



第7章 函数  
任务1 简单函数的定义及应用

主讲人 |

知识  
目标



1. 简单函数的语法格式。
2. 简单函数的定义与引用。

技能  
目标



1. 会定义与调用函数。
2. 会使用函数进行模块化算法设计与实现。
3. 会调试和运行程序。

素质  
目标



1. 学习过程中形成乐于探究的意识和敢于创新的精神。
2. 养成主动与他人交流合作的精神。

# 提纲

1.函数的概述

2.函数的定义

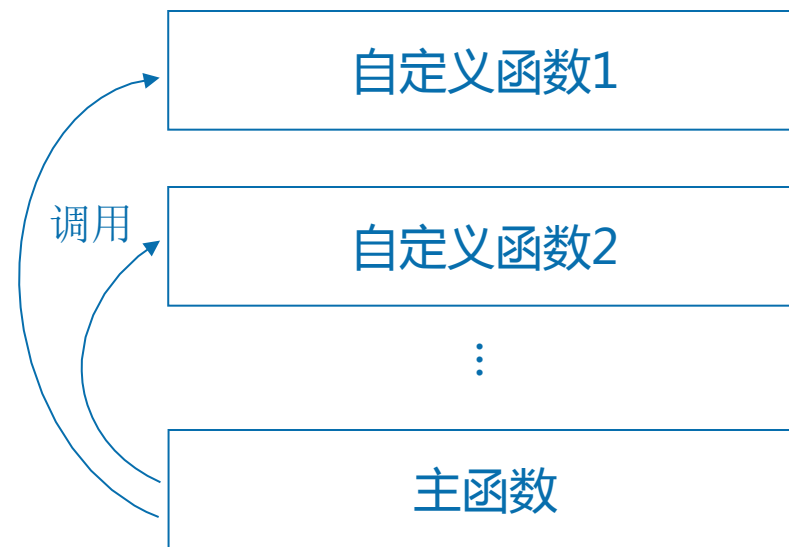
3.函数的调用



### 注意说明:

- (1) C源程序是由函数组成的且只有一个主函数main()。
- (2) 主函数可以调用其它函数，而不允许被其它函数调用。
- (3) 程序由主函数开始执行，并且由主函数结束。
- (4) 主函数可以放在程序的任意位置。

### 源程序





## 库函数和自定义函数：

### (1) 库函数

C系统提供，用户无须定义，只需在程序前包含有该函数原型的头文件即可在程序中直接调用。

例如：

printf、scanf、getchar、putchar、gets、puts等函数，都需要加上头文件#include "stdio.h"。



求  $\sqrt{x}$  的值。

头文件math.h 中包含了sqrt函数，其功能就是开方。

### 语句编写

```
#include "stdio.h"
#include "math.h"
main()
{
    int x;
    double y;
    scanf("%d",&x);
    y=sqrt(x);
    printf("%lf\n",y);
}
```

### 运行结果

```
5 ✓
2.236068
```



库函数和自定义函数：

## (2) 自定义函数

由用户按实际需要自己编写的函数，需要被调用才能实现其功能。



求两个整数之和。

### 语句编写

```
int sum(int x,int y) //自定义函数
{
    return x+y;
}
main() //主函数
{
    int s;
    s=sum(2,3);
    printf("s=%d\n",s);
}
```

### 运行结果

s=5



一般形式:

```
void 函数名()  
{  
    函数体;  
}
```



注意说明:

- (1) 其中void和函数名称为函数头。函数名后有一个空括号，其中无参数，但括号不可少。
- (2) {}中的内容称为函数体。
- (3) void代表函数无类型，即无返回值，不需要return。

语句编写

```
void Hello( )      //自定义函数  
{  
    printf("Hello world!\n");  
}  
main()            //主函数  
{  
    Hello();  
}
```

运行结果

Hello world!



