C++语言及编程技巧

姚庭宝 编著

国防科技大学出版社
NATIONAL UNIVERSITY OF DEFENSE TECHNOLOGY PRESS

C++语言及编程技巧

姚庭宝　编著

国防科技大学出版社
·湖南长沙·

图书在版编目(CIP)数据

C++语言及编程技巧/姚庭宝编著. —长沙:国防科技大学出版社,2003.10
ISBN 7-81099-021-7

Ⅰ.C… Ⅱ.姚… Ⅲ.C语言—程序设计 Ⅳ.TP312

中国版本图书馆 CIP 数据核字(2003)第 093046 号

电话:(0731)4572640 邮政编码:410073
E-mail:gfkdcbs@public.cs.hn.cn
责任编辑:何 晋 责任校对:陈 靖
新华书店总店北京发行所经销
国防科技大学印刷厂印刷
*
787×1092 1/16 印张:29 字数:724 千
2003 年 10 月第 1 版第 1 次印刷 印数:1—3000
ISBN 7-81099-021-7/TP·2
定价:35.00 元

前　言

C++语言起源于广泛流行的C语言，是C语言的有机继承与发展。它由C语言派生而来，既兼容了C语言的大部分特性，又包含了许多新的特性，使其成为比C语言更灵活、功能更强大的计算机程序设计语言。C++语言是20世纪90年代以来最热门的软件开发平台之一，它体现了从结构化程序设计方法到面向对象程序设计方法的演变，而面向对象的分析与设计概念正成为最受欢迎的程序设计方法学。C++语言发展至今，业已成为国内外越来越多软件开发人员的首选语言。在国内，近几年不少高校已陆续将C++语言作为大学本科学习计算机编程的入门语言，本书即是为此目的而编写的。大学低年级是很重要的打基础阶段，其所选计算机语言应能充分体现适宜于加强基础、培养与训练计算机语言编程及应用能力、拓宽知识面的特色，并能适应信息技术飞速发展的需求，C++语言能很好地充任这一角色。

全书主要内容分为上、下两编。上编《C++面向过程程序设计》共八章。主要介绍C++语言的特点与基本要素、程序结构和过程化基础。在上编中，较详尽地阐述了C++语言的一系列基本概念与特性，提供丰富的示例和程序加以应用，同时还推介有关程序设计的基本方法与技术、编程技巧以及良好的编程风格与习惯。希望读者经过仔细阅读与理解，逐渐领会并掌握C++语言的各种基本概念与特性，能运用C++语言的基本要素和程序设计的基本方法进行面向过程的结构化程序设计，编写各类C++源程序，并通过上机实践进行调试、运行，检验、巩固所学概念与知识，逐步培养用C++语言编程和进行程序开发的能力。下编《C++面向对象程序设计》共七章，它是上编的有机发展，在熟悉C++面向过程程序设计的基础上，从体现软件工程思想的角度，阐述C++面向对象程序设计的基本特性和使用方法。希望读者通过仔细阅读与理解，能正确领会C++语言功能强大、使用灵活的特色，学会用面向对象程序设计方法与技术去思考、分析、设计、编程，努力提高C++语言程序设计水平，为能在后续课程（首要的是《数据结构》）实际应用以及今后开发应用软件中得心应手地、充分地展示自己的良好编程能力和分析问题、解决问题的本领构筑一座坚实的桥梁。

作者在编写本书的过程中，力求做到全书内容翔实，信息量大，结构严谨，编排合理，文字流畅，示例众多，便于自学，实用性强。文字叙述努力体现启发性、针对性、直观性、技巧性。为了更能体现大学生教材的特点，每章末均安排有小结。近期还将编著出版与本教材相配套的《C++语言程序设计习题与解析》一书，以有利于广大学生进一步培养、检验阅读理解程序示例、独立编程以及上机操作的能力，提高并加强计算机应用的整体水平。

