**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。

1.

文本

描述已自动生成

#include<iostream>

using namespace std;

class Time

{

int hour;

int minute;

int sec;

public:

void settime(int h, int m, int s)

{

hour = h;

minute = m;

sec = s;

}

void showtime()

{

cout << "现在的时间是" << hour << ":" << minute << ":" << sec << endl;

}

};

int main()

{

Time tl; //定义t1为Time类对象

tl.settime(2, 3, 4);

tl.showtime();

system("pause");

return 0;

}

**我认为对类的整体影响不大的成员或者需要与外界交互的函数等成员应作为公有成员出现，而比较重要的参数等成员应设置为私有，仅留公有的函数与外界交互。当与私有成员有关的函数及其用到私有成员的函数应在类内定义，而与类内私有成员函数无关并且需要在没有定义对象的时候使用的函数应该在类外定义**

2.2

文本

低可信度描述已自动生成

#include <iostream> //将类声明头文件包含进来

#include "student.h"

using namespace std;

int main()

{

Student stud; //定义对象

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

stud.set\_value(001, "zhy", 'm');

stud.display(); //执行stud对象的display函数

stud1.display();

system("pause");

return 0;

}

class Student //类声明

{public:

Student() {};

Student(int n, const char\* na, char se)

{

num = n;

sex = se;

int a = 0;

while (na[a] != '\0' && a < 19) {

name[a] = na[a];

a++;

}

name[19] = '\0';

}

void set\_value(int n, const char\* na, char se)

{

num = n;

sex = se;

int a = 0;

while (na[a] != '\0' && a < 19) {

name[a] = na[a];

a++;

}

name[19] = '\0';

}

public:

void display();

private:

int num;

char name[20];

char sex;

};

#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;

}

5.3#include<iostream>

using namespace std;

class cupboid

{

int v;

int c;

int k;

int g;

public :

void set(int C, int K, int G)

{

c = C;

k = K;

g = G;

v = c \* k \* g;

}

void show()

{

cout << "长方体的体积为" << v<<endl;

}

};

int main()

{

int a, b, c,d,e,f,g,h,i;

cin >> a >> b >> c >> d >> e >> f >> g >> h >> i;

cupboid a1;

cupboid a2;

cupboid a3;

a1.set(a, b, c);

a2.set(d, e, f);

a3.set(g, h, i);

a1.show();

a2.show();

a3.show()

;

system("pause");

return 0;

}

文本

描述已自动生成

5.4

文本

描述已自动生成

#include<iostream>

using namespace std;

class student

{

public:

int num;

int grade;

};

void max( student \*b)

{

student a[5];

for (int h = 0; h <= 4; h++)

{

a[h].grade = b[h].grade;

}

int c = 0;

for (int i = 0; i <= 4; i++)

{

if (i!=4)

{

if (a[i].grade <= a[i + 1].grade)

{

int b = a[i].grade;

a[i].grade = a[i + 1].grade;

a[i + 1].grade = b;

c++;

}

}

else

{

if (c == 0)

{

i = 5;

}

else

{

i = -1;

c = 0;

}

}

}

for (int j = 0; j <= 4; j++)

{

if (a[0].grade == b[j].grade)

{

cout << b[j].num;

}

}

}

int main()

{

student a[5];

a[0].num = 1;

a[0].grade = 100;

a[1].num = 2;

a[1].grade = 99;

a[2].num = 3;

a[2].grade = 101;

a[3].num = 4;

a[3].grade = 110;

a[4].num = 5;

a[4].grade = 140;

max(a);

system("pause");

return 0;

}5.5

文本

描述已自动生成

#include<iostream>

using namespace std;

class point

{

int x;

int y;

public:

point()

{

x = 60;

y = 80;

}

void setpoint(int i, int j)

{

x += i;

y += j;

display();

}

void display()

{

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

}

};

int main()

{

point a;

a.display();

a.setpoint(5, 6);

system("pause");

return 0;

}