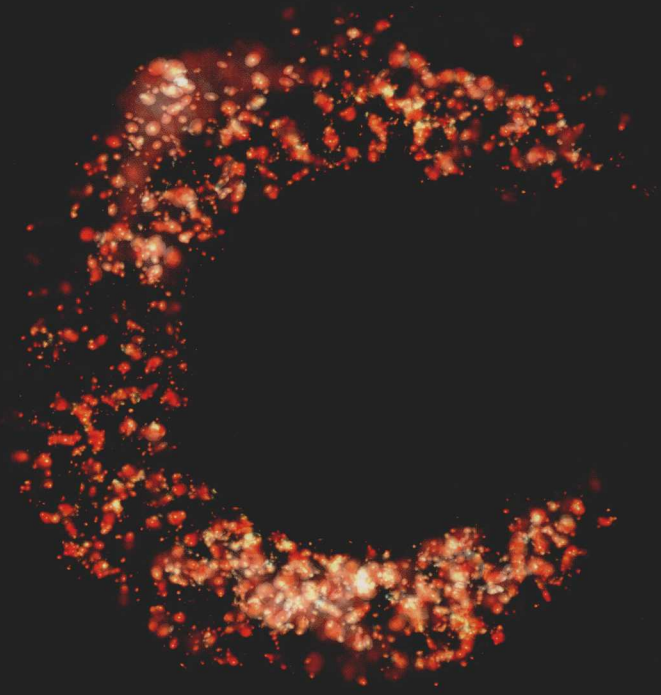
Broadview
www.broadview.com.cn



C++语法详解

本书全方位阐述了C++语法知识点,语法示例短小精悍，案头必备，方便速查
细致的图解分析，让读者更易理解语法原理，确保读者知其然更知其所以然

黄勇 / 编著

中国工信出版集团
電子工業出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
http://www.phei.com.cn

C++语法详解

黄勇 / 编著

電子工業出版社
Publishing House of Electronics Industry
北京•BEIJING

内 容 简 介

本书对C++的语法进行了全面介绍和深入讲解，内容包括：C++整型、字符型、浮点型、声明、定义、typedef、运算符、表达式、左值、选择语句、循环语句、指针、数组、函数和标识符的作用域、类基础、类作用域及相关运算符、构造函数、复制构造函数、析构函数、名称空间、类中的成员、运算符（操作符）重载、继承、虚函数、多态性、对象模型、虚函数表、模板、I/O、异常、预处理器、typeid、强制类型转换和string类等。本书层次分明，由浅入深，各章节相对独立，语法示例短小精悍，方便对有疑惑的语法进行速查。学习完本书，读者不会再对C++的各种语法感到困惑。

本书适合有一定C++基础、对C++的语法有疑惑、想深入了解C++语法细节的人员阅读。本书同时也可以作为解决C++语法问题的参考书；对于学习过C++或已精通C++的人员，也是一本不错的资料查阅手册。

图书在版编目（CIP）数据

C++语法详解 / 黄勇编著. —北京：电子工业出版社，2017.7
ISBN 978-7-121-31655-5

Ⅰ.①C… Ⅱ.①黄…Ⅲ. ①C 语言－程序设计 Ⅳ.①TP312.8

中国版本图书馆 CIP 数据核字(2017)第 119321 号

策划编辑：安　娜
责任编辑：葛　娜
印　　刷：三河市良远印务有限公司
装　　订：三河市良远印务有限公司
出版发行：电子工业出版社
　　北京市海淀区万寿路 173 信箱　　邮编：100036
开　　本：787×980　1/16　印张：31.25　字数：697 千字
版　　次：2017 年 7 月第 1 版
印　　次：2017 年 7 月第 1 次印刷
定　　价：89.00 元

凡所购买电子工业出版社图书有缺损问题，请向购买书店调换。若书店售缺，请与本社发行部联系，联系及邮购电话：（010）88254888，88258888。

质量投诉请发邮件至 zlts@phei.com.cn，盗版侵权举报请发邮件至 dbqq@phei.com.cn。

本书咨询联系方式：010-51260888-819，faq@phei.com.cn。

投稿联络：安 娜
微信&QQ：80303489
邮箱：anna@phei.com.cn

前　言

本书具有如下特点：

（1）本书是专门讲解C++语法规则的书籍，因此书中不会介绍任何有关程序设计的内容（比如编写一个计算规形面积的程序等）。本书将语法问题分离出来，避免既不像写语法的书，也不像写程序设计的书。

