1．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的确定方法。

（4）理解变量的作用范围。

2．实验内容

（1）上机调试下面的程序，记录系统给出的出错信息，并指出出错原因。

#include<stdio.h>

int main( )

{

int x,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，输入系统a、b、c，求一元二次方程ax2+bx+c=0的根，包括主函数和如下子函数。

①判断a是否为零。

②计算判别式b2-4ac。

③判断根的情况。

④输出根。

（3）输入下面的程序，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;

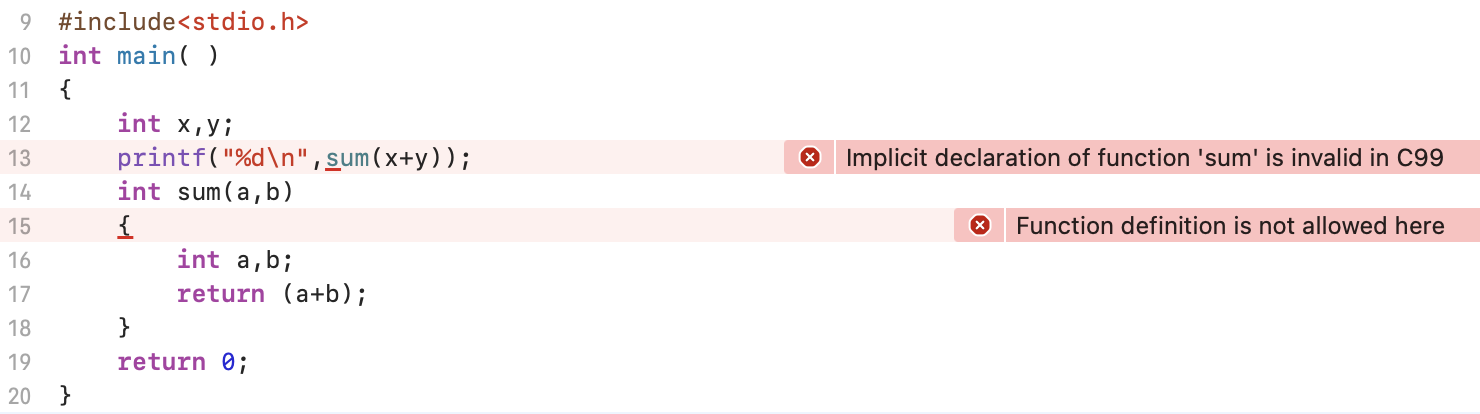
return (m);

}

1. 问代码输入进Xcode后显示运行失败



显示有如下图所示错误：



总的来说，未定义函数“sum”，函数定义不能出现在main()函数内部。除此以外，x和y的值未明确。

修改后程序如下：

|  |
| --- |
| #include<stdio.h>  int sum(int,int);  int main( )  {  int x,y;  scanf("%d %d",&x,&y);  printf("%d\n",sum(x,y));    return 0;  }  int sum(int a,int b)  {  return (a+b);  } |

并测试输入3和5的结果，如下：

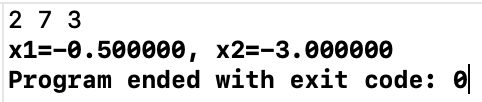


（2）问代码如下：

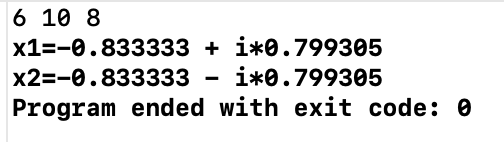
|  |
| --- |
| #include<stdio.h>  #include <math.h>  int iszero(double);  double delta(double,double,double);  int root(double);  void output\_root(int,double,double,double);  int main( )  {  double a,b,c;  scanf("%lf %lf %lf",&a,&b,&c);  if(iszero(a)==0)  printf("不是一元二次方程，请重新输入！\n");  else  output\_root(root(delta(a, b, c)), a, b, delta(a, b, c));  return 0;  }  //判断二次项系数是否为0  int iszero(double a)  {  if(a==0)  return 0;  else  return 1;  }  //计算判别式  double delta(double a,double b,double c)  {  return b\*b-4\*a\*c;  }  //根据判别式判断根的情况  int root(double r)  {  if (r==0)  return 0;  else if (r>0)  return 1;  else  return 2;  }  //输出根  void output\_root(int n,double a,double b,double r)  {  double x1,x2,real,image;  if(n==0)  {  x1=-b/(2\*a);  printf("x1=x2=%lf\n",x1);  }  if (n==1)  {  x1=(-b+sqrt(r))/(2\*a);  x2=(-b-sqrt(r))/(2\*a);  printf("x1=%lf, x2=%lf\n",x1,x2);  }  if (n==2)  {  real=-b/(2\*a);  image=sqrt(-r)/(2\*a);  printf("x1=%lf + i\*%lf\n",real,image);  printf("x2=%lf - i\*%lf\n",real,image);  }  } |

