21世纪高等院校计算机系列教材

C/C++程序设计

CHENGXU SHEJI

沈克永 刘肃平 钟文峰 ◎ 主编

北京邮电大学出版社
www.buptpress.com

21 世纪高等院校计算机系列教材

C/C++ 程序设计

主　编　沈克永　刘肃平　钟文峰
副主编　杨　波　陈　强　张　朋

北京邮电大学出版社
·北京·

内 容 简 介

本书结合最新的C/C++标准，对C/C++程序设计语言进行深入浅出的介绍。从最基本的概念出发，介绍C/C++作为语言的来龙去脉，并且一步步地进入语言的更深层次开发。本书前部分为C的基础部分，在C语言中适当的引入C++的相关知识，以便让学者学习后部分C++打下扎实的基础。通过结构化程序设计的学习，读者可具备软件开发所需要的基本知识。针对面向对象理论的编程方法，本书对封装、继承和多态、运算符重载语法现象等重点部分提供了明确而细致的阐述。

本书分为两大块：第1～6章具体讲述C语言的基本概念和一般编程方法；第7～12章具体讲述C++的基本知识，介绍C++对C的扩充和面向对象编程的基础知识。

本书适合高等院校计算机类学生及计算机相关工作人员使用。

图书在版编目(CIP)数据

C/C++程序设计/沈克永主编. —北京：北京邮电大学出版社，2005
ISBN 7-5635-0974-7

Ⅰ.C... Ⅱ.沈... Ⅲ.C语言—程序设计 Ⅳ.TP312

中国版本图书馆CIP数据核字(2005)第000141号

书　　名：C/C++程序设计
主　　编：沈克永
责任编辑：方　瑜　李欣一
出版发行：北京邮电大学出版社
社　　址：北京市海淀区西土城路10号(邮编：100876)
电话传真：010-62282185(发行部) 010-62283578(FAX)
电子信箱：publish@bupt.edu.cn
经　　销：各地新华书店
印　　刷：北京源海印刷有限责任公司
开　　本：787 mm×1 092mm　1/16
印　　张：19.75
字　　数：493千字
印　　数：1—5 000册
版　　次：2004年11月第1版　2004年11月第1次印刷

ISBN 7-5635-0974-7/TP·137　　定　价：29.00元

前　言

新的编程语言不断涌现，同时一些旧的语言因为不能满足需求而被淘汰，而C语言因其与主流操作系统和新兴语言的密切联系（Windows、UNIX和Linux的大部分代码都是用C语言写的，而Perl和Python等语言“新锐”，最初也是用C语言实现的），故一直保有较多的用户。在国内外的大多数高校，C语言也一直是主要的计算机教学语言。但是，笔者通过对C语言教材多年的关注，发现国内C语言教材的编写存在两个问题：

问题一　C语言不是FORTRAN

许多作者在编写教材时，仅仅把C语言当作一种更好的FORTRAN，这种观念是否正确，本文不作讨论。书中的多数习题都是数学问题。在把C语言引入大学教学的早期，这样做并不奇怪。因为在我国计算机科学发展的早期，计算机主要是为大规模数值运算服务的。许多教授C语言的老师都有用FORTRAN解决数学问题的经历，所以他们在编写教材时，自然把注意力集中在数学领域。而且，Matlab和Maple等数学软件那时尚未进入我国，使用C语言解决数学问题可以说是顺应环境的。但是同10年前相比，现在的大学生很容易得到各种数学工具软件，在这种情况下，C语言教材的习题仍然没有多大变化，这就有些耐人寻味了。在书店里，随意翻阅几本C语言教材，几乎总可以发现求最大公约数、矩阵转置等“经典”习题。一些所谓“创新”的教材，不过是在引入数学问题时转换了一个角度。可以说，这些C语言教材仅仅是在教授学生用C语言解决数学问题，而书中的数学问题用Matlab甚至Excel只需一个命令就能得到结果。这样的习题对学生能有多少意义呢？C语言的设计者Ritchie说过，学习一种语言的最好方法是编程。而一味地让学生编程解决没有吸引力的问题，恐怕他们的学习热情很快会被耗尽。在这方面，作者们应该多多参考国外的优秀教材，例如Kernighan和Ritchie写的《C程序设计语言》（The C Programming Language 2ndEd），在设计习题时，始终考虑C语言的特性：它确实是一种通用性极强的语言，但它的各种特点使它在编写操作系统和编译器时具有极大的优势。因此书中大多数习题都与这几个领域的重要概念密切相关，而纯粹的数学题只在第1章出现了几道。国外还有一些较新的C语言教材，全书习题以编写一个小型的语言编译器为主线，学生把习题从头到尾做下来，一个较简单的编译器也就构造出来了。这样更容易让学生掌握C语言的精髓，同时也能让他们更快地获得“专业感”。

