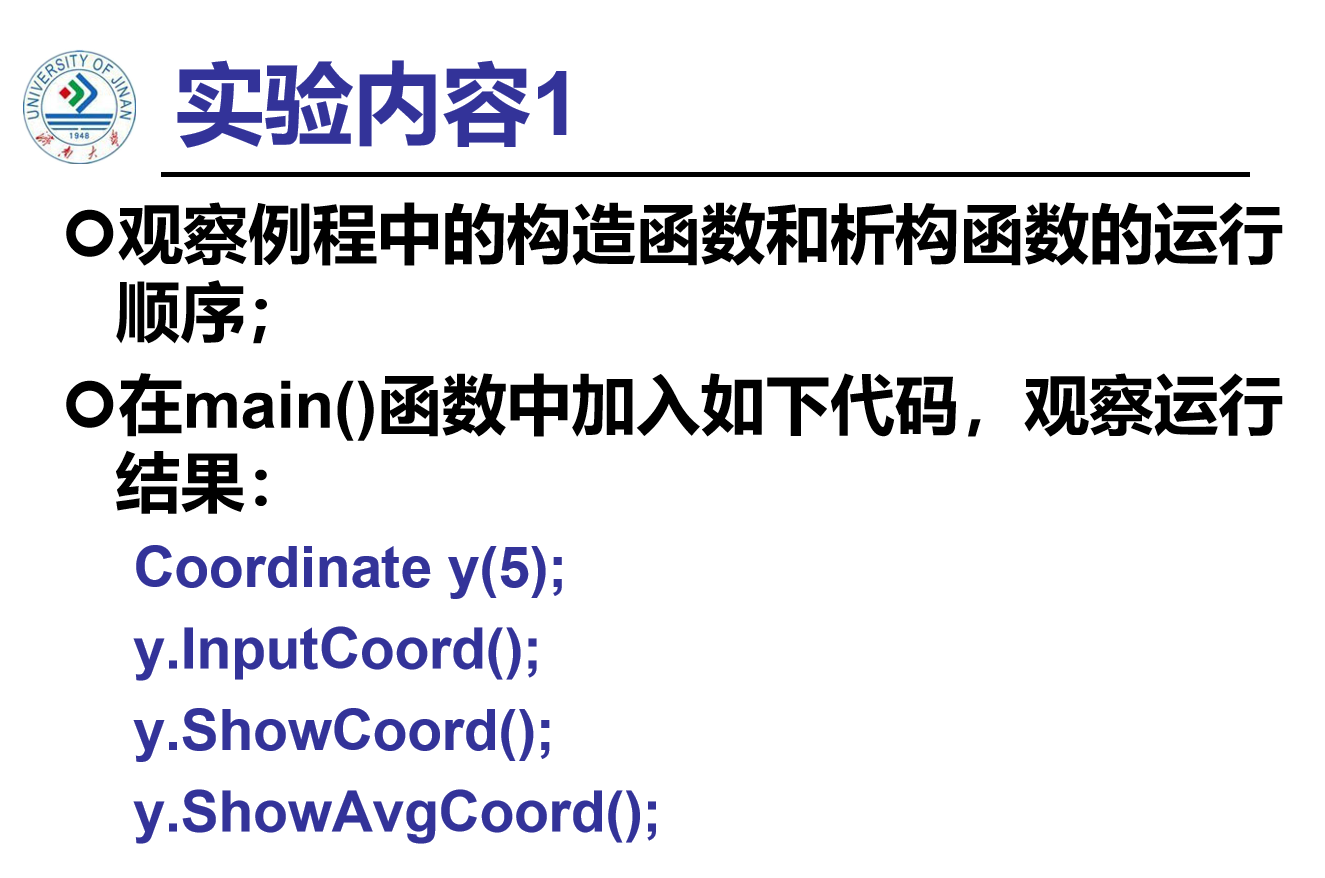