全书共列举了近200个完整的C++程序，它们均已在Microsoft Visual C++ 6.0系统下经编译通过并正确运行。

作为讲授《C++语言及编程技巧》课程的大学教材，本书计划教学时数为64学时，分两学期（大学一年级下半学期与二年级上半学期）讲授，每学期32学时。还应安排充足的上机实践机时。在高年级时还宜安排32学时的《程序设计实践》课程，使广大学生能藉此进一步发展程序设计技巧，提高灵活运用C++语言解决实际应用问题的本领，并深入体现并挖掘C++面向对象程序设计的潜力。

作者在著述时，参阅了国内外一些有关 C++ 语言及程序设计的教材、书籍，受益匪浅。在此，谨向这些教材、书籍的作者表示感谢。作者衷心感谢亲人杜秀荣女士，是她的一贯关心与支持，使我能忙中偷闲、集中精力编写并完成此书。还应感谢曾与我合作编著《C 语言及编程技巧》教材及教学参考书的陆勤副教授、封孝生副教授，以及国防科技大学人文与管理学院的邓苏教授、肖卫东副教授、刘忠副教授、陈洪辉副教授、蔡建国讲师，他们给予作者不少帮助并提出很好的建议。国防科技大学人文与管理学院常兆诚副院长等领导一直关心和支持学校在新形势下的教材编写并鼓励作者。不少工作人员付出了辛勤的劳动。在此一并致谢。

对于本书的内容选择、深度难度以及文字叙述上的不当之处，热诚欢迎广大读者特别是使用本书作为教材或教学参考书的大学生们提出批评、建议，也衷心希望能得到各高校教师及各界同行的指教与帮助。

姚庭宝*
2003 年 9 月于国防科技大学

* 联系地址：湖南省长沙市国防科技大学人文与管理学院，邮政编码：410073，电话：(0731)4575578(办)，(0731)4575557(家)。

目　录

◁ 上编　C++面向过程程序设计 ▷

上编　C++面向过程程序设计

上编主要介绍C++程序设计语言的特点与基本要素、程序结构和过程化基础。在上编中，将详尽地阐述C++语言的一系列基本概念与特性，并提供丰富的示例和程序加以应用，同时还推介有关程序设计的基本方法与技术、编程技巧以及良好的编程风格与习惯。希望读者通过仔细阅读与理解，逐渐领会并掌握C++语言的各种基本概念与特性，能运用C++语言的基本要素和程序设计的基本方法进行面向过程的结构化程序设计，编写各类C++源程序，并通过上机实践进行调试、运行，检验、巩固所学概念与知识，逐步培养用C++语言编程和进行程序开发的能力。在学习方法上，宜勤思索，多实践，先模仿，再创新；主动进取，促进思维，举一反三，灵活运用。不仅着力于培养与提高使用C++这一优秀语言工具进行程序设计的兴趣，而且还追求充分挖掘C++语言强大的潜在能力，不断加强用计算机编程解决应用问题的本领。这也必将为进一步学习下编《C++面向对象程序设计》打下较扎实的基础。

上编主要内容

第一章　C++编程基础

C++与C一样，是一种优秀的程序设计语言。对于初次接触并下决心学习C++语言及程序设计的读者来说，本章内容是入门开篇，是进入C++神奇殿堂的基石。本章通过展示一些简单示例介绍C++程序的基本结构，以及构成C++程序若干最基本、最常用的元素概念及使用规则。

本章主要内容包括：关于C和C++；C++程序的基本结构；基本字符集、标识符与关键字；基本数据类型；运算符与表达式；赋值语句及输入输出简述；程序设计风格。

1.1　关于C和C++

C语言是国际上广泛流行的一种结构化语言。它是作为UNIX操作系统的开发语言而推出的。C语言刚问世时，其主要目的只是为了描述和实现UNIX操作系统而提供一种较为灵活、实用的工作语言，其使用对象主要是既有扎实程序设计理念、又有丰富实际应用经验的编程人员和软件工程师们。由于UNIX操作系统的成功，加上C语言本身的诸多优点，使得C语言在系统软件领域得到了广泛的应用。C语言具有很强的功能和高度的灵活性。它和其他流行的结构化语言一样，能够提供丰富的数据类型、广泛使用的软件以及一组很丰富的、供计算和数据处理的运算符。由于C语言的优异特性及强大功能，在不断扩充和完善的过程中，它已逐渐成为设计、开发系统软件与应用软件的主流语言之一，为广大微机用户和编程人员所喜用。C语言实际上已成为工业界所选用的系统实现语言。

