

**《C++语言程序设计》实验报告**

**实验三：面向过程与面向对象**

**姓 名：李剑寒**

**学 号：Y01914541**

**座 位 号： 92**

**专业班级： 网络空间安全**

**实验日期：2021/5/6**

**安徽大学互联网学院**

**实验内容**

**一、实验要求：**

1.一元二次型方程的求解是数学中的常见问题，其中一种求解方法是“公式法”。在此次实验中，分别利用面向过程与面向对象的思路去求解一元二次型方程的实数根。具体要求如下：公式法求解一元二次型方程主要分为三步，

(1)判断是否为一元二次方程；(2)判断是否有实数根；(3)公式法求解实根。

面向过程：编写3个函数，求解一元二次型方程的实根，若不满足一元二次方程的条件或无法求出实根，则输出原因；

面向对象：编写一个一元二次型方程的类，并将上述三个功能作为成员函数，求解一元二次型方程的实根，若不满足一元二次方程的条件或无法求出实根，则输出原因。

**二、设计思路及程序代码**

面向过程：

#include<iostream>

#include<cmath>

using namespace std;

int isfun(float a)

{

if (a == 0)

{

cout << "这不是一个一元二次函数！" << endl;

return 0;

}

else

{

cout << "这是一个一元二次函数" << endl;

return 1;

}

}

int haveroot(float a,float b, float c)

{

if (b\*b - 4 \* a\*c >= 0)

{

cout << "方程有实根" << endl;

return 1;

}

else

{

cout << "方程无实根!" << endl;

return 0;

}

}

void solve(float a, float b, float c)

{

cout <<"x1= "<< (-b + sqrt(b\*b - 4 \* a\*c)) / (2 \* a)<<endl;

cout <<"x2= "<< (-b - sqrt(b\*b - 4 \* a\*c)) / (2 \* a)<<endl;

}

int main()

{

float a, b, c;

cout << "请输入一个一元二次函数: " << endl;

cout << "ax^2+bx+c=0" << endl;

cout << "a= ";

cin >> a;

cout << "b= ";

cin >> b;

cout << "c= ";

cin >> c;

if (isfun(a))

{

haveroot(a, b, c);

if (haveroot(a, b, c))

solve(a, b, c);

}

system("pause");

return 0;

}

面向对象：

#include<iostream>

#include<cmath>

using namespace std;

class fun

{

public:

fun(float a, float b, float c);

fun();

int isfun(float a, float b, float c);

int haveroot(float a, float b, float c);

void solve(float a, float b, float c);

private:

float a, b, c;

};

fun::fun(float a, float b, float c) :a(a), b(b), c(c){};

int fun::isfun(float a, float b, float c)

{

if (a == 0)

{

cout << "这不是一个一元二次函数！" << endl;

return 0;

}

else

{

cout << "这是一个一元二次函数" << endl;

return 1;

}

}

int fun::haveroot(float a, float b, float c)

{

if (b\*b - 4 \* a\*c >= 0)

{

cout << "方程有实根" << endl;

return 1;

}

else

{

cout << "方程无实根!" << endl;

return 0;

}

}

void fun::solve(float a, float b, float c)

{

cout << "x1= " << (-b + sqrt(b\*b - 4 \* a\*c)) / (2 \* a) << endl;

cout << "x2= " << (-b - sqrt(b\*b - 4 \* a\*c)) / (2 \* a) << endl;

}

int main()

{

float a, b, c;

cout << "请输入一个一元二次函数: " << endl;

cout << "ax^2+bx+c=0" << endl;

cout << "a= ";

cin >> a;

cout << "b= ";

cin >> b;

cout << "c= ";

cin >> c;

fun x(a,b,c);

if (x.isfun(a,b,c))

{

x.haveroot(a, b, c);

if (x.haveroot(a, b, c))

x.solve(a, b, c);

