**实验报告四**

1. **程序设计思路**

在本次实验中，我们在头文件中建立了一个名为“shuxing”的类，相当于建立了一个英雄的模板，这样以后无论建立何种英雄只要改变类中参数即可，不需要再重新编写每个英雄函数，在“shuxing”类下建立了显示英雄状态的“show”函数，完成击打动作的“gongji”函数，在调用“gongji”函数，只需要写入英雄的伤害值即可完成英雄的初始化，显示英雄被击败的析构函数“~gongji”。本次实验是两个英雄分摊伤害，所以只需要调用两次构造函数写入不同的属性即可。

1. **程序代码**
2. Main.cpp

#include <iostream>

#include "gongji.h"

using namespace std;

int main()

{

cout<<"》 亲爱的召唤师：你好，我是你的助手妲己！！"<<endl;

cout<<"》 欢迎来到王者峡谷！！"<<endl;

cout<<"》 敌军还有五秒钟到达战场！！"<<endl;

cout<<"》 让我们一起战斗吧！！"<<endl;

cout<<"》 敌方英雄血量："<<endl;

cout<<"》 100"<<endl;

cout<<"》 敌方英雄魔抗:"<<endl;

cout<<"》 100"<<endl;

cout<<"》 敌方英雄物抗："<<endl;

cout<<"》 100"<<endl;

cout<<"》 英雄蓝量："<<endl;

cout<<"》 100"<<endl;

cout<<"》 英雄红量："<<endl;

cout<<"》 100"<<endl;

cout<<"》 让我们按“1”键释放一技能击溃敌人吧！！"<<endl;

int q;

cin>>q;

shuxing direnjie(100,100,100,100,100);

shuxing houyi(100,100,100,100,100);

while(q>0)

{

switch (q)

{

case 1:if (direnjie.b1>0)

{

direnjie.gongji(10,10,15,15,25);

direnjie .show();

if (direnjie .b1<=0)

{

direnjie.~shuxing ();

}

};

if (houyi.b1>0)

{

houyi.gongji(15,10,15,15,20);

houyi .show();

if (houyi .b1<=0)

{

houyi.~shuxing ();

}

};

break ;

default:cout<<"无效键"<<endl;break;

}

cin>>q;

}

return 0;

}

1. Gongji.cpp

#include <iostream>

#include "gongji.h"

using namespace std;

shuxing::shuxing( int c1,int c2,int c3,int c4,int c5)

{

b1=c1;

b2=c2;

b3=c3;

b4=c4;

b5=c5;

}

1. Gongji.h

#include <iostream>

using namespace std;

class shuxing

{

public:

shuxing();

shuxing( int a1,int a2,int a3,int a4,int a5);//血量：a1 魔抗：a2 物抗：a3 蓝耗：a4 红耗：a5

int b1; int b2;int b3;int b4;int b5;

void show()

{

cout<<"英雄属性："<<endl;

cout<<"敌方英雄血量:";

cout<<b1 <<endl;

cout<<"敌方英雄魔抗:";

cout<< b2 <<endl;

cout<<"敌方英雄物抗:";

cout<< b3<<endl;

cout<<"英雄蓝量:";

cout<< b4<<endl;

cout<<"英雄红量:";

cout<< b5 <<endl;

}

void gongji(int f2,int f3,int f4,int f5,int f1)//攻击对英雄造成的伤害

{

b1 =b1-f1;

b2 =b2-f2;

b3 =b3-f3;

b4 =b4-f4;

b5 =b5-f5;

};

~shuxing()