问题二　Turbo C 2.0已过时

在选择编程环境时，Turbo C 2.0始终是多数作者的“最爱”。Turbo C 2.0是美国Borland公司于1988年推出的产品，在DOS时代，它是C语言开发环境中当之无愧的佼佼者。但是，16年过去了，在这16年中，DOS早已被Windows和Linux取代，C语言也经历了两次大的修订（分别在1989年和1999年）。因此，无论是在使用的便捷性上，还

是在对语言标准的支持上，Turbo C 2.0 都已经“廉颇老矣”。作者们之所以抱着 Turbo C 2.0 不放，固然说明这个软件确实优秀，但更主要的，恐怕还是图省事，不愿意在教材修订上下大力气。实际上，在 Windows 环境中，目前有许多 C 语言编译器和开发环境可以免费下载，如 Dev-Cpp(http://www.bloodshed.net)，LCC-Win32(http://www.cs.virginia.edu/~lcc-win32/)，Mingw(http://mingw.sourceforge.net)。同 Turbo C 2.0 相比，这些开发环境提供了更方便的使用特性、更新的语言支持以及更多的帮助文档。对 Borland 或者微软公司的产品情有独钟的作者，还可以登录它们的网站，免费下载最新的 C 编译器(均可以编译 C 语言源程序)，用它们作为教学环境。如果学校使用的操作系统是 Linux，那就更方便了，因为几乎所有的 Linux 发行版本(distributions)都提供了 C 语言编译器 gcc。对于有经验的 C 语言用户来说，熟悉以上编程工具并不困难。作者为什么不考虑把它们介绍给学生呢？

目前教材市场竞争已趋白热化，同时越来越多的国外优秀教材进入中国。如果我们的 C 语言教材作者仍然墨守陈规，不敢或不愿大幅度更新教材内容，那么他们的作品终将乏人问津。

众所周知：全国计算机等级考试、全国计算机应用技术证书(NIT)和全国各个地区组织的大学生计算机统一考试都将 C 语言列入了考试范围。学习 C 语言是广大计算机应用人员和广大学生的迫切要求。我们已经知道 C 的重要性了，因此也针对近几年的市场调查和多年来的教学经验，组织了一批具有丰富教学经验的骨干教师根据学校的实际情况对多年来使用的教材进行了大幅度的调整及内容的更新，精心编写出了《C/C++ 教程》这本教材，全书共分 11 章，概念清晰、内容详尽、例题丰富、深入浅出、通俗易懂、一环紧扣一环。最后在 C 的基础上，为学生朋友今后学习 C++ 做铺垫，专门增加了第 11 章 C++ 的初步知识，供读者选学。

本书由沈克永教授组织编写，钟文峰、刘肃平主审。其中，目录由刘肃平老师编写，第 1 章由钟晓燕老师编写，第 2 章由李芳老师编写，第 3 章由陈黎艳老师编写，第 4 章由王和平老师编写，第 5 章由胡思娟老师编写，第 6 章由章志明老师编写，第 7 章由王立平老师编写，第 8 章由钟文峰老师编写，第 9 章由王芳老师编写，第 10 章由夏晓娣老师编写，第 11 章由冯芳、王施媛、刘雅娟老师编写，附录由杨波老师编写。同时陈强、张朋、孙素敏等老师也参与了教材的编写工作。

