实验报告

1. 实验目的

在析构函数中显示这是哪个对象的析构函数，实现一个技能的基本功能。

**二、设计思路**

上一个作业中定义了函数类HERO，和构造函数HERO（），并且程序执行过程中调用了类的对象a，这里需要在类中再定义一个析构函数，作为对象a的析构函数，并输出文字“对象a的析构函数被调用”，定义类HERO2，且定义对象c和对一些函数进行声明，然后定义HERO的数组对象并给他赋值，在下面的输出对象中调用HERO2中的函数返回输出他的值。释放技能后，对改变后的属性进行运算，在如此输出它的属性值。

**三、作业程序代码**

**HERO.H**

#include<iostream>

using namespace std;

class HERO

{

public:

HERO()

{};

~HERO()

{

cout<<"对象a的解析函数被调用"<<endl;

}

}a;

**HERO2.H**

#include<iostream>

using namespace std;

class HERO2

{

public:

int x,y;

HERO2()

{};

HERO2 (int n)

{

x=n;

y=n;

}

int get\_x()

{

return x;

}

int get\_y()

{

return y;

}

void print2()

{

cout<<" 扁鹊二技能为善恶诊断：扁鹊朝指定方向释放药剂，对触碰的敌人造成法术伤害，并对触碰的友军英雄施加恢复效果，恢复其生命值。扁鹊身上产生标记并恢复一定血量。"<<endl;

cout<<" 假设扁鹊二技能攻击到白起,攻击前白起的属性为："<<endl;

}

}c;

**MAIN.CPP**

#include<iostream>

#include"HERO.h"

#include"HERO2.h"

#include<conio.h>

using namespace std;

int main()

{

c.print2();

HERO2 a[5]={8000,400,200,100,450};

cout<<" 血量："<<a[0].get\_x()<<endl;

cout<<" 蓝量："<<a[1].get\_x()<<endl;

cout<<" 物攻："<<a[2].get\_x()<<endl;

cout<<" 法强："<<a[3].get\_x()<<endl;

cout<<" 速度："<<a[4].get\_x()<<endl;

cout<<" 状态："<<"正常"<<endl;

cout<<"二技能命中白起后："<<endl;

getch();

a[0]=8000-400-1000\*0.75;

a[4]=250;

cout<<" 血量："<<a[0].get\_y()<<endl;

cout<<" 蓝量："<<a[1].get\_y()<<endl;

cout<<" 物攻："<<a[2].get\_x()<<endl;

cout<<" 法强："<<a[3].get\_y()<<endl;

cout<<" 速度："<<a[4].get\_y()<<endl;

cout<<" 状态："<<"被标记并减速"<<endl;

return 0;

}

**四、作业程序出现错误及改正**

**1、出现错误**



1. **改正**

这是因为类 HERO2 的参数放在外部定义出现的错误。将在HERO2()上面一行定义的参数int n 移到HERO2（）括号内既可。

**五、作业程序运行结果**

