**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。

姓名：周涛

班级：软工2404

学号：8209240402

**【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。

1. #include<iostream>

using namespace std;

class Time

{

private:

int hour;

int minute;

int sec;

public:

void settime(int HOUR, int MINUTE,int SECOND)

{

hour = HOUR;

minute = MINUTE;

sec = SECOND;

}

void setout()

{

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

}

};

int main()

{

Time t1;

int a, b, c;

cout << "Please enter the hour:"<<endl;

cin >> a;

cout << "Please enter the minute:"<<endl;

cin >> b;

cout << "Please enter the second" << endl;;

cin >> c;

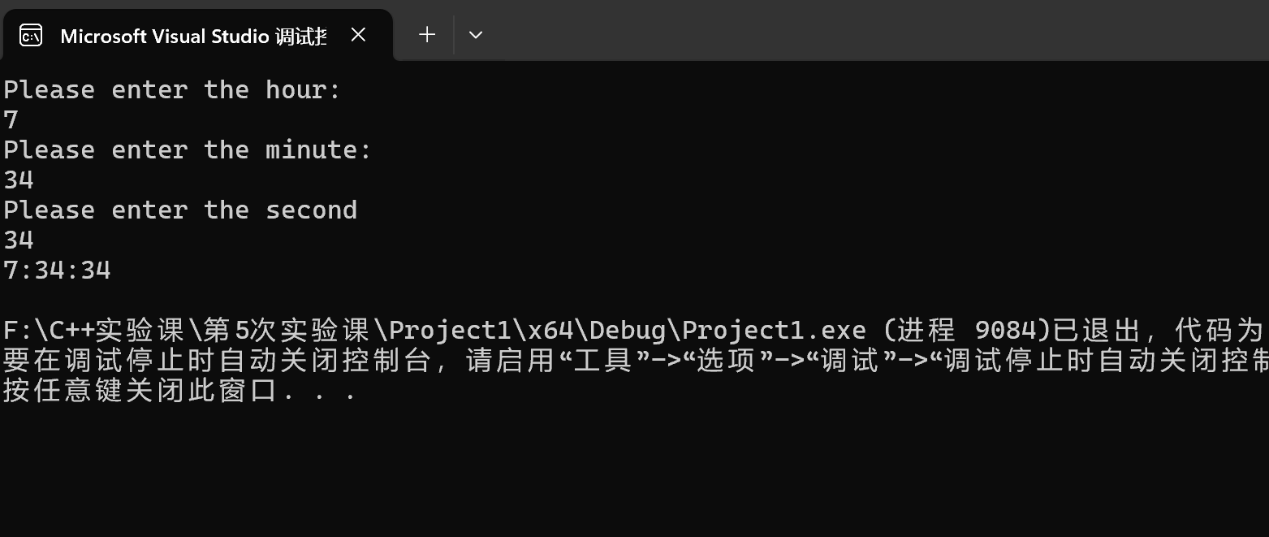
t1.settime(a, b, c);

t1.setout();

return 0;

}

hour,minute,sec设为私有实现封装和保护，提供输入，输出的成员函数以及其输入的参数设为公有，用来提供交互的接口，输入输出这种简单的函数最好在类内定义，可以直接调用类内成员，复杂的，重复使用的最好在类外定义，方便重复使用。

2.project2.h

class Student

{

public:

Student();

Student(int n, const char\* nam, char s);

void display();

void set\_value(int n, const char\* nam, char s);

private:

int num;

char name[20];

char sex;

};

Project2student.cpp

#include <iostream>

#include "project2.h"

Student::Student()

{

num = 0;

name[0] = '\0';

sex = '\0';

}

Student::Student(int n, const char\* nam, char s)

{

num = n;

int i = 0;

while (nam[i] != '\0' && i < 19) {

name[i] = nam[i];

i++;

}

name[i] = '\0';

sex = s;

}

void Student::display()

{

std::cout << "num: " << num << std::endl;

std::cout << "name: " << name << std::endl;

std::cout << "sex: " << sex << std::endl;

}

void Student::set\_value(int n, const char\* nam, char s)

{

num = n;

int i = 0;

while (nam[i] != '\0' && i < 19) {

name[i] = nam[i];

i++;

}

name[i] = '\0';

sex = s;

}

Project2源.cpp

#include <iostream>

#include "project2.h"

int main()

{

Student stud;

stud.set\_value(101, "Bob", 'M');