可以说这次对教材的更新是为适合学校的特点而作的努力，写好一本书不是一件简单的事情。写书不只是简单地把有关的技术内容告诉读者而已，还要考虑怎样写才能使读者更容易理解，这要下很大的功夫才能做到。由于水平有限及时间仓促，本书肯定会有不少缺点和不足，热切期望得到专家和广大读者的批评指正。

作　者

2004.11

目　录

第1章 C语言概述

1.1 C语言的发展历史及其基本特征

C语言是目前世界上最流行、使用最广泛、很有发展前途的一种优秀的高级程序设计语言。它的语言功能比较丰富，表达能力强，使用起来非常灵活方便。在对操作系统和系统使用程序以及需要对硬件进行操作的场合，用C语言明显优于其他高级语言，许多大型应用软件都是用C语言编写的。除此之外，C语言还具有绘图能力强、可移植性高、数据处理能力强大的优点，因此是一门适于编写系统软件，三维、二维图形和动画的高级语言。可见，C语言适合于作为系统描述语言，既可用来写系统软件，也可用来写应用软件。C语言的应用如此广泛，那么它又是如何发展至今呢？

1.1.1 C语言的发展历史

C语言的发展颇为有趣。它的原型是早期的ALGOL 60语言。ALGOL 60是在1960年出现的，它是一种面向问题的高级语言，但是它离计算机硬件比较远，不适合用来编写系统程序。1963年，剑桥大学将ALGOL 60语言发展成为CPL (Combined Programming Language) 语言。CPL语言在ALGOL 60语言的基础上更接近硬件，但是它的规模比较大，难以实现。1967年英国剑桥大学的Matin Richards对CPL语言做了简化，并推出了BCPL (Basic Combined Programming Language)语言，1970年，美国贝尔实验室的Ken Thompson将BCPL进行了修改，并为它起了一个有趣的名字“B语言”。意思是将CPL语言煮干，做了进一步的简化，提炼出了它的精华。并且他用B语言写了第一个UNIX操作系统，在PDP-7上实现。但B语言过于简单、功能也极其有限。而在1972年至1973年间，B语言也被人给“煮”了一下，美国贝尔实验室的D. M. Ritchie在B语言的基础上最终设计出了一种新的语言，他取了BCPL的第二个字母作为这种语言的名字，这就是早期的C语言。C语言既保持了BCPL和B语言的优点(精练、接近硬件)，又克服了他们的缺点(过于简单、数据无类型等)。

继C语言诞生后，为了使UNIX操作系统推广，1977年Dennis M. Ritchie发表了不依赖于具体机器系统的C语言编译文本《可移植的C语言编译程序》，使C移植到其他机器时所需做的工作大大简化了。1978年Brian W. Kernighian和Dennis M. Ritchie出版了名著《The C Programming Language》，从而使C语言成为目前世界上流行最广泛的高级程序设计语言。1978年以后，C语言已先后移植到大、中、小、微型机上，并独立于UNIX和PDP的

发展。1987 年,随着微型计算机的日益普及,出现了许多 C 语言版本。由于没有统一的标准,这些 C 语言之间出现了一些不一致的地方。为了改变这种情况,美国国家标准研究所(ANSI)为 C 语言制定了一套 ANSI 标准,成为现行的 C 语言标准。1990 年,国际标准化组织 ISO(International Standard Organization)接受 1987 年 ANSI C 为 ISO C 的标准(ISO 9899—1990)。目前所广泛使用的 C 编译系统大致都是以它为基础的,但也有些不同。在微型机上常见使用的有 Microsoft C、Turbo C、Quick C、Borland C 等版本。

