高 等 学 校 教 材

C 语言程序设计

张树粹　孟佳娜　编著

人 民 邮 电 出 版 社

图书在版编目（CIP）数据

C 语言程序设计 / 张树粹，孟佳娜编著．—北京：人民邮电出版社，2006.2
高等学校教材
ISBN 7-115-13942-3

Ⅰ．C… Ⅱ．①张…②孟… 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（2005）第 142240 号

内 容 提 要

本书全面介绍 C 语言的基本概念、语法规则和程序设计方法。全书共分 9 章，主要包括：C 语言概述，基本数据类型及运算符，程序控制结构，数组，函数，编译预处理，指针，结构型、共用型和枚举型，文件等；各章均附有习题和实验。

基于多年的教学经验，本书注重 C 语言本身的系统性与认知规律的结合。在写法上，针对初学者的特点，深入浅出，通俗易懂；在结构和内容上，准确定位，合理取舍，精选例题，强化实验。

本书适合作为大专院校“程序设计”课程的入门教材，同时也可作为计算机水平考试培训及各类成人教育等教学用书，还可以供计算机爱好者自学。

高等学校教材

C 语言程序设计

◆ 编　　著　张树粹　孟佳娜
责任编辑　张孟玮

◆ 人民邮电出版社出版发行　　北京市崇文区夕照寺街 14 号
邮编　100061　　电子函件　315@ptpress.com.cn
网址　http://www.ptpress.com.cn
北京艺辉印刷有限公司印刷
新华书店总店北京发行所经销

◆ 开本：787×1092　1/16
印张：16.75
字数：401 千字　　2006 年 2 月第 1 版
印数：1 – 3 000 册　　2006 年 2 月北京第 1 次印刷

ISBN 7-115-13942-3/TP・4929

定价：22.00 元

读者服务热线：（010）67170985　印装质量热线：（010）67129223

编 者 的 话

C 语言是国际上广泛应用的计算机程序设计语言。C 语言以其功能强大、表达灵活、代码效率高和可移植性好而著称。因此，它广泛应用于编写各种系统软件和应用软件。

程序设计语言，是学习计算机科学与技术专业大多数学科分支的入门课程，因此，高校都选择 C 语言作为学科教学的先修课程。近年来，高等学校在非计算机专业也都普遍开设了“C 语言程序设计”课程。另外，越来越多的非计算机专业的学生，参加全国计算机等级考试也都选择了 C 语言程序设计模块。学习 C 语言已经成为广大青年学生的迫切要求。但由于 C 语言涉及的概念多、数据类型多，并且使用灵活，容易出错。不少初学者觉得 C 语言难学，迫切希望有一本易于快速入门的 C 语言教材。在 C 语言课程组老师们的共同努力下，我们编写的《C 语言程序设计》已经在校内出版试用，得到了广大学生认可和学校有关部门的充分肯定。在此基础上，笔者对此书又进行认真细致地修改，力图将 C 语言中的有关概念表述得更加通俗、更加简练和更有条理，以便使学生快速入门，熟练掌握。

本书主要讲述 C 语言程序设计的基本原理和基本思想方法。主要特点概括如下。

1．定位准确，取舍合理。本书是针对高校计算机及其相关专业、非计算机专业计算机教育的本科、高职高专计算机及相关专业的“程序设计”基础课程而编写的。根据不同层次的教学要求，本书内容可灵活取舍，而不失其教材内容的科学性与系统性。

2．精选例题，通俗易懂。为使 C 语言程序设计的基本概念、基本理论叙述更加通俗易懂，本书选用了作者多年积累下来的很多教学实例；为了使学生熟练掌握 C 语言程序设计的基本方法，作者精心选编和调试了书中的所有示例，并把重点始终放在分析求解问题的思想方法上。

3．程序设计是一门实践性很强的课程，不仅要讲授程序设计的基本概念和基本理论，而且更要着力培养学生的设计和编程能力。为此，每一章后面都选编了与其教学内容紧密相关的实验题目，方便了教与学。

4．合理设计综合实例。结合数组、函数、结构型等章节内容，本书在部分章节设计了一个综合实例，以利于循序渐进地培养学生的综合应用能力。

5．本书提供配套光盘供教师教学参考，内容包括电子教案、例题源代码、习题参考答案等。

本书可供普通高等院校、各类成人教育院校作为开设“程序设计”基础课程的教材，也可作为编程人员和参加计算机等级考试（C 语言模块）自学者的参考书。一般情况下，建议计划 78 学时，其中讲授 48 学时，实验 30 学时。书中标有*号的内容可根据教学要求进行取舍。

本书的第 1，2，3，6，9 章由张树粹编写，第 4，5，7，8 章由孟佳娜编写。在编写过程中，谭征、卢云宏、李玲、刘迎军等 4 位老师为本书进行了认真地核校。

在本书的编写过程中，参考了大量有关 C 语言程序设计的书籍和资料，编者在此对这些参考文献的作者表示感谢。任满杰和陈守孔教授也为本书提出了宝贵建议，并给予了热情地鼓励与支持，在此一并表示最诚挚的谢意!