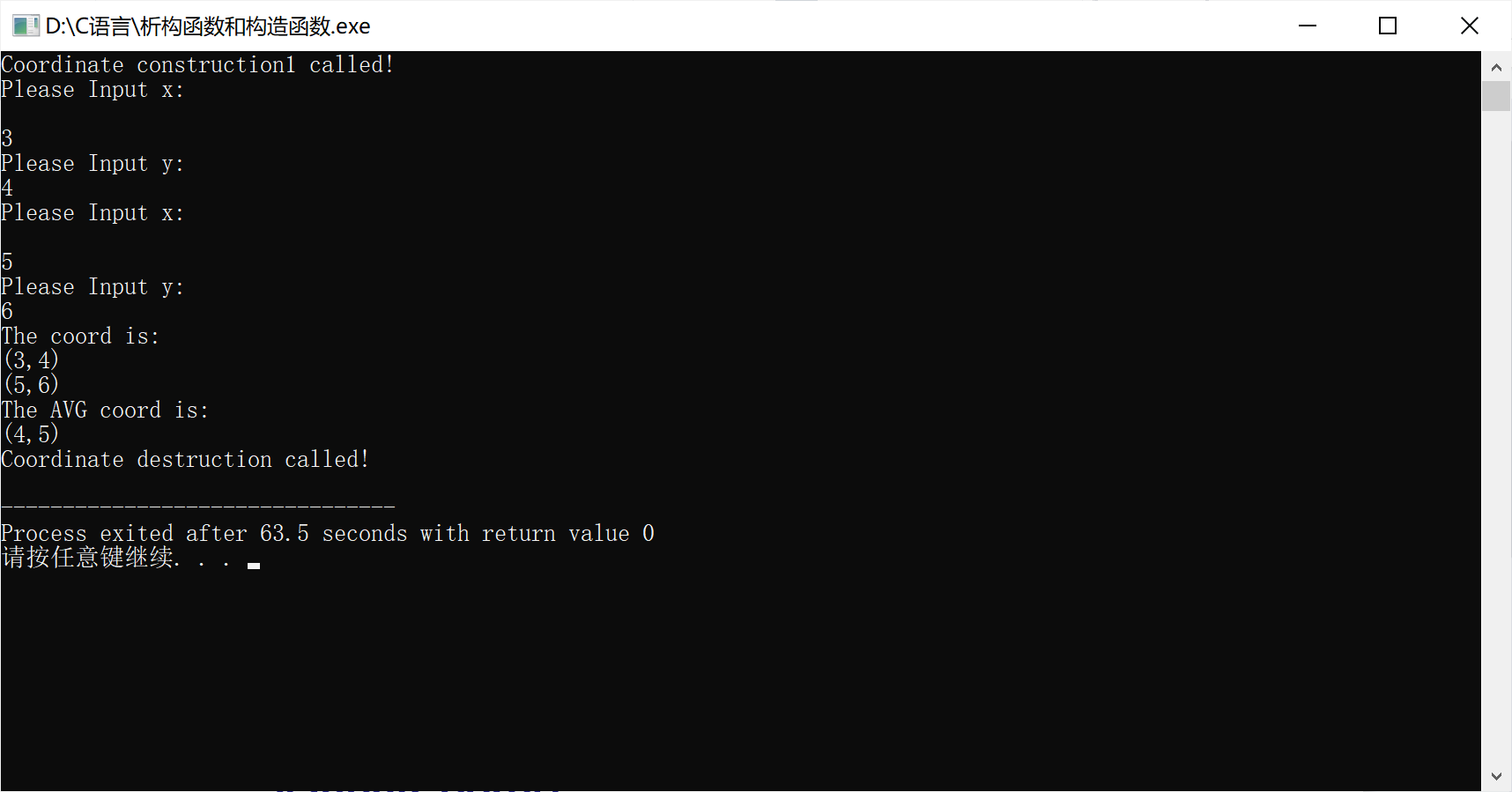
上机实验三