对 C 语言和其他传统的高级语言作比较后发现,从掌握语言的难易程度来看,C 语言比其他语言难一些。BASIC 是初学者入门的较好的语言,FORTRAN 也比较好掌握。对科学计算多用 FORTRAN 或 PL/I;对商业和管理等数据处理领域,用 COBOL 为宜。C 语言虽然也可以用于科学计算和管理领域,但 C 语言的特长并不在于此而是在于 C 语言能对操作系统和系统实用程序以及需要对硬件进行操作的场合。从教学角度看,由于 PASCAL 是世界上第一个结构化语言,所以曾一度被认为是计算机专业的比较理想的教学语言,但 C 语言也是理想的结构化语言,且描述能力强,也适于教学。因此,C 语言已经成为被广泛使用的教学语言。

1.1.2 C 语言的基本特征

C 语言发展如此迅速,而且成为最受欢迎的语言之一,主要因为它具有强大的不同于其他语言的功能。许多著名的系统软件,如 DBASE Ⅲ PLUS、DBASE Ⅳ 都是由 C 语言编写的。用 C 语言加上一些汇编语言子程序,就更能显示语言的优势了,像 PC-DOS、WORD-STAR 等就是用这种方法编写的。归纳起来 C 语言具有下列特点:

1. C 语言是中级语言

C 语言允许直接访问物理地址,能进行位(bit)操作,能实现汇编语言的大部分功能,可以直接对硬件进行操作。C 语言可以像汇编语言一样对位、字节和地址进行操作,而这三者是计算机最基本的工作单元。它把高级语言的基本结构和语句与低级语言的实用性结合起来,使得 C 语言既是成功的系统描述语言,又是通用的程序设计语言。

一般的将各类语言分类如下:

高级	BASIC
	FORTRAN
	COBOL
	PASCAL
	Ada
	Modula-2
中级	C
	FORTH
	宏汇编
低级	汇编语言

2. 简洁紧凑、灵活方便

C 语言一共只有 32 个关键字,9 种控制语句。程序书写自由,主要用小写字母表示。

下面将C语言与PASCAL语言做一比较,如表1.1所示。

表1.1 C语言与PASCAL语言的对比

C语言	PASCAL语言	含义
{ }	BEGIN … END	复合语句
if(e) S;	IF (e) THEN S	条件语句
int i;	VAR i:INTEGRE;	定义i为整型变量
int a[10];	VAR a: ARRAY [1..10] OF INTEGER;	定义a为整型一维数组
int f();	FUNCTION f():INTEGER;	定义f为返回整型值的函数
int * p;	VAR p: ↑INTEGER;	定义p为指向整型变量的指针变量
i+=2;	i:=i+2	赋值语句,使i+2→i
i++, ++i	i:=i+1	i自增值1,i+1→i

从比较中我们可以看到,C程序比PASCAL简练,源程序短,压缩了一切不必要的成分,使得输入程序的工作量更少。

3. 运算符丰富

C的运算符包含的范围很广泛,共有34个运算符。C语言把括号、赋值、强制类型转换等都作为运算符处理,从而使C的运算类型极其丰富,表达式类型多样化,灵活使用各种运算符可以实现在其他高级语言中难以实现的运算。

4. 数据结构丰富

C的数据类型有:整型、实型、字符型、数组类型、指针类型、结构体类型、共用体类型等。它具有现代化语言的各种数据结构,能用来实现各种复杂的数据类型(如链表、树、栈等)的运算。在C语言中引入了指针概念,使用起来比PASCAL更为灵活、多样,使程序效率更高。另外C语言具有强大的图形功能,支持多种显示器和驱动器,且计算功能、逻辑判断功能强大。

5. C语言是结构式语言

结构式语言的显著特点是代码及数据的分隔化,即程序的各个部分除了必要的信息交流外彼此独立。这种结构化方式可使程序层次清晰,便于使用、维护以及调试。C语言是以函数形式提供给用户的,这些函数可方便地调用,并具有多种循环、条件语句控制程序流向,从而使程序完全结构化。它层次清晰,便于按模块化方式组织程序,易于调试和维护。