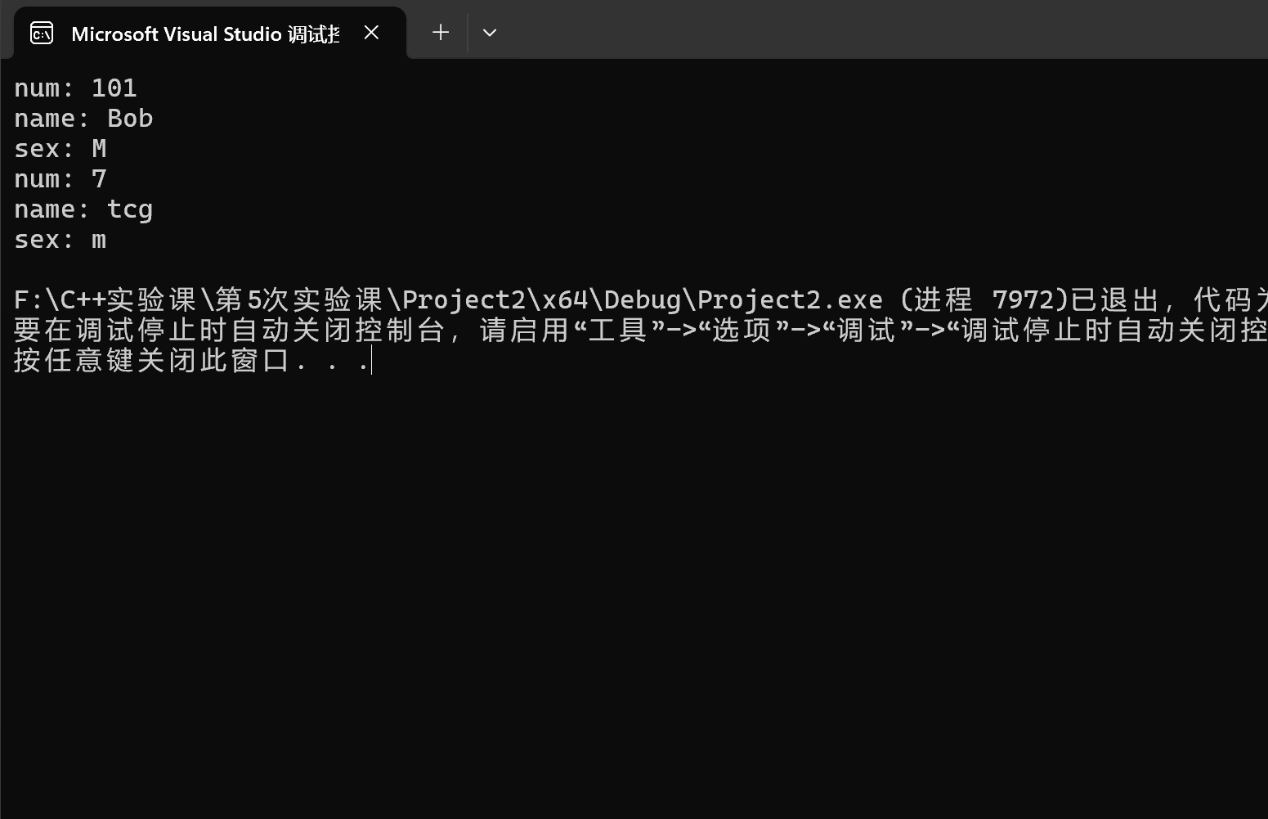
stud.display();

Student stud1(007, "tcg", 'm');

stud1.display();

return 0;

}

3. #include<iostream>

using namespace std;

class cft

{

private:

double length;

double height;

double width;

public:

void set()

{

cout << "Please enter the length :" << endl;

cin >> length;

cout << "Please enter the width :" << endl;

cin >> width;

cout << "Please enter the height :" << endl;

cin >> height;

}

int get()

{

return length \* width \* height;

}

};

int main()

{

cft cft1;

cft cft2;

cft cft3;

cft1.set();

cft2.set();

cft3.set();

