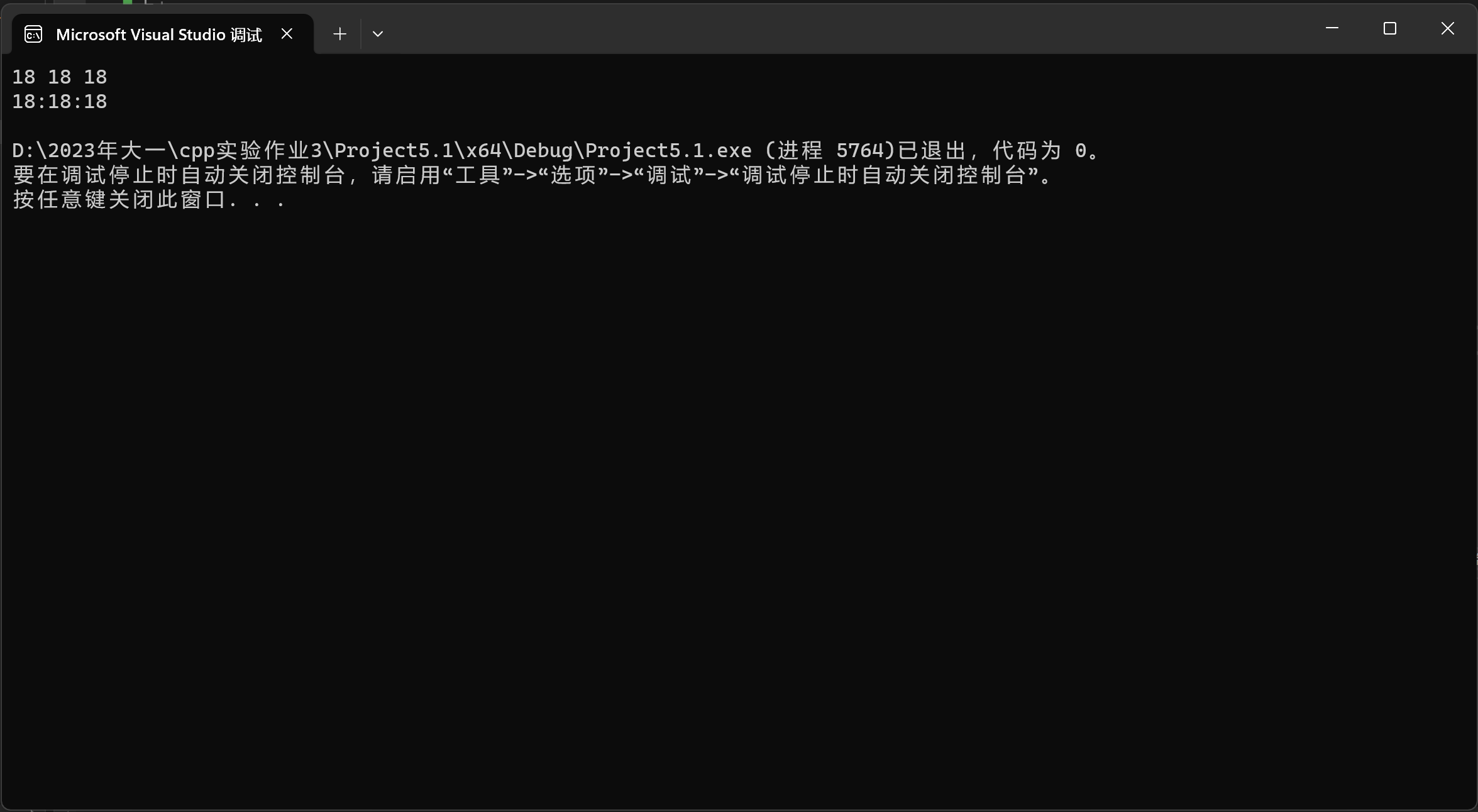
int h = 0, m = 0, s = 0;

t1.set\_time(h,m,s);

t1.showtime();

return 0;

}



成员函数应指定为公用的

成员变量应制定为私有的

简单函数最好放在类中定义

复杂函数最好放在类外定义

5.2

Student.h

//student.h (这是头文件，在此文件中进行类的声明)

#pragma once

class Student //类声明

{

public: //公用成员函数原型声明

void display();

void set\_value(int x, const char y[20], char z);

private:

int num;

char name[20];

char sex;

};

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(int x,const char y[20], char z)

{

num = x;

strcpy\_s(name, y);

sex = z;