6. C语法限制不太严格、程序设计自由度大

一般的高级语言语法检查比较严,能够检查出几乎所有的语法错误,而C语言允许程序编写者有较大的自由度。例整型量与字符型数据以及逻辑型数据可以通用。

7. C语言程序生成代码质量高,程序执行效率高

C语言程序生成的代码一般只比汇编程序生成的目标代码效率低10%~20%。

8. C语言适用范围大,可移植性好

C语言有一个突出的优点就是适合于多种操作系统,如DOS、UNIX。同时,C语言也适用于多种计算机机型。

由于C语言的这些优点,C语言的应用领域越来越广,许多大型的软件都采用C语言

编写，这主要是因为C语言良好的可移植性和较强的硬件控制能力。除此之外，C语言已经成为被广泛使用的教学语言。而且，C语言强调程序设计的灵活性，放宽了对语法定义的限制，使得C语言的应用面更为宽广。

以上只是介绍了C语言的最容易理解的一般特点，至于C语言内部的其他特点，在深入了解了C语言以后，各位自己就会慢慢发现和领会。

1.2 简单的C程序介绍

本节主要介绍几个简单的C程序，通过从这几个程序的分析中感受C程序的一些特性。

【例1.1】

```
#include<stdio.h>
main ( )
{
  printf("\nHello World!\n");
}
```

程序运行结果为：

```
Hello World!
```

虽说这个程序只有5行，但是这是一个C语言的经典的程序。它牵涉到了C语言的函数调用、文件的预先声明处理，以及一个完整的“main()”主函数。在这个程序中，第一行的“#include<stdio.h>”是一个对stdio.h文件的预先声明和调用，因为以后的“printf()”函数是在这个文件中描述和定义的。在C语言中，为了程序的顺利执行，所有的变量和函数在使用之前都必须声明。第二行的“main()”是主函数名，在用C语言编写程序时，每一个程序都必须有一个“main()”函数（主函数）且只能有一个主函数。在对程序进行编译时，编译程序会找到“main()”函数作为程序的入口来编译程序。第三行和第五行的大括号“{”“}”是主函数范围限定符，它们规定了主函数的范围。“{”和“}”必须成对出现，否则会出错。第四行的“printf("\nHello World!\n");”是一个对输出函数的调用，双引号内的字符按原样输出，“\n”是换行符，即在输出“Hello World!”后回车换行，语句最后有一分号。在C语言里没有专用的输入、输出语句，而是调用函数“printf()”来实现输出的。"\nHello World!\n"则是传给这个函数的参数。

【例1.2】

```
main(  )                /*主函数*/
{
  int x,y,sum;          /*定义变量*/
  x=46;y=100;
  sum=x+y;              /*求两数之和*/
  printf ("sum is %d\n", sum) ;
}
```

此程序的功能是求出两个已给出的整数的和。为了便于理解，在程序中用/* …… */加注汉字表示解释，但在实际的程序运行中，注释是不起作用的。

该程序首先定义 x，y 为整型变量，然后通过赋值运算符“=”分别将 x，y 赋值为 46，100。在程序的第五行中，将 x+y 的值赋给变量 sum，在第六行中“%d”是输入输出的“格式字符串”，它主要用来指定输入输出时的数据类型和格式，在以后的章节中将详细介绍与学习。本程序中，“%d”表示输出的是“十进制整数类型”的数，而 printf 函数中的 sum 是要输出的变量。

最终程序运行后的结果为

```
sum is 146
```

【例 1.3】

```
main( )                          /* 主函数 */
{  int x,y,z;                    /* 变量说明 */
   scanf("%d%d",&x,&y);          /* 输入变量 x 和 y 的值 */
   z=max(x,y);                   /* 调用 max 函数 */
   printf("max=%d",z);           /* 输出 =max(x,y) */
}

int max(int a,int b)             /* 定义 max 函数 */
{  int c;                        /* 声明部分，定义变量 */
   if (a>b) c=a;
   else c=b;
   return (c);                   /* 返回 c 的值，把结果返回主调函数 */
}
```

