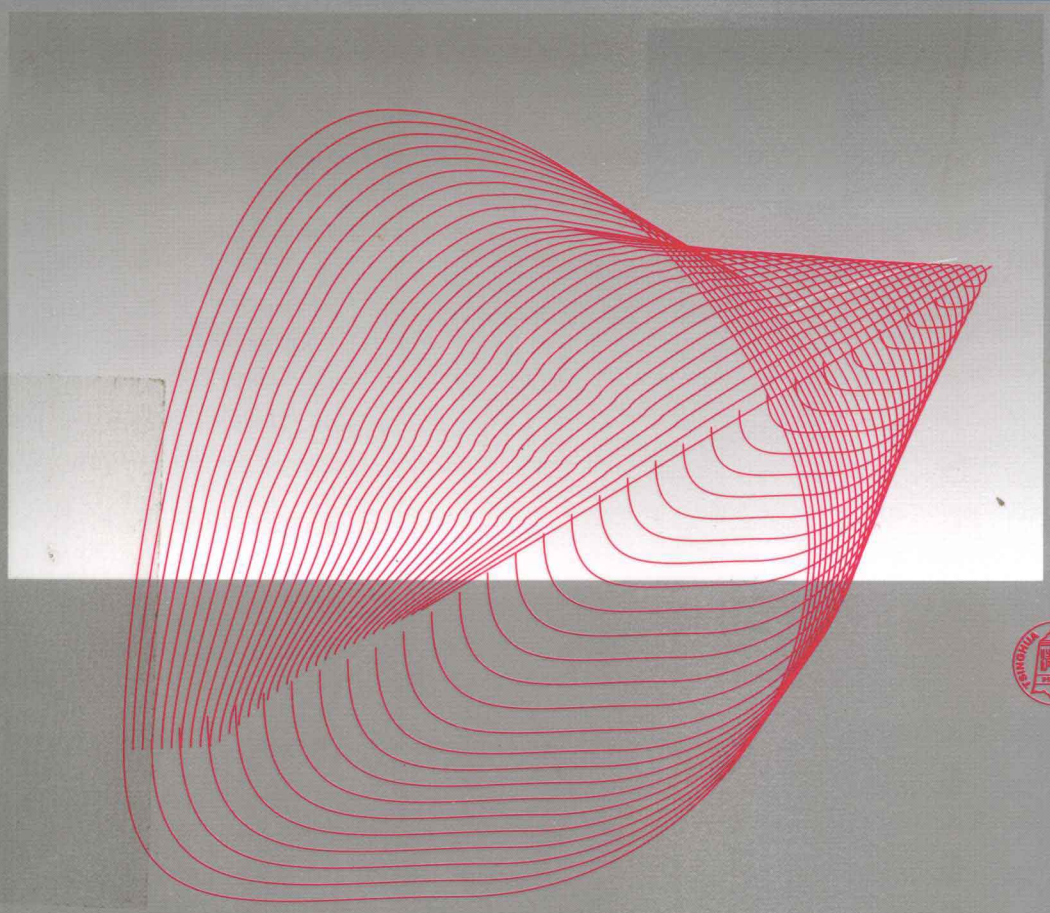
21 世纪高等学校计算机教育实用规划教材

C语言程序设计实用教程

周屹 李建勋 主编
詹晓娟 张谢群 姚登举 副主编
李朴 主审



清华大学出版社

21 世纪高等学校计算机教育实用规划教材

C语言程序设计实用教程

周 屹 李建勋 主编
詹晓娟 张谢群 姚登举 副主编

清华大学出版社
北京

内 容 简 介

本书从实用出发，详细地介绍了 C 语言概述、数据类型、运算符与表达式、基本控制结构、数组、函数、编译预处理、指针、结构体和其他类型、文件等程序设计内容。本书系统全面，层次清晰，例题丰富，实用性强，面向应用，注重培养应用技能和编程能力，每章都配有小结、习题与实训部分，便于教学组织和实践操作。

本书是为高等院校第一门程序设计课程而编写的教材，适合作为普通高等本科院校理工类专业学生的程序设计教材，也可以作为高职等不同层次学生学习计算机语言的入门教材。

图书在版编目（CIP）数据

C 语言程序设计实用教程/周屹等主编. —北京：清华大学出版社，2012.6
（21 世纪高等学校计算机教育实用规划教材）
ISBN 978-7-302-28478-9