}

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()

{

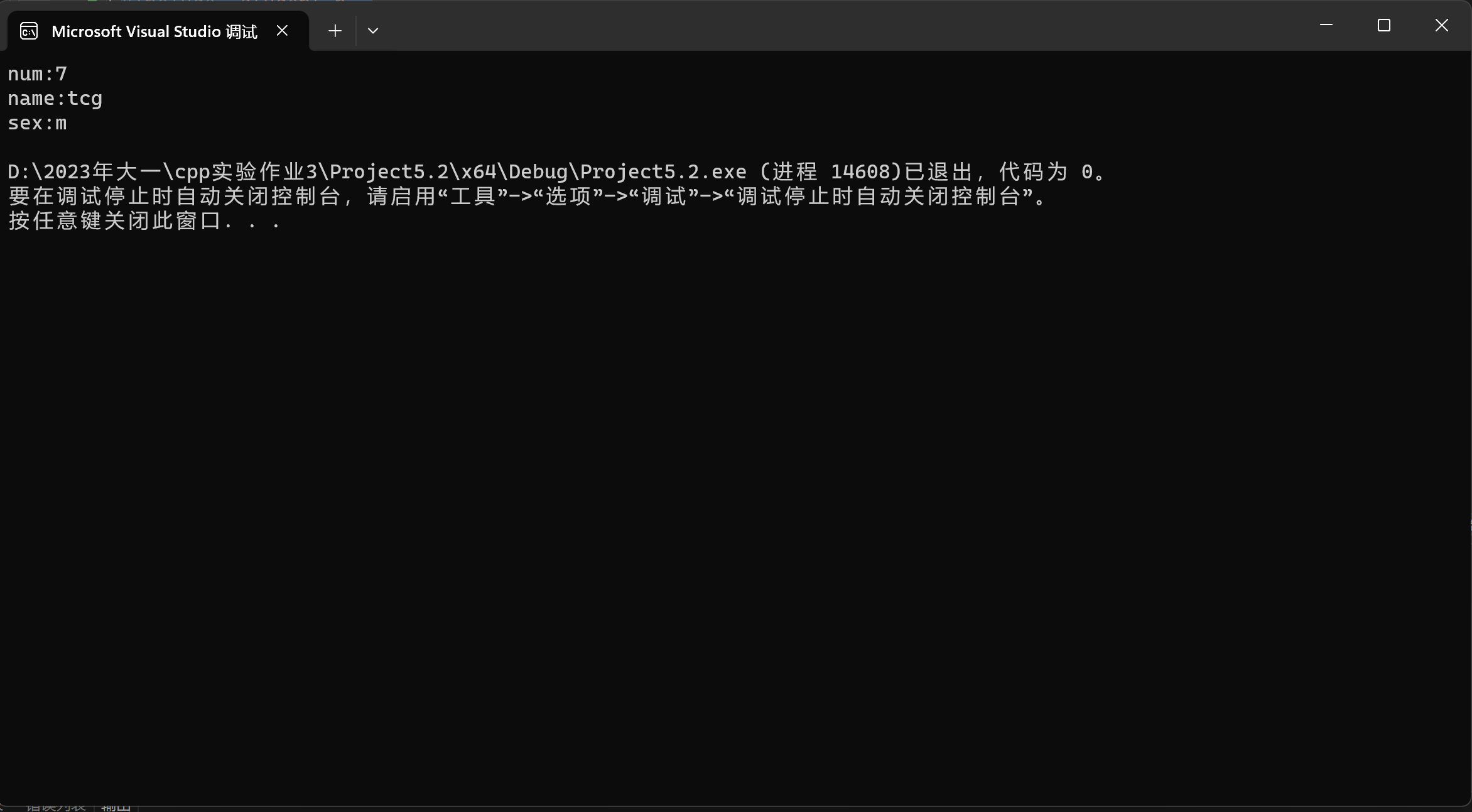
Student stud; //定义对象

stud.set\_value(007,"tcg", 'm');

stud.display(); //执行stud对象的display函数

return 0;

}



5.3

#include<iostream>

class cuboid

{

private:

int length, width, height, V;

public:

void cin\_data()

{

std::cin >> length >> width >> height;

}

void count()

{

V = length \* width \* height;

}

void print\_V()

{

std::cout << V << std::endl;

}

};

int main()

{

cuboid c1, c2, c3;

c1.cin\_data();

c1.count();

c1.print\_V();

c2.cin\_data();

c2.count();

