高职高专计算机专业系列教材

C 语言程序设计

刘振安　编著

清华大学出版社

(京)新登字 158 号

内 容 简 介

本书以培养 C 语言应用能力为主线，强调理论教学与实训密切结合。同时，注意介绍 ANSI C 已经更新的内容，并与 C++ 接轨，例如引入函数原型、void 关键字及 const 限定符的使用方法力图使 C 与 C++ 完美衔接。在重点介绍基本理论、基本知识和基本技能的基础上，注意帮助读者熟练掌握编译工具，以便为后续课程的学习打下基础。

各章均有例题和错误分析，并结合本章内容给出实训题和习题，同时从实用的观点出发，专门开设一章介绍 C 程序结构化设计实例，结合实例详细介绍头部文件、多个 C 语言文件及工程文件的编制方法，以培养学生的实际应用能力。

本书取材新颖、结构合理、概念清楚、语言简洁、通俗易懂、实用性强，易于教学。本书特别适合作为高职高专的教材，也可作为培训教材、自学教材及工程技术人员的参考书。

图书在版编目(CIP)数据

C 语言程序设计/刘振安编著. —北京：清华大学出版社，2002
(高职高专计算机专业系列教材)
ISBN 7-302-06000-2

Ⅰ. C… Ⅱ. 刘… Ⅲ. C 语言—程序设计—高等学校—教材 Ⅳ. TP312

中国版本图书馆 CIP 数据核字(2002)第 081658 号

出 版 者：清华大学出版社(北京清华大学学研大厦，邮编 100084)
http://www.tup.tsinghua.edu.cn
责任编辑：徐跃进
印 刷 者：北京顺义振华印刷厂
发 行 者：新华书店总店北京发行所
开　　本：787×1092 1/16　**印张**：18.25　**字数**：419 千字
版　　次：2002 年 12 月第 1 版　2002 年 12 月第 1 次印刷
书　　号：ISBN 7-302-06000-2/TP・3580
印　　数：0001～8000
定　　价：24.00 元

高职高专计算机专业系列教材

序

1999年10月，教育部高教司主持召开了全国高职高专教材工作会议，会议要求尽快组织规划和编写一批高质量的、具有高职高专特色的基础和专业教材。根据会议精神，在清华大学出版社的支持下，于2000年1月在上海召开了由来自全国各地的部分高职、高专、成人教育及本科院校的代表参加的“高职高专计算机专业培养目标和课程设置体系研讨会”。与会的专家和教师一致认为，在当前教材建设严重滞后同高职教育迅速发展的矛盾十分突出的情况下，编写一套适应高等职业教育培养技术应用型人才要求的、真正具有高职特色的、体系完整的计算机专业系列教材十分必要而且迫切。会议成立了高职高专计算机专业系列教材编审委员会，明确了高职计算机专业的培养目标，即掌握计算机专业有关的基本理论、基本知识和基本技能，尤其要求具有对应用系统的操作使用、维护维修、管理和初步开发的能力。

根据上述目标，编委会拟定了本套教材的编写原则。在教材内容安排上，以培养计算机应用能力为主线，构造该专业的课程设置体系和教学内容体系；从计算机应用需求出发进行理论教学，强调理论教学与实验实训密切结合，尤其突出实践体系与技术应用能力的实训环节的教学；教材编写力求内容新颖、结构合理、概念清楚、实用性强、通俗易懂、前后相关课程有较好的衔接。与本科教材相比，本套教材在培养学生的应用技能上更有特色。

根据目前各高职高专院校计算机专业的课程设置情况，编委会确定了首批出版的十几本教材。这些教材的作者多是在高职高专院校或本科院校的职业技术学院任教的、具有多年教学经验的教师，每本书均由计算机专业的资深教授或专家主审把关。我们还将在此基础上，陆续征集出版第二、三批教材，力争在3到5年内完成一套完整的高职高专计算机专业教材。

应当说明的是，凡是高等职业教育、高等专科教育和成人高等教育院校的计算机及其相关专业均可使用本套教材。各学校可以根据实际需要，在教学中适当增删一些内容、实训项目和练习题，从而更有针对性地帮助学生掌握计算机专业知识，并形成相关的应用能力。