C语言具有下列特点：

(1)C是中级语言，语言表达能力强；

(2)C是结构化语言，层次清晰，结构紧凑；

(3)运算符、数据类型丰富，功能齐全，灵活多样；

(4)语言简洁，使用方便、灵活；

(5)生成的目标代码质量高，快捷高效；

(6)脱离具体机器，可移植性强，适用范围广，能有效提高软件生产率。

C语言自问世以来，以其表达简洁、实现高效、灵活小巧、功能齐全而风靡系统软件领域，C语言得到了极为广泛的应用。但是，随着时代的变迁，软件规模越趋庞大，复杂性也越来越增加。软件技术不断更新，更为强调模块性、抽象性，要求软件可扩充性强，可重用性高，易于维护，便于移植。特别是软件工程的兴起，面向对象技术的出现，都给各种程序设计语言提出了更新、更高的要求。C语言在盛行的同时，也逐渐暴露出它自身的局限性。比如，C语言不是强类型语言，语法限制少，程序中很多问题在编译期间不能发现，而在运行期间爆发，这对保证程序的正确性、可靠性和安全性上将带来隐患。又如，C语言缺乏支持代码重用的语言机制，对于大型程序的开发，难以控制程序的复杂性。特别是，C语言是一种面向过程的编程语言，已不能满足运用面向对象程序设计方法开发现代软件的需求。C++语言应运而生。

C++语言是AT&T公司贝尔实验室的Bjarne Stroustrup博士开发的。他在作为系统程序

员积累了长期实践经验的基础上，结合面向对象思想，在考虑到C语言已取得巨大成就、应用广泛并存在大量软件资源的现实下，将C语言进行了改善与扩充，增强了C语言的许多特性，特别是为了有效地提高软件生产率，确保软件的质量和可重用性，提供了面向对象的程序设计能力。因此，C++语言是从C语言发展演变而来，C++语言的开发者所走的是与C语言相兼容的道路，而不是从头来起开发一个新语言。这从为该语言所起的名称亦可见一斑。取名C++，即在原语言名称C字后面加上了一个C语言特性中的自增运算符++，以表明它是C语言的一个增强版本。

Stroustrup博士一开始研制开发的是一种带类的C，亦即是扩充了类机制的C语言。1985年，C++ 1.0定型。1989年，推出C++ 2.0。1993年，C++ 3.0问世。C++语言的标准化工作从1989年开始，1994年制定了ANSI C++标准草案。C++语言仍处在发展之中。现已制订了ISO C++标准(同时也是ANSI C++标准)。

C++语言是在C语言基础上，扩充了面向对象的程序设计发展而成的一种全新计算机高级语言。采用C++语言编程能够更自然地体现抽象数据类型的概念，从而更本质地描述数据结构和算法，在设计、开发系统软件与应用软件中，展现了强大的生命力。C++有可能成为今后软件产业的主导语言。而Visual C++则更是其佼佼者。

学习C++语言程序设计不一定非要以先学习C语言程序设计为前提。然而，由于C++被看做是更好的C，它是C语言的有机继承与发展，已有C语言程序设计基础的人们所掌握的C语言诸多特性以及面向过程程序设计的能力，对进一步学习、理解C++语言程序设计必将大有裨益。C++语言是C语言的超集，它既支持面向过程程序设计，又支持面向对象程序设计。在支持面向过程程序设计方面，C++语言与C语言并无本质上的差异，由于C++与ANSI C相兼容，众多C语言的程序设计特性可不经修改或稍加修改在C++中使用，用C语言编写的众多标准库函数和实用软件亦可以用于C++中。因此，C语言程序设计的经验将为尽快进入C++语言面向对象程序设计打下坚实的基础。有这方面经验的读者可将本书上编视作温故知新编。而尚未接触过C语言程序设计的读者，则可从头开始仔细阅读本书，直接学习C++语言程序设计。

