



北京大学
PEKING UNIVERSITY

信息科学技术学院

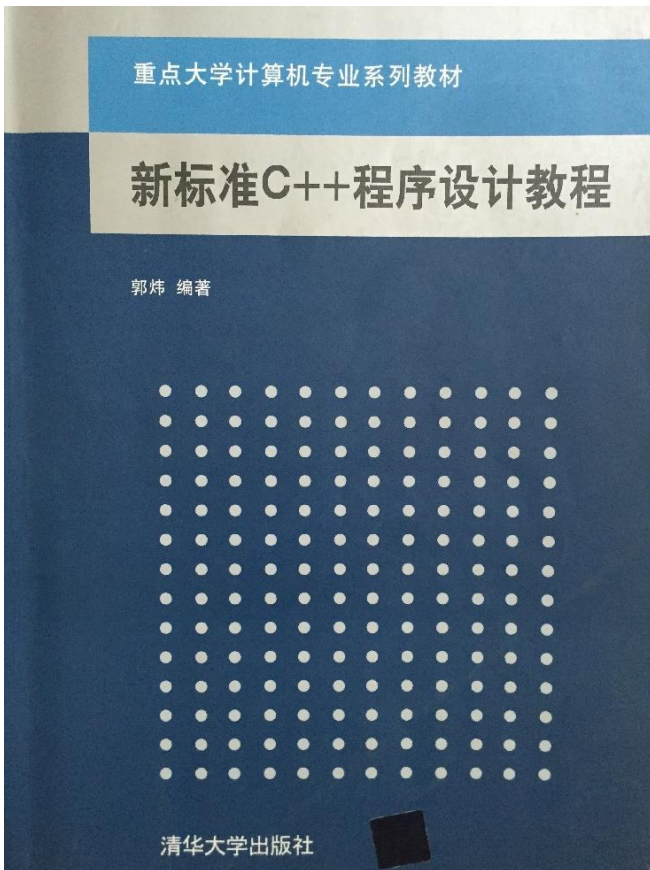
指定教材:

程序设计与算法(一)

李文新 郭炜

主讲教师互动微博:

<http://weibo.com/guoweiofpku>





北京大学
PEKING UNIVERSITY

信息科学技术学院《程序设计与算法》

函数

为什么需要函数

- 写了一段牛顿迭代法求平方根的代码，程序里面无数地方都要求平方根，难道需要的地方都把这段代码拷贝一遍？

为什么需要函数

- 写了一段牛顿迭代法求平方根的代码，程序里面无数地方都要求平方根，难道需要的地方都把这段代码拷贝一遍？
- 一个数十万行的程序，都写在 `main` 里面？数百个程序员如何合写一个 `main`？

为什么需要函数

- “函数” 可以将实现了某一功能，并需要反复使用的代码包装起来形成一个功能模块（即写成一个“函数”），那么当程序中需要使用该项功能时，只需写一条语句，调用实现该功能的“函数”即可。

为什么需要函数

- “函数” 可以将实现了某一功能，并需要反复使用的代码包装起来形成一个功能模块（即写成一个“函数”），那么当程序中需要使用该项功能时，**只需写一条语句，调用实现该功能的“函数”即可。**
- 不同的程序员可以分别写不同的函数，拼起来形成一个大程序

函数的定义

```
返回值类型 函数名(参数1类型 参数1名称, 参数2类型 参数2名称……)
{
    语句组(即“函数体”)
}
```

- 如果函数不需要返回值，则“返回值类型”可以写“**void**”

函数调用和return语句

- 调用函数：

函数名（参数1, 参数2, ……）

函数调用和return语句

- 调用函数：

函数名（参数1, 参数2, ……）

- 对函数的调用，也是一个表达式。函数调用表达式的值，由函数内部的return语句决定。return语句语法如下：

return 返回值；

函数调用和return语句

- 调用函数：

函数名（参数1, 参数2, ……）

- 对函数的调用，也是一个表达式。函数调用表达式的值，由函数内部的return语句决定。return语句语法如下：

return 返回值；

- return语句的功能是结束函数的执行，并将“返回值”作为结果返回。“返回值”是常量、变量或复杂的表达式均可。如果函数返回值类型为“void”，return语句就直接写：

return ；

函数调用和return语句

- `return` 语句作为函数的出口，可以在函数中多次出现。多个`return`语句的“返回值”可以不同。在哪个`return`语句结束函数的执行，函数的返回值就和哪个`return`语句里面的“返回值”相等。