由于各地区各学校在教学水平、培养目标理解等方面有所不同，加上这套教材编写时间仓促，难免会出现这样或那样的错误，敬请各学校在使用过

程中及时将修改意见或好的建议返回给教材编审委员会，以便我们及时修订、改版，使该系列教材日趋完善。

我们恳切地希望高职高专院校任课的专业教师和专家对后续教材的编写提出建设性的意见，并真诚地希望各位教师参与我们的工作。

高职高专计算机专业
系列教材编审委员会
2000 年 5 月

前　言

C语言的通用性和无限制性使得它比一般的程序设计语言更加通俗，更加有效。无论是系统软件（操作系统，编译系统等）、应用软件（图形处理），还是数据处理（如企业管理）以及数值计算等都可以很方便地使用C语言。各大专院校及成人教育机构都开设了C语言课程，C语言的学习班则更为普遍。

C语言的特点是多方面的。简单说来，有如下几个优点：

- 吸取了汇编语言的精华，使C语言对高级语言来讲是“低级”语言，由于它具有描述准确和目标程序质量高的优点，所以有很强的生命力；
- 继承和发扬了高级语言的长处，使C语言相对汇编语言来讲又是“高级”语言；
- C语言的规模适中，语言简洁，其编译程序简单而紧凑，它在运行时所需要的支持少，占用的存储空间也小；
- C语言的可移植性好，这是指程序从一个环境不加改动或稍加改动就可搬到另一个完全不同的环境上运行。汇编程序因依赖机器硬件，所以根本不可移植，而其他一些高级语言（如FORTRAN等）编译的程序也是不可移植的。

由于上述几个突出优点使越来越多的人加入到学习、使用和研究C语言的行列之中。

随着C++的普及，读者可能会问：既然C++是C语言的超集，为什么不直接学习C++？我们认为C语言比C++更适合作为编程的入门语言。C++确实比C语言有更多的优点，但是如果程序不是非常庞大复杂，就显示不出C++的优点。另外，即使是目前流行的Visual C++，MFC及OWL库，它们的底层仍是C语言编写的Windows API函数，C语言结构化程序设计方法仍然适用。正如中国台湾地区的著名教育家侯俊杰先生在介绍如何学习MFC时所说：“程序设计领域里，每个人都想飞。但是，还没学会走之前，连跑都别想！”因此他首先把大家领回C语言的Windows API编程。全国计算机等级考试、全国计算机应用技术证书考试（NIT）和大学生编程竞赛，都有C语言方面的内容。甚至在学生求职时，用人单位还会问你是

否学过C语言，著名的微软公司，也会出一道C语言的题目考考应聘者，可见C语言仍然是必须掌握的一门语言。C语言在人才培养的过程中，起到很重要的作用。UNIX操作系统也是用C语言实现的，掌握C语言已经是非计算机专业学生的追求目标，作为计算机专业的学生，更需要打下坚实的基础。

对于初学者来说，另一个重要的因素就是开发程序过程中编译器所产生的错误和警告信息集合的可理解性。尽管现在多数编译系统都可以同时编译C和C++代码，但C++的错误注释通常含糊不清，没有相当的知识根本无法理解。虽然C语言中的错误信息也不够完美，但却简单得多。

事实上，计算机科学工程领域的学生大多需要同时掌握C和C++，首先教授C语言可以为随后介绍的C++奠定理论基础，使学生更容易理解抽象的数据类型。不过，本书虽然使用C语言讲解，但与现有的教材也有很大不同。首先是ANSI C有了新的发展，舍去一些陈旧的规定，引入新的规定，尤其是引入C++的一些内容。例如引入函数原型，void关键字及const限定符的使用方法，引入void指针及使用const修饰指针，并修改了对自动数组和结构初始化的规定，等等。本书虽然是在笔者原来的几本C语言教材的基础上改写的，但已经删除了陈旧的内容，引入新的内容并向C++过渡，从而为学习C++奠定基础。

本书还有如下特点：

- 以实例为蓝线，以通俗的语言和简要的内容阐述了C语言的程序设计方法；
- 理论联系实际，书中列举的例题简单易懂，针对性强，实例由浅入深，有一定工程背景，能起到学以致用的效果；
- 取材新颖，内容丰富，还精选了一些综合应用实例以加深对C语言的理解；
- 结合具体的集成开发环境实验手段，以及结合课文列举调试的具体方法，不仅给学习者提供了极大的方便，还使他们能对编辑、编译、查错及连接运行C程序有完整的认识，以便学生掌握编制实用程序的基本技能和使用集成环境及调试程序的具体方法；
- 介绍了大程序的设计方法、C语言头部文件、工程文件、库管理及多个文件的编制和调试技术，使读者能很快掌握C语言的实用技术，达到培养其应用技能的目的；
- 通过作者自己的经验，结合课文介绍了C语言编程错误及预防方法，从另一个角度加深读者对概念的理解，这不仅对初学者大有好处，对具有一定经验的C语言程序设计者也大有益处；
- 根据ANSI C的新规则，删除陈旧过时的内容，引入新的内容并向C++过渡，例如有关函数原型、void关键字、const限定符、const修饰指针、const修饰函数参数及自动数组和结构初始化等方面的新知识；
- 突出了结构化、模块化设计的基本思想；
- 配备一定数量的基础习题和实验训练题。