1.2 C++程序的基本结构

学习C++程序设计语言就是为了能顺利地使用C++语言的诸多优良特性，编写C++源程序并上机调试，以解决具体应用问题。C++语言和其他流行的计算机程序设计语言一样，按其规定的构成和格式以及所提供的语句形式，由程序员编写程序，进行人与计算机的通信，实现指定的功能。就面向过程的程序设计而言，C++可以说是一种严谨的结构化程序设计语言。C++语言使用计算机能够识别和检验(通过C++编译器)的、类似于日常英语的专用词汇、术语及规则。它们构成了C++语言的语法。程序员使用它们编写C++程序时，可不必去管它们究竟是如何在计算机内部转换并实现的，而只需致力于正确地理解和使用这些词汇、术语并严格遵循既定的规则。

下面先从几个简单的C++程序示例入手，以使欲入门的读者对C++程序的基本结构有个概貌性的了解。以后随着本书展开阐述，将由简到繁，逐步深入，陆续引出一些新的特性、规则，以解决新的问题。

1.2.1 几个简短的 C++ 程序

例 1 编写并运行一个简短的 C++ 程序。

```
// program c1_01.cpp
#include <iostream.h>
void main()
{
  cout<<"Welcome to C++ world!\n";
}
```

这个程序所要实现的功能是在标准输出设备即显示屏上输出一行指定的文字信息。

程序第一行是注释行。它由"//"开始,后跟注释内容。注释内容由程序员根据需要书写。一般是为了提高程序的可读性而使用注释,故注释内容以能有助于理解程序为主。本程序中的注释仅给出了为该 C++ 程序所取的一个源程序文件名。

程序第二行是一个特殊行,这是一个在编译前由预处理程序识别的包含命令,称作预处理命令。它的作用是将括在一对尖括号内中名为 iostream.h 的文件代码在编译前嵌入到源程序该命令所在处,以便程序员能够引用其中所包含的若干标准库函数(如本示例中的函数 cout)。iostream 系 input/output stream(输入/输出流)的缩写。文件 iostream.h 中含有说明 C++若干标准输入输出设备的信息。后缀.h 表明这是个头文件,这是因为这类文件常被嵌入至程序开始处。

程序第三行是主函数的起始行。main 是 C++ 语言中标识主函数的专用名。C++ 语言规定必须用 main 作为主函数名。任一 C++ 程序中,必须有且仅有一个名为 main 的主函数,它是每个 C++ 程序的入口。main 后面的一对圆括号不能省略,因为它表明 main 是一个函数。一个 C++ 程序总是从主函数 main 开始执行。main()称为函数 main 的首部。main 前面的 void 表明此函数是一个无返回值函数。程序中允许不写 void,C++ 编译器将给出一个警告信息,但不影响运行结果。void 是系统预定义的专用名词,称为关键字。

程序第四至六行是主函数 main 的函数体。函数体自左花括号符"{"开始,至右花括号符"}"结束。这里"{"和"}"分别表示程序的打开和关闭。这一对花括号符通常被称作语句括号,在语句括号内可包含一组用以描述数据的说明部分以及一组用以规定操作的语句执行部分。本示例的函数体内仅包含一个用于输出文字信息的语句。cout 是系统实现的一个标准输出函数,它在头文件 iostream.h 中被定义为标准输出流。cout 系 console output(控制台输出)的编写。"<<"是输出操作符。本语句的作用是将紧随"<<"其后括在两个双引号之内的字符串信息输出到标准输出设备即显示器上。但换行符 \n(记作'\n')除外。'\n'是 C++ 语言中的一个转义符,其作用是回车换行,故当输出前述字符串信息之后,光标会自动跳到下一行即新行的起始位置。cout 语句的末尾有个分号符。在 C++ 语言中,分号符是语句终止符,它是每一个语句的必要组成部分。

C++ 编译器在任何时候都要求并期待一个符合语法规则的完整程序。对于这样的程序,将能顺利地通过编译并可执行。否则,编译器将会指出编译出错信息,并期待予以改正。只有改正了程序中所有语法错误,使其通过编译,才能开始执行程序。对于本示例程序,将存储在名为 c1_01.cpp 的源文件中,后缀.cpp 表明这是一个包含 C++ 程序的 C++ 源文件。经过编译

