**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?

#include<iostream>

using namespace std;

class Time

{

private:

int hour;

int minute;

int sec;

public:

void settime()

{

cout << "enter hour:";

cin >> hour;

cout << "enter minute:";

cin >> minute;

cout << "enter seconds:";

cin >> sec;

}

void displaytime()

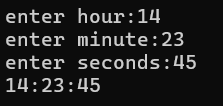
{

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

}

};

int main()

{

Time t1;

t1.settime();

t1.displaytime();

return 0;

}

1. 一些会被调用的类的接口函数 在本题中为settime与 displaytime函数应为public
2. 简单的成员函数，因为这些函数可读性高，放在类中容易维护
3. 复杂的函数应该放在类外定义

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。

//main.cpp

#include <iostream>

#include "student.h"

using namespace std;

int main()

{

Student stu1;

Student stu2;

stu1.set\_value(1,"Lihua",'M');

stu2.set\_value(2, "Hanhong", 'F');

cout << "Student 1 information" << endl;

stu1.display();

cout << "Student 2 information" << endl;

stu2.display();

}

//student.cpp

#include <iostream>

#include "student.h"

#include <cstring>

using namespace std;

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

{

strncpy\_s(name,20,nm,\_TRUNCATE);

name[20] = '\0';

num = n;

sex = s;

}

void Student::display()

{

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

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

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

}

//student.h

#pragma once

class Student

{

public:

void display();

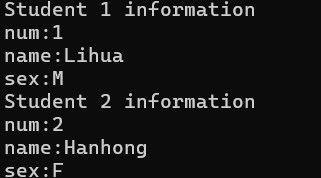
void set\_value(int n, const char\* nm, char s);

private:

int num;

char name[20];

char sex;

};

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。

#include <iostream>

using namespace std;

class CFT

{

private:

double height;

double length;

double width;

public:

double calculate()

{

return length \* height \* width;

}

void input()

{

cout << "enter height:";

cin >> height;

cout << "enter width:";

cin >> width;

cout << "enter length:";

cin >> length;

}

};

int main()

{

CFT changfangti[3];

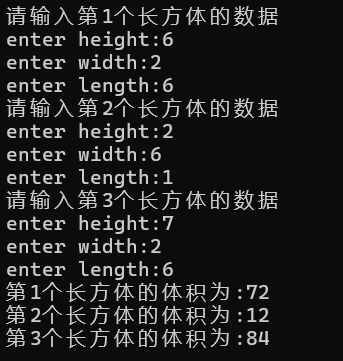
for (int i = 0; i < 3; ++i)

{

cout << "请输入第" << i + 1 << "个长方体的数据" << endl;

changfangti[i].input();

}

 for (int i = 0; i < 3; ++i)

{

double v = changfangti[i].calculate();

cout << "第" << i + 1 << "个长方体的体积为:" << v << endl;

}

return 0;

}

1. 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。

#include <iostream>

using namespace std;

class Student

{

private:

int id;

double score;

public:

void setinfo(int stuid, double stuscore)

{

id = stuid;

score = stuscore;

}

int get\_id()

{

return id;

}

double get\_score()

{

return score;

}

};

void max(Student\* students, int count)

{

Student\* maxstudent = students;

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

{

if ((students + i)->get\_score() > maxstudent->get\_score())

{

maxstudent = students + i;

}

}

cout << "成绩最好的学生信息为" << endl;

cout << "ID:" << maxstudent->get\_id() <<"\t" << "SCORE:" << maxstudent->get\_score();

}

int main()

{

const int studentcount = 5;

Student students[studentcount];

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

{

int id;

double score;

cout << "请输入第" << i + 1 << "个学生的id和score:";

cin >> id >> score;

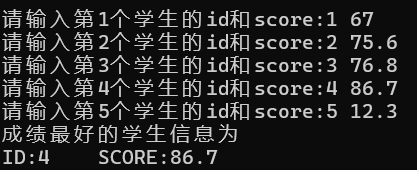
students[i].setinfo(id, score);

}

max(students, studentcount);

return 0;

}



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

#include <iostream>

using namespace std;

class Point

{

private:

int x,y;

public:

Point()

{

x = 60, y = 80;

}

void setPoint(int i, int j)

{

x = x + i;

y = y + j;

}

void display()

{

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

}

};

int main()

{

Point p;

cout << "初始化后的坐标为";

p.display();

 p.setPoint(30, 10);

cout << "修改后的坐标为:";

p.display();

return 0;

}

# 遇到的问题与解决方法

1. 在写项目2时，出现了形参与实参不匹配的情况，后来将name的参数类型修改成为const char\*成功解决
2. 项目2，输出name出现乱码，经查阅发现是name数组没有初始化的原因，使用csrting中的strncpy方法进行修改，编译器提示不安全，后修改为strncpy\_s方法成功解决
3. 项目3在调用成员函数时用 实例名.函数名失败 后使用->方法调用
4. 项目4中对指针的调用理解不清晰，不明白指针含义，后经与同学交流和资料查阅，在max函数中使用指针来找出成绩最好的学生

# 体会

对类的使用基本掌握，对构造函数的理解不够清晰，对指针的应用不够熟练，概念不清晰，需要更多的实例来增加对指针的理解与应用进度，对数组的使用不够熟练，初始化很重要。

在cpp中对指针的运用非常重要，应该下更大功夫对其得到更熟练的掌握与应用来提高程序的效率，优化简洁程度.