教育部计算机课程指导委员会副主任委员、中国科学技术大学计算机系高性能计算中心主任陈国良教授，以及原安徽大学副校长、计算机系程慧霞教授在百忙之中审阅了书稿并提出许多宝贵意见，在此深表感谢。写作中还参考了大量的文献资料，特此对这些作

者表示感谢。

彭程、孙忱、鲁磊、徐菁、葛愿、李祖奎、叶刚、田钢、林育、郭小敏等参加了本书的编写工作。在定稿之后，彭程、孙忱和鲁磊又各自将全部程序重新验证一遍，以保证程序的正确性。尤其是彭程和孙忱，不仅逐字校对，还反复推敲，提出意见供我修改，为书稿花费了大量心血。

刘振安

2002 年 6 月于中国科学技术大学

目　录

第 1 章　C 语言程序设计基础

本章简要介绍 C 语言的特点，并通过简单而典型的 C 语言程序（后面简称 C 程序）实例，引入 C 程序的构成方式，C 程序所使用的基本数据结构和表达式，以及实现输出和输入的方法，从而建立 C 程序设计的基本概念。

1.1　C 语言特点

C 语言是 20 世纪 70 年代初期美国贝尔（Bell）实验室 Dennis M. Ritchie 设计的一种程序设计语言，正式发表于 1978 年。

1970 年，Ken Thompson 在早期编程语言 BCPL 的基础上开发了一种新的语言，取名叫 “B”。Dennis M. Ritchie 在 “B” 的基础上，于 1971 年开发了第一个 C 编译程序，并于 1972 年开始使用（主要是在贝尔实验室内部使用）。以后，C 语言又经过多次改进，直到 1975 年用 C 语言编写的 UNIX 操作系统第 6 版公诸于世之后，C 语言才引起广泛的重视。目前，其应用领域已不再限于系统软件的开发，成为了最流行的程序设计语言之一。

1978 年，Brian Kernighan 和 Dennis M. Ritchie 在《The C Programming Language》一书中对 C 语言作了详尽的描述。随着微型计算机的日益普及，大量的 C 语言工具相继问世。然而这些工具没有统一的标准，存在不一致的现象。为了改变这种情况，ANSI 于 1983 年成立了一个专门委员会，为 C 语言制定了 ANSI（美国国家标准协会）标准。当时比较流行的有 TURBO C，它不仅满足 ANSI 标准，还提供了一个集成开发环境，同时也按传统方式提供了命令行编译程序版本以满足不同用户的需要。随着 Windows 编程的兴起，Borland C 和 Microsoft C 受到用户的欢迎。目前比较流行的是兼容 C 语言的 Microsoft Visual C ++ 6.0 及 Borland C ++ 集成环境。

C 语言是一种通用的程序设计语言。C 语言的通用性和无限制性，使得它对许多程序设计者来说都显得更加通俗，更加有效。目前 C 语言已用于各个方面的程序设计，无论系统软件设计（操作系统，编译系统等）或应用软件设计（图形处理），还是数据处理（如企业管理）或数值计算等都可以很方便地使用它。

C 语言有如下特点。

（1）C 语言吸取了汇编语言的精华，使它对高级语言来讲是“低级”语言（汇编语言是一种面向机器的程序设计语言，尽管它的编程相对高级语言来要麻烦得多，但由于它具有

描述准确和目标程序质量高的优点，所以它仍然有很强的生命力）。

① C 语言提供了对于位、字节以及地址的操作，使程序可以直接对内存及指定寄存器进行操作。

② C 语言吸取了宏汇编技术中的某些灵活的处理方法，提供宏代换 #define 和文件蕴含 #include 的预处理命令。目前 ANSI 新标准采用了 C ++ 的 const 类型限定符，更增加了编程的可靠性。