**班级：电自2004 学号：202030310161 姓名：徐海宸**

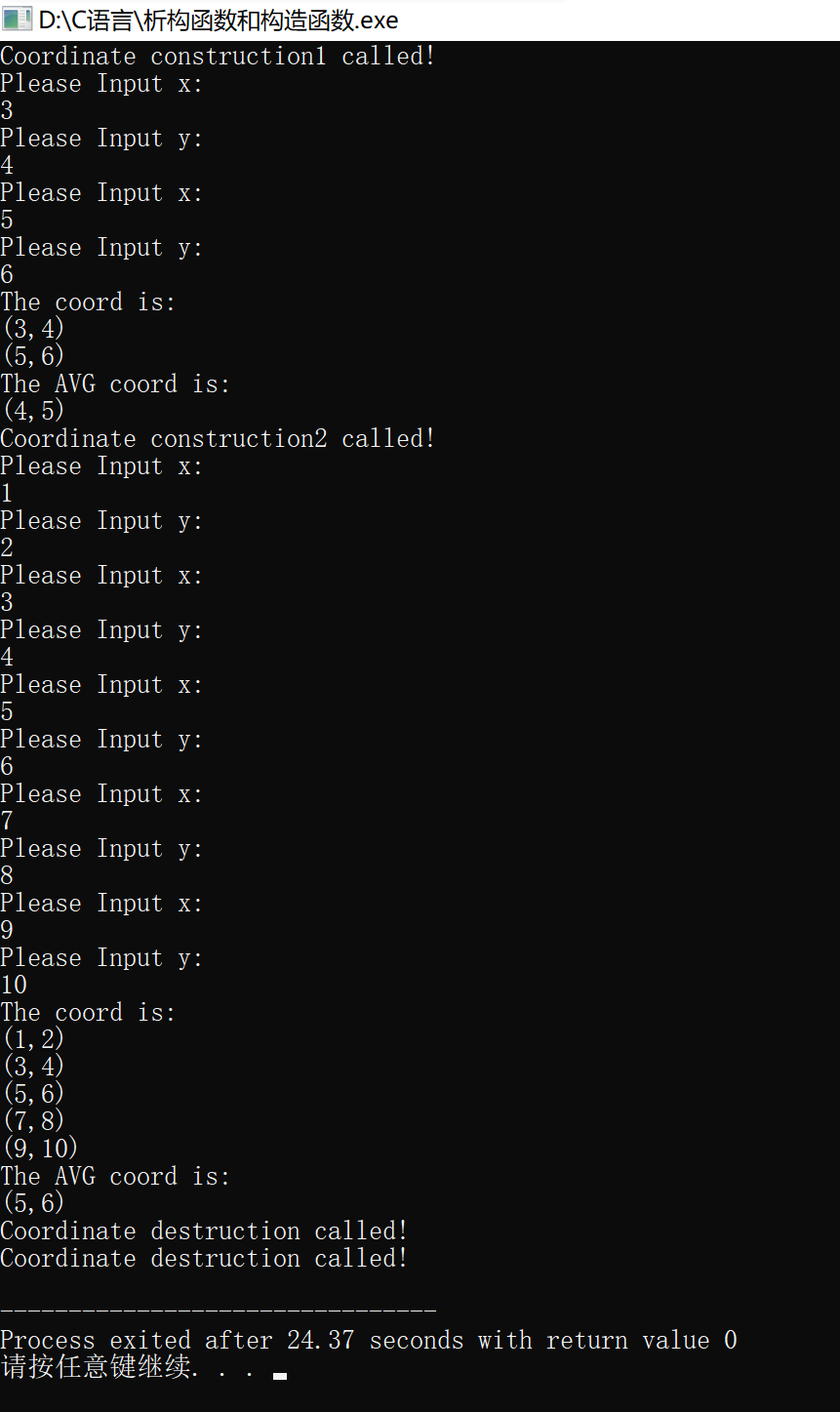


**○运行结果：**

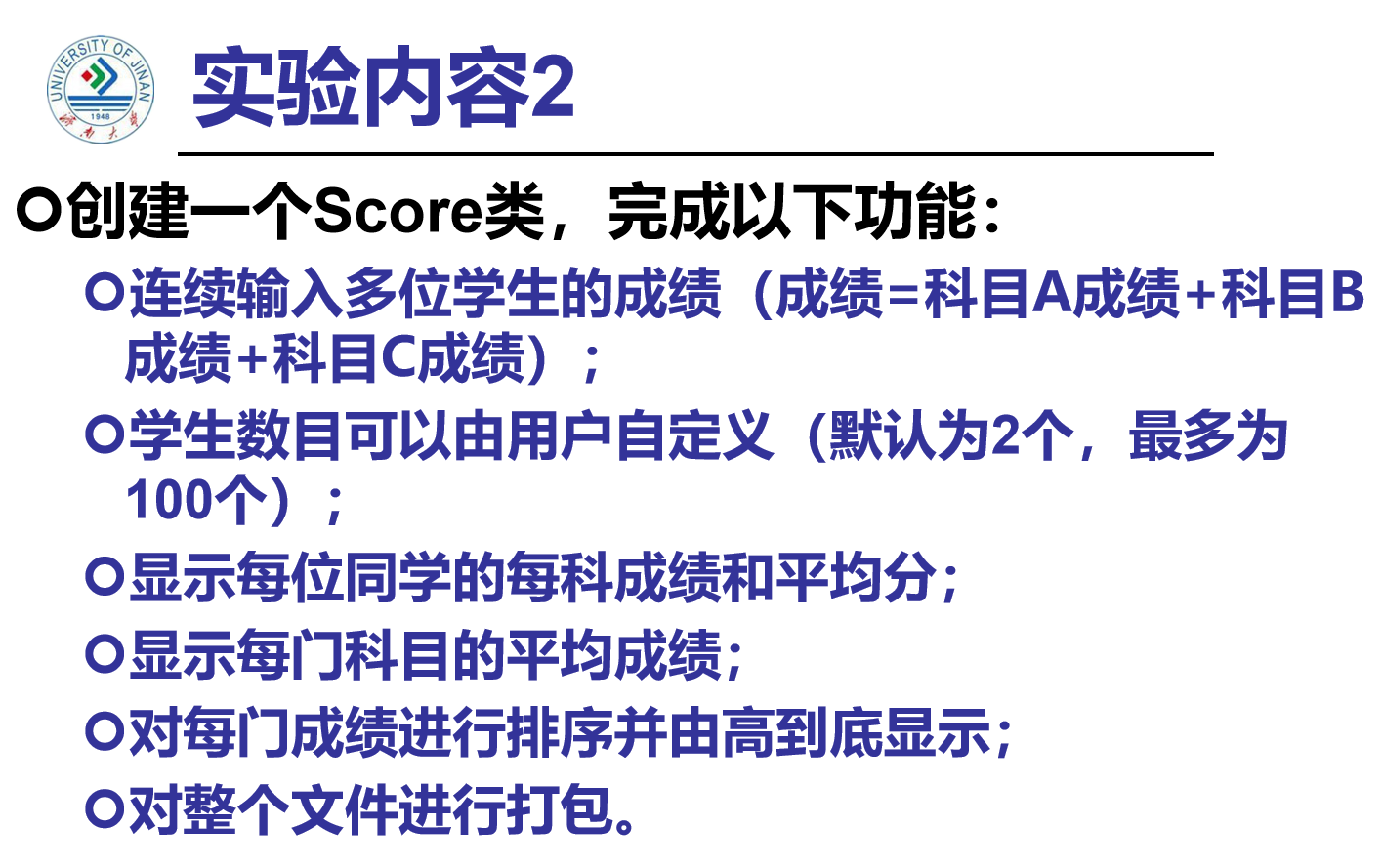


发现程序先执行了不含参数的构造函数Coordinate，在输入完两个坐标值并求出平均坐标值后显示出来，最后再执行析构函数~Coordinate

**○加入代码后运行结果：**



可以看到程序先执行了没有参数的构造函数Coordinate，输入完两个坐标并显示该坐标，求出平均坐标后也显示出来；然后执行了有参数的构造函数Coordinate，参数为5，之后调用成员函数inputCoord将五个坐标输入并显示了出来，求出其均坐标后显示了出来；最后先调用了有参数的构造函数的析构函数，后调用了不含参数的构造函数Coordinate的析构函数。



