**Copyright ©2021-2099 liangxiankui. All rights reserved**

**4.1**

# 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;

}

程序结果：调用函数sqr\_it前, 实参对象obj的数据成员i的值为:10

在函数sqr\_it内，形参对象ob的数据成员i的值为:100

调用函数sqr\_it后, 实参对象obj的数据成员i的值为:10

**4.2**

#include "iostream.h"

#include <string.h>

class TStudent

{

private:

char m\_Name[6];

static float m\_ClassMoney;

public:

void InitStudent(char \*);

void ExpendMoney(float);

static void ShowMoney();

};

float TStudent::m\_ClassMoney=1000;

void TStudent::InitStudent(char name[])

{

strcpy(m\_Name,name);

}

void TStudent::ExpendMoney(float money)

{

m\_ClassMoney-=money;

cout<<m\_Name<<"花费班费"<<m\_ClassMoney<<endl;

}

void TStudent::ShowMoney()

{

cout<<"班费还剩余"<< m\_ClassMoney<<endl;

}

void main()

{

TStudent stu[3];//定义三个学生

stu[0].InitStudent("A");

stu[1].InitStudent("B");

stu[2].InitStudent("C");

stu[0].ExpendMoney(50);

TStudent::ShowMoney();

stu[1].ExpendMoney(98.5);

TStudent::ShowMoney();

stu[2].ExpendMoney(500.53);

TStudent::ShowMoney();

}

程序结果：A花费班费950  
班费还剩余950  
B花费班费851.5  
班费还剩余851.5  
c花费班费350.97  
班费还剩余350.97

感想：函数只是把a,b的值通过赋值传递给了x,y，函数里头操作的只是x,y的值并不是a,b的值。在向函数传递对象时，是通过“传值调用”传递给函数的，只由实参传递给形参，而不能由形参传递给实参。函数中对对象的任何修改均不影响调用该函数的对象。