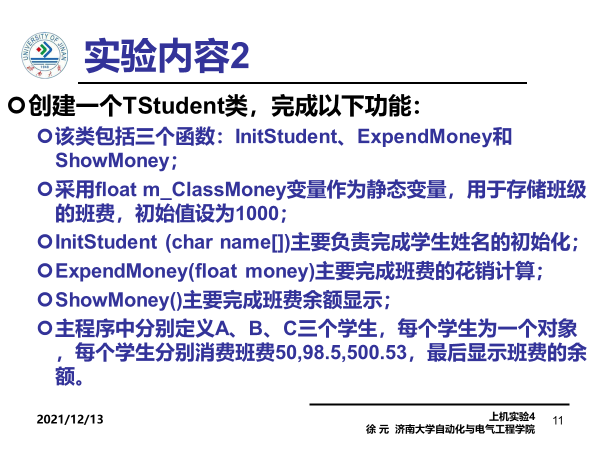
Copyright ©2021-2099 xuchao. All rights reserved

**实验要求：**



**程序代码：**

#include <iostream>

using namespace std;

static float m\_classMoney = 1000;

class TStudent

{

char\* name;

public:

void InitStudent(char name[])

{

name = name;

}

void ExpendMoney(float money, char name[]);

void showMoney();

};

void TStudent::ExpendMoney(float money, char name[])

{

m\_classMoney -= money;

cout << name << "花费班费" << money << "元" << endl;

}

void TStudent::showMoney()

{

cout << "班费还剩余" << m\_classMoney <<"\n" << endl;

}

int main()

{

TStudent stu[3];

char name[3][100];

int i;

double money[3] = { 50,98.5,500.53 };

for (i = 0; i < 3; i++)

{

cout << "请输入姓名:";

cin >> name[i];

stu[i].InitStudent(name[i]);

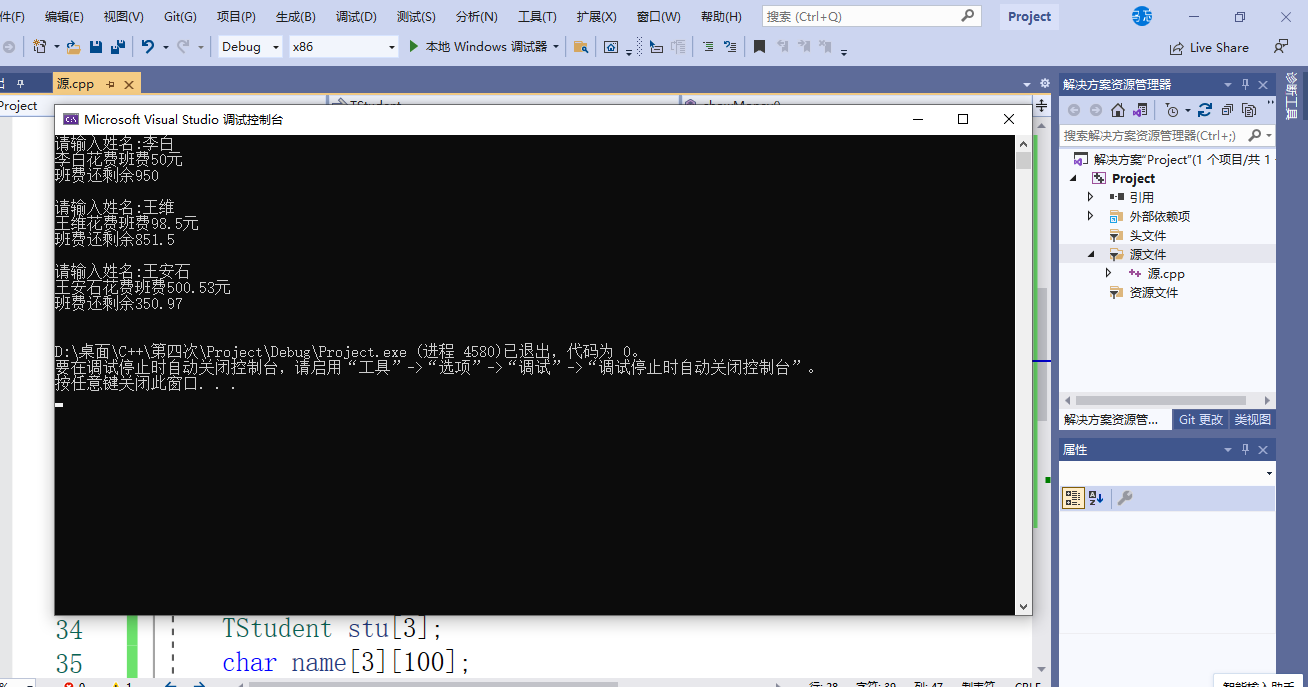
stu[i].ExpendMoney(money[i],name[i]);

stu[i].showMoney();

}

}

**程序结果：**



**感想心得：**

由于一开始时认为静态变量属于每一个对象，结果输出结果怎么都不对，查询资料后明白了静态变量属于类而不是某一个对象。

静态数据成员是类的成员，而不是对象的成员，所有该类对象都共用该数据成员，可以实现同类对象之间进行数据共享。

静态数据成员是静态存储的，它是静态生存期，必须对它进行初始化。

1、静态数据成员在定义或声明时前面加关键字static。

2、静态变量的初始化f方式:

<数据类型><类名>::<静态数据成员名>=<值>

初始化在类外实现，不能在头文件中，前面不加static，需要使用作用域运算符来标明它所属类；