一般形式:

```
void 函数名(形式参数表列)
{
    函数体;
}
```



注意说明:

- (1) 有参函数比无参函数多了一个内容, 即形式参数表列。
- (2) 形式参数可以是各种类型的变量, 参数之间用逗号间隔。
- (3) 进行函数调用时, 主调函数将传递给形式参数实际的值。

语句编写

```
void sum(int a,int b) //自定义函数
{
    int s;
    s=a+b;
    printf("s=%d\n",s);
}
main() //主函数
{
    int x=2,y=3;
    sum(x,y);
}
```

运行结果

s=5





一般形式:

```
类型标识符 函数名()  
{  
    函数体 ;  
    return 表达式 ;  
}
```



注意说明:

- (1) 函数的类型实际上是函数返回值的类型。
- (2) return语句的作用是把值作为函数的值返回给主调函数。
- (3) 有返回值的函数中至少应有一个return语句。

### 语句编写

```
int sum()                //自定义函数  
{  
    int a,b,s;  
    scanf("%d%d",&a,&b);  
    s=a+b;  
    return s;  
}  
main()                  //主函数  
{  
    int s;  
    s=sum();  
    printf("s=%d\n",s);  
}
```

### 运行结果

```
2 3 ✓  
s=5
```



## 一般形式:

```
类型标识符 函数名(形式参数表列)
{
    函数体 ;
    return 表达式 ;
}
```



## 注意说明:

- (1) 在C程序中,一个自定义函数的定义可以放在任意位置。
- (2) 如果放在main之后,在调用之前,需要对它进行说明。  
例如: `int sum(int a,int b);`

## 语句编写

```
int sum(int a,int b ) //自定义函数
{
    int s;
    s=a+b;
    return s;
}
main() //主函数
{
    int x,y,s;
    scanf("%d%d",&x,&y);
    s=sum(x,y);
    printf("s=%d\n",s);
}
```

## 运行结果

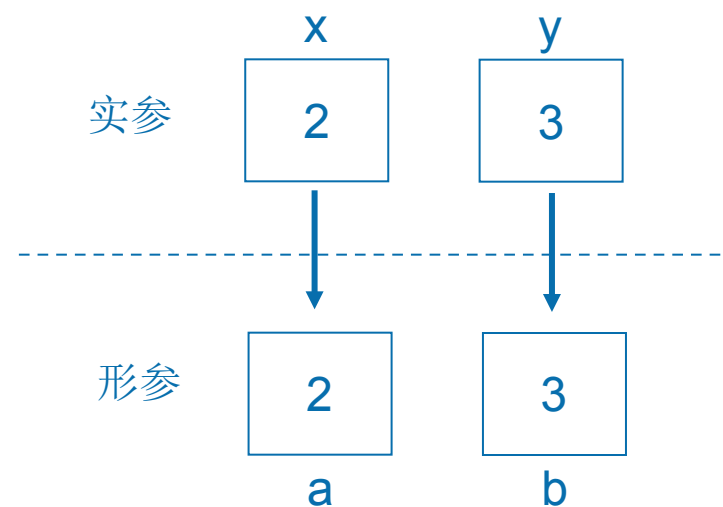
```
2 3 ✓
s=5
```



## 注意说明:

- (1) 形参变量只有在被调用时才分配内存单元，在调用结束时，即刻释放所分配的内存单元。
- (2) 实参可以是常量、变量、表达式等，在进行函数调用时，把具有确定的值传送给形参。
- (3) 函数调用中发生的数据传送是单向的。
- (4) 实参和形参在数量、类型和顺序上应严格一致。

## 内存





return语句一般形式:

```
return 表达式;  
或者为:  
return (表达式);
```



注意说明:

- (1) 在函数中允许有多个return语句，但每次调用只能有一个return语句被执行。
- (2) 函数返回值的类型和函数定义中函数的类型应保持一致。
- (3) 不返回函数值的函数，可以明确定义为“空类型”，类型说明符为“void”。

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：  
<https://d.book118.com/975232114342011222>