

电子科技大学信息与软件工程学院

标准实验报告

(实验) 课程名称: 面向对象程序设计 C++

电子科技大学教务处制表

1、阅读一段代码，分析其执行过程，然后上机运行，对比输出结果；修改上面的程序，增加一个 fun 函数，改写 main 函数。在 main 函数中调用 fun 函数，在 fun 函数中调用 change 和 display 函数。在 fun 函数中使用对象的引用（Student&）作为形参。

```

#include <iostream>
using namespace std;
class Student
{
public:
    Student(int n,float s):num(n),score(s){}
    void change(int n,float s){num=n;score=s;}
    void display(){cout<<num<<" "<<score<<endl;}
private:
    int num;
    float score;
};

int main()
{
    Student stud(101, 78.5); //创建一个student类的对象，
    stud.display();          //调用对象的成员函数，输出
    stud.change(101,80.5);   //调用对象的成员函数，修改
    stud.display();
    return 0;
}

```

2、设计一个立方体类 Box，它能计算并输出立方体的体积和表面积，同时提供私有数据成员存储边长，提供函数返回数据成员的引用，并在 main() 中调用该函数，实现左值赋值，实现成员函数对边长进行改变，要求提供指针、引用、传值三种参数的函数重载。要求写出 Box 类的构造函数和析构函数。

3、阅读一段代码，将程序中的 display 函数不放在 Time 类中，而作为类外的普通函数，然后分别在 Time 和 Date 类中将 display 声明为友元函数。在主函数中调用 display 函数，display 函数分别引用 Time 和 Date 两个类的对象的私有数据输出 年，月，日和时，分，秒。修改后上机调试和运行。

```

#include<iostream>

using namespace std;

class Date;           //对 Date 类的提前引用声明

class Time             //定义 Time 类
{
public:
    Time(int, int, int);    //构造函数

    void display(Date &); //display 是成员函数，形参是 Date 类对象的引用

```

```

    private:
        int hour;
        int minute;
        int sec;
};

class Date                                //声明 Date 类
{
    public:
        Date(int, int, int);
        friend void Time::display(Date &); //声明 Time 中的 display 函数为友元成员函数

    private:
        int month;
        int day;
        int year;
};

Time::Time(int h, int m, int s)           //类 Time 的构造函数
{
    hour=h;
    minute=m;
    sec=s;
}

void Time::display(Date &d)               //display 的作用是输出年，月，日和时，分，秒
{
    cout<<d.month<<"/"<<d.day<<"/"<<d.year<<endl;
    cout<<hour<<":"<<minute<<":"<<sec<<endl;
}

Date::Date(int m,int d,int y)             //类 Date 的构造函数
{

```

```
    month=m;
    day=d;
    year=y;
}
int main()
{
    Time t1(10,13,56);           //定义 Time 类对象 t1
    Date d1(12,25,2004);         //定义 Date 类对象 d1
    t1.display(d1);              //调用 t1 中的 display 函数，实参是 Date 类对象 d1
    return 0;
}
```

七、实验器材（设备、元器件）：

PC 机

八、实验步骤：

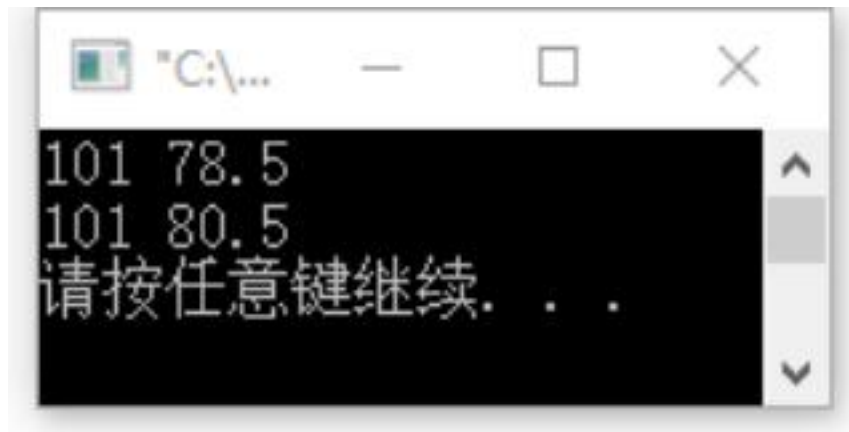
- 1) 创建工程
- 2) 编辑程序
- 3) 编译链接程序
- 4) 调试程序
- 5) 运行程序，分析结果