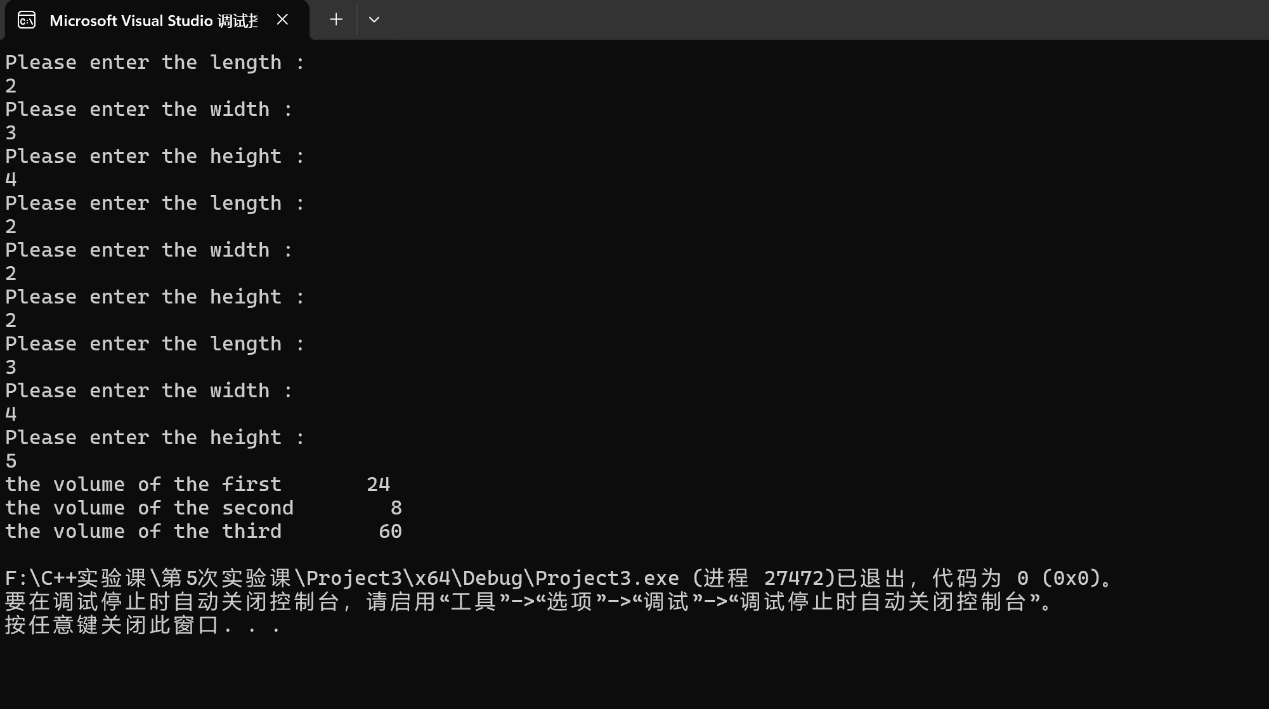
cout << "the volume of the first " << cft1.get() << endl;

cout << "the volume of the second " << cft2.get() << endl;

cout << "the volume of the third " << cft3.get() << endl;

return 0;

}

4. #include<iostream>

using namespace std;

class student

{

public:

int num;

int score;

student(int a, int b)

{

num = a;

score = b;

}

};

student max(student\* p,int length)

{

for (int i = 0; i < length-1; i++)

{

if (p[i].score >p[i + 1].score)

{

student abf= p[i];

p[i] = p[i + 1];

p[i + 1] = abf;

}

}

return p[4];

}

int main()

{

student shuzu[5] = { student(1,97),student(2,96),student(3,98),student(4,99),student(5,100)};

student\* ptr = shuzu;

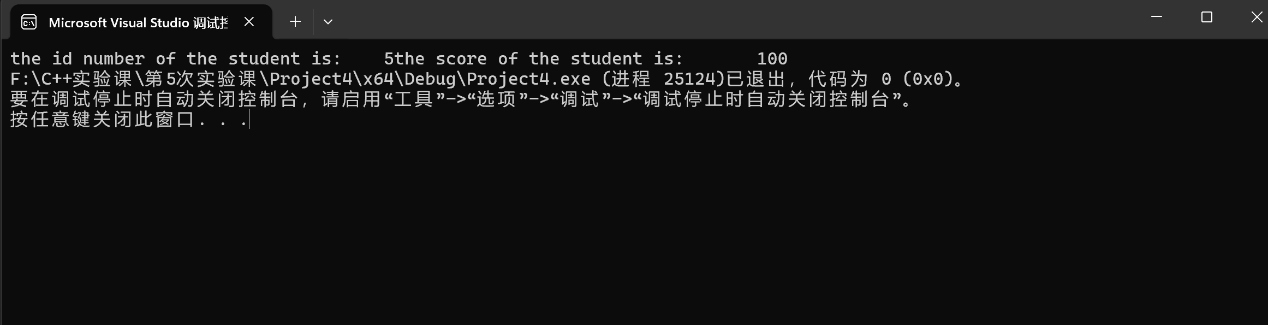
student adf=max(ptr, 5);

cout << "the id number of the student is: " << adf.num;

cout << "the score of the student is: " << adf.score;

return 0;

}

5. #include<iostream>

using namespace std;

class point

{

private:

int x;

int y;

public:

point(int a, int b)

{

x = a;

y = b;

}

void setpoint(int i,int j)

{

x += i;

y += j;

}

void display()

{

cout << "x=" << x << endl;

cout << "y=" << y << endl;

}

};

int main()

{

int m;

cout << "please enter i:" << endl;

cin >> m;

int n;

cout << "please enter j:" << endl;

cin >> n;

point point1(60, 80);

point1.setpoint(m, n);

point1.display();

return 0;

}