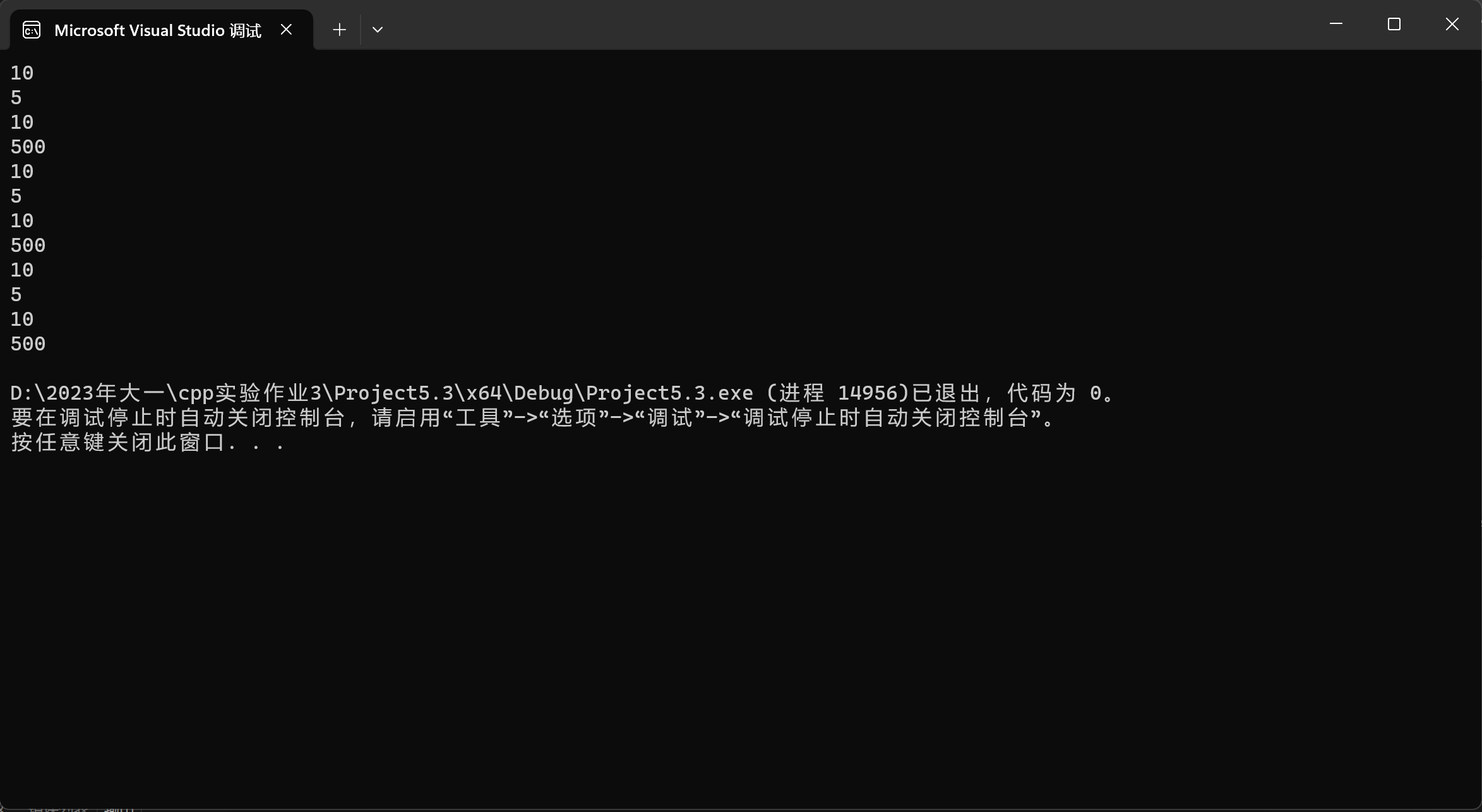
c2.print\_V();

c3.cin\_data();

c3.count();

c3.print\_V();

return 0;

}

5.4

class Student

{

public:

Student(string n, double s) : num(n), score(s) {}

string num;

double score;

};

int main()

{

void max(Student\*);

Student s[5] =

{

Student("1",60),

Student("2",70),

Student("3",80),

Student("4",90),

Student("5",100)

};

max(s);

return 0;

}

void max(Student\* p)

{

Student\* t;

Student\* c;

double max;

for (t = p, c = t, max = t->score; t < (p + 5); ++t)

{

if (t->score > max)

{

max = t->score;