（2）书中的示例程序都使用简短的名字，比如a, b, A等，以便于记忆，而不会使用很长的名字。

（3）一个知识点能用一段话讲解清楚的，尽量不使用两段话。一个知识点一个标号，方便查阅和增补。

（4）一个知识点列举一个单独的简短易懂的程序作为示例。大多数教材都喜欢在第1章开头定义一个变量，然后一直到章尾都在使用那个变量作为示例。本书打破传统，一个知识点就是一个单独的示例，不与上一个知识点的示例拉上关系，更不会与上一章的内容拉上关系，让读者能够随时独立复习每个知识点，而不用再去复习不必要的章节内容。

（5）本书的示例程序主要是针对语法问题的，示例程序每行都有注释，尽量做到把每个语法问题都反映出来。

（6）本书引用了大多数教材上没有提到的一些概念，并对这些概念做了深入介绍。

（7）本书对某些难点内容做了细致的图解分析，让读者更容易明白难点的原理。书中的图是专门针对语法问题的，尽量做到让读者看图就能明白其原理。

（8）本书对指针和数组的理解有独到的见解，学完数组和指针章节会给读者耳目一新的感觉。

（9）本书尽量做到用最少的文字、最少的篇幅描述清楚知识点，是一本真正的含金量高的图书。

由于能力有限，书中难免有错漏之处，望广大读者指出更正，不胜感激。

读者服务

轻松注册成为博文视点社区用户（www.broadview.com.cn），扫码直达本书页面。

- **下载资源**：本书提供的附录 A 和附录 B 文件，可在 下载资源 处下载。
- **提交勘误**：您对书中内容的修改意见可在 提交勘误 处提交，若被采纳，将获赠博文视点社区积分（在您购买电子书时，积分可用来抵扣相应金额）。
- **交流互动**：在页面下方 读者评论 处留下您的疑问或观点，与我们和其他读者一同学习交流。

页面入口：http://www.broadview.com.cn/31655

目 录

第1章

C++快速入门

（1）C++是大小写敏感的语言，即在程序中大写字母C和小写字母c是不同的。

（2）关键字（保留字）：关键字是C++使用的名字，任何自定义的名字都不能与关键字相同。

（3）注意：C++中的所有符号都应在英文状态下输入，比如分号应使用“;”，而不是“；”。

示例如下：

```
/*此符号是C++的多行
注释符号*/
#include<iostream>
using namespace std;    //双斜杠符号是C++的单行注释符号
int main()
{cout << "hy" <<endl;   //将右边的表达式输出到由<<指向的左边的设备cout上。cout一般表示控制台
   cin.get();           //暂停等待输入，或按下回车键继续执行，此句可以省略
   return 0;
}
```

程序说明如下。

1．解决执行后的结果一闪而过

有些编译器在运行C++程序时，会在独立的窗口中运行，程序执行完之后窗口就关闭了（程序执行得很快，一般情况下，窗口都是一闪而过），这样就无法看到程序的执行结果。解决方法就是在程序中想要暂停的地方加上 cin.get();语句，这样程序就会停在这里等按下回车键。一般情况下，只需将cin.get();放在main函数的结尾或者return语句之前就行。

2．注释

（1）注释：注释的内容是不会被编译器编译的，注释就是程序中对代码的说明性文字，这些说明性文字是为提高程序的可读性而写的，可以在注释中写入任何内容，当然也可以没有注释。

（2）单行注释使用“//”符号：C++使用“//”作为程序注释的开头，“//”后面的内容为程序的注释。但使用“//”只能注释单行的程序，若注释超过一行，到第二行又必须以“//”作为

注释的开头。

（3）多行注释使用/*...*/符号：这是 C 风格的注释，但 C++同样支持，/*和*/之间的内容为注释，以/*开头，以*/结束，这种注释方法可以跨越多行。

3．预处理器和#include<iostream>

（1）#include 是预处理器指令，预处理器就是编译器在把代码编译为机器指令之前执行的一个过程，它会在编译程序的时候自动运行，所有的预处理器都是以“#”开头的，其中 include 是预处理器的一个指令。

（2）#include<iostream>的作用：将 iostream 文件的内容添加到程序中。在将代码编译为机器指令之前，会将 iostream 文件的内容添加到程序中，这时 iostream 文件的内容将取代代码行#include<iostream>。

