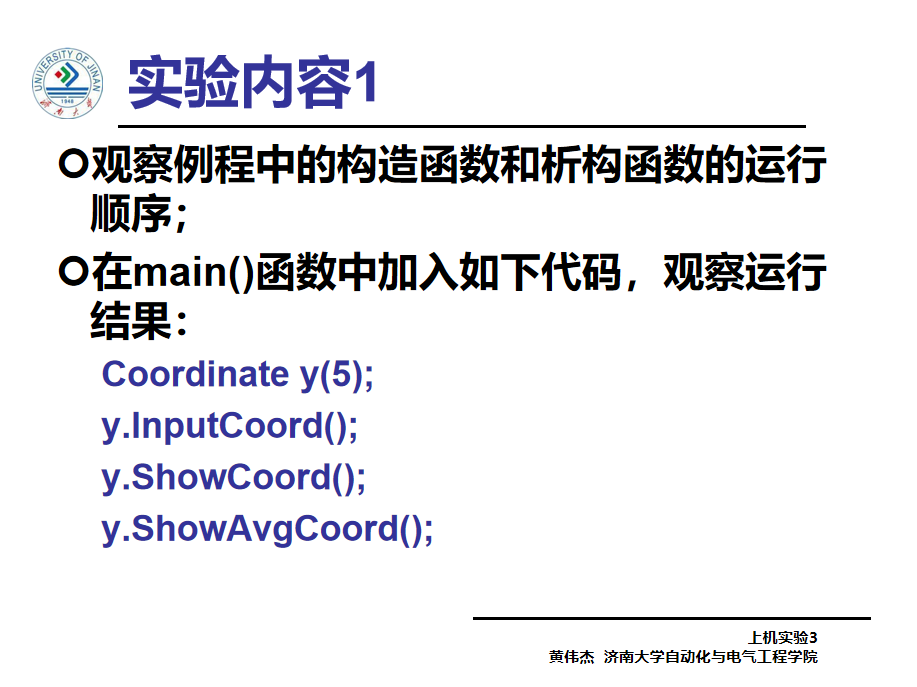
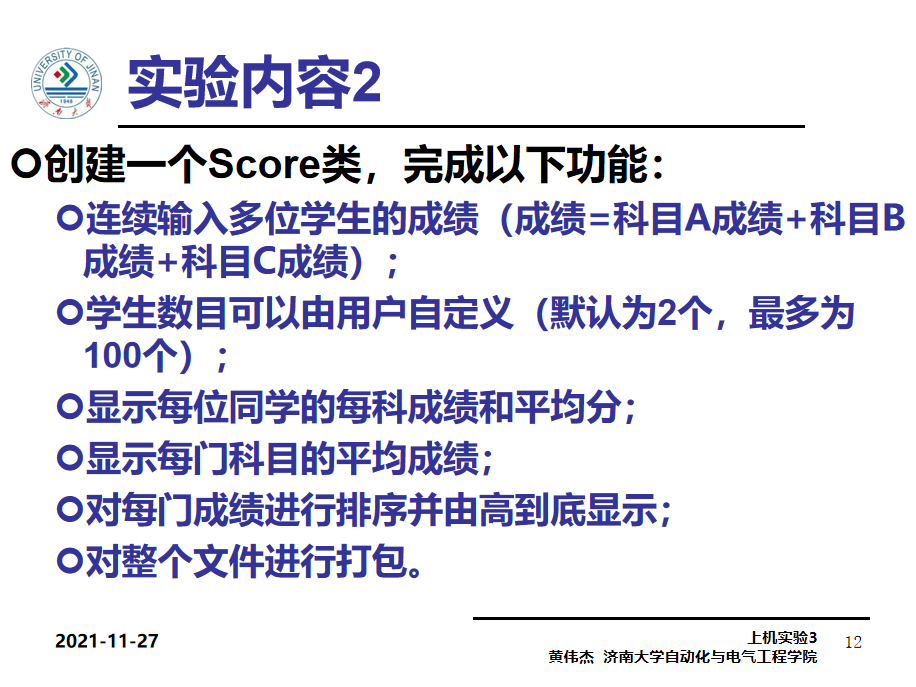
一、实验题目