**程序代码：**

#include<iostream>

#include<string>

using namespace std;

class Score {

public:

Score()

{

times = 2;

}

Score(int times1)

{

times = times1;

}

void InputNameAndScore()

{

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 >> SScore[i][1];

cout << "请输入科目B成绩：" << endl;

cin >> SScore[i][2];

cout << "请输入科目C成绩：" << endl;

cin >> SScore[i][3];

}

}

void ShowNameAndScore()

{

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

{

cout << "姓名：" << Name[i] << "科目A成绩" << SScore[i][1] << "科目B成绩"

<< SScore[i][2] << "科目C成绩：" << SScore[i][3] << endl;

}

}

void ShowStdentAvgScore(int Sid)

{

float avg = 0;

avg = (SScore[Sid][1] + SScore[Sid][2] + SScore[Sid][3]) / 3;

cout << "姓名：" << Name[Sid] << "平均成绩：" << avg << endl;

}

void ShowClassAvgScore(string ClassName)

{

int Cid;

float avg = 0;

if (ClassName == "A")Cid = 1;

if (ClassName == "B")Cid = 2;

if (ClassName == "C")Cid = 3;

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

{

avg = avg + SScore[i][Cid];

}

avg = avg / times;

cout << "课程名称：" << ClassName << "平均成绩：" << avg << endl;

}

void OderScore(string ClassName)

{

int Cid;

if (ClassName == "A")Cid = 1;

if (ClassName == "B")Cid = 2;

if (ClassName == "C")Cid = 3;

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

{

SScore1[i] = SScore[i][Cid];

}

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

{

Name1[i] = Name[i];

}

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

{

if (SScore1[i] > SScore1[i - 1])

{

float temp = SScore1[i - 1];

SScore1[i - 1] = SScore1[i - 1];

SScore1[i] = temp;

string temp1;

temp1 = Name1[i - 1];

Name1[i - 1] = Name1[i];

Name1[i] = temp1;

}

}

cout << "课程名称：" << ClassName << endl;

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

{

cout << "姓名：" << Name1[i] << "成绩：" << SScore1[i] << endl;

}

}

private:

float SScore[100][3], SScore1[100];

string Name[100], Name1[100];

int times;

};

int main()

{

Score x;

x.InputNameAndScore();

x.ShowNameAndScore();

x.ShowStdentAvgScore(1);

x.ShowClassAvgScore("A");

x.OderScore("B");

return 0;

}

**运行结果：**



**问题总结及个人感想：**

本次试验我学到了如何创建类并且在类中定义析构函数、成员函数，但是编写每个成员函数的函数体是我最头疼的，由于逻辑关系很复杂，一开始我找不到头绪，后来由老师给的例子中构造函数的表达以及向同学寻求了帮助后我开始逐步编写每一个成员函数的函数体，后来经过好几次编译失败，原因包括for循环中i我没有先定义再使用，还有一些函数体的大括号没扩对地方，最后经过仔细的检查和纠正，总算能运行程序了。回过头来看整个程序，我发现主函数中语句很少，大部分都是调用的类中的成员函数，和C语言不一样，整个程序中最复杂的地方都在成员函数的函数体里，内容逻辑上是和C语言差不多的。通过本次实验我学会了怎样定义一个类并且编写成员函数，实现对构造函数、析构函数、成员函数的调用。

Copyright IMG_2562021-2099 HaiChenXu. All rights reserved.