本实验 4 个小题目都分别按上述步骤进行。

九、实验数据及结果分析：

题目 1：

首先输入题目给出代码，分析给出代码的输出结果：



第一排 101 78.5 是对象初始化即调用构造函数的时候的私有变量 num 和 score 的值；

第二排 101 80.5 是运用 change 函数改变后的私有变量 num 和 score 的值；

重写 fun 函数，修改源代码，如下所示：

```
#include<iostream>
using namespace std;
class Student
{
public:
    Student(int n,float s):num(n),score(s){}
    void change(int n,float s){num=n;score=s;}
    void display(){cout<<num<<" "<<score<<endl;}
private:
    int num;
    float score;
};
void fun(Student &a){
    a.change(17,7);
    a.display();
}

int main()
{
    Student stud(101,78.5);
    stud.display();
    stud.change(101,80.5);
    stud.display();
    fun(stud);
    return 0;
}
```

如上给出的代码所示，我书写了一个 fun 函数，并且在 fun 函数中使用对象的引用 (Student&) 作为形参，之后再 main 函数中运用已经定义好的对象 stud 作为实参带入 fun 函数中，运用 change 函数修改 stud 这个对象私有数据 num 和 score 的值，并且运用 display 打印出来。运行的结果如下图所示：

运行如下图所示：



第一排 101 78.5 是对象初始化的时候的私有变量 num 和 score 的值；
第二排 101 80.5 是运用 change 函数改变后的私有变量 num 和 score 的值；
第三排是将对象 stud 作为实参带入 fun 函数后改变后的私有变量 num 和 score 的值。

题目 2:

Box 类包含三个私有数据成员 a(立方体边长)、volume(体积)和 area(表面积)，
另有两个构造函数以及 geta() (获取边长的引用)、seta() (设置立方体边长)、
getvolume() (计算体积)、getarea() (计算表面积)和 disp() (输出结果)。