③ C 语言能很方便地与汇编语言连接。在 C 程序中引用汇编语言程序与引用 C 语言函数一样，这为某些特殊功能程序的设计提供了方便。

（2）C 语言继承和发扬了高级语言的长处，使它相对汇编语言来讲又是“高级”语言。

① 吸取了 ALGOL 语言的分程序结构思想。在 C 程序中，可用一对花括号 { } 把一串语句括起来而成为复合句（分程序），在括号内可定义变量。它还继承了 PASCAL 语言的数据类型，提供了相当完备的数据结构。

② 吸取了 FORTRAN 语言的模块结构思想。在 C 程序中，它的每一个函数都是独立的，可以单独编译。这对设计一个大的程序来说，有利于分工编程和调试。

③ C 程序中的任何函数都允许递归，这使某些算法实现起来就十分方便。

（3）C 语言的规模适中、语言简洁，其编译程序简单、紧凑。C 语言在表示上非常简洁（比如用一对花括号“{ }”代替 Begin_End，运算符尽量缩写等）。C 语言本身没有提供输入、输出工具以及并行操作，它的许多成分都是通过显式函数调用来完成的，而且运行时所需要的支持少，因而占用的存储空间也小。

（4）C 语言的可移植性好，其语言程序不加或稍加改动就可以从一个环境搬到另一个完全不同的环境下运行。汇编语言程序因依赖机器硬件，所以根本不可移植；一些高级语言，如 FORTRAN 等语言的程序也是不可移植的。

（5）生成的代码质量高，在代码效率方面可以和汇编语言相媲美。

C 语言的优点很多，但也有些不足之处。例如：运算符优先级太多，不便记忆，有些还与常规约定有所不同；类型检验较弱，转换比较随便，不太安全等。尽管如此，由于上述几个突出的优点，C 语言仍不失为一个实用的通用程序设计语言，因而学习和使用它的人越来越多。

1.2 C 程序入门

1.2.1 简单的 C 程序结构及函数

用 C 语言编写的程序称为 C 语言源程序，简称 C 程序。C 程序一般是由一个或若干个函数组成，这些函数可以保存在一个或几个源程序文件中，这些文件都以 C 作为文件扩展名。组成一个程序的若干函数中必须有一个且只能有一个名为 main 的函数（主函数），运行 C 程序时总是从 main 函数开始执行。在一个函数名字之后一定要有一对圆括

号，圆括号中是否有参数由编程者决定。目前只介绍无参数的 main 函数，下面是一个简单的 C 程序实例。

【例 1.1】 打印字符串。

```
/* 功能：打印字符串 */
#include <stdio.h>                          /* 包含头部文件 */
int main( )
{
    printf ( " Hello! How are you ? " );    /* 打印字符串 */
    return 0;                               /* 主函数 main( )的返回值 */
}
```

这是一个完整的 C 程序。“/*”与“*/”之间的内容是注释，注释在编译时不产生目标代码。所谓目标代码，就是程序可以执行的代码。

在【例 1.1】的程序中有一对花括号“{ }”，可以把它看做程序体括号。还可以用一对花括号括起任何一组语句构成一个复合句(分程序)。要注意在一个函数中至少应有一对花括号。C 程序的一般函数或主函数 main( )之后应有一个“{”，函数的最后应是一个“}”。在一个 C 程序或一个函数中，“{”和“}”必须成对出现。语句

```
printf ( "Hello ! How are you ?" );
```

是一个函数调用语句，它调用名叫 printf 的库函数，圆括号内由双引号括起来的部分是该语句所带的参数，即要打印的内容。printf 是标准输出函数，对应于输出设备终端显示器，上述语句表示要在其上输出字符串“Hello ! How are you ?”在右括号“)”之后的分号“;”是语句结束标志。也就是说，C 语言必须用“;”号作为语句的结束标志。

C 语言函数分为两类：系统本身提供的库函数和自定义函数。库函数又称标准函数，例如标准输出函数 printf。标准函数定义在相应的头部文件(头部文件的后缀是. h)中。如果要调用这些标准函数，要先在主函数之前使用 #include 语句将相应的头部文件包含，在需要调用标准函数的地方直接写上函数名，带上参数即可。例如 printf 函数的头部文件(简称头文件)是 stdio. h，可使用如下语句将其包含。

```
#include <stdio.h>
```

然后就可以在程序中使用 printf 库函数实现输出功能。注意，#include 语句结束没有“;”号。

C 语言有非常丰富的库函数，应该尽量利用它们。例如，库函数 abs 用来求一个整数的绝对值，它定义在头文件 math. h 中，使用时需要包含 math. h。

【例 1.2】 使用库函数求一个整数的绝对值。

```
#include <stdio.h>
#include <math.h>
int main( )
{
```