上机实验3：构造函数和析构函数





二、实验过程

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* 实验一 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

———————————————原始代码：———————————————

#include<iostream>

using namespace std;

class Coordinate {

public:

Coordinate()

{

times = 2;

cout << "Coordinate construction1 called!" << endl;

}

Coordinate(int times1)

{

times = times1;

cout << "Coordinate construction2 called!" << endl;

}

~Coordinate()

{

cout << "Coordinate destruction called!" << endl;

}

void InputCoord()

{

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

{

cout << "Please Input x:" << endl;

cin >> Coord[i][1];

cout << "Please Input y:" << endl;

cin >> Coord[i][2];

}

}

void ShowCoord()

{

cout << "The coord is:" << endl;

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

{

cout << "(" << Coord[i][1] << "," << Coord[i][2] << ")" << endl;

}

}

void ShowAvgCoord()

{

float avgx = 0;

float avgy = 0;

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

{

avgx = avgx + Coord[i][1];

avgy = avgy + Coord[i][2];

}

avgx = avgx / times;

avgy = avgy / times;

cout << "The AVG coord is:" << endl;

cout << "(" << avgx << "," << avgy << ")" << endl;

}

private:

float Coord[100][100];

int times;

};

int main()

{

Coordinate x;

x.InputCoord();

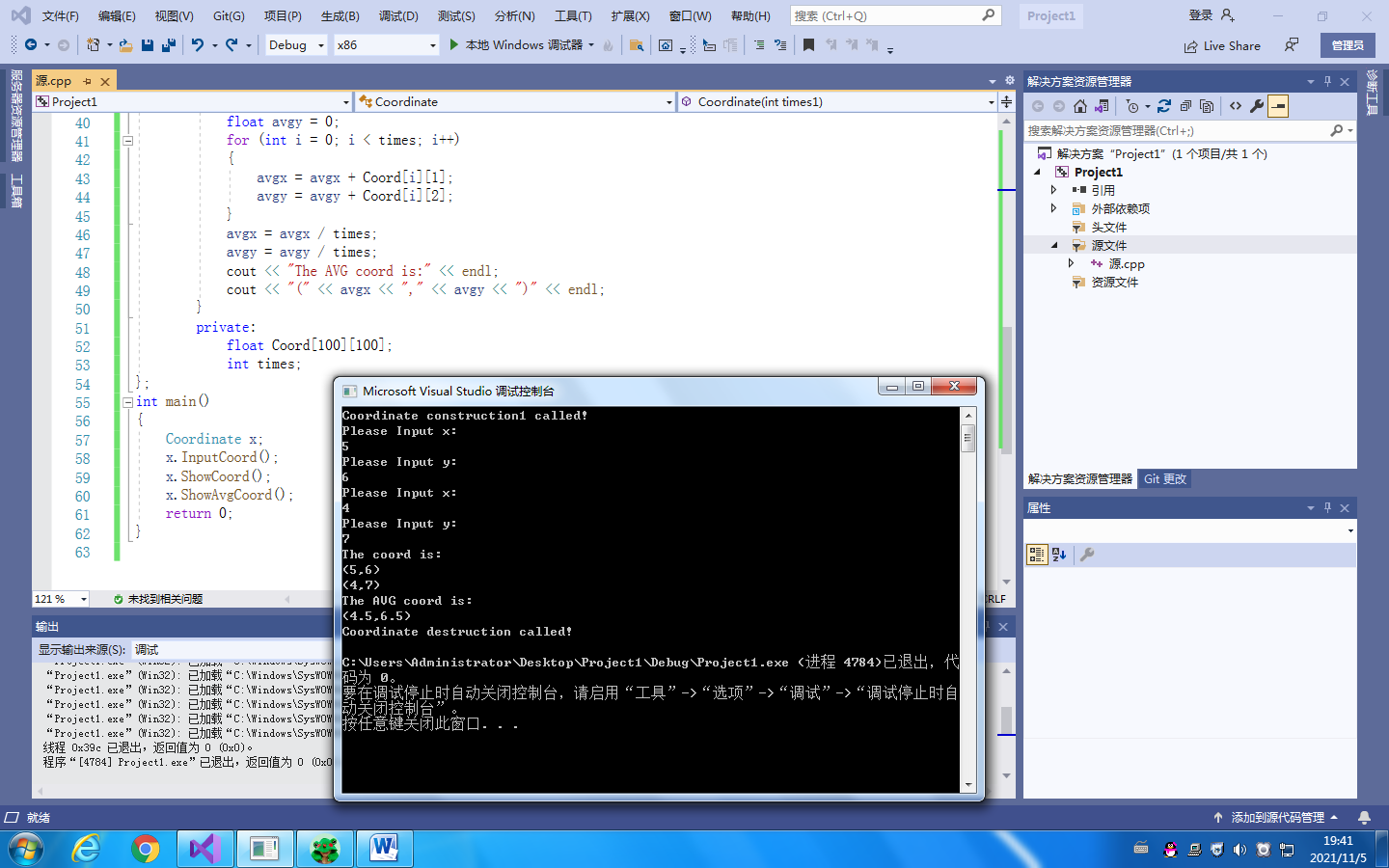
x.ShowCoord();