{

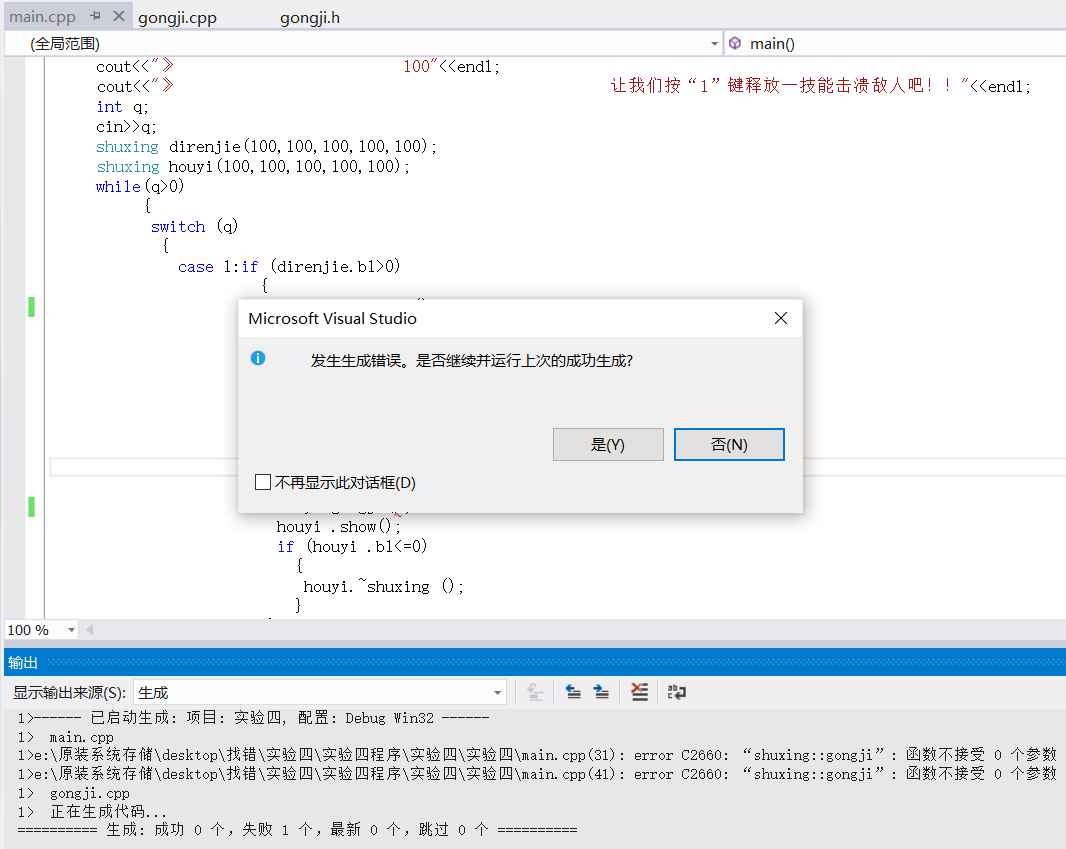
cout<<"》 ###敌方英雄被击败！！###"<<endl ;

};

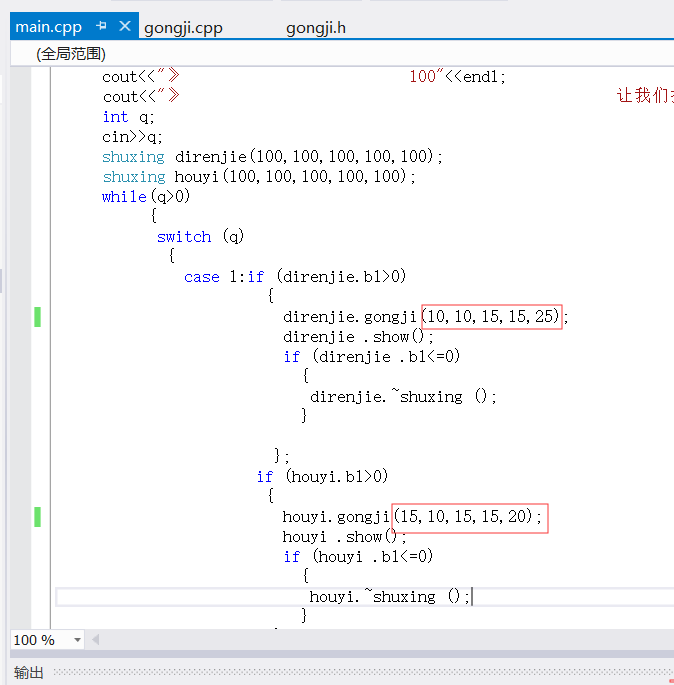
};

1. **程序错误及解决方法**

程序错误：程序无法正常编译，现象如下：



错误更正：经过观察系统提示发现，“gongji”函数不接受0个参数，是因为没有给“gongji”函数赋值，只要对其进行赋值即可，改正如下：



1. **程序运行结果**