系统的编译通过并经连接后，产生了后缀为.exe 的可执行文件。若装载并运行，则将在显示屏上输出

```
Welcome to C++ world!
```

例 2 下述 C++程序中有多个 cout 语句，但多个字符串信息将输出在同一行上，也就是说运行结果与例 1 中的 C++程序相同。

```
// program c1_02.cpp
#include <iostream.h>
void main()
{
  cout<<"Welcome ";
  cout<<"to ";
  cout<<"C++ world!\n";
}
```

例 3 下述 C++程序中虽仅有一个 cout 语句，但字符串信息将分别输出在多行上。这是因为有多个转义符'\n'。

```
// program c1_03.cpp
#include <iostream.h>
void main()
{
  cout<<"Welcome\nto\nC++ world!\n";
}
```

运行示例

```
Welcome
to
C++ world!
```

例 4 运行下述 C++程序

```
// program c1_04.cpp
#include <iostream.h>
main()
{
  cout<<"Life is dear indeed,\n";
  cout<<"love is priceless too;\n";
  cout<<"But for freedom's sake \n";
  cout<<"I may part with the two.\n";
}
```

将在显示屏上输出匈牙利著名诗人裴多菲的一首诗

```
Life is dear indeed,
love is priceless too;
But for freedom's sake
```

```
I may part with the two.
```

若欲将每两句输出在同一行上，仅需将上述程序第一、三个 cout 语句中的转义符'\n'去掉即可。

例 5 下述 C++程序实现：从键盘输入两个整数值，计算其和，并输出结果值。

```
// program c1_05.cpp
#include <iostream.h>
void main()
{
  int i,j,sum;
  cout<<"Enter two integers: ";
  cin>>i>>j;
  sum=i+j;
  cout<<"The sum is "<<sum<<".\n";
}
```

程序第五行是变量说明语句，它是主函数体中的说明部分，其中 int 是个系统预定义的关键字，它表明其后的变量是整数类型，而 i、j、sum 均是变量名，故该说明语句表明变量 i、j、sum 均被说明为整型变量。

程序第六行用于显示提示信息，希望用户键入两个整数值。

程序第七行是个用于读取数据的输入语句，它让系统等待从标准输入设备即键盘输入两个整数值，并分别赋予整型变量 i 和 j。cin 是系统实现的一个标准输入函数，它在头文件 iostream.h 中被定义为标准输入流。cin 系 console input(控制台输入)的缩写。">>"是输入操作符。

程序第八行是个赋值语句。它实现把 i 和 j 相加的结果赋予变量 sum。

程序第九行仍是调用标准库函数 cout，它向标准输出设备即显示器输出文字信息以及刚求得的整型变量 sum 的值。

运行示例

```
Enter two integers:63 58↵
The sum is 121.
```

例 6 运行下述 C++程序将获得与例 1、例 2 中程序相同的输出结果。

```
// program c1_06.cpp
#include <iostream.h>
void display_message1()
{
  cout<<"Welcome ";
}
void display_message2()
{
  cout<<"to ";
}
```

```
void display_ message3()
{
  cout<<"C++ world!\n";
}
void main()
{
  display_ message1();
  display_ message2();
  display_ message3();
}
```

运行示例

```
Welcome to C++ world!
```

本程序除了主函数 main 以外，还定义了三个函数，分别取名为 display_ message1、display_ message2 和 display_ message3。每一个函数其函数体内均仅包含一个输出语句。主函数体内则包含了分别对这三个函数调用的语句。每次进行函数调用时，控制将转移到该函数定义的函数体中去执行那里的输出语句，输出若干文字信息，然后将控制返回至原调用处，继续执行 main 主函数体中的下一语句。就这三个函数体本身而言，均并无返回值。它们都是无返回值函数。但在显示器上，调用每一函数都将显示一个字符序列。