Ⅰ. ①C…　Ⅱ. ①周…　Ⅲ. ①C 语言-程序设计-高等学校-教材　Ⅳ. ①TP312

中国版本图书馆 CIP 数据核字（2012）第 064607 号

责任编辑：闫红梅　赵晓宁
封面设计：傅瑞学
责任校对：梁　毅
责任印制：何　芊

出版发行：清华大学出版社
网　　址：http://www.tup.com.cn，http://www.wqbook.com
地　　址：北京清华大学学研大厦 A 座　　邮　　编：100084
社 总 机：010-62770175　　邮　　购：010-62786544
投稿与读者服务：010-62776969，c-service@tup.tsinghua.edu.cn
质 量 反 馈：010-62772015，zhiliang@tup.tsinghua.edu.cn
印 刷 者：北京季蜂印刷有限公司
装 订 者：三河市溧源装订厂
经　　销：全国新华书店
开　　本：185mm×260mm　　**印　　张**：21.75　　**字　　数**：537 千字
版　　次：2012 年 6 月第 1 版　　**印　　次**：2012 年 6 月第1 次印刷
印　　数：1～3000
定　　价：33.00 元

产品编号：039601-01

出版说明

随着我国高等教育规模的扩大以及产业结构调整的进一步完善，社会对高层次应用型人才的需求将更加迫切。各地高校紧密结合地方经济建设发展需要，科学运用市场调节机制，合理调整和配置教育资源，在改革和改造传统学科专业的基础上，加强工程型和应用型学科专业建设，积极设置主要面向地方支柱产业、高新技术产业、服务业的工程型和应用型学科专业，积极为地方经济建设输送各类应用型人才。各高校加大了使用信息科学等现代科学技术提升、改造传统学科专业的力度，从而实现传统学科专业向工程型和应用型学科专业的发展与转变。在发挥传统学科专业师资力量强、办学经验丰富、教学资源充裕等优势的同时，不断更新教学内容、改革课程体系，使工程型和应用型学科专业教育与经济建设相适应。计算机课程教学在从传统学科向工程型和应用型学科转变中起着至关重要的作用，工程型和应用型学科专业中的计算机课程设置、内容体系和教学手段及方法等也具有不同于传统学科的鲜明特点。

为了配合高校工程型和应用型学科专业的建设和发展，急需出版一批内容新、体系新、方法新、手段新的高水平计算机课程教材。目前，工程型和应用型学科专业计算机课程教材的建设工作仍滞后于教学改革的实践，如现有的计算机教材中有不少内容陈旧（依然用传统专业计算机教材代替工程型和应用型学科专业教材），重理论、轻实践，不能满足新的教学计划、课程设置的需要；一些课程的教材可供选择的品种太少；一些基础课的教材虽然品种较多，但低水平重复严重；有些教材内容庞杂，书越编越厚；专业课教材、教学辅助教材及教学参考书短缺，等等，都不利于学生能力的提高和素质的培养。为此，在教育部相关教学指导委员会专家的指导和建议下，清华大学出版社组织出版本系列教材，以满足工程型和应用型学科专业计算机课程教学的需要。本系列教材在规划过程中体现了如下一些基本原则和特点。

（1）面向工程型与应用型学科专业，强调计算机在各专业中的应用。教材内容坚持基本理论适度，反映基本理论和原理的综合应用，强调实践和应用环节。

（2）反映教学需要，促进教学发展。教材规划以新的工程型和应用型专业目录为依据。教材要适应多样化的教学需要，正确把握教学内容和课程体系的改革方向，在选择教材内容和编写体系时注意体现素质教育、创新能力与实践能力的培养，为学生知识、能力、素质协调发展创造条件。

（3）实施精品战略，突出重点，保证质量。规划教材建设仍然把重点放在公共基础课和专业基础课的教材建设上；特别注意选择并安排一部分原来基础比较好的优秀教材或讲义修订再版，逐步形成精品教材；提倡并鼓励编写体现工程型和应用型专业教学内容和课程体系改革成果的教材。

（4）主张一纲多本，合理配套。基础课和专业基础课教材要配套，同一门课程可以有多本具有不同内容特点的教材。处理好教材统一性与多样化，基本教材与辅助教材，教学参考书，文字教材与软件教材的关系，实现教材系列资源配套。

（5）依靠专家，择优选用。在制订教材规划时要依靠各课程专家在调查研究本课程教材建设现状的基础上提出规划选题。在落实主编人选时，要引入竞争机制，通过申报、评审确定主编。书稿完成后要认真实行审稿程序，确保出书质量。

繁荣教材出版事业，提高教材质量的关键是教师。建立一支高水平的以老带新的教材编写队伍才能保证教材的编写质量和建设力度，希望有志于教材建设的教师能够加入到我们的编写队伍中来。