本书配套电子教案及书中相关源程序均可从人民邮电出版社网站（www.ptpress.com.cn）下载区中下载。

由于水平有限，书中疏漏之处在所难免，请各位读者不吝指正。

编者

2005 年 10 月于烟台大学

目　　录

第 1 章　概　　述

本章主要介绍程序与程序设计的基本概念，算法与程序基本结构，C 语言的特点及发展史，C 语言字符集、基本词类和基本句类。另外，还介绍 C 语言标准输入/输出函数，以及使用 Turbo C 3.0 调试 C 程序的方法和步骤。

1.1　程序设计与高级语言

1.1.1　程序与程序设计

程序是使用计算机语言解决某个问题的方法和步骤的描述。计算机程序设计是在某一程序语言环境下，编写出能够使计算机理解并执行的程序代码。程序的特点是有始有终，每个步骤都能操作，所有步骤执行完后则对应问题得到解决。

例如，求两个整数和的方法和步骤如下：

第一步，获取两个整数 a 和 b；

第二步，计算 c=a+b；

第三步，输出 c；

第四步，结束。

接下来使用计算机语言编写程序，例如用 C 语言描述。

【例 1.1】 求两个数的和。程序清单如下：

```
#include <stdio.h>                          /*包含头文件*/
main()                                      /*主函数*/
{                                           /*主函数开始*/
   int a,b,c;                               /*定义 a,b,c 为整数*/
   scanf("%d,%d",&a,&b);                    /*给整型变量 a 和 b 赋值*/
   c=a+b;                                   /*计算 c=a+b */
   printf("sum=%d\n",c);                    /*输出计算结果 c */
}                                           /*程序结束*/
```

该程序包括了两部分，一是对变量 a,b,c 进行数据类型说明，二是确定了求两个数和的计算方法，最后将计算结果打印输出。

1.1.2　高级语言

在程序设计的发展过程中，出现了各种计算机语言，最早期是用二进制代码编写程序，称为“机器语言”，又称为“低级语言”。很快又出现了汇编语言，汇编语言是用符号来代表二进制代码，所以又称为“符号语言”。机器语言和汇编语言都是“面向机器的语言”，它们的特点是很难脱离硬件，难以记忆。

由于程序设计的关键是将问题解决的算法过程描述出来，所以，一种描述算法过程很

方便，同时脱离了机型要求，并且能够面向问题的计算机程序设计语言很快研制成功，这就是现在使用的“高级语言”。高级语言的特点是更加接近自然语言和数学语言，非常容易掌握和普及。

使用计算机语言编写的程序叫源程序。源程序不能在计算机上直接运行，计算机只能接受 0 和 1 组成的二进制程序（又称二进制机器指令），高级语言源程序必须通过系统软件将其翻译成二进制程序后才能执行。翻译程序有两种执行方式：一种是通过“解释程序”将源程序翻译一句执行一句，这种执行方式称为“解释执行”方式；另一种是通过“编译程序” 将源程序全部翻译成二进制程序后再执行，此种执行方式称为“编译执行”方式。大多数高级语言采用“编译执行”方式，C 语言就是其中之一。从源程序到在计算机上得到运行结果，其操作过程（以 C 语言为例）如图 1.1 所示。

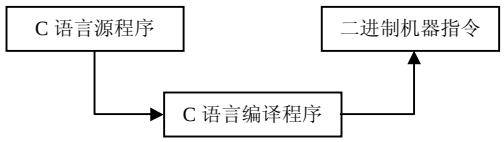

图 1.1　C 程序编译过程

1. 源程序的编辑

一个 C 源程序（又称 C 源文件）是一个编译单位，它是以文本格式保存的。源文件名用户自定，文件的扩展名（或后缀名）为“.c”。例如 myfile.c 是一个 C 源程序文件名。

2. 编译

源程序建立好，经检查无误后就可以进行编译。经过编译后，系统会自动生成二进制程序（.obj），称为“目标文件”。例如 myfile.c 源程序文件编译后生成 myfile.obj 文件。

