**实验报告**

**专业班级：软件工程2401**

**学号：8209240101**

**姓名：蔡颖思**

**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。

1. 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 // 定义Time类

{

private: // 将数据成员改为私有

int hour;

int minute;

int sec;

public:

void settime(int Hour,int Minute,int Sec) {

hour = Hour;

minute = Minute;

sec = Sec;

} //创建一个输入时间的成员函数

void gettime() {

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

} //创建一个输出时间的成员函数

};

int main()

{

Time t1; //定义t1为Time类对象

t1.settime(10,10,10); //外部输入设定时间

t1.gettime();

return 0;

}

/\*分析：

一般函数成员为公有，变量成员为私有

含有私有数据的函数在内部定义

被多个类共享或者需要经常测试维护的函数放在外部定义

\*/

2，#pragma once

//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 setvalue(int m\_num, const char\* m\_name, char m\_sex) {

num = m\_num;

name = m\_name; //如何使用char类型的name[20]来接收一个const char\*类型的变量

sex = m\_sex;

}

private:

int num;

const char\* name; //不知道怎么用char name[20]就直接改了

char sex;

};

//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

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

stud.display(); //执行stud对象的display函数

return 0;

}

//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;

}

3，#include<iostream>

using namespace std;

class cuboid {

public:

int length;

int width;

int height; //定义长宽高

void setcuboid(int m\_l,int m\_w,int m\_h) {

length = m\_l;

width = m\_w;

height = m\_h; //输入长方体信息

}

int culculateV(){

return length\*width\*height;

} //计算长方体体积

void getV() {

cout << "长方体的体积是" << culculateV() << endl;

} //输出长方体体积

};

int main() {

cuboid c1;

cuboid c2;

cuboid c3;

cout << "请输入第一个长方体的长宽高:" << endl;

cin >> c1.length >> c1.width >> c1.height;

c1.getV();

system("pause");

cout << "请输入第二个长方体的长宽高:" << endl;

cin >> c2.length >> c2.width >> c2.height;

c2.getV();

system("pause");

cout << "请输入第三个长方体的长宽高:" << endl;

cin >> c3.length >> c3.width >> c3.height;

c3.getV();

}

4，#include<iostream>

using namespace std;

class Student {

public:

int number;

double score;

Student(int m\_number, double m\_score) {

number = m\_number;

score = m\_score;

}

};

int max(Student\*p,int size) {

int maxIndex = 0; // 存储最高成绩学生的索引

for (int i = 1; i < size; ++i) {

if (p[i].score >p[maxIndex].score) {

maxIndex = i;

}

}

// 输出最高成绩学生的学号

cout << "最高成绩的学生学号是: " << p[maxIndex].number << endl;

return 0;

}

int main() {

Student\* student= new Student[5] {

Student(1,90.0),

Student(2,80.0),

Student(3,70.0),

Student(4,60.0),

Student(5,50.0),

};

max(student,5);

delete[] student;

}

5，#include<iostream>

using namespace std;

class point {

private:

int x=60;

int y=80; //初始化xy

public:

void setPoint(int i, int j) {

x += i;

y += j; //修改坐标

}

void display() {

cout << "修改后的坐标值为：" << " ( " << x << " , " << y << " ) " << endl;

} //输出修改后坐标

};

int main() {

point p;

p.setPoint(1, 1); //验证

p.display();

return 0;

}

### 【遇到的问题和解决办法】

问题：1，实验2中不知道如何用char类型的name[20]来接收const char\*类型的“tcg”；

1. 实验4中用指针来接收对象数组；
2. 实验4中怎么输出成绩最高者的学号；

解决：1，直接把char name[20]改成了const char\*name;

1. 用new ，delect
2. 冒泡

### 【体会】