**实验报告**

— 上机四

* 实验内容一（了解三种不同的对象传递方式）

1. 程序代码：

对象ob作为函数sqr\_it形参：

# include<iostream>

using namespace std;

class Tr{

public :

Tr(int n)

{ i = n;}

void set\_i(int n)

{ i = n;}

int get\_i()

{return i;}

private:

int i;

};

void sqr\_it(Tr ob) // 对象ob作为函数sqr\_it的形参

{ ob.set\_i(ob.get\_i()\*ob.get\_i());

cout<<"在函数sqr\_it内，形参对象ob的数据成员i的值为:"<<ob.get\_i();

cout<<endl;

}

int main()

{ Tr obj(10);

cout<<"调用函数sqr\_it前, 实参对象obj的数据成员i的值为:";

cout<<obj.get\_i()<<endl;

sqr\_it(obj);

cout<<"调用函数sqr\_it后, 实参对象obj的数据成员i的值为:";

cout<<obj.get\_i()<<endl;

return 0;

}

对象指针ob作为函数sqr\_it的形参：

void sqr\_it(Tr\* ob) // 对象指针ob作为函数sqr\_it的形参

{ ob->set\_i(ob->get\_i()\*ob->get\_i());

cout<<“在函数sqr\_it内，形参对象ob的数据成员i的值为:”<<ob->get\_i();

cout<<endl;

}

对象的引用作为函数sqr\_it的形参：

void sqr\_it(Tr& ob)

{ ob.set\_i(ob.get\_i()\*ob.get\_i());

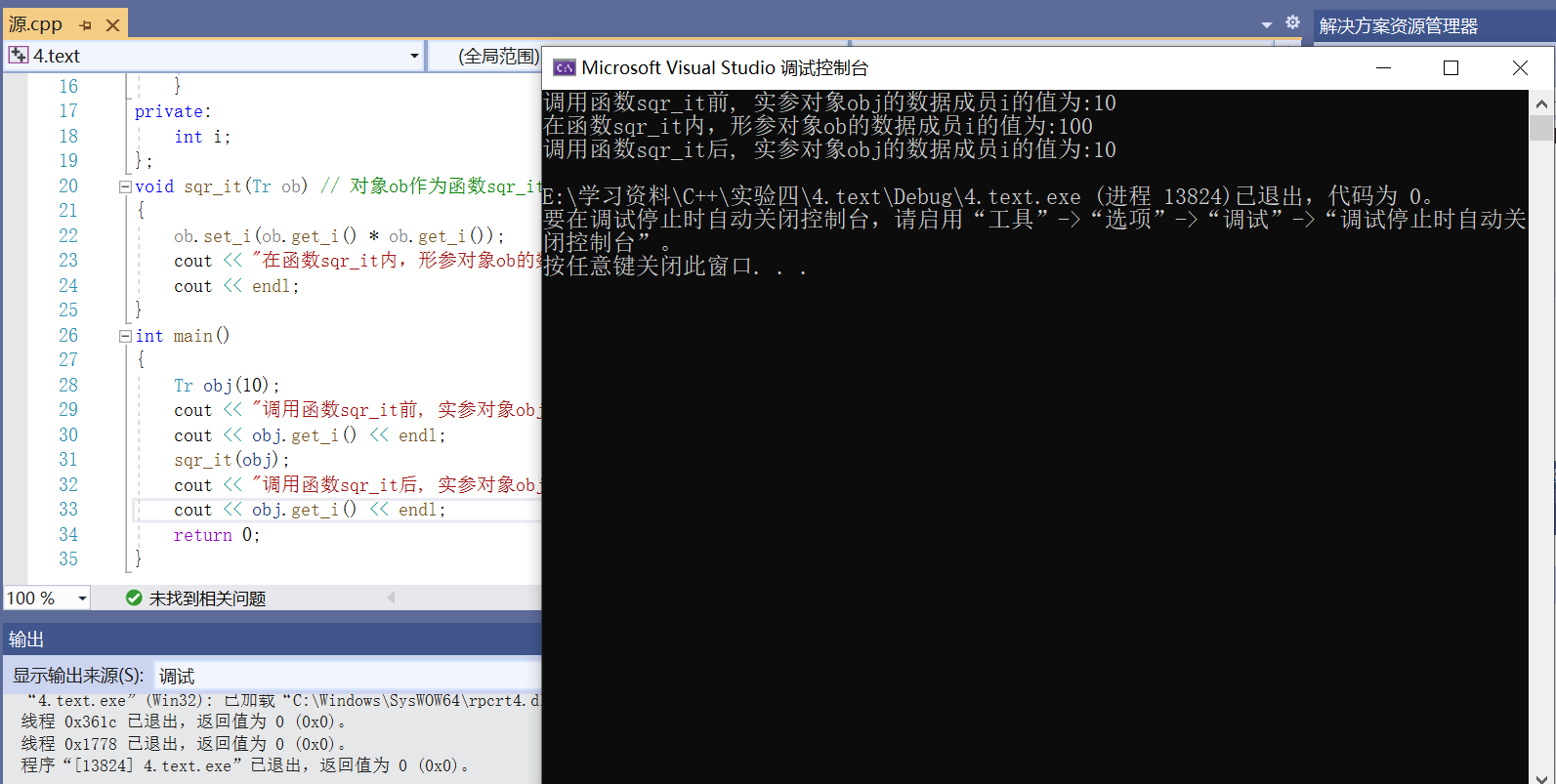
cout<<“在函数sqr\_it内，形参对象ob的数据成员i的值为:”<<ob.get\_i();