3. 连接

源程序经过编译后所生成的目标文件（.obj）是相对独立的模块，但不能直接执行，用户必须用连接编辑器将它和其他目标文件以及系统所提供的库函数进行连接，生成可执行文件（.exe）才能执行。例如 myfile.c 源程序文件编译、连接后生成 myfile.exe 文件。

4. 执行

可执行文件生成后，可直接执行。若执行结果达到预想的结果，则说明程序编写正确，否则，修改源程序直到得出正确结果为止。

1.2　算　　法

学习计算机程序设计语言的目的，是要用语言作为工具，设计出计算机能够运行的程序。设计一个程序首先需要做两方面的工作，一是组织合理结构的数据，二是设计解决问题的算法。这样，就可用任何一种计算机语言编写程序，即计算机源程序。因此，人们用以下公式来表示程序：

程序 = 算法 + 数据结构

1.2.1　算法的特性

算法是指为了解决某个特定问题而采用的确定的方法和步骤。计算机算法可分为两大类别：数值运算和非数值运算。数值运算的目的是求数值解，例如，求方程的根、求圆的

面积、求 *n* 的阶乘等，都属于数值运算。非数值运算包括的面十分广泛，主要用于事务管理，例如，人事管理、图书管理、学籍管理等。算法具有以下 5 个特性。

1. 有穷性

算法应包含有限个操作步骤。也就是说，在执行若干个操作之后，算法将结束，而且每一步都在合理的时间内完成。

2. 确定性

算法中的每一条指令必须有确切的含义，不能有二义性，对于相同的输入必须能得到相同的结果。

3. 可行性

算法中的每一步都应当有效执行，通过基本运算后能够实现目标。

4. 有零个或多个输入

在计算机上实现算法，所需的处理数据要在程序执行时通过输入得到。也有的程序不需要输入数据。

5. 有一个或多个输出

算法的目的是求解（结果），结果要通过输出得到。

1.2.2　算法表示

算法可以用各种描述方法来进行描述，常用的有自然语言、伪代码、传统流程图和 N-S 流程图。使用流程图（又称框图）将算法描述出来，然后，根据流程图编写程序代码是计算机程序设计常常采用的方法。

1. 用伪代码表示算法

伪代码是用介于自然语言和计算机语言之间的文字和符号来描述算法。

【例 1.2】 用伪代码描述输出 x 的绝对值。

```
IF x is positive THEN
   print   x
ELSE
   print –x
```

也可以中英文混用，将上例改写成：

```
若 x 为正
   打印 x
否则
   打印 –x
```

2. 用传统流程图表示算法

传统流程图也是很好的描述算法的工具，流程图的图形符号如图 1.2 所示。

图 1.2　传统流程图的图形符号

其中，流程线包括 4 个方向线，即→←↓↑。

【例 1.3】 求两个整数之和的算法的流程图，如图 1.3 所示。

用传统流程图表示算法，形象直观，简单方便，但是这种流程图对于流程线走向没有任何限制，可以任意转向，在描述复杂的算法时容易混乱，不易阅读。

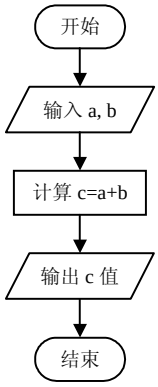

图 1.3 求两个整数之和的传统流程图

3. 用 N-S 流程图表示算法

随着结构化程序设计方法的出现，1973 年美国学者 I .Nassi 和 B.Shneiderman 提出了一种新的流程图形式。这种流程图去掉了流程线，算法的每一步都用一个矩形框来描述，一个完整的算法就是按用户设计的执行顺序连接起来的一个大矩形。人们把这种流程图称为 N-S 流程图，如图 1.4 所示。

例如，求两个整数之和的算法的 N-S 流程图如图 1.5 所示。

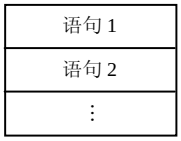

图 1.4 N-S 流程图

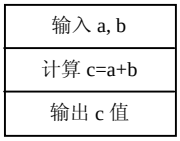

图 1.5 求两个整数之和的 N-S 流程图

1.3 C 语言的发展史与特点

1.3.1 C 语言的发展史

C 语言是目前国际上广泛流行的一种结构化的程序设计语言，计算机专业人员使用它来开发系统软件，软件开发人员使用它来编写应用软件，特别是近些年来，不仅是计算机专业人员和软件开发人员，而且广大计算机爱好者也越来越青睐 C 语言。

