Copyright ©2021-2099 Aiting Chen. All rights reserved

上机作业3

#include<iostream>

using namespace std;

class Score {

public:

Score()

{

grade= 2;

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

}

Score(int grade1)

{

grade = grade1;

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

}

Score(float grade2)

{

grade = grade2;

cout << "Score construction3 called!" << endl;

}

~Score()

{

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

}

void InputCoord()

{

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

{

for (j = 0; j < grade; j++)

{

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

cin >> Coord[i][j][1];

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

cin >> Coord[i][j][2];

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

cin >> Coord[i][j][3];

}

}

}

void ShowCoord()

{

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

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

{

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

{

cout << "(" << Coord[i][j][1] << "," << Coord[i][j][2] <<","<<Coord[i][j][3]<< ")" << endl;

}

}

}

void ShowAvgCoord()

{

float avgx = 0;

float avgy = 0;

float avgz = 0;

for (int i = 0; i < grade; 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][100];

int grade;

};

int main()

{

Score x;

x.InputCoord();

x.ShowCoord();

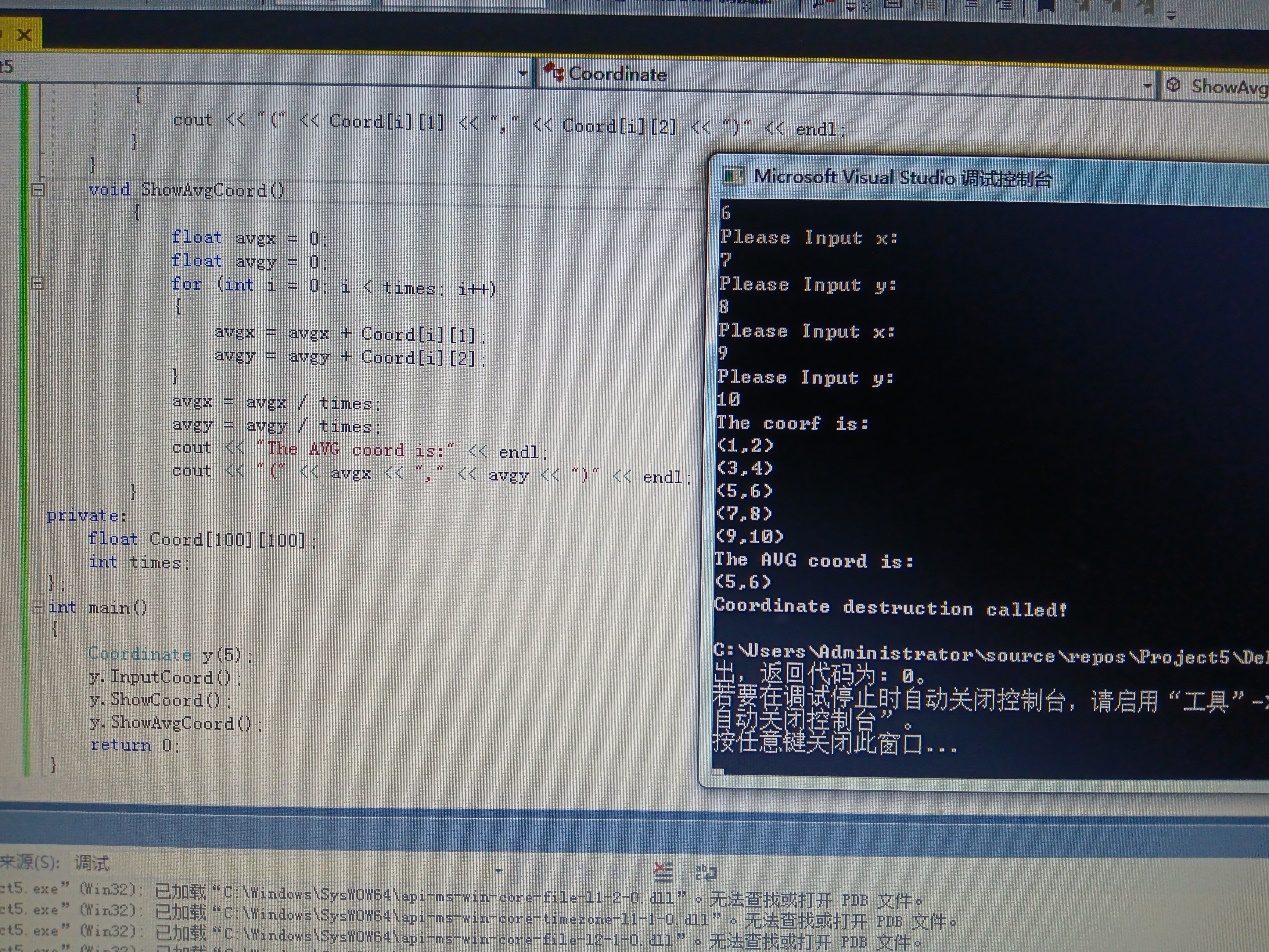
x.ShowAvgCoord();