本题程序如下：

```
#include<iostream>
using namespace std;
class Box
{
public:
    Box(int n){
        l=n;
    };
    ~Box(){} //析构函数
    int &L(){return l;} //返回数据成员 l 的引用
    void change(int a){ //传值
        l=a;
    }
    void change(int *p){ //指针
        l=*p;
    }
};
```

```

    }
    void change(int &c){           //引用
        l=c;
    }
    void display_S(){             //打印表面积的函数
        int s=l*l*6;
        cout<<"边长为:"<<l<<"表面积为:"<<s<<endl;
    }
    void display_V(){             //打印体积的函数
        int v=l*l*l;
        cout<<"边长为:"<<l<<"体积为:"<<v<<endl;
    }
private:
    int l;
};

int main()
{
    int b=2;
    int c=3;
    int m=6;
    Box box(0);
    box.L()=m;                    //左值赋值给 l
    box.display_S();
    box.display_V();
    cout<<"\n"<<endl;

    box.change(1);
    box.display_S();
    box.display_V();
    cout<<"\n"<<endl;

    box.change(&b);
    box.display_S();
    box.display_V();
    cout<<"\n"<<endl;

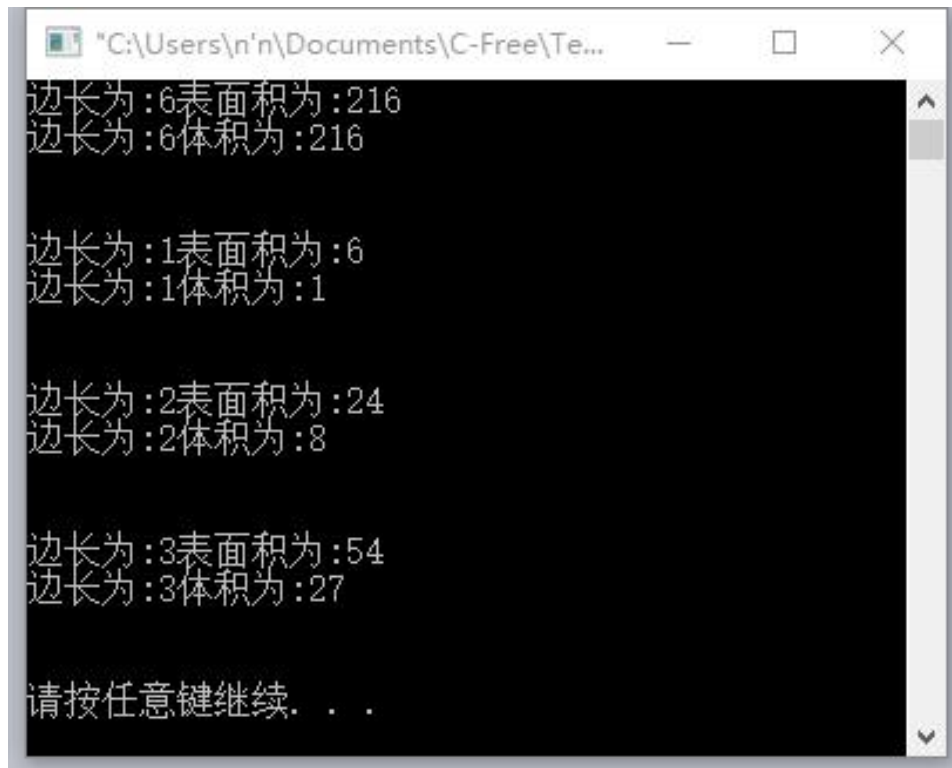
    box.change(&c);
    box.display_S();
    box.display_V();
    cout<<"\n"<<endl;
    return 0;
}

```

代码中，我分别运用了左值赋值的方式和成员函数改变私有数据的方式来对边长

1 进行了改变，每一次改变后我都运用了 display_S() 和 display_V() 两个函数来打印了改变后的边长的值、表面积和体积的值。

本程序的执行结果如下：



```
"C:\Users\n'n\Documents\C-Free\Te...  
边长为:6表面积为:216  
边长为:6体积为:216  
  
边长为:1表面积为:6  
边长为:1体积为:1  
  
边长为:2表面积为:24  
边长为:2体积为:8  
  
边长为:3表面积为:54  
边长为:3体积为:27  
  
请按任意键继续. . .
```

如前两排结果所示，运用左值赋值的方式把 m 的值赋给了边长 1，并且打印出了此时的表面积和体积的值分别为 6、216、216；

如第三排和第四排所示，运用传值的方式把边长 1 的值修改为了 1，并且打印出了此时的表面积和体积的值分别为 1、6、1；

如第五排和第六排所示，运用指针的方式把边长 1 的值修改为了 2，并且打印出了此时的表面积和体积的值分别为 2、24、8；

如第七排和第八排所示，运用引用的方式把边长 1 的值修改为了 3，并且打印出了此时的表面积和体积的值分别为 3、54、27。

题目 3：

首先输入原始给出的代码：

```
#include<iostream>  
using namespace std;  
class Date;           //对 Date 类的提前引用声明  
class Time             //定义 Time 类  
{  
public:  
    Time(int, int, int);    //构造函数  
    void display(Date &); //display 是成员函数，形参是 Date 类对象的引用  
private:
```

```

        int hour;
        int minute;
        int sec;
};

class Date                                //声明 Date 类
{
    public:
        Date(int, int, int);
        friend void Time::display(Date &); //声明 Time 中的 display 函数为友元成员函数
    private:
        int month;
        int day;
        int year;
};

Time::Time(int h, int m, int s)           //类 Time 的构造函数
{
    hour=h;
    minute=m;
    sec=s;
}

void Time::display(Date &d)               //display 的作用是输出年，月，日和时，分，秒
{
    cout<<d.month<<"/"<<d.day<<"/"<<d.year<<endl;
    cout<<hour<<":"<<minute<<":"<<sec<<endl;
}

Date::Date(int m,int d,int y)             //类 Date 的构造函数
{
    month=m;
    day=d;
    year=y;
}

int main()
{
    Time t1(10,13,56);                    //定义 Time 类对象 t1
    Date d1(12,25,2004);                  //定义 Date 类对象 d1
    t1.display(d1);                       //调用 t1 中的 display 函数，实参是 Date 类对象 d1
    return 0;
}