C 语言的前身是 ALGOL 语言。ALGOL 语言是 1960 年开发出的一种面向问题的高级语言，用 ALGOL 来描述算法很方便，但它的缺点是离计算机硬件系统比较远，不宜用来编写系统程序。1963 年英国剑桥大学在 ALGOL 语言基础上增添了处理硬件的能力，推出了 CPL（Combined Programming Language）语言。CPL 语言比 ALGOL 语言接近硬件一些，但由于规模较大，学习和掌握困难，没有流行开来。1967 年英国剑桥大学的 Dennis M.Ritchie 对 CPL 语言进行了简化，推出了 BCPL 语言。1970 年美国贝尔实验室的 Ken Thompson 对 BCPL 语言又做了进一步简化，设计出了既简单又接近硬件系统的 B 语言（以 BCPL 的第一个字母命名），同时，使用 B 语言编写出 UNIX 操作系统，在 PDP-7 上实现。1972 年美国贝尔实验室的 Dennis.M.Ritchie 在 B 语言的基础上设计出 C 语言（以 BCPL 的第二个字母命名）。C 语言在保留 BCPL 语言和 B 语言的强大的硬件处理功能的基础上，扩充了数据类型，恢复了通用性。1973 年 Dennis M.Ritchie 和 K.Thompson 两个人合作将 UNIX 操

作系统用C语言重写了一遍（即 UNIX V5）。系统的代码量比以前的版本大了三分之一，加进了多道程序设计功能，特别是整个 UNIX 操作系统（包括C编译程序本身）都建立在C语言的基础之上，而 UNIX V5 便奠定了 UNIX 操作系统的基础。随着 UNIX 的日益广泛使用，C 语言也得到迅速推广，可以说 C 语言和 UNIX 在发展过程中相辅相成，目前，C 语言早已成为世界上公认的应用最广泛的计算机语言。

1977 年 K.Thompson 和 Dennis M.Ritchie 撰写了《C 程序设计语言》一书，对 C 语言进行了规范化的描述，成为当时的标准，称为 K&R 标准。随着微型机的普及，出现了不同的 C 语言标准版本，为了统一标准，美国标准化协会（ANSI）于 1983 年制定了一套 ANSI C 标准，此后又相继推出 87 ANSI、C99 标准。

目前在计算机上使用的 C 编译程序有 Turbo C、Microsoft C 和 Quick C。本书将以 ANSI C 为标准，以“Turbo C 3.0”为编译程序介绍 C 语言的内容，讲述结构化程序的基本结构以及程序设计方法。本书中的例题均已在 Turbo C 3.0 的环境下调试通过。

1.3.2　C 语言的特点

1. 结构化语言

C 语言是结构化程序设计语言。每一类语言都有它的特点，结构化语言的一个显著特点是代码和数据的分离化，即程序的各部分除了必要的信息交流外，彼此互不影响，相互隔离。体现 C 语言的主要特点的是函数。

C语言的程序是由函数构成的，一个函数为一个“程序模块”。一个C源程序至少包含一个函数，就是 main 函数（主函数），也可以包含一个 main 函数和若干个其他函数（子函数）。所以说，函数是C程序的基本单位。同时,C 语言系统也提供了丰富的库函数（又称系统函数），用户可以在程序中直接引用相应的库函数，根据需要编制和设计用户自己的函数。所以说，一个 C 程序由用户自己设计的函数（以下简称用户函数）和库函数两部分构成。

2. 简洁、紧凑、灵活

Turbo C 只有 32 个保留字（关键字），9 种控制语句，程序书写自由，主要以小写字母表示，C编译程序的体积很小。另外，C是一种自由格式的语言，没有像 FORTRAN 语言那样的书写格式的限制，故用C语言书写程序可以随心所欲，自由方便。

3. 运算符丰富

C语言的运算符种类很多，共有 34 种运算符（见附录 2）。C 语言可以进行字符、数字、地址、位等运算，并可完成通常由硬件实现的普通算术运算、逻辑运算。灵活使用各种运算符可以完成许多在其他高级语言中难以实现的运算或操作。

4. 中级语言

我们通常称面向问题的语言为“高级语言”，而面向机器的语言为“低级语言”，C语言既具有高级语言的功能，又具有低级语言的许多功能。C语言能够对内存单元中的二进制位（bit）操作，实现汇编语言的大部分功能，直接对硬件进行操作。由于C语言的这种双重性，使它既是成功的系统描述语言，又是通用的程序设计语言，所以有人称它为中级语言。

5. 移植性好

可移植性是指程序可以从一个环境不加或稍加改动就可移到另一个完全不同的环境上

运行。对汇编语言而言，由于它只面向特定的机器，所以根本不可移植。而一些高级语言（比如 FORTRAN）其编译程序也不可移植，而只能根据国际标准重新实现。但 C 语言在许多机器上的实现是通过将 C 编译程序移植得到的。据统计，不同机器上的 C 编译程序 80%的代码是共同的。