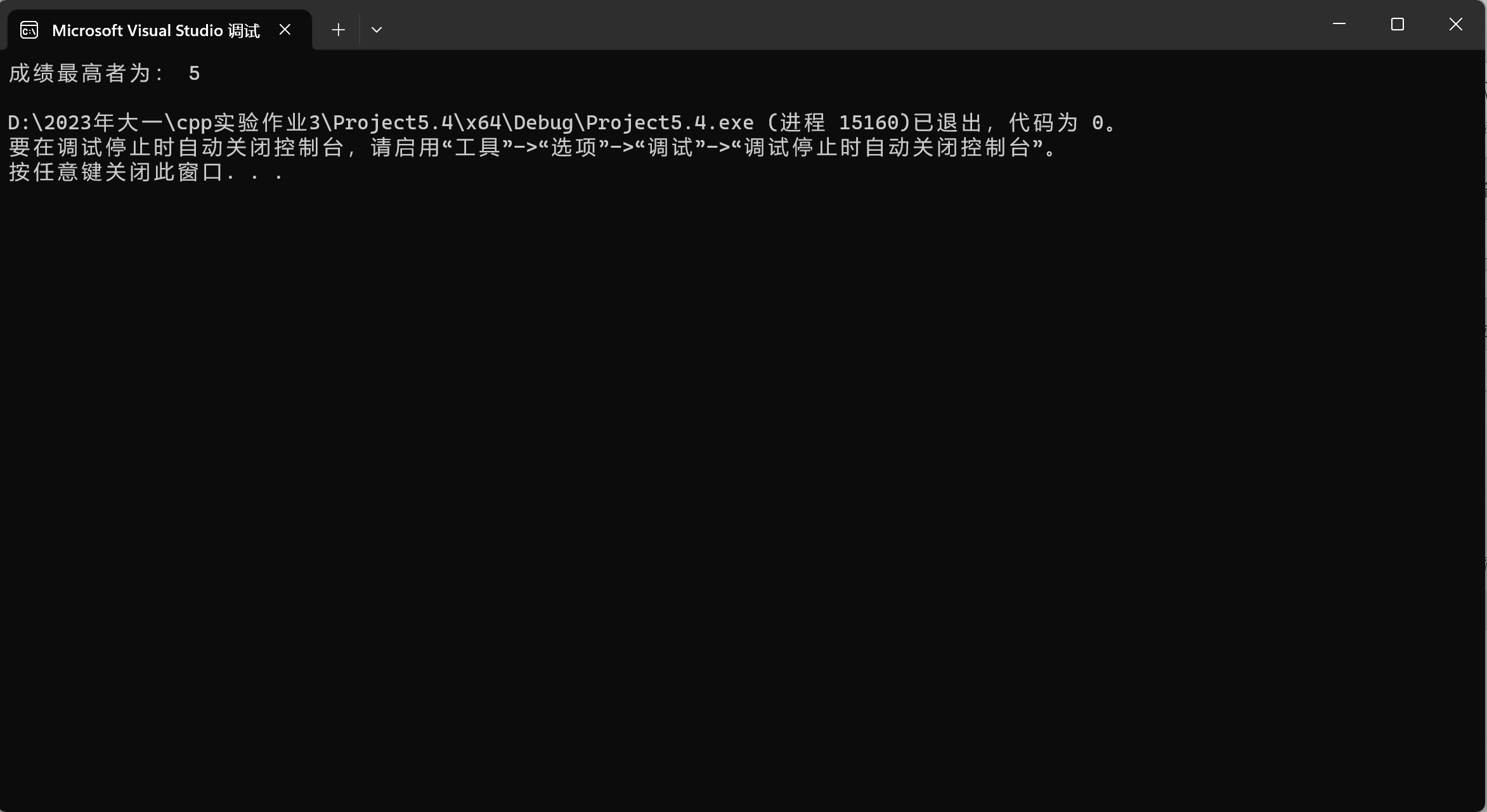
c = t;

}

}

cout << "成绩最高者为： " << c->num << endl;

}



5.5

#include<iostream>

class Point

{

private:

int x, y;

public:

Point(int xa = 0,int ya = 0)

{

x = xa;

y = ya;

}

void setPoint()

{

int x1,y1;

std::cin >> x1 >> y1;

x = x + x1;

y = y + y1;

}

void display()

{

std::cout << "(" << x << "," << y << ")";

}

};

int main()

{

Point p(60, 80);

p.setPoint();

p.display();

return 0;

}