x.ShowAvgCoord();

return 0;

}

———————————————运行结果：———————————————



———————————————主函数更换为—————————————

int main()

{

Coordinate y(5);

y.InputCoord();

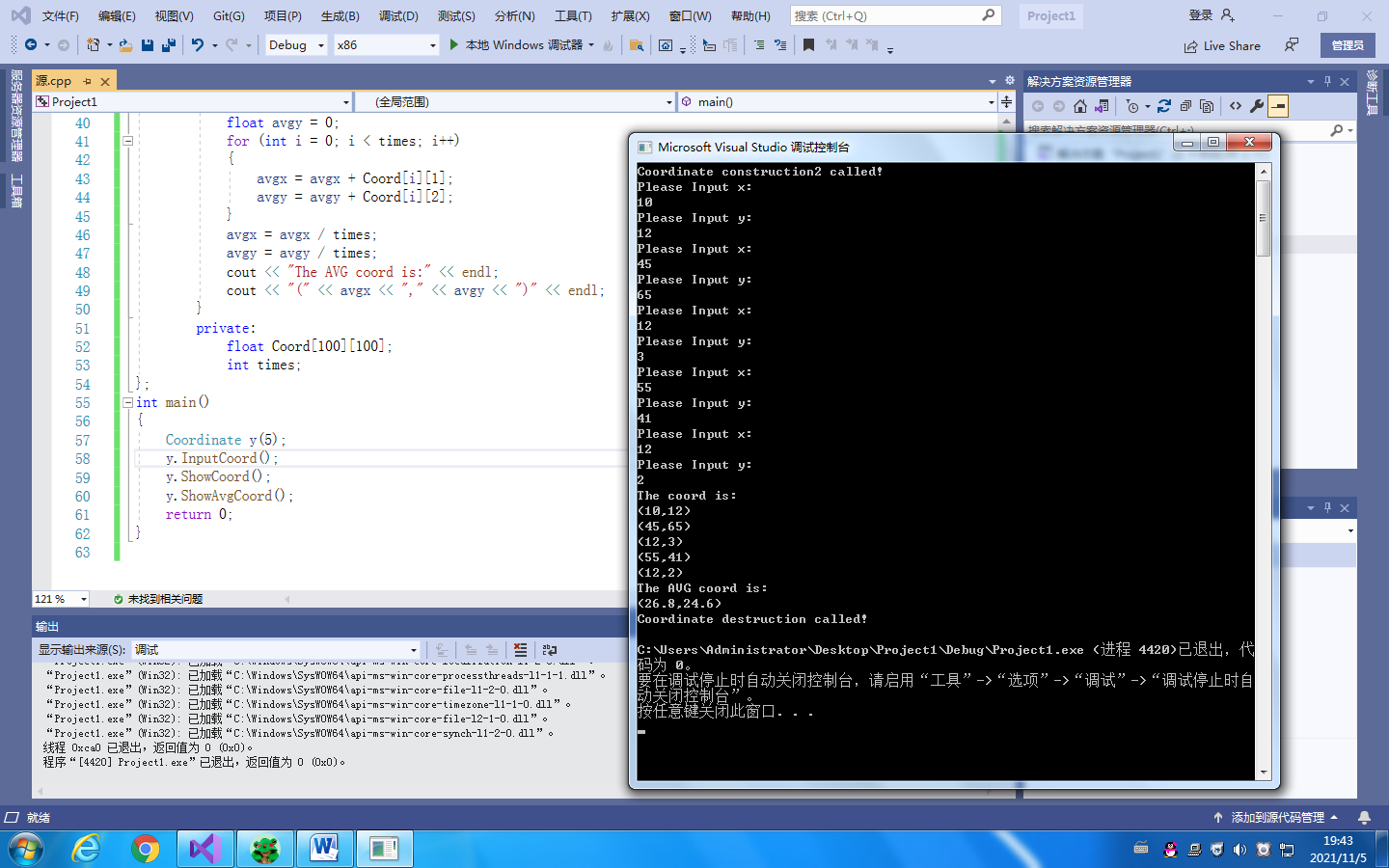
y.ShowCoord();