```

输出结果如下：



display() 为 Time 类中的成员函数同时也是 Date 类中的友元函数，所以它可以同时调用两个类中的私有成员并且打印出来，如上图为输出结果。

修改后的代码如下：

```
#include<iostream>
using namespace std;
class Date;           //对 Date 类的提前引用声明

class Time             //定义 Time 类
{
public:
    Time(int, int, int);    //构造函数
    friend void display(Date &, Time &); //display 是友元函数，形参是 Date 类
和 Time 类对象的引用
private:
    int hour;
    int minute;
    int sec;
};

class Date             //声明 Date 类
{
public:
    Date(int, int, int);
    friend void display(Date &, Time &); //display 是友元函数，形参是 Date 类
和 Time 类对象的引用
private:
    int month;
    int day;
    int year;
```

```

};
Time::Time(int h, int m, int s)    //类 Time 的构造函数
{
    hour=h;
    minute=m;
    sec=s;
}
void display(Date &d,Time &t)    //display 的作用是输出年，月，日和时，分，秒
{
    cout<<d.month<<"/"<<d.day<<"/"<<d.year<<endl;
    cout<<t.hour<<":"<<t.minute<<":"<<t.sec<<endl;
}
Date::Date(int m,int d,int y)    //类 Date 的构造函数
{
    month=m;
    day=d;
    year=y;
}
int main()
{
    Time t1(10,13,56);           //定义 Time 类对象 t1
    Date d1(12,25,2004);         //定义 Date 类对象 d1
    display(d1,t1);              //调用 t1 中的 display 函数，实参是 Date 类对象 d1
    return 0;
}

```

在如上的代码中,将 display 函数作为类外的普通函数,然后分别在 Time 和 Date 类中将 display 声明为友元函数。在主函数中调用 display 函数,display 函数分别引用 Time 和 Date 两个类的对象的私有数据输出 年,月,日和时,分,秒。

修改后的输出结果为:



display 函数分别打印除了 d1 和 t1 两个对象的私有成员值。

十、实验结论：

在题目一中，在 fun 函数中使用对象的引用（Student&）作为形参，其中的形参应该运用引用的方式，并且课题通过这种方式来改变对象的私有变量的值。

在题目二中，如果提供函数返回数据成员的引用的话，就可以运用左值赋值来给私有数据即边长进行赋值，并且同时也可以运用类的成员函数来调用私有数据进行改变，同时可以用传值、指针和引用三种方式来设置形参。构造函数在定义对象的时候就会默认使用，而析构函数也会隐式调用。

在题目三中，源程序是把 display 函数作为 Time 类的成员函数，并且作为 Date 类的友元函数，所以它可以同时调用两个类中的私有成员数据并且打印出来。而修改后的函数是把 display 函数作为一个普通的类外函数，但同时作为 Time 类和 Date 类的友元函数，这样的话也可以通过 display 函数来定义对象访问两个对象的私有成员。

十一、总结及心得体会：

我在实验后更加了解到了 C++ 中的一些基本的概念和操作，获益匪浅。题目一中的 fun 函数中的形参应该用对象的引用，若不是引用则会编译报错，在题目二中运用函数返回数据成员的引用这里必须返回的是引用方式，不然数据不会得到改变，因为引用是从地址上来修改数据的。题目三中友元函数的用法方式很多，无论是把 display 函数定义在类中还是类外都有办法实现调用，有时候就要看情况来看运用哪一种友元方式。

十二、对本实验过程及方法、手段的改进建议：

我认为本实验充分的利用了一些 C++ 的基本程序操作手段，无修改意见。

报告评分：

指导教师签字：