6. 功能强大

高级语言是否适用于编写系统软件，除了语言表达能力之外，还有一个很大的因素是该语言的代码质量。如果代码质量低，则系统开销就会增大。一般说来，语言越低级其代码质量就越高，但编程的工作量也越大。由于 C 语言具有低级语言的一些功能，所以，现在许多系统软件都用 C 语言来描述，从而大大提高了编程效率。

7. 编译语言

C 语言是编译语言，用 C 语言编写的源程序必须经过编译后（生成.obj 文件），再与库文件连接生成可执行文件（.exe 文件），执行可执行文件的过程称为运行程序。

8. 语法限制不严格，程序设计自由度大

用 C 语言所编写的程序的正确性和合法性在很大程度上要由程序员而不是由 C 编译程序来保证。例如，C 编译程序对数组下标不做越界检查，数据类型检验功能较弱且转换比较随便等。故对 C 语言不熟悉的人员，编写一个正确的 C 程序可能要比编写其他高级语言程序难一些。所以用 C 语言编写程序，要对程序进行认真的检查，而不要过分依赖 C 编译程序的查错功能。正是因为 C 语言放宽了语法的限制，所以换来了程序设计的较大自由度和灵活性。使用得当，便会体会到这并不是 C 语言的缺点，而恰恰是其一大优点。

1.4 C 程序结构及书写规则

1.4.1 C 程序的基本结构

C 程序是由一个主函数和若干个（或 0 个）用户函数组成的，主函数和用户函数的位置是任意的。但它们的调用关系是一定的，即主函数可以调用任何用户函数，用户函数间可以互相调用，但不能调用主函数。用户函数甚至可以调用自己，这种调用称为递归调用。

C 程序总是从 main()函数开始执行，而不论 main()函数在整个程序中的位置如何。从主函数的第一条语句开始执行，直到主函数的最后一条语句结束。非主函数要通过“函数调用”得到执行。

一个函数由两部分组成，即函数说明部分和函数体部分。

1. 函数说明部分（又称函数头）

函数说明部分包括函数返回值类型、函数存储类型、函数标识符、函数的参数和参数的类型，其格式是函数名后紧接着一对圆括号。

2. 函数体部分

在函数中用大括号括起来的部分是函数体部分。函数体内有若干条语句，它们能完成各种操作，具体函数的功能都写在函数体中。C 语言允许函数体内为空。

1.4.2　程序的书写规则

C 程序书写格式随意，除了保留字外，任何地方都可以插入空格、回车换行符。

为了便于阅读程序，建议采用格式化的书写格式，采用缩格纵向对齐方式。可以在程序的任何一处插入“注释”。注释是以“/*”开始，到“*/”结束的任意字符序列。注释语句是非执行语句，不参加编译，也不会出现在目标文件中，只起到帮助阅读程序的作用，如同读文章加上注释一样。

1.5　C 语言的基本词法

学习自然语言要学习词汇和语法规则，根据语法规则使用词汇书写句子或文章。C 语言是一种计算机语言，用计算机语言编写程序就是使用其基本字符、基本词类根据它的语法规则，按照算法描述出解决问题的方法和步骤。

1.5.1　字符集

C 程序允许出现的所有基本字符的组合称为 C 语言的字符集，C 语言的字符集就是 ASCII 字符集，主要分为下列 4 类。

1. 大小写英文字母

   A，B，C，…，Z，a，b，c，…，z。

2. 数字

   0，1，2，3，4，5，6，7，8，9。

3. 键盘符号

如表 1.1 所示。

表 1.1　　键盘符号

符　号	含　义	符　号	含　义	符　号	含　义
~	波浪号	）	右圆括号	：	冒号
‘	重音号	?	下划线	；	分号
!	叹号	?	减号	"	双引号
@	a 圈号	+	加号	’	单引号
#	井号	=	等号	<	小于号
$	美元号	|	或符号	>	大于号
%	百分号	\	反斜杠	，	逗号
^	异或号	{	左花括号	.	小数点
&	与符号	}	右花括号	?	问号
*	星号	[	左方括号	/	（正）斜杠
(	左圆括号	]	右方括号		空格符号

注意：有些运算符是由两个字符共同构成的。如＆＆，||，<=，>=，==，<<，>>，!=，++，??等，在 C 程序中应将它们看成一个整体，而不要当成两个字符来对待（见附录 2）。