21 世纪高等学校计算机教育实用规划教材编委会
联系人：魏江江 weijj@tup.tsinghua.edu.cn

前言

C语言是目前流行的通用程序设计语言，之所以成为许多计算机专业人员和程序爱好者学习程序设计的首选入门语言，除了C语言的众多优点外，最主要的还是C语言的实用性。

在教授C语言的过程中，不但要注重知识的讲授，更要强调基本技能的训练；不仅要让学生学习程序设计的基本概念和方法，掌握编程技术，更重要的是培养学生分析和解决问题的能力以及灵活应用编程思想和方法的能力。本书在详细阐述C语言基础知识的基础上，着重讨论了程序设计的基本原理、概念和方法，穿插演示性案例于理论讲解中，使理论变得更容易理解、容易接受。此外，还在每章末尾安排了实训内容，目的是提高学生综合利用所学知识解决实际问题的能力。

本书紧扣教学大纲，密切结合实际，以应用为目的，结构清晰，内容系统翔实，布局合理，加强基本概念和基本分析方法的介绍，尽量用实例解释概念，用案例演绎方法和原理，叙述采用启发式，逻辑性强，文字流畅、通俗易懂。便于教学的展开，也便于学生学习，是学习C语言的理想教材。

本书由长期承担程序设计基础课教学，具有丰富教学经验的一线教师编写，根据作者多年从事程序设计课程教学活动以及软件开发的经验，针对多数学生的认知规律，在详细介绍、解析C语言语法知识的同时，教材内容充分考虑到知识和技能的结合，由浅入深、循序渐进，把枯燥的概念、语法、算法融汇在生动、有趣的案例中，以调动学生的学习积极性。本书还配有多媒体电子教案、PPT等教学辅助课件。本书程序较多，所涉及的变量均用正体表示。

本书由周屹、李建勋主编，詹晓娟、张谢群、姚登举副主编，李朴主审。第1和第3章由周屹编写，第2和第4章由詹晓娟编写，第5章由姚登举编写，第6章由运海红编写，第7章由张谢群编写，第8章和附录由王娜和钟玉峰编写，第9章由王丁编写，第10章由李建勋编写，全书由周屹统稿。

本书在编写过程中，得到了各方面有关专家的大力支持和帮助，在此对所有人的工作与支持表示衷心的感谢。由于编者水平有限，书中难免存在不足，敬请广大读者批评指正。编者E-mail地址是zhouyi_1@163.com。

编　者

2012年2月

目 录

第1章 C语言概述

1.1 C语言历史及其特点

C语言是在B语言的基础上发展起来的，它的根源可以追溯到ALGOL 60（Algorithmic Language 60）。1960年出现的ALGOL 60是一种面向问题的高级语言，离硬件比较远，不宜用来编写系统程序。1963年，英国的剑桥大学推出了CPL（Combined Programming Language）语言。CPL语言在ALGOL 60的基础上接近硬件一些，但规模比较大，难以实现。

1967年，英国剑桥大学的Matin Richards对CPL语言作了简化，推出了BCPL（Basic Combined Programming Language）语言。1970年，美国贝尔实验室的Ken Thompson以BCPL语言为基础，又作了进一步简化，使得BCPL能挤压在8KB内存中运行，这个很简单的而且很接近硬件的语言就是B语言（取BCPL的第一个字母），并用它写了第一个UNIX操作系统，在DEC PDP-7上实现。1971年，在PDP-11/20上实现了B语言，并写了UNIX操作系统。但B语言过于简单，功能有限，并且和BCPL都是“无类型”的语言。

1972—1973年，贝尔实验室的D. M. Ritchie在B语言的基础上设计出了C语言（取BCPL的第二个字母）。C语言既保持了BCPL和B语言的优点（精练，接近硬件），又克服了它们的缺点（过于简单，数据无类型等）。

1973年，C语言已经可以用来编写UNIX操作系统的内核，这是第一次用C语言来编写操作系统的内核。Dennis Ritchie和Brian Kernighan在1978年出版了《C程序设计语言》（The C Programming Language，经常简称为“白皮书”或K&R）。

1980年以后，贝尔实验室使得C语言变得更为广泛地流行，C语言一度成为操作系统和应用程序编程的首选。甚至到今天，C语言仍被用于编写操作系统以及作为广泛的计算机教育的语言。