（3）iostream 文件的作用：这里的 io 指的是输入（input）/输出（output），使用 cout 进行输出或使用 cin 进行输入的程序必须包含 iostream 文件，因为 cout、cin 这些内容是在 iostream 文件中定义的。

4．头文件（或包含文件）和 iostream

（1）头文件（或包含文件）：像#include<iostream>中的 iostream 文件，被包含在其他文件起始处的文件被称为头文件或包含文件。

（2）可以将任何合法的 C++文件包含进其他文件，只需像上面那样使用#include<...>即可。

（3）库文件：C++编译器本身自带了很多头文件，这些头文件都有一定的功能，例如 math.h、iostream.h，其中 math.h 文件有很多关于数学计算方面的函数（函数见后文解释）。C++自带的这些标准头文件被称为库文件。

（4）在 C 语言中头文件使用扩展名.h，而 C++则不再使用扩展名了，因此老式的 C++编译器应使用#include<iostream.h>并将后面的 using namespace std;去掉，若还有更老的编译器，则应使用#include<stream.h>。因此，建议使用比较新的编译器。

（5）C++转换了 C 中的一些头文件，有些头文件还做了一些修改，并将这些文件重新命名，新命名的文件去掉了扩展名.h，并在文件名前面加上了前缀 c，比如 math.h 的 C++版本为 cmath，转换后的文件可以使用 C 所不具有的 C++特性。

5．名称空间

（1）名称空间：就是提供一个声明名称的区域，比如在某个区域声明了一些名称，那么拥有这些名称的这个区域就是名称空间。当然，你可以为这个区域取一个名字，有了名称空间就能更好地控制名称的作用范围，同时一个名称空间中的名称不会与另一个名称空间中声明的相同名称相冲突。比如重庆市有一个合川市，广西也有一个合川市，在这里“重庆市”和“广西”

就是两个名称空间。若直接说合川市并不能知道它的具体位置，但说广西合川市就能知道它是广西的，这时并不会与重庆市的合川市相冲突。因此，在使用名称空间之前应先告诉程序，将使用哪一个名称空间中的名称，这样当以后出现这个名称时就知道它来自哪个名称空间了。

（2）名称空间的创建：名称空间使用 namespace 来创建，比如 namespace sss{...}就创建了一个名称空间，该名称空间的名称为“sss”，在大括号{}中可以声明名称。

（3）使用名称空间中的名称。

- 可以使用::（作用域解析运算符）来完全限定名称空间中的名称，比如 std::cout 就表示使用的是 std 名称空间中的名称 cout。
- 使用 using 编译指令：其语法格式为“using namespace 名称空间的名称;”，表示指定名称空间中的所有名称都可用。比如使用 using namespace std;语句，就表示位于 std 名称空间中的所有名称都可直接使用。

6．std 名称空间和 C++标准库文件的关系

（1）std 是 C++的标准名称空间，所有 C++标准库文件中的名称都是在 std 名称空间中声明的。

（2）凡是需要使用 C++标准库文件（需要使用#include 将库文件包含进来）的地方都必须使用 std 名称空间，也就是应使用 using namespace std;语句。当然，也可以通过上面讲的作用域解析运算符“::”来使用 std 名称空间中的名称。

7．函数

（1）函数基础。

① 函数：函数就是一种操作或一种功能。比如使用某函数实现对一些数据进行排序，再用另一个函数进行求和运算等。

② 函数的形式为：返回类型 函数名(形参列表){函数体}。比如 int f(int a, int b){...}，表示定义了一个名称为 f，返回类型为 int（整型），且带两个 int 类型的形参变量 a 和 b 的函数。

③ 函数若没有返回值，则只需使用 void 表示函数的返回类型即可。比如 void f(){...}，关键字 void 表示没有返回类型，也就是函数没有返回值。

④ 函数的形参列表也可以为空，比如 int f(){...}。注意，即使函数的形参列表为空，也不能省略函数名后面的小括号。

⑤ 函数由两部分组成：函数头和函数体。比如 int main(){...}，函数头是 int main()，函数体则是大括号{}中的内容，在函数体中可以编写程序以使函数实现一种操作或一种功能。

（2）函数的调用。

① 在一个函数内部可以调用其他函数。

② 在一个函数中调用另一个函数只需使用另一个函数的函数名即可，当程序执行到被调用