函数使用实例1：Max函数

```
#include <iostream>
using namespace std;
int Max(int x,int y) //求两个整型变量中的较大值
```

```
{
    if( x > y )
        return x;
    return y;
}
```

形参

=

形参实参类型需兼容！

```
int main()
{
    int n = Max(4,6);
    cout << n << ", " << Max(20,n) << endl;
    return 0;
} => 6,20
```

实参

函数使用实例2：判断是否是素数的函数

```
#include <iostream>
using namespace std;
bool IsPrime(unsigned int n)
{
    if( n <= 1 )
        return false;
    for( int i = 2; i < n; ++i )    //看看有没有n的因子
        if( n % i == 0 )
            return false;
    return true;
}

int main() {
    cout << IsPrime(2) << ", " << IsPrime(4) << ", " << IsPrime(5);
    return 0;
} => 1,0,1
```

返回值为void的函数

```
void DrawCircle(double x,double y,double r)
{
    //下面的代码在屏幕上以(x,y)点为圆心，r为半径画圆
    .....
    return;
}
```

调用:

```
DrawCircle(0,0,1);
```

函数使用实例3：已知三角形三个顶点位置, 求边长

给定平面上不共线的三个点，其坐标都是整数，编写程序，求它们构成的三角形的三条边的长度。输入是6个整数： $x_1, y_1, x_2, y_2, x_3, y_3$ 代表三个点的坐标，以任意顺序输出三条边的长度均可。

```
#include <iostream>
using namespace std;
#define EPS 0.001 //用以控制计算精度
double Sqrt(double a)
{
    //求a的平方根
    double x = a/2, lastX = x + 1 + EPS; //确保能够进行至少一次迭代
    while( x - lastX > EPS || lastX - x > EPS) {
        //只要精度没有达到要求，就继续迭代
        lastX = x;
        x = (x + a/x)/2;
    }
    return x;
}

double Distance(double x1, double y1, double x2, double y2)
{
    //求两点(x1,y1), (x2,y2) 的距离
    return Sqrt((x1-x2)*(x1-x2)+(y1-y2)*(y1-y2));
}
```



```
int main()
{
    int x1,y1,x2,y2,x3,y3;
    cin >> x1 >> y1 >> x2 >> y2 >> x3 >> y3;
    cout << Distance(x1,y1,x2,y2) << endl;
    //输出(x1,y1)到(x2,y2)距离
    cout << Distance(x1,y1,x3,y3) << endl;
    cout << Distance(x3,y3,x2,y2) << endl;
    return 0;
}
```

返回值为void的函数

```
void DrawCircle(double x,double y,double r)
{
    //下面的代码在屏幕上以(x,y)点为圆心，r为半径画圆
    .....
    return;
}
```

=

调用:

```
DrawCircle(0,0,1);
```

函数的声明

- 一般来说函数的**定义**必须出现在函数调用语句之前，
否则调用语句编译出错

如果过函数A内部调用了B, B内部调用了A, 哪个写前面？

函数的声明

- 函数的调用语句前面有函数的**声明**即可，不一定要有定义！

返回值类型 函数名(参数1类型 参数1名称, 参数2类型 参数2名称.....);

例如:

```
int Max(int a,int b);  
double Sqrt(double);  
double Distance(double,double,double,double);
```

参数名称可以省略。函数声明也称为“函数的原型”

函数的声明

```
void FunctionB(); //声明
void FunctionA() {
    .....
    FunctionB();
    .....
    return;
}
void FunctionB() {
    .....
    FunctionA();
    .....
    return;
}
```

main函数

C/C++程序从main函数开始执行，执行到main中的return则结束。

```
#include <iostream>
using namespace std;
int main()
{
    cout << "Hello,world" << endl;
    return 0;
}
```

函数参数的传递

- 函数的形参是实参的一个拷贝，且形参的改变不会影响到实参
(除非形参类型是数组、引用或对象 —— 引用或对象的内容本课不涉及)

```
#include <iostream>
using namespace std;
void Swap(int a,int b)
{
    int tmp;
    //以下三行将a,b值互换
    tmp = a ;
    a = b;
    b = tmp;
    cout << "In Swap: a=" << a << " b=" << b << endl;
}
```

