



石家莊鐵道大學
SHIJIAZHUANG TIEDAO UNIVERSITY

在线开放课程

C语言程序设计—函数

习题解答

主讲：石玉晶

◆ 1、函数定义正确的是_____：

A. double fun(int x , int y)
{
 z = x + y ;
 return z ;
}

B. fun(int x , y)
{
 int z ;
 return z ;
}

C. fun(x , y)
{
 int x , y ;
 double z ;
 z = x + y ;
 return z ;
}

D. double fun(int x , int y)
{
 double z ;
 z = x + y ;
 return z ;
}

答案： D

- ◆ 2、以下说法**不正确**的是_____。
- A. 实参可以是常量、变量或表达式
- B. 形参可以是常量、变量或表达式
- C. 形参可以为任意类型
- D. 形参应与其对应的实参类型一致

答案： B

习题

- ◆ 3、一个函数形参的作用域是_____。
- A. main函数
 - B. 形参所在函数体
 - C. 从定义处到文件尾
 - D. 整个程序

答案： B

习题

- ◆ 4、C语言规定，简单变量做实参时，它和对应形参之间的数据传递方式是_____。
- A. 地址传递
- B. 单向值传递
- C. 由实参传递给形参，再由形参传回给实参
- D. 由用户指定传递方式

答案： B

- ◆ 5、若用一维数组名作为函数调用的实参，传递给形参的是_____。
- A. 数组的地址
- B. 数组第一个元素的值
- C. 数组中全部元素的值
- D. 数组元素的个数

答案：A

- ◆ 6、下面程序的运行结果是_____。

```
#include<stdio.h>
int main( )
{
    void increment( );
    increment( );
    increment( );
    increment( );
    return 0 ;
}
```

```
void increment( )
{
    int x = 0 ;
    x += 1 ;
    printf( "%d\t" , x ) ;
}
```

答案： 1 1 1

- ◆ 7、下面程序的运行结果是_____。

```
#include<stdio.h>
int main( )
{
    int fun( int n );
    printf( "%d\n" , fun( 3 ) );
    return 0 ;
}
```

```
int fun( int n )
{
    if( n ) return fun( n - 1 ) + n ;
    else return 0 ;
}
```

答案： 6

习题

- ◆ 8、下面程序的运行结果是_____。

```
#include<stdio.h>
int a[3];
int main( ){
    void f1( );
    void f2( );
    void print( );

    f1( );
    print( );
    f2( );
    print( );
return 0;}
```

```
void f1( ){
    int i;
    for ( i = 0 ; i < 3 ; i++ ) a[i] = i;
}
void f2( ){
    int i ,a[3];
    for ( i = 0 ; i < 3 ; i++ ) a[i] = i*2;
}
void print( ){
    int i;
    for ( i = 0 ; i < 3 ; i++ )printf( "%3d" , a[i] );
}
```

答案： 0 1 2 0 1 2

习题

- ◆ 9、编写一个判断素数的函数，返回0表示不是素数，返回1表示是素数。在主函数输入1个正整数，输出是否为素数的信息。
- ◆ 数学原理： 只能被1和本身整除的数为素数

习题

```
int f(int a)
{
    int i;
    for(i=2;i<a;i++)
        if(a%i==0) return 0;
    return 1;
}
```

```
int main()
{
    int x;
    printf("请输入一个大于1的正整数: ");
    scanf("%d",&x);
    if(f(x)) printf("%d是素数\n",x);
    else printf("%d不是素数\n",x);
}
```

谢谢！