}

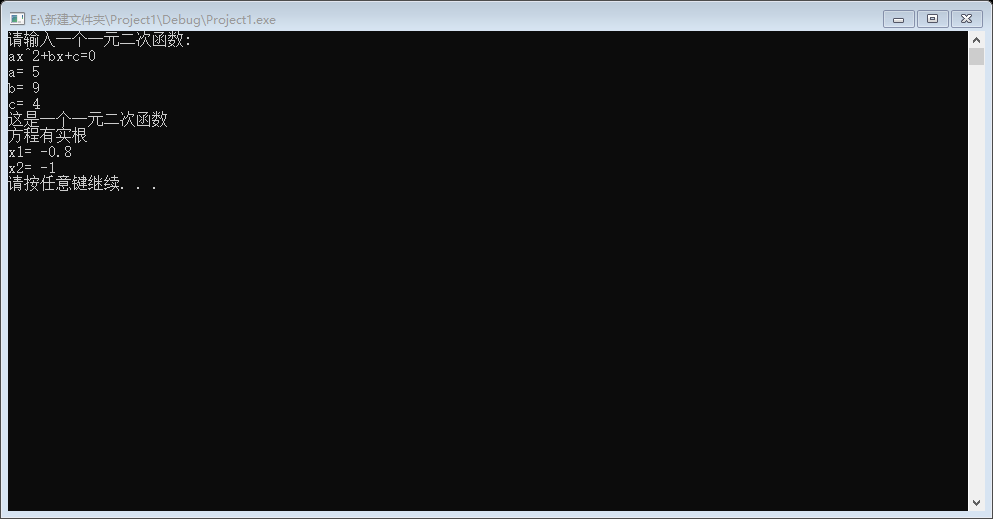
system("pause");

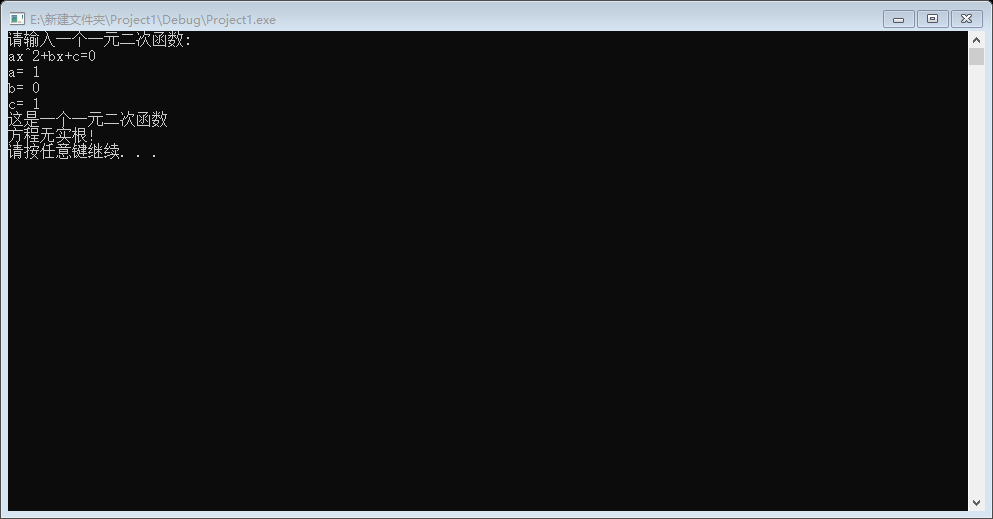
return 0;

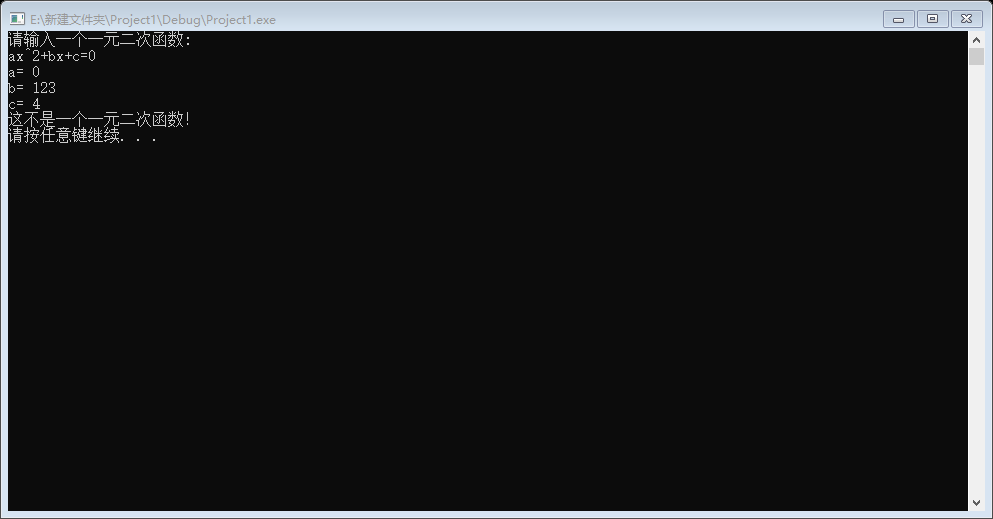
}

**三、运行结果及分析**

**面向过程：**







面向对象：