y.ShowAvgCoord();

return 0;

}

—————————————————运行结果：——————————————————



\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* 实验二 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

—————————————————代码：—————————————

#include<iostream>

using namespace std;

class Coordinate {

public:

Coordinate()

{

times = 2;

cout << "Coordinate construction1 called!" << endl;

}

Coordinate(int times1)

{

times = times1;

cout << "Coordinate construction2 called!" << endl;

}

~Coordinate()

{

cout << "Coordinate destruction called!" << endl;

}

void Input()//输入学生姓名，成绩

{

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

{

cout << "请输入学生姓名：" << endl;

cin >> name[i];

cout << "请输入科目A成绩：" << endl;

cin >> Coord[i][0];

cout << "请输入科目B成绩：" << endl;

cin >> Coord[i][1];

cout << "请输入科目C成绩：" << endl;

cin >> Coord[i][2];

}

}

void ShowScore()//每位同学成绩，平均成绩

{

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

{

cout << "姓名：" << name[i] << " " ;

cout << "科目A成绩：" <<Coord[i][0]<< " " ;

cout << "科目B成绩：" <<Coord[i][1]<< " " ;

cout << "科目C成绩：" << Coord[i][2]<< " " ;

cout << "平均成绩：" << (Coord[i][0]+ Coord[i][1]+Coord[i][2])/3.0 << " "<<endl;

}

}

void ShowAvgscore()//每门科目平均成绩

{

float sum;

for (int j = 0; j < 3; j++) {

sum = 0;

cout << "课程名称：" <<(char)(j+65)<<" ";

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

sum = sum + Coord[i][j];

}

cout << "平均成绩：" << sum / times << endl;

}

}

void ShowOrder() {

int k;

float r;

string p;

for (int t = 0; t < 3; t++) {

cout << "科目" << (char)(t+65)<< "成绩排序：" << endl;

for (int i = 0; i < times - 1; i++) {

k = i;

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

if (Coord[k][t] < Coord[j][t]) {

k = j;

}

if (k != i) {

r = Coord[k][t];

Coord[k][t] = Coord[i][t];

Coord[i][t] = r;

p = name[k];

name[k] = name[i];

name[i] = p;

}

}

cout << "姓名：" << name[i] << " 成绩：" << Coord[i][t] << endl;

}

cout << "姓名：" << name[times - 1] << " 成绩：" << Coord[times - 1][t] << endl;

}

}

private:

string name[100];

float Coord[100][3];

int times;

};

int main()

{

Coordinate y;

y.Input();

y.ShowScore();