在Xcode中输入上面代码，测试四组数据结果如下：

1. （2，7，3）



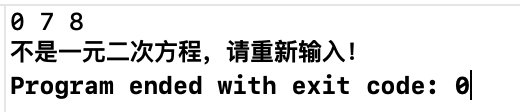
1. （6，10，8）



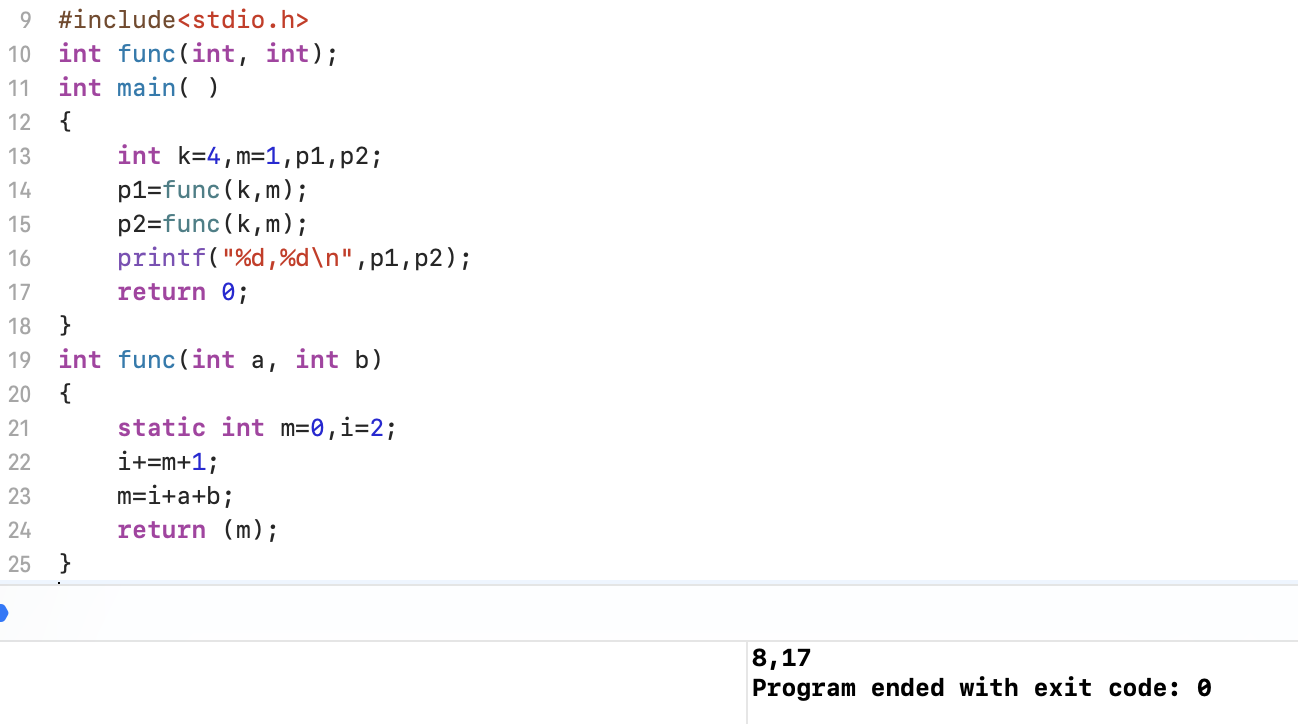
1. （2，4，2）



1. （0，7，8）



（3）输入程序后结果为



在这里，static函数用于修饰局部变量，即在一个函数内部，只能在该函数内被调用。

这样定义的变量通常被称为**局部静态变量**，它的值不会因为函数调用的结束而被清除，当函数再次被调用时，它的值是上一次调用结束后的值。

4．分析与讨论

1. 针对以上实验内容写出相应的参数传递过程并分析结果。
2. 函数在定义时要注意什么？
3. 讨论静态局部变量的传递性。