Scorey(5);

y.InputCoord();

y.ShowCoord();

y.ShowAvgCoord();

return 0;

#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 ShowStudentAvgScore(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 OrderScore(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 (int i = 0; i < times; i++)

{

Name1[i] = Name[i];

}

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

{

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

{

float temp = SScore1[i - 1];

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

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();

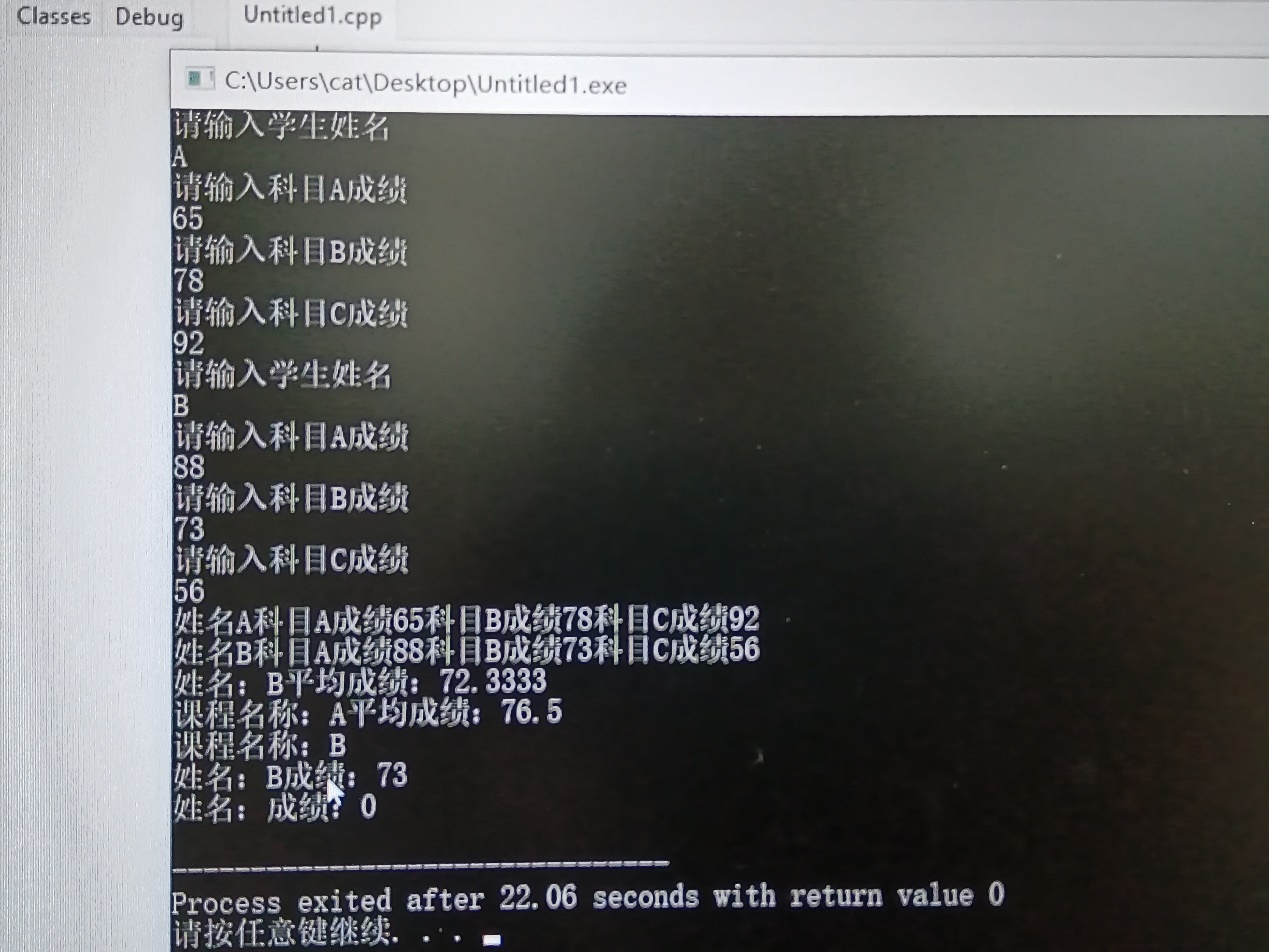
x.ShowNameAndScore();

x.ShowStudentAvgScore(1);

x.ShowClassAvgScore("A");

x.OrderScore("B");

return 0;

}

总结：理解了类和对象的概念，掌握了声明类和定义对象的方法；掌握了构造函数和析构函数的实现方法，初步掌握使用类和对象编写C++程序。