**第三次上机报告**

**上机内容：**

**实验目的**：

1.理解类的对象的概念，掌握声明类和定义对象的方法

2.掌握析构函数和构造函数的实现方法

3.初步掌握使用类和对象编制C++程序

**实验程序：**

hero.h:

#include<iostream>

using namespace std;

class Hero

{

public:

Hero();

Hero(int x1, int x2, int x3);

Hero(const Hero & name);

~Hero();

Hero & operator=(const Hero &rhs);

private:

int skill1;

int skill2;

int skill3;

};

hero.cpp:

#include"hero.h"

Hero::Hero()

{

cout << "This is a default constructor!" << endl;

skill1=0;

skill2=0;

skill3=0;

}

Hero::Hero(int x1, int x2, int x3)

{

cout << "This is a overloaded constructor!" << endl;

skill1=x1;

skill2=x2;

skill3=x3;

}Hero::Hero(const Hero & name)

{

cout << "This is a copy constructor!" << endl;

skill1= name.skill1;

skill2= name.skill2;

skill3= name.skill3;

}

Hero::~Hero()

{

cout << "This is a destructor!" << endl;

}

Hero & Hero::operator=(const Hero &rhs)

{

if( this == &rhs)

{

return \*this;

}

this->skill1 = rhs.skill1;

this->skill2 = rhs.skill2;

this->skill3 = rhs.skill3;

return \*this;

}

**main.h:**

#include<iostream>

#include"hero.h"

using namespace std;

int main()

{

Hero Libai;

Hero Sunwukong = Hero(10, 10, 10);

Hero Hanxin(Sunwukong);

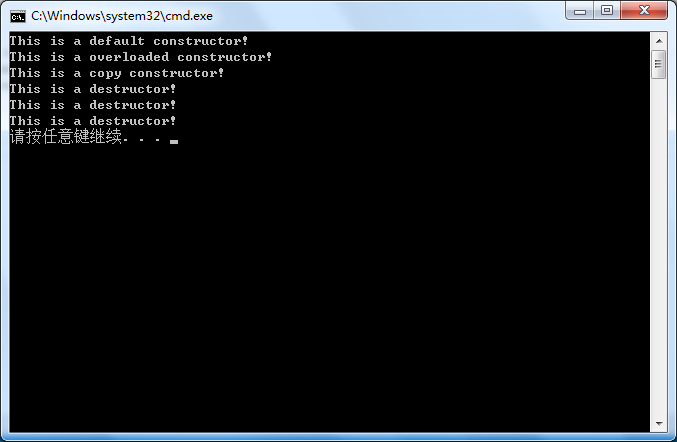
Libai = Sunwukong;

return 0;

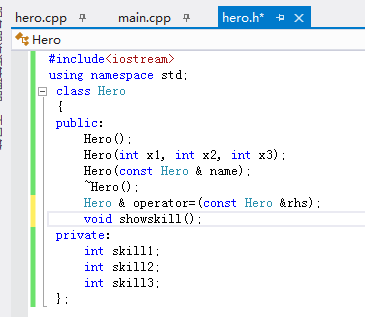
}

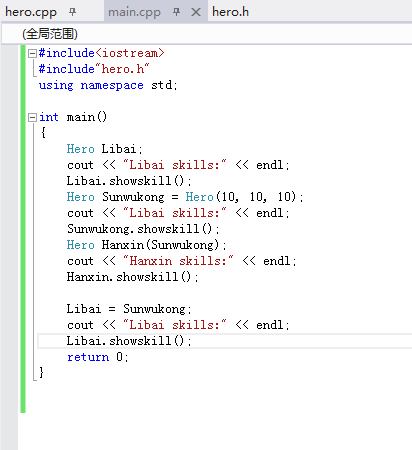
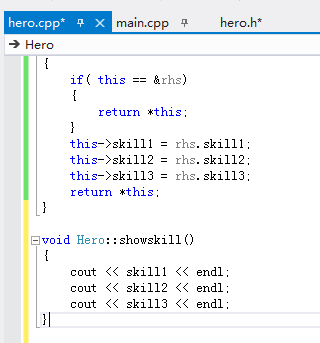
运行结果：