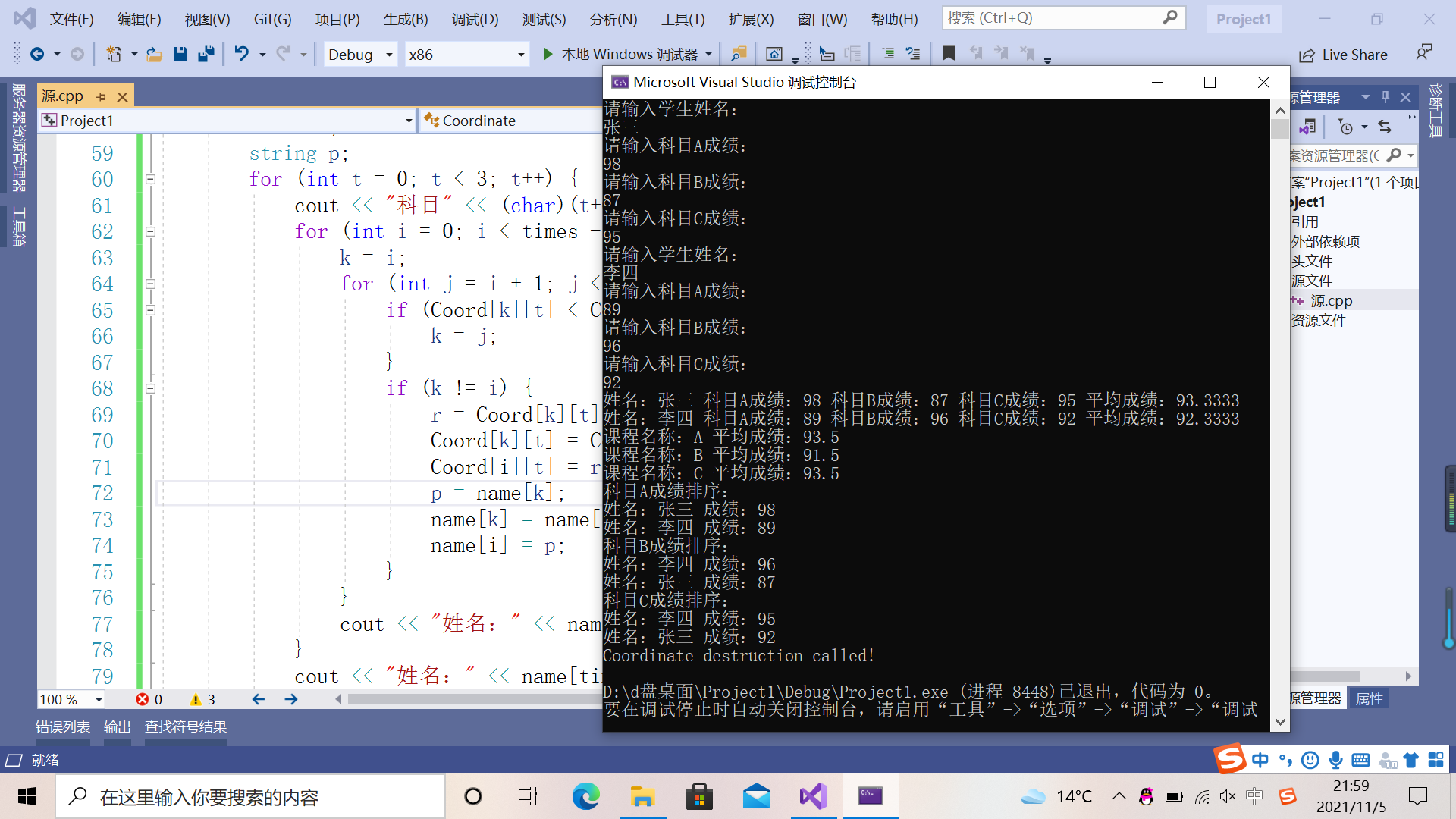
y.ShowAvgscore();

y.ShowOrder();

return 0;

}

————————————————运行结果：——————————————



三、问题分析与感想

1、开始时，我不知道如何用一个函数显示ABC三门课，于是重复写了三遍一样的程序，但之后发现可以使用ASCII码来显示，使用强制类型转换将ASCII码数值转换成字符，例如65转换为"A"，这样三个科目可以合并为一个循环，减少了代码量。之后看了老师的讲解，发现ABC课程名称是用实参赋值给形参的，这样的写法更加适合大多数情况。

2、在机房电脑输入姓名成绩时出现问题，导致只能输入一个姓名就程序结束，可以把姓名单独用一个循环进行输入。在我自己的电脑上尝试时，没有出现这样的问题，以上实验二代码和运行结果均是我自己电脑运行的。

3、一维的字符数组char无法储存每个同学的姓名，需要使用字符串数组string或者用二维的字符数组，然后输入时用name[1][]确保输入一个姓名

电自2002 许吴瑕 202030310293

Copyright ©2021-2099 WuxiaXu. All right reserved