**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。

**【算法分析，程序结果】**

1.

#include<iostream>

using namespace std;

class Time

{

private:

int hour;

int minute;

int sec;

public:

Time(int h, int m, int s)

{

hour = h;

minute = m;

sec = s;

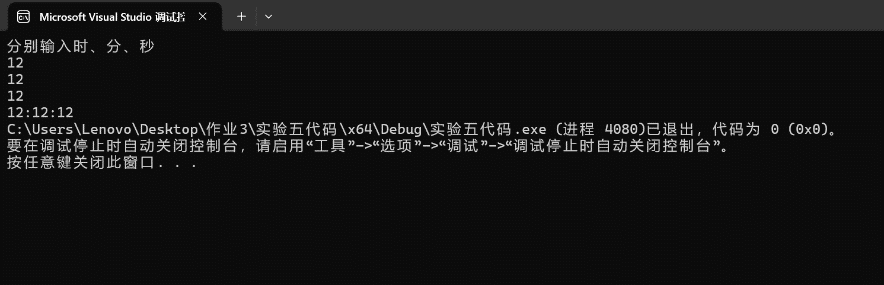
}

void SetTime()

{

int a;

int b;

 int c;

cin >> a;

cin >> b;

cin >> c;

hour = a;

minute = b;

sec = c;

}

void ShowTime()

{

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

}

};

int main()

{

int a = 0, b = 0, c = 0;

cout << "分别输入时、分、秒" << endl;

Time t1(a,b,c);

t1.SetTime();

t1.ShowTime();

return 0;

}

利用构造函数来赋予成员变量初值，然后通过函数SetTime来通过键盘输入具体变量值。

2.

//student.cpp

#include <iostream>

#include"student.h"

using namespace std;

void Student::display()

{

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

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

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

}

//student.h

class Student

{

public:

void display();

void SetValue(int num,const char name[], char sex)

{

int m = 0;

while (name[m] != '\t')

{

m++;

}

this->num = num;

for (int a = 0; a < m; a++)

{

this->name[a] = name[a];

}

this->sex = sex;

}

private:

int num;

char name[20];

char sex;

};

//main.cpp

#include<iostream>

#include"student.h"

using namespace std;

int main()

{

Student stud;

stud.SetValue(007,"tcg", 'm');

stud.display();

return 0;

}

通过SetValue的一个共有成员函数来赋予函数初值，其中通过while (name[m] != '\t')来判断名字有几个字符，从而进行精准赋值。

3.

#include<iostream>

using namespace std;

class Rectangle

{

private:

int length;

int width;

int height;

public:

int volume;

Rectangle(int l, int w, int h)

{

length = l;

width = w;

height = h;

}

void Set()

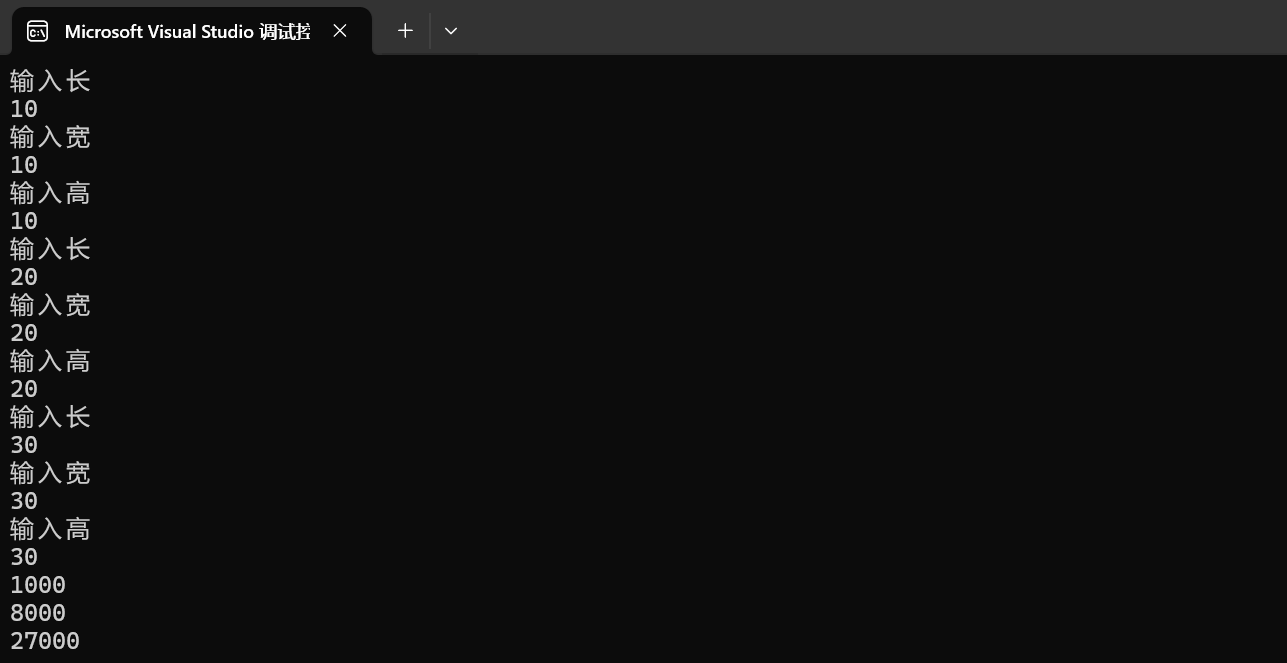
{

int a, b, c;

cout << "输入长" << endl;

cin >> a;

cout << "输入宽" << endl;

 cin >> b;

cout << "输入高" << endl;

cin >> c;

length = a;

width = b;

height = c;

}

int Volume()

{

int c = length \* width \* height;

return c;

}

void Cout()

{

cout << length \* width \* height << endl;

}

};

int main()

{

Rectangle a(0, 0, 0);

a.Set();

Rectangle b(0, 0, 0);

b.Set();

Rectangle c(0, 0, 0);

c.Set();

a.Cout();

b.Cout();

c.Cout();

}

通过Set共有成员函数来赋予成员变量初值，通过Cout函数来输出体积。

4.