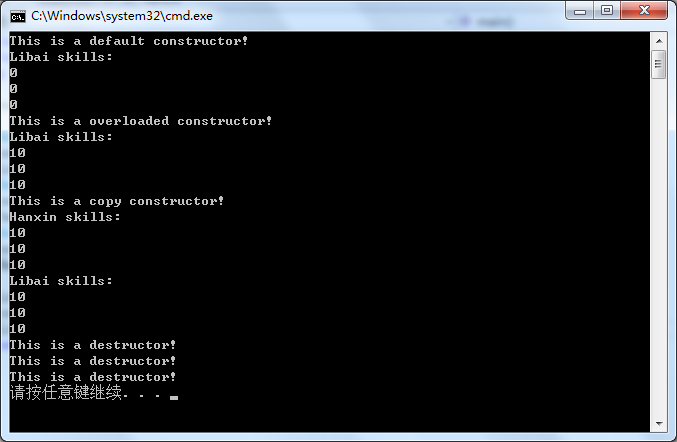




添加程序：







**问题：**

**一、类和对象的声明和定义对象**

1.对象的类型称为类，类代表一批对象的共性和特征。先声明类类型，再用它去定义若干个同类型的对象。类是对象的模板。

类是抽象的，不占用内存。对象是具体的，占用内存。

2.声明类类型的方法跟声明结构体类型相似。格式如下：

class 类型名 //类头

{private: //类体，内容是类的成员表，包括数据和函数

私有的数据和成员函数；

public:

公用的数据和成员函数；

}； //分号结束

注意：

1）一般是把数据隐蔽起来，而把成员函数作为对外界的接口；

2）被声明为私有的成员，只能被本类中的成员函数引用，类外不能调用（友元类除外）；公用的成员，外界可以调用

3）除了private和public外，还有一种成员访问限定符，protected，称为受保护的成员，它不能被类外访问，和私有成员类似，但可以被派生类的成员函数访问。

4）不指定为private或public系统默认为private

5)通常先写public部分，再写private部分

6）为了用户方便，c++编译系统往往向用户提供类库，内装常用的基本类，也可以把自己常用的类放在一个专门的类库中，需要时直接调用。

3.定义对象的方法：

1）先声明类类型，再定义对象 格式：类名 对象名

2）声明类的同时定义对象

3）不出现类名，直接定义对象(不提倡）

class{

private:

public:

}stud1,stud2;

4.类中不包含成员函数，相当于结构体

5.一般的做法是把需要被外界调用的函数指定为public，指定为private的函数称为工具函数，它是被本类的成员函数调用的，支持其他函数的操作

6.在类外定义成员函数

可以在类体内进行函数声明，把定义放在类体外。（函数较长时推荐使用）

注意：成员函数在类外定义的格式为 类名：：函数名（）

“：：”是作用域限定符

函数名前面啥都没有的，是全局函数。

**二、构造函数和析构函数实现方法**

**构造函数：**

1.构造函数定义

类的构造函数是类的一种特殊的成员函数，它会在每次创建类的新对象时执行。

构造函数的名称与类的名称是完全相同的，并且不会返回任何类型，也不会返回 void。构造函数可用于为某些成员变量设置初始值。

构造函数主要用于为对象分配空间，进行初始化。

2.性质

（1）构造函数的名字必须与类名相同

（2）构造函数可以有任意类型的参数,但不能指定返回类型。它有隐含的返回值,该值由系统内部使用。

（3）构造函数是特殊的成员函数,函数体可写在类体内,也可写在类体外。

（4）构造函数可以重载,即一个类中可以定义多个参数个数或参数类型不同的构造函数。

（5）构造函数被声明为公有函数,但它不能像其他成员函数那样被显式地调用,它是在定义对象的同时被调用的。

**析构函数：**

1.析构函数的定义

类的析构函数是类的一种特殊的成员函数，它会在每次删除所创建的对象时执行。

析构函数的名称与类的名称是完全相同的，只是在前面加了个波浪号（~）作为前缀，它不会返回任何值，也不能带有任何参数。析构函数有助于在跳出程序（比如关闭文件、释放内存等）前释放资源。

2.性质

（1）析构函数与构造函数名字相同,但它前面必须加一个波浪号(~);

（2）析构函数没有参数,也没有返回值,而且不能重载。因此在一个类中只能有一个析构函数;

（3）当撤消对象时,编译系统会自动地调用析构函数。