#include<iostream>

#include<string>

using namespace std;

class score {

public:

score()

{

times = 2;

cout << "默认学生为2" << endl;

}

score(int times1)

{

times = times1;

cout << "自定学生人数" << endl;

}

~score()

{

cout << "~score" << endl;

}

void inputscore()

{

for (int i = 0; i < times; i++)

{

cout << "输入学生姓名" << endl;

cin >> name[i];

cout << "输入科目A成绩" << endl;

cin >> grade[i][0];

cout << "输入科目B成绩" << endl;

cin >> grade[i][1];

cout << "输入科目C成绩" << endl;

cin >> grade[i][2];

}

}

void showscore()

{

for (int i = 0; i < times; i++)

{

cout << "学生姓名是" << name[i] << endl;

cout << "科目A成绩是" << grade[i][0] << endl;

cout << "科目B成绩是" << grade[i][1] << endl;

cout << "科目C成绩是" << grade[i][2] << endl;

}

}

void showStuAvgScore()

{

//float avg = 0;

for (int i = 0; i < times; i++)

{

float avg = 0;

for (int j = 0; j <= 2; j++)

{

avg = avg + grade[i][j];

}

avg = avg / 3;

cout << name[i] << "同学的平均值是" << avg << endl;

}

}

void showSubAvgScore()

{

float avgA = 0;

float avgB = 0;

float avgC = 0;

for (int i = 0; i < times; i++)

{

avgA = avgA + grade[i][0];

avgB = avgB + grade[i][1];

avgC = avgC + grade[i][2];

}

avgA = avgA / times;

avgB = avgB / times;

avgC = avgC / times;

cout << "科目A的平均分是" << avgA << endl;

cout << "科目B的平均分是" << avgB << endl;

cout << "科目C的平均分是" << avgC << endl;

}

void sortSub()

{

float temp;

for (int i = 0; i <= 2; i++)

{

for (int j = 1; j < times; j++)

{

if (grade[j][i] > grade[j - 1][i])

{

temp = grade[j][i];

grade[j][i] = grade[j - 1][i];

grade[j - 1][i] = temp;

}

}

for (int j = 0; j < times; j++)

{

cout << grade[j][i] << " ";

}

cout << endl;

}

}

private:

float grade[100][100];

int times;

string name[100];

};

int main()

{

score x;

x.inputscore();

x.showscore();

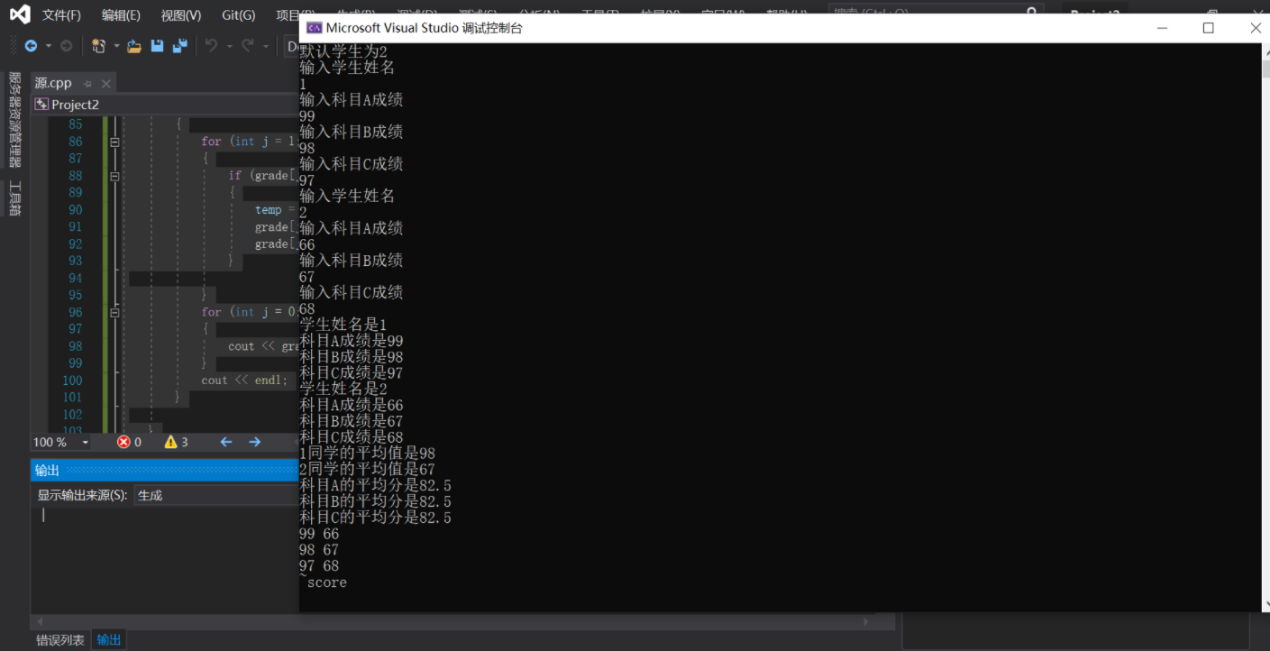
x.showStuAvgScore();

x.showSubAvgScore();

x.sortSub();

}

**运行结果**



**总结：**

用到了冒泡法，之前C语言学过。

冒泡法排序：  
从第一个和第二个开始比较，如果第一个比第二个大，则交换位置，然后比较第二个和第三个，逐渐往后

经过第一轮后最大的元素已经排在最后，所以重复上述操作的话第二大的则会排在倒数第二的位置。

那重复上述操作n-1次即可完成排序，因为最后一次只有一个元素所以不需要比较。