此函数的功能是输入两个整数，输出其中的最大数。上面例中程序的功能是由用户输入两个整数，程序执行后输出其中较大的数。本程序由两个函数组成，主函数和 max 函数。函数之间是并列关系，可从主函数中调用其他函数。max 函数的功能是比较两个数，然后把较大的数返回给主函数。max 函数是一个用户自定义函数，因此在主函数中要给出说明。可见，在程序的说明部分中，不仅可以有变量说明，还可以有函数说明。关于函数的详细内容将在以后的学习中详细介绍。在程序的每行后用/* 和 */括起来的内容为注释部分，程序不执行注释部分。

例 1.3 中程序的执行过程是，首先请用户输入两个数，回车后由 scanf 函数语句接收这两个数送入变量 x，y 中，然后调用 max 函数，并把 x，y 的值传送给 max 函数的参数 a，b。在 max 函数中比较 a，b 的大小，把大者返回给主函数的变量 z，最后在屏幕上输出 z 的值。

通过例子，我们得出 C 源程序的结构特点：

(1) 一个 C 语言源程序可以由一个或多个源文件组成。每个源文件可由一个或多个函数组成。一个源程序不论由多少个文件组成，都有一个且只能有一个 main 函数，即主函数。也可以包含一个 main 函数和若干个其他函数，函数是 C 语言的基本组成单位。被调用的函数可以是系统提供的库函数(例如 printf 和 scanf 函数)，也可以是用户根据需要自己编写设

计的函数(例如例 1.3 中的 max 函数)。

(2) 源程序中可以有预处理命令(include 命令仅为其中的一种),预处理命令通常应放在源文件或源程序的最前面。

(3) C 程序的执行总是先从 main 主函数开始,而不论 main 函数在整个程序中的确切位置(main 函数可以放在程序的最前面,也可以放在程序的最后面,或者是放在其他的一些函数的前面或后面)。

(4) 每一个说明、每一个语句都必须以分号结尾。例如:sum = x + y; 分号是 C 语句的必要组成部分,即使是程序中最后一个语句也应该包含分号,但注意预处理命令、函数头和花括号"}"之后不能加分号。

(5) C 语言本身并没有输入输出的语句,有关输入输出的操作都是通过调用库函数 scanf 和 printf 等函数来完成的。由于输入输出操作牵涉到具体的计算机硬件设备,把输入输出操作放在函数中处理,使 C 对输入输出实行"函数化",就可以使 C 语言本身的规模较小,编译程序更为简单,程序的可移植性增强,从而能够方便、容易地在各种机器上实现。

(6) 标识符、关键字之间必须至少加一个空格以示间隔。若已有明显的间隔符,也可不必再加空格来间隔。

(7) 在程序语句旁边可以用/* …… */来对 C 程序中的语句作必要的注释,以增强程序本身的可读性。

另外,初学者从书写清晰,便于阅读、理解、维护的角度出发,在书写程序时应遵循以下规则:

(1) 一个说明或一个语句占一行。

(2) 用{ }括起来的部分,通常表示了程序的某一层次结构。{ }一般与该结构语句的第一个字母对齐,并单独占一行。

(3) 低一层次的语句或说明可比高一层次的语句或说明缩进若干格后书写。以便看起来更加清晰,增加程序的可读性。

在编程时应力求遵循这些规则,以养成良好的编程风格。

本章小结

在本章中介绍了当今被广泛使用的优秀高级程序设计语言 C 语言的发展历程。

(1) C 语言由早期的 ALGOL 60 语言发展而来,先后经历了 CPL 语言、BCPL 语言、B 语言的发展。1973 年美国贝尔实验室的 D. M. Ritchie 在 B 语言的基础上最终设计出了一种新的语言,这就是早期的 C 语言。

(2) C 语言自诞生以来一直被广泛地使用至今,主要是因为 C 语言具有强大的生命力和不同于或优于其他的语言的特点。在本章的 1.1.2 节中详细阐述了 C 语言的 8 个特点。

(3) 在本章的 1.2 节中介绍了 3 个简单的 C 程序,简要说明了 C 语言构成特点和书写习惯等知识。

总的来说,本章从总体的角度介绍了 C 语言的发展历史和 C 语言基本特征,但有关 C