Copyright ©2021-2099 FengJiaHe. All rights reserved

**例题/练习1：**

#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 construction 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;

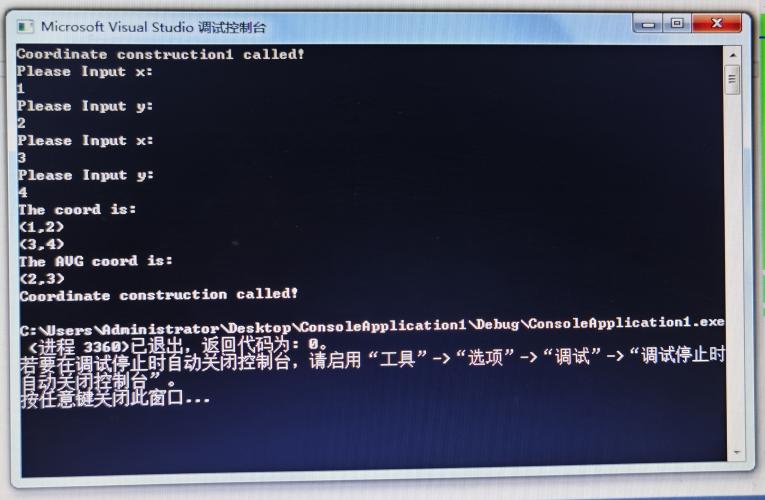
x.InputCoord();

x.ShowCoord();

x.ShowAvgCoord();

return 0;

}



**练习2：**

#include <iostream>

#include <string>

using namespace std;

class Score{

public:

Score()

{

times = 2;

}

Score(int times1)

{

times = times1;

}

void Input()

{

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

{

cout << "请输入学生姓名：" << endl;

cin >> name1[i];

cout << "请输入科目A成绩：" << endl;

cin >> Coord[i][1];

cout << "请输入科目B成绩：" << endl;

cin >> Coord[i][2];

cout << "请输入科目C成绩：" << endl;

cin >> Coord[i][3];

}

}

void Show()

{

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

{

cout << "姓名:" << name1[i] << " 科目A成绩:" << Coord[i][1] << " 科目B成绩:" << Coord[i][2] << " 科目C成绩:" << Coord[i][3] << endl;

}

}

void gerenavg()

{

float avg = 0;

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

{

avg = (Coord[i][1] + Coord[i][2] + Coord[i][3]) / 3;

cout << "姓名:" << name1[i] << " 平均成绩:" << avg << endl;

}

}

void kechengavg()

{

float avgA = 0,avgB = 0,avgC = 0;

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

{

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

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

avgC = avgC + Coord[i][3];

}

avgA = avgA / times;

avgB = avgB / times;

avgC = avgC / times;

cout << "课程名称:A 平均成绩:" << avgA << endl;

cout << "课程名称:B 平均成绩:" << avgB << endl;

cout << "课程名称:C 平均成绩:" << avgC << endl;

}

void paiming(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++)

{

Coord1[i] = Coord[i][Cid];

}

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

{

name2[i] = name1[i];

}

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

{

if (Coord1[i] > Coord1[i - 1])

{

float temp = Coord1[i - 1];

Coord1[i - 1] = Coord1[i];

Coord1[i] = temp;

string temp1;

temp1 = name2[i - 1];

name2[i - 1] = name2[i];

name2[i] = temp1;

}

}

cout << "课程名称：" << ClassName << endl;

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

{

cout << "姓名：" << name2[i] << " 成绩：" << Coord1[i] << endl;

}

}

private:

float Coord[100][100], Coord1[100];

int times;

string name1[100],name2[100];

};

int main()

{

Score x;

x.Input();

x.Show();

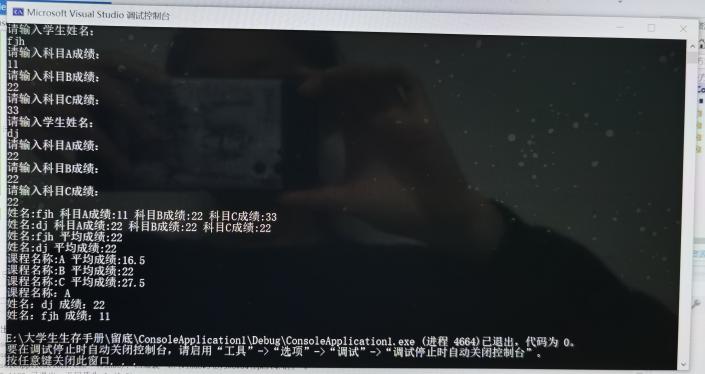
x.gerenavg();

x.kechengavg();

x.paiming("A");

return 0;

}



**感想心得：**

本次实验实践书写构造函数和析构函数，检验理论知识的学习成果。最初很难理解的理论概念，在实践中逐渐内化。

通过分析例题，举一反三就可以做出练习。我出现的问题在于大量变量名干扰了逻辑，还存在输出内容显示上的一些细节，通过理清思路并调整循环函数解决了问题。

我理解了类和对象的概念，掌握声明类和定义对象的方法，知道了构造函数和析构函数的运行顺序。