cout<<endl;

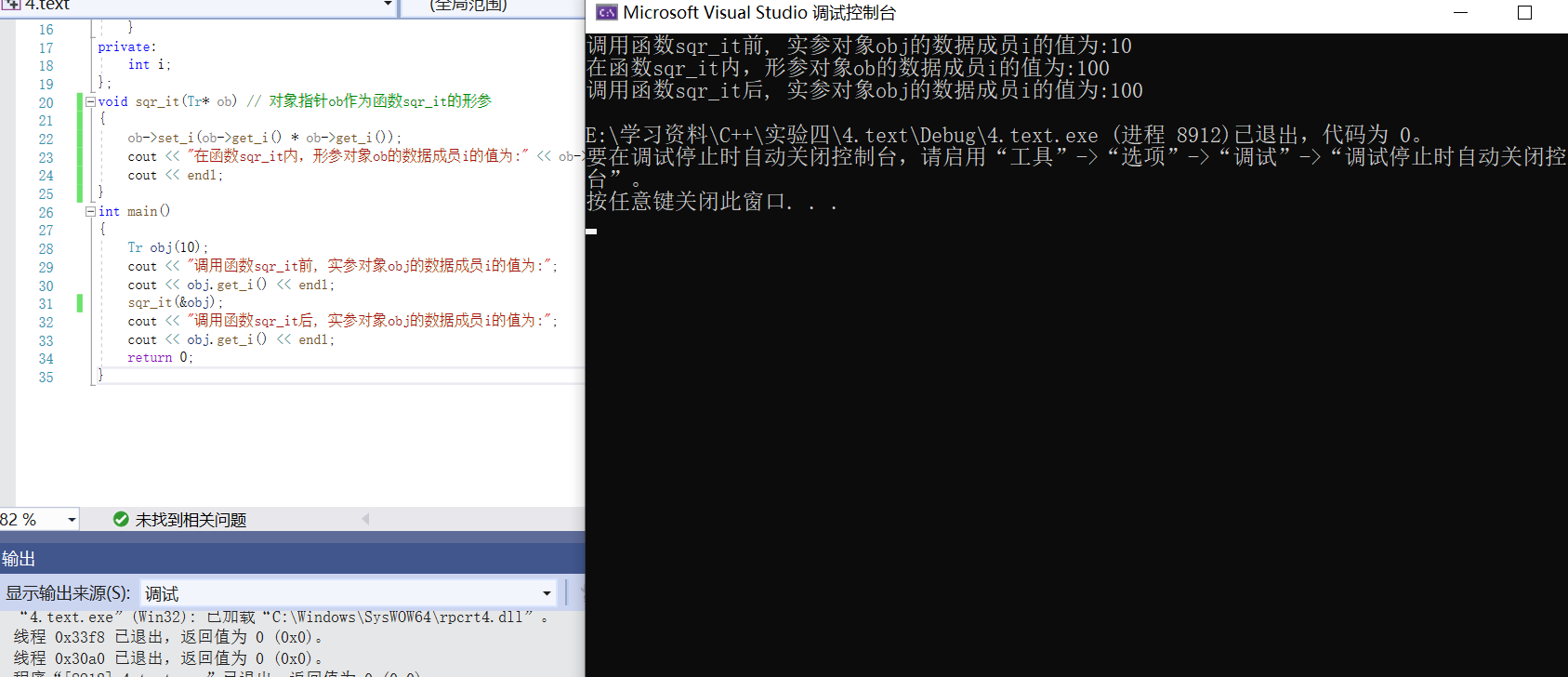
}

* 程序结果：

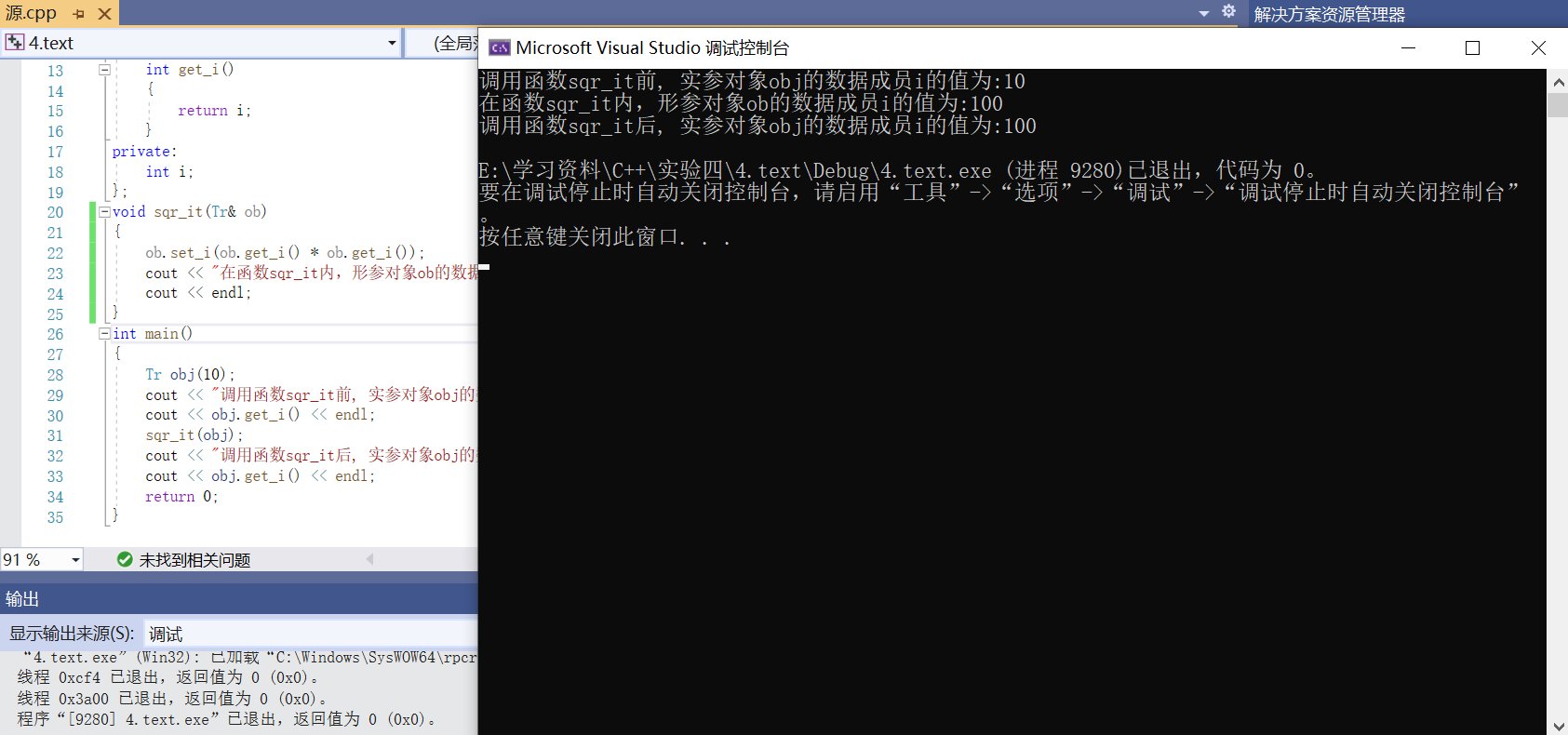
对象ob作为函数sqr\_it形参：



对象指针ob作为函数sqr\_it的形参：

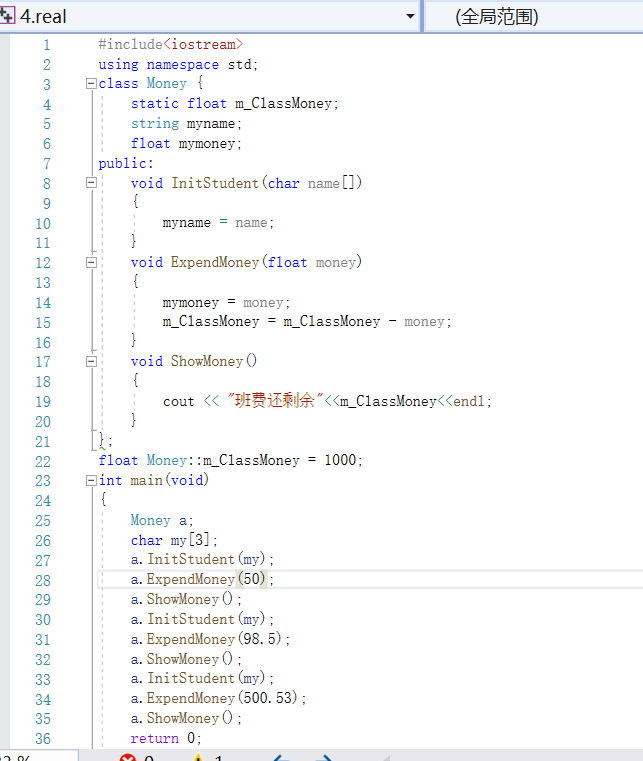


