**实验五 函数程序设计**

**一、实验目的**

（1）掌握C语言函数定义及调用的规则。

（2）理解参数传递的过程。

（3）掌握函数返回值的大小和类型确定的方法。

（4）理解变量的作用范围。

**二、实验内容**

（1）上机调试下面的程序，修改系统给出的出错信息，并指出出错原因。

#include<stdio.h>

int main()

{

int,y;

printf("%d\n",sum(x+y));

int sum(a,b)

{

int a,b;

return(a+b);

}

return 0;

}

（2）编写一个程序，输入系数*a*，*b*，*c*，求一元二次方程*ax*2*+bx+*c=0的根，包括主函数和如下函数。

① 判断*a*是否为零。

② 计算判别式。

③ 计算根的情况。

④ 输出根。

（3）输入下面程序，分析运行结果。

#include<stdio.h>

int func (int ,int);

int main ( )

{ int k=4,m=1,p1,p2;

p1=func(k,m) ;

p2=func(k,m) ;

printf("%d,%d\n",p1,p2) ;

return 0;

}

int func (int a,int b)

{ static int m=0,i=2;

i+=m+1;

m=i+a+b;

return (m);

}

**三、实验步骤和实验结果**。

分别贴上以上三个程序代码，并给出程序运行结果的截图

（1）程序中错误：函数为定义在主函数外、主函数中对函数的调用不正确、函数中重复声明变量、没有输入函数

修改后的程序如下：

#include<stdio.h>

int sum(int a,int b)

{

return(a+b);

}

int main()

{

int x, y;

scanf("%d %d", &x, &y);

printf("%d\n", sum(x, y));

return 0;

}

运行截图：

形状

低可信度描述已自动生成

（2）代码：

//编写一个程序，输入系数a，b，c，求一元二次方程ax2+bx+c=0的根，包括主函数和如下函数。

//① 判断a是否为零。

//② 计算判别式 。

//③ 计算根的情况。

//④ 输出根。

//编写一个程序，输入系数a，b，c，求一元二次方程ax2+bx+c=0的根，包括主函数和如下函数。

//① 判断a是否为零。

//② 计算判别式 。

//③ 计算根的情况。

//④ 输出根。

#include <iostream>

#include <cmath>

using namespace std;

bool isZero(double a){

if (a == 0.0){

return true;

}

return false;

}

double judge(double a, double b, double c){

return pow(b, 2) - 4 \* a \* c;

}

int main()

{

double a, b, c;

cin >> a >> b >> c;

if (isZero(a)){cout << "不是一元二次方程！"; return 0;}

double j = judge(a, b, c);

double x1, x2;

if (j < 0){

cout << "不存在实根" << endl;

} else{

if (j == 0){

x1 = x2 = -b / 2 / a;

cout << "x1 = x2 = " << x1;

return 0;

} else{

x1 = (-b + sqrt(j)) / 2 / a;

x2 = (-b - sqrt(j)) / 2 / a;

cout << "x1 = " << x1 << " x2 = " << x2;

return 0;

}

}

}运行截图：

电脑萤幕

中度可信度描述已自动生成 文本

中度可信度描述已自动生成

文本

描述已自动生成

（3）运行结果



分析：由于函数func（）中声明i，m这两个变量，使用了static，只有在第一次调用该函数会给i，m赋初值，之后再次调用会保留之前的值，对该程序进行调试可以看到该过程：

第一次调用：

图形用户界面, 应用程序

描述已自动生成

第二次调用：

图形用户界面, 应用程序

描述已自动生成