1983年，美国国家标准委员会（ANSI）对C语言进行了标准化，于1983年颁布了第一个C语言标准草案（83 ANSI C），后来于1987年又颁布了另一个C语言标准草案（87 ANSI C）。最新的C语言标准是在1999年颁布并在2000年3月被ANSI采用的C99。20世纪80年代晚期，布贾尼·斯特劳斯特卢普（Bjarne Strou-strup）和贝尔实验室为C语言添加了面向对象的特性，C语言扩展出C++语言。目前C++现在广泛应用于Microsoft Windows下运行的商业应用程序的编制，然而C语言仍然是UNIX世界的热门编程语言。

由于C语言具有严格的设计，与具体的硬件无关及其他许多优点，使它很快就超越了贝尔实验室的范围，迅速地在全球传播。C语言之所以备受青睐，是和它具有的许多优点分不开的。这些优点主要有：

（1）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。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。 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。

（2）运算符丰富。C 的运算符包含的范围很广泛，共有 13 种 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。

（3）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，用来实现各种复杂的数据类型的运算，并引入了指针概念，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。

（4）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，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。

（5）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。

（6）C 语言允许直接访问物理地址，可以直接对硬件进行操作。 因此既具有高级语言的功能，又具有低级语言的许多功能，能够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，可以用来编写系统软件。

（7）C 语言程序生成代码质量高，程序执行效率高。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。

正因为 C 语言具有诸多优点，它才能够风靡全球，成为世界上应用最广泛的几种计算机语言之一。它不仅可用于编写系统软件，而且也可以用于编写应用软件。

1.2 简单 C 程序设计示例

C 语言所用的表述方式对没有编写过计算机程序的人来说可能是陌生的，因此通过一些简单的例子来初识 C 语言。

【例 1-1】 在屏幕上显示“Hello，C!”。

```
#include<stdio.h>
void main()                    /*主函数*/
{
printf("Hello, C!\n");         /*输出语句*/
}
```

main 是主函数的函数名，表示这是一个主函数。每一个 C 源程序都必须有且只能有一个主函数。一对大括号“{}”是主函数的定界符，在主函数中只有一个语句；在该语句中调用了格式输出库函数 printf，用于向屏幕上输出一个字符串。函数 printf 的功能是把要输出的内容送到标准输出设备上，默认的标准输出设备是显示器。由“/*”和“*/”括起来

的任何文字是注释，程序不执行注释部分。语句用分号结束，一行可以写多个语句，关键字用小写字母，书写采用自由格式。

主函数体分为两部分，一部分为说明部分；另一部分为执行部分。说明是指变量的类型说明。例 1-1 中未使用任何变量，因此无说明部分。

C 语言规定，源程序中所有用到的变量都必须“先说明，后使用”，否则会出错。这一点是编译型高级程序设计语言的一个特点，说明部分是 C 源程序结构中很重要的组成部分。

【例 1-2】 输出两个变量的和。

```
#include<stdio.h>
void main()
{
    int x,y;                          /*变量定义语句：定义 2 个整型变量 x、y*/
    x=3;                              /*可执行的赋值语句：将 3 赋值给变量 x*/
    y=6;                              /*可执行的赋值语句：将 6 赋值给变量 y*/
    printf("%d\n",x+y);
}
```

本例中使用了两个变量 x、y，用类型说明符 int 来说明这两个变量。说明部分后的三行为执行部分或称为执行语句部分，用来完成程序的功能。执行部分的第一、二行是赋值语句，printf 函数在显示器上输出表达式 x+y 的值，至此程序结束。

【例 1-3】 利用函数实现输出最大值。

```
#include<stdio.h>
void main()                           /*主函数*/
{
    int x,y,z;                        /*变量说明*/
    int max(int a,int b);             /*函数说明*/
    printf("input two numbers: \n");
    scanf("%d%d",&x,&y);              /*输入 x,y 值*/
    z=max(x,y);                       /*调用 max 函数*/
    printf("max mumber=%d",z);        /*输出*/
}
int max(int a,int b)                  /*定义 max 函数*/
{
    if(a>b)
        return a;
    else
        return b;                     /*将结果返回主调函数*/
}
```

例 1-3 程序中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 main 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第 5 章中介绍。

例 1-3 程序的执行过程是，首先在屏幕上显示提示串，请用户输入两个数，按 Enter 键后由 scanf 函数接收这两个数并送入变量 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。