函数参数的传递

```
int main()
{
    int a = 4, b = 5;
    Swap(a,b);
    cout << "After swaping: a=" << a << " b=" << b;
    return 0;
}
```

In Swap: a=5 b=4

After swaping: a=4 b=5

一维数组作为函数的参数

- 一维数组作为形参时的写法如下：

类型名 数组名[]

不用写出数组的元素个数。例如：

```
void PrintArray( int a[ ]) { }
```

一维数组作为函数的参数

- 一维数组作为形参时的写法如下：

类型名 数组名[]

不用写出数组的元素个数。例如：

```
void PrintArray( int a[ ]) { }
```

- 数组作为函数参数时，是传引用的，即形参数组改变了，实参数组也会改变。

一维数组作为函数的参数

●编写一个求整型数组最大值的函数

```
#include <iostream>
using namespace std;
int a1[4] = {4,15,6,9};
int a2[] = {3,18,56,40,78};
int FindMax( int a[] ,int length)  { //length是数组长度
    int mx = a[0];
    for(int i = 1;i < length; ++i)
        if( mx < a[i])
            mx = a[i];
    return mx;
}
int main() {
    cout << FindMax(a1,sizeof(a1)/sizeof(int)) << endl;
    cout << FindMax(a2,sizeof(a2)/sizeof(int)) << endl;
    return 0;
}
```

一维数组作为函数的参数

- 编写一个把int数组所有元素置0的函数

```
#include <iostream>
using namespace std;
int a1[4] = {4,15,6,9};
void SetToZero(int a[] ,int length)
{
    for(int i = 0;i < length; ++i)
        a[i] = 0;
}
int main()
{
    SetToZero(a1,4);
    for(int i = 0;i < 4; ++i)
        cout << a1[i] << "," ;
    return 0;
} => 0,0,0,0
```

二维数组作为函数的参数

- 二维数组作为形参时，必须写明数组有多少列，不用写明有多少行：

```
void PrintArray( int a[][5])  
{  
    cout << a[4][3];  
}
```

- 必须要写明列数，编译器才能根据下标算出元素的地址。

$a[i][j]$ 的地址：

数组首地址 $+i \times N \times \text{sizeof}(a[0][0]) + j \times \text{sizeof}(a[0][0])$ (N是数组列数)

形参数组的首地址就是实参数组的首地址

库函数和头文件

难道真的求平方根都要自己写一个？

库函数和头文件

难道真的求平方根都要自己写一个? No!!

```
#include <iostream>
```

```
#include <cmath> // 头文件<cmath>中包含许多数学库函数
```

```
using namespace std;
```

```
int main()
```

```
{
```

```
    double a;
```

```
    cin >> a;
```

```
    if( a < 0 ) {
```

```
        cout << "Illegal input" << endl;
```

```
        return 0;
```

```
    }
```

```
    cout << sqrt(a); //调用标准库函数求平方根
```

```
    return 0;
```

```
}
```

库函数和头文件

库函数：C/C++标准规定的，编译器自带的函数

头文件：C++编译器提供许多“头文件”，如

`iostream`

`cmath`

`string`

头文件内部包含许多库函数的声明以及其他信息，如`cin`,`cout`的定义

```
#include <iostream>
```

即可将头文件包含到程序中，此后即可使用头文件中定义的库函数及其他信息

库函数和头文件示例

●数学函数

数学库函数声明在`cmath`中，主要有：

<code>int abs(int x)</code>	求整型数 x 的绝对值
<code>double cos(double x)</code>	求 x (弧度) 的余弦
<code>double fabs(double x)</code>	求浮点数 x 的绝对值
<code>int ceil(double x)</code>	求不小于 x 的最小整数
<code>double sin(double x)</code>	求 x (弧度) 的正弦
<code>double sqrt(double x)</code>	求 x 的平方根
.....	

库函数和头文件示例

●字符处理函数

这些库函数在ctype中声明，主要有：

int isdigit(int c)	判断 c 是否是数字字符
int isalpha(int c)	判断 c 是否是一个字母
int isalnum(int c)	判断 c 是否是一个数字或字母
int islower(int c)	判断 c 是否是一个小写字母
int islower(int c)	判断 c 是否是一个小写字母
int isupper(int c)	判断 c 是否是一个大写字母
int toupper(int c)	如果 c 是一个小写字母，则返回对应大写字母
int tolower (int c)	如果 c 是一个大写字母，则返回对应小写字母