

- C++ 教程
- C++ 简介
- C++ 环境设置
- C++ 基本语法
- C++ 注释
- C++ 数据类型
- C++ 变量类型
- C++ 变量作用域
- C++ 常量
- C++ 修饰符类型
- C++ 存储类
- C++ 运算符
- C++ 循环
- C++ 判断
- C++ 函数
- C++ 数字
- C++ 数组
- C++ 字符串
- C++ 指针
- C++ 引用
- C++ 日期 & 时间
- C++ 基本的输入输出
- C++ 数据结构

- C++ 面向对象
- C++ 类 & 对象
- C++ 继承

## C++ 数字

通常，当我们需要用到数字时，我们会使用原始的数据类型，如 int、short、long、float 和 double 等等。这些用于数字的数据类型，其可能的值和数值范围，我们已经在 C++ 数据类型一章中讨论过。

## C++ 定义数字

我们已经在之前章节的各种实例中定义过数字。下面是一个 C++ 中定义各种类型数字的综合实例：

实例

```
#include <iostream>
using namespace std;

int main ()
{
    // 数字定义
    short s;
    int i;
    long l;
    float f;
    double d;

    // 数字赋值
    s = 10;
    i = 1000;
    l = 1000000;
    f = 230.47;
    d = 30949.374;

    // 数字输出
    cout << "short s : " << s << endl;
    cout << "int i : " << i << endl;
    cout << "long l : " << l << endl;
    cout << "float f : " << f << endl;
    cout << "double d : " << d << endl;

    return 0;
}
```

当上面的代码被编译和执行时，它会产生下列结果：

```
short s :10
int i :1000
long l :1000000
```

- HTML / CSS
- JavaScript
- 服务端
- 数据库
- 数据分析
- 移动端
- XML 教程
- ASP.NET
- Web Service
- 开发工具
- 网站建设



C++ 重载运算符  
和重载函数

C++ 多态

C++ 数据抽象

C++ 数据封装

C++ 接口（抽象  
类）

## C++ 高级教程

C++ 文件和流

C++ 异常处理

C++ 动态内存

C++ 命名空间

C++ 模板

C++ 预处理器

C++ 信号处理

C++ 多线程

C++ Web 编程

## C++ 资源库

C++ STL 教程

C++ 标准库

C++ 有用的资源

C++ 实例

```
float f : 230.47
double d : 30949.4
```

## C++ 数学运算

在 C++ 中，除了可以创建各种函数，还包含了各种有用的函数供您使用。这些函数写在标准 C 和 C++ 库中，叫做**内置函数**。您可以在程序中引用这些函数。

C++ 内置了丰富的数学函数，可对各种数字进行运算。下表列出了 C++ 中一些有用的内置的数学函数。

为了利用这些函数，您需要引用数学头文件 `<cmath>`。

序号	函数 & 描述
1	<b>double cos(double);</b> 该函数返回弧度角（double 型）的余弦。
2	<b>double sin(double);</b> 该函数返回弧度角（double 型）的正弦。
3	<b>double tan(double);</b> 该函数返回弧度角（double 型）的正切。
4	<b>double log(double);</b> 该函数返回参数的自然对数。
5	<b>double pow(double, double);</b> 假设第一个参数为 x，第二个参数为 y，则该函数返回 x 的 y 次方。
6	<b>double hypot(double, double);</b> 该函数返回两个参数的平方总和的平方根，也就是说，参数为一个直角三角形的两个直角边，函数会返回斜边的长度。
7	<b>double sqrt(double);</b> 该函数返回参数的平方根。
8	<b>int abs(int);</b> 该函数返回整数的绝对值。
9	<b>double fabs(double);</b> 该函数返回任意一个浮点数的绝对值。
10	<b>double floor(double);</b> 该函数返回一个小于或等于传入参数的最大整数。

下面是一个关于数学运算的简单实例：

### 实例

```
#include <iostream>
#include <cmath>
using namespace std;
```



```
int main ()
{
    // 数字定义
    short s = 10;
    int i = -1000;
    long l = 100000;
    float f = 230.47;
    double d = 200.374;

    // 数学运算
    cout << "sin(d) :" << sin(d) << endl;
    cout << "abs(i) :" << abs(i) << endl;
    cout << "floor(d) :" << floor(d) << endl;
    cout << "sqrt(f) :" << sqrt(f) << endl;
    cout << "pow( d, 2) :" << pow(d, 2) << endl;

    return 0;
}
```

当上面的代码被编译和执行时，它会产生下列结果：

```
sin(d) :-0.634939
abs(i) :1000
floor(d) :200
sqrt(f) :15.1812
pow( d, 2 ) :40149.7
```

## C++ 随机数

在许多情况下，需要生成随机数。关于随机数生成器，有两个相关的函数。一个是 **rand()**，该函数只返回一个伪随机数。生成随机数之前必须先调用 **srand()** 函数。

下面是一个关于生成随机数的简单实例。实例中使用了 **time()** 函数来获取系统时间的秒数，通过调用 **rand()** 函数来生成随机数：

### 实例

```
#include <iostream>
#include <ctime>
#include <cstdlib>

using namespace std;

int main ()
{
    int i,j;

    // 设置种子
    srand( (unsigned)time( NULL ) );

    /* 生成 10 个随机数 */
    for( i = 0; i < 10; i++ )
    {
        // 生成实际的随机数
        j= rand();
        cout <<"随机数:  " << j << endl;
    }
}
```



```
return 0;  
}
```

当上面的代码被编译和执行时，它会产生下列结果：

```
随机数: 1748144778  
随机数: 630873888  
随机数: 2134540646  
随机数: 219404170  
随机数: 902129458  
随机数: 920445370  
随机数: 1319072661  
随机数: 257938873  
随机数: 1256201101  
随机数: 580322989
```

← C++ 函数

C++ 数组 →



5 篇笔记

写笔记

#### 在线实例

- HTML 实例
- CSS 实例
- JavaScript 实例
- Ajax 实例
- jQuery 实例
- XML 实例
- Java 实例

#### 字符集&工具

- HTML 字符集设置
- HTML ASCII 字符集
- HTML ISO-8859-1
- HTML 实体符号
- HTML 拾色器
- JSON 格式化工具

#### 最新更新

- CSS clip-path 属性
- CSS @charset 规则
- CSS grid-row 属性
- CSS grid-templa...
- CSS grid-templa...
- CSS grid-templa...
- CSS grid-templa...

#### 站点信息

- 意见反馈
- 免责声明
- 关于我们
- 文章归档

#### 关注微信