1.3 C 程序的结构特点和书写格式

1.3.1 C 语言程序的结构特点

通过上面 3 个简单的 C 语言程序，可以看出 C 语言程序的基本结构具有以下几个特点。

（1）C 语言程序为函数模块结构，所有的 C 语言程序都是由一个或多个函数构成的，其中 main 函数必须有且只能有一个。函数是 C 语言程序的基本单位。

（2）C 语言程序总是从主函数开始执行，当执行到调用函数的语句时，程序将控制转移到被调函数中执行，执行结束后，再返回到调用函数继续执行，直到程序执行结束为止。

（3）C 语言程序的函数包括编译系统提供的标准函数（如 printf()、scanf()等）和由用户自己定义的函数。

（4）源程序中的预处理命令通常放在源文件或源程序的最前面。

（5）每一个说明和每一个语句都必须以分号结尾。但是预处理命令、函数头和函数体的定界符“{”和“}”之后不能加分号。

（6）标识符、关键字之间必须至少加一个空格以示分隔。若已有明显的分隔符，也可以不再加空格。

（7）可以在程序的任何位置用“/*注释内容*/或“//注释内容”的形式对程序或语句进行注释。

1.3.2 C 程序的书写格式

C 语言程序的书写格式非常自由，但从书写清晰，便于阅读、理解、维护的角度出发，建议在书写 C 语言程序时遵循以下几个规则。

（1）通常一个说明或一个语句占一行，每个语句由分号结束；一个语句可以分行写在多行上，一行内也可以写几个语句。

（2）可以在函数与函数之间加空行，以清楚地分出程序中有几个函数。

（3）用{}括起来的部分，通常表示程序的某一层次结构。{}一般与该结构语句的第一个字母空两格，并单独占一行。

（4）低一层次的语句比高一层次的语句向后缩进若干空格书写，同一个层次的语句左对齐，以便看起来更加清晰，增加程序的可读性。

（5）对于数据的输入，运行时最好要出现输入提示；对于数据输出，也要有一定的提示和格式。

（6）在程序中可以用 / *……*/加上必要的注释，以增加程序的可读性。

在编程时应力求遵循上述规则，以养成良好的编程习惯。

1.4 C 程序的开发过程

用 C 语言编写的程序称为源程序（Source Program），计算机本身并不能直接理解这样的语言，需要经过解释程序或编译程序将其翻译成机器语言，计算机才能理解并运行程序。将源程序翻译成机器语言的过程称为编译，编译的结果是得到源程序的目标代码（Object Program），最后还要将目标代码与系统提供的函数和自定义的过程（或函数）链接起来，就可得到机器可执行的程序。计算机可执行的程序称为可执行程序或执行文件。

1.4.1 源程序翻译

C 语言源程序的后缀名为.c。它是不能直接在计算机上运行的，必须通过机器翻译成目标代码，再将目标代码链接成可加载模块（可执行文件），才能在计算机上运行。这种把源程序翻译成目标代码的程序称为编译器或翻译器。适合 C 语言的编译器还不止一种，不同的机器、不同的操作系统可能会有 1 种或多种不同的编译器。C 语言源程序的翻译过程如图 1-1 所示，由词法分析器、语法分析器和代码生成器 3 部分组成。

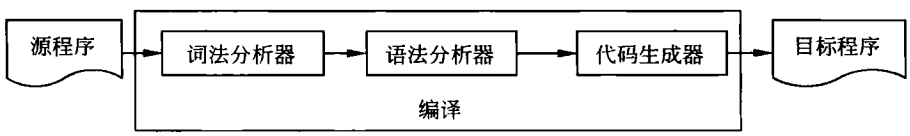


图 1-1 程序语言的翻译过程

1．词法分析器

词法分析器（Lexical Analyzer）主要是对源程序进行词法分析，是按单个字符的方式阅读源程序，并且识别哪些符号的组合可以代表单一的单元，并根据它们是否是数字值、单词（标识符）、运算符等，将这些单词分类。词法分析器将词法分析结果保存在一个结构单元里，这个结构单元称为记号（Token），并将这个记号交给语法分析器，词法分析会忽略源程序中的所有注释。

2．语法分析器

语法分析器（Parser）直接对记号进行分析，并识别每个成分所扮演的角色。这些语法规则也就是程序设计语言的语法规则。

3．代码生成器

代码生成器（Code Generator）将经过语法分析后没有语法错误的程序指令转换成机器语言指令。

例如，假定编写了一个名为 mytest 的程序，源程序的全名为 mytest.c，用 microsoft C 编译器，在命令方式下，可采用下面这样的方式对 mytest.c 进行编译：

```
cl  -c mytest.c
```

如果源程序没有错误，就会生成一个名为 mytest.obj 目标代码程序。其他程序语言也会有类似的命令将源程序翻译成目标代码，具体的命令与每种程序语言的编译器有关。