对象的引用作为函数sqr\_it的形参：

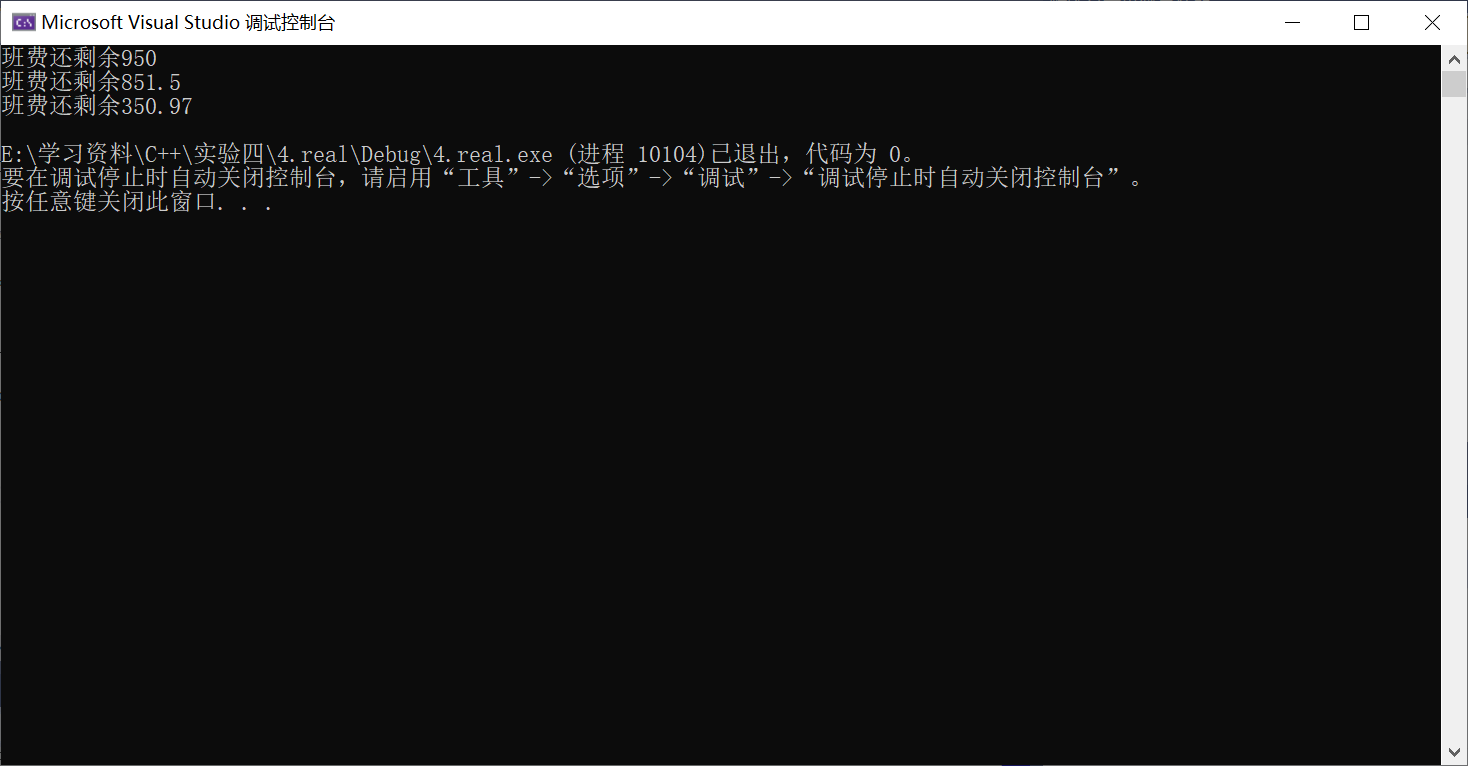


* 实验内容二（创建一个TStudent类）

1. 程序代码：



1. 程序结果：



问题总结：

当定义函数让对象指针ob作为函数sqr\_it的形参时直接运行发生错误，因为指针作为函数的形参在调用函数时应该加上&符号。

感想心得：

通过第四次上机实验，我进一步加深对类和对象的理解，掌握几种对象传递的使用方法并且掌握静态成员的概念和使用。

我温故了指针作为形参时函数应如何调用，也巩固了C++中的对C的拓展“引用&”；对其进一步理解了。

此外我对于静态成员有了属于自己的理解，初步学会了什么时候设置成静态成员，哪些成员应该设置为静态成员。

Copyright ©2021-2099 MengyaoChen. All rights reserved.