#include<iostream>

using namespace std;

class Student

{

private:

int num;

int score;

public:

void Set(int num,int score)

{

this->num = num;

this->score = score;

}

int num1()

{

return num;

}

int score1()

{

return score;

}

};

int max(Student a[5])

{

int b[5];

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

{

b[i] = a[i].score1();

}

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

{

bool m = 1;

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

{

if (i == c)

{

}

else

{

if (b[i] < b[c])

m = 0;

}

}

if (m == 1)

{

return b[i];

}

}

}

int maxnum(Student a[5])

{

int b[5];

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

{

b[i] = a[i].score1();

}

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

{

bool m = 1;

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

{

if (i == c)

{

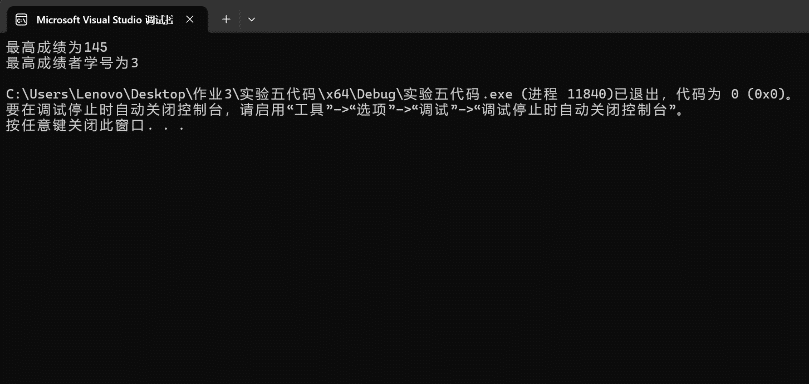
}

else

{

if (b[i] < b[c])

m = 0;

 }

}

if (m == 1)

{

return i;

break;

}

}

}

int main()

{

Student a[5];

a[0].Set(1, 143);

a[1].Set(2, 136);

a[2].Set(3, 137);

a[3].Set(4, 145);

a[4].Set(5, 132);

int max1 = max(a);

cout << "最高成绩为" << max(a) << endl;

cout << "最高成绩者学号为" << maxnum(a) << endl;

}

先在主函数中建立五个成员的数组，分别通过构造函数给他们赋予初值，通过num（）score（）成员函数返回值来调用成员变量从而进行max（）函数的编写，maxnum（）函数返回值则是返回学号。

5.

#include<iostream>

using namespace std;

class Point

{

private:

int x;

int y;

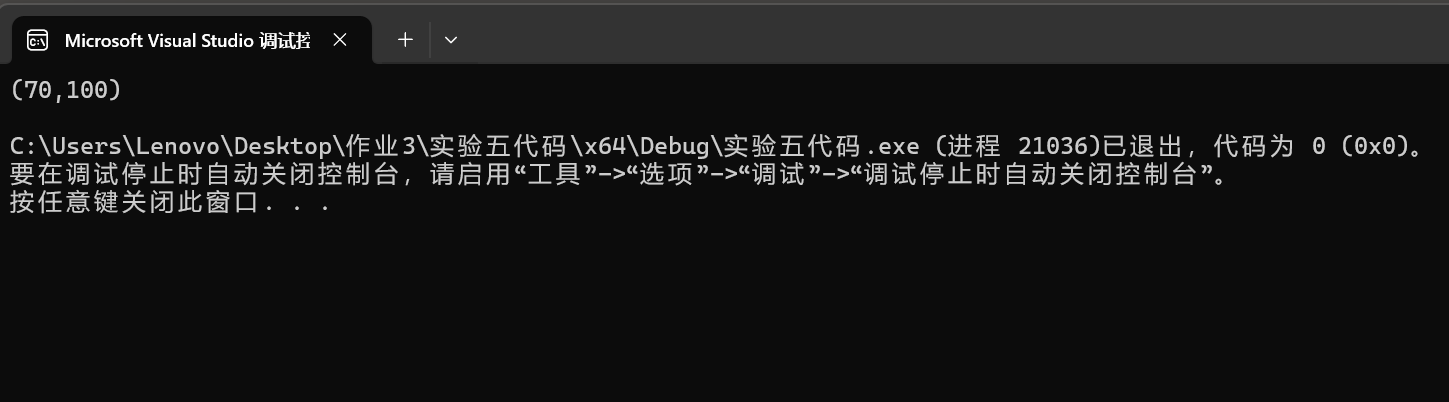
public:

Point(int x, int y)

{

this->x = x;

this->y = y;

 }

void setPoint(int i, int j)

{

x = x + i;

y = y + j;

}

void display()

{

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

}

};

int main()

{

Point a(60, 80);

a.setPoint(10, 20);

a.display();

}

通过setPoint函数来向坐标加对应的值，display来输出。

**【遇到的问题与解决方法】**

1. 在写第二题时忘记char是单个字符，然后构造函数没有写char[ ]而是直接写的char，从而报错，后来用string发现不能解决问题，最后结合数组与循环结构来计算输入的字符数来完成。
2. 编写第四题时不知道怎么用一个max函数同时返回最大值和最大值对应学号，后来用一个新函数返回学号来解决。

**【体会】**

1. char类型是单个字符，要输入一串字符需要用字符数组char[ ]
2. 一个函数只能有一个返回值，需要两个对应返回值就建立两个函数。