**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。

代码实现：

1. #include<iostream>

using namespace std;

class Time // 定义Time类

{

public:

//输入小时

void sethour(int h)

{

hour = h;

}

//获取小时

int gethour()

{

return hour;

}

//输入分钟

void setminute(int m)

{

minute = m;

}

//获取分钟

int getminute()

{

return minute;

}

//输入秒

void setsecond(int s)

{

sec = s;

}

//获取秒

int getsecond()

{

return sec;

}

private: // 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

int h, m, s;

cout << "请输入小时：";

cin >> h;

cout << endl << "请输入分钟：";

cin >> m;

cout << endl << "请输入秒：";

cin >> s;

tl.sethour(h);

tl.setminute(m);

tl.setsecond(s);

cout << tl.gethour() << ' ' << tl.getminute() << ' ' << tl.getsecond() << endl;

//cin >> tl.hour; //输入设定的时间

//cin >> tl.minute;

//cin >> tl.sec;

//cout << tl.hour << "；" << tl.minute << "；" << tl.sec << endl;

return 0;

}

(3) 应将成员函数设置为公有且在类内进行定义，数据成员应该为私有，全局函数在类外定义

2.

Student.h

#pragma once

#include<iostream>

using namespace std;

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;

};

Student.cpp

#include<iostream>

#include "Student.h"

using namespace std;

//在类外定义display函数

void Student::display()

{

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

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

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

}

void Student::set\_value()

{

cout << "请输入姓名：";

cin >> name;

cout << endl << "请输入学号：";

cin >> num;

cout << endl << "请输入性别：";

cin >> sex;

}

Main.cpp

#include<iostream>

#include"Student.h"

using namespace std;

int main()

{

Student stud; //定义对象

stud.set\_value();

stud.display(); //执行stud对象的display函数

return 0;

}

3.

#include<iostream>

using namespace std;

//定义长方体对象

class Cft

{

public:

//设置长

void setlength()

{

int l;

cout << "请输入长：";

cin >> l;

length = l;

cout << endl;

}

//设置宽

void setwidth()

{

int w;

cout << "请输入宽：";

cin >> w;

cout << endl;

width = w;

}

//设置高

void setheight()

{

int h;

cout << "请输入高：";

cin >> h;

cout << endl;

height = h;

}

//获取体积

int getV()

{

return width \* length \* height;

}

private:

int length;

int width;

int height;

};

int main()

{

Cft c1;

Cft c2;

//输入第一、二个长方体的长宽高

cout << "第一个长方体：" << endl;

c1.setlength();

c1.setwidth();

c1.setheight();

cout << "第二个长方体：" << endl;

c2.setlength();

c2.setwidth();

c2.setheight();

//输出体积

cout << "第一个长方体的体积为：" << c1.getV() << endl;

cout << "第二个长方体的体积为：" << c2.getV() << endl;

system("pause");

return 0;

}

4.

#include<iostream>

#include<string>

using namespace std;

//定义学生类

class Student

{

public:

Student(int iid, string nname, int sscore)

{

id = iid;

name = nname;

score = sscore;

}

int id;

string name;

int score;

};

//获取最高成绩者的学号 （函数max）

void max(Student\* p,int len)

{

int mark = p->id; //用于记录最高成绩者的学号

int maxscore = p->score; //用于记录最高分

Student temp(0, " ", 0); //用于暂存对象

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

{

i++;

if (maxscore < p->score)

{

maxscore = p->score;

mark = p->id;

}

}

cout << "最高成绩者的学号是：" << mark << endl;

}

int main()

{

//创建对象数组

Student stu[5] =

{

Student(1,"张三",65),

Student(6,"李四",23),

Student(5,"王五",87),

Student(10,"小明",46),

Student(9,"小红",99),

};

max(stu, 5);

system("pause");

return 0;

}

5.

#include<iostream>

using namespace std;

//定义Point类

class Point

{

public:

Point(int xx = 60, int yy = 80) //构造函数初始化数据

{

x = xx;

y = yy;

}

void setPoint(int i, int j) //修改类内坐标值

{

x = x + i;

y = y + j;

}

void display()

{

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

}

private:

int x; //横坐标

int y; //纵坐标

};

int main()

{

Point a;

//验证初始化

a.display();

//验证setPoint

a.setPoint(10, 8);

a.display();

system("pause");

return 0;

}