在一个 C++ 程序中，程序员可根据实际应用问题的需要，编写若干个自定义函数。当然需要依据一定的规则来编写函数定义，并正确地使用它们。

例 7　最简单的、语法正确从而能编译通过并顺利运行的 C++ 程序编码是

```
main()
{
}
```

或者写在同一行上

```
main(){}
```

运行该程序将不输出任何内容。

1.2.2　C++ 程序的基本结构

以上几个程序简例初步勾画出一个 C++ 程序的基本结构。概括如下：

(1)C++ 程序是一种函数结构，函数是 C++ 程序的基本构件。C++ 程序是函数驱动的。C++ 用函数组织程序。可执行的 C++ 程序系由一个或多个函数组成，每个函数有其自己的名字，实现指定的功能。任何 C++ 程序必须有一个名为 main 的主函数。运行 C++ 程序总是从 main 函数开始，结束于 main 函数最后一个花括号。但 main 函数不一定非得出现在程序的开头。

(2)C++ 程序中的函数定义由函数首部与函数体组成。函数首部说明了该函数的名字和类型特征等。函数体是一个由左、右花括号符括起来的复合结构，其中可包含若干变量说明语句及执行语句。变量必须先说明后使用。执行语句包括输入语句、输出语句、赋值语句等。

(3)C++ 程序中的函数有标准库函数和用户定义函数两种。在前面几个 C++ 程序简例中，cout、cin 都是标准库函数，而 display_message1、display_message2、display_message3 均为

用户定义函数。标准库函数由系统提供,可供任何程序使用。在C++程序中不用再定义它们,但应写上相应的包含命令,如带有头文件 iostream.h 的#include 命令。在C++程序中,主函数可以调用其他函数,其他函数间也可以相互调用。但其他函数不能调用主函数。

(4)C++程序通过标准输入/输出流进行输入/输出。标准输入流 cin 用于从标准输入设备即键盘上读取适当的数据并赋予相应的变量。标准输出流 cout 用于向标准输出设备即显示器按指定格式输出若干文字信息、数据等。

(5)在C++程序中,任一语句必须以分号符作为其终止符。分号符不仅仅起语句分隔符的作用,而且它还是语句的必要组成部分,故不能省略。但应注意程序中有些场合不应随意加上分号符,比如#include 命令末尾、main()之后以及花括号对的后面等。

(6)C++程序中的大多数计算都是通过执行赋值语句来完成的。在往后所举的程序示例中,将有大量赋值语句。在这里需记住的是,像在例5中出现的赋值语句

```
sum=i+j;
```

其含义为:将等号符右部整型变量 i 与 j 的值相加,其结果值赋予等号符左部的整型变量 sum。这里的等号符称做赋值运算符,或简称赋值符。

(7)为了增加C++程序的可读性,可在程序中适当位置插写注释。注释以"//"开始,后跟注释内容。注释自动在行尾结束,即它在单行内有效。

(8)为便于理解并修改程序,宜按一定的缩进格式来书写并键入程序。对于层次较多、结构较为复杂的程序而言,这样做就更显其重要性。每次缩进的空格字符数,并无统一规定。本书中所有C++程序,均遵循每次缩进两个ASCII码空格字符的原则。

1.3 基本字符集、标识符与关键字

C++程序系由一行行字符序列所组成。字符集是构成C++程序的基本元素。用C++语言进行程序设计,必将涉及诸多运算对象。在C++程序中,以标识符代表某些运算对象。C++语言有关键字、标准标识符、程序员自定义标识符三类标识符。关键字又称保留字。程序员自定义标识符又称用户标识符。

1.3.1 基本字符集

和其他计算机编程语言一样,C++语言有其自身的基本字符集及使用规则。一个C++程序是由C++语言基本字符构成的一个有机序列。C++语言的基本字符集是ASCII字符集的一个子集。ASCII是American Standard Code for Information Interchange(美国国家信息交换标准代码)的词头缩短语。ASCII字符集的标准编码模式为大多数计算机系统所采用。标准的ASCII字符集以7位二进制代码形式存放于8位二进制数中,故它一共有128个ASCII代码,其序号为0至127。在这128个不同的ASCII代码中,有95个可见字符和33个控制字符。ASCII码值为0~31以及127所对应的字符均为控制字符,它们用于数据通信和让设备实现某些功能。ASCII码值为32~126所对应的字符均为可见字符。附录一给出了常用字符与ASCII代码对照表。

C++语言使用一组基本字符集来构造C++语句以及整个C++源程序。C++语言的基本字符集所包含的字符有