21世纪高等学校计算机教育实用规划教材

邹姝稚 主 编
陈 斌 副主编
徐向英 王丽爱 编著

C语言程序设计

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

邹姝稚 主 编
陈 斌 副主编
徐向英 王丽爱 编著

C语言程序设计

清华大学出版社
北京

内 容 简 介

本书共 11 章。分别介绍 C 程序的构成与 C 语言的特点；基本数据类型、运算符及表达式以及各类算符的优先级、结合性与求值规则；结构化程序设计的三种方法，即顺序结构 C 程序的开发方法、分支结构 C 程序的开发方法和循环结构 C 程序的开发方法；数组的概念、相关语法及其数组的应用；函数调用技术及其变量存储类别；编译预处理技术；指针技术在程序开发中的应用；结构体、共用体和枚举类型的数据结构，并讨论了链表技术；C 的数据文件的概念及其与文件相关的库函数的应用。

本书凝聚了编者近 30 年 C 语言教学经验。本书的体系结构和内容组织，具有理论适度、内容完整、重点突出、概念清楚、深入浅出、通俗易懂等特点。本书还具有两大特色，一是首创了语法图的讲解方法，能帮助读者更直观、准确地理解各种 C 语法；二是在指针一章提出了一组关于各类指针的通式，将极大降低指针技术的学习难度。

本书可作为本科院校、高职高专相关专业的教材，也可供准备参加 C 语言等级考试、资格和水平考试的读者阅读参考，同时也可作为工程技术人员和计算机爱好者的参考资料。

图书在版编目（CIP）数据

C 语言程序设计 / 邹姝稚主编. --北京：清华大学出版社，2013.2
21 世纪高等学校计算机教育实用规划教材
ISBN 978-7-302-29743-7

Ⅰ. ①C…　Ⅱ. ①邹…　Ⅲ. ①C 语言－程序设计　Ⅳ. ①TP312

中国版本图书馆 CIP 数据核字（2012）第 189171 号

责任编辑：魏江江　赵晓宁
封面设计：常雪影
责任校对：时翠兰
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
　　课 件 下 载：http://www.tup.com.cn，010-62795954
印 刷 者：北京季蜂印刷有限公司
装 订 者：三河市李旗庄少明印装厂
经　　销：全国新华书店
开　　本：185mm×260mm　　**印　张**：25　　**字　　数**：609 千字
版　　次：2013 年 2 月第 1 版　　**印　　次**：2013 年 2 月第1 次印刷
印　　数：1～3000
定　　价：39.50 元

产品编号：041656-01

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，不断更新其教学内容、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，不能满足按新的教学计划、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 语言是一种长期并广泛使用的结构化程序设计语言，兼具高级语言的特点和低级语言的功能，是计算机应用人员、系统编程人员、硬件产品开发和底层程序开发人员应该掌握的程序设计工具之一。

C 语言程序设计是理工科各专业计算机应用能力培养的重要技术基础课程，不仅开设于计算机相关专业，在许多普通高等学校、高职院校的非计算机专业也普遍开设了 C 语言课程。此外，全国计算机等级考试、省级计算机等级考试、全国计算机应用技术证书考试（NIT）等也都将 C 语言列入考试科目。

由于 C 语言语法现象众多，功能丰富，使用灵活，使不少初学者感到难以入门、学习困难。鉴于此，笔者将多年从事 C 语言教学的经验及有关教学研究成果融入书中，力求以合理的编写体系和教材内容，配合丰富的实例帮助读者学习 C 语言。

本书具有如下特点：

（1）面向多层次、多学科专业，体系结构合理，强调实用性。本书在章节编排上充分考虑初学者的特点和认知规律，力求层次分明、循序渐进，既考虑内容的完整性，又精心安排各部分的前后关系，以期分散难点，降低学习难度。在教材内容组织上坚持基本理论适度、重点突出。文字表述上力求深入浅出、通俗易懂。能满足普通高等学校及高职院校等各层次对 C 语言的教学需求。

（2）首创语法图的讲解方法。C 语言语法概念众多，为提高语法讲解的直观性，同时也避免自然语言可能造成的歧义。本书对较为复杂或烦琐的语法均采用了语法图的讲解方法，使语法表述更准确、更全面。将语法图引入 C 语言的讲解，这也是本书的一大特色。

（3）提出了关于指针规则的一组通式。指针是 C 语言的重要特色，使用指针编程也是 C 语言的主要风格，同时，指针的学习历来是 C 语言的学习难点。本书主编集多年 C 语言教学及研究心得，将提出的一组关于指针的通式引入本书，作为对各种指针规则的提炼和总结，将帮助读者对指针的学习。

（4）用例题贯穿整个教学过程，将理论和实践有机结合。C 语言是一门实践性很强的课程，对编程能力和调试能力的训练都非常重要。本书结合实例阐述理论，通过大量精心编写的例题，将正确的语法规范介绍给读者。此外，每章都配有大量用以练习的课后习题，从不同角度同步训练读者的程序阅读、程序完善和程序开发能力。

（5）严格介绍标准 C 的语法。本书中所有例题及课后习题均在 Turbo C 2.0 平台上编译通过。在本书的第 1 章及相关附录中较为详细地介绍了 Turbo C 2.0 的菜单设置及其使用方法。虽然有许多其他编译平台兼容了 C 及 C++，但由于标准 C 所开发的程序一定能在 C++

环境中运行，但在C++环境下运行的C程序不一定能在Turbo C 2.0平台上通过。因此，本书及其后继配套的实验教材，都要求所有程序使用Turbo C 2.0编译通过。

（6）注重教材的立体化配套。除提供教材这一基本形式外，还同步提供电子课件。后继将配套出版习题解答和上机实验指导，以方便教学。

本书讲课时数为48～64学时，另设实验课32学时。学习本教材后，应安排为期一周的课程设计，以独立完成一个小型应用系统的设计与开发。

本书由邹姝稚任主编，编写第1、第2和第9章，陈斌任副主编，编写第6～第8章，徐向英编写第3～第5章，王丽爱编写第10和第11章。附录由邹姝稚和徐向英共同编写。全书由邹姝稚统稿。

由于编者水平有限，书中难免存在缺点和错误，恳请专家和读者批评指正。

编　者

2012年6月

目 录

第1章 C语言概述

C语言属于计算机高级语言。C语言的出现是基于为UNIX操作系统提供开发工具的需求。问世后的C语言由于其简洁紧凑、表达能力强、目标代码质量高、可移植性好等诸多优点，成为使用最广泛的程序设计语言。其兼具高级语言的优点和低级语言的许多特点，不仅可用于编写系统软件，而且也可以用于编写应用软件，被美誉为“高级语言中的低级语言”或“超高级语言”。

本章主要介绍C语言的起源和发展、C语言的特点、C程序的结构和C程序设计必须遵循的规范，最后简要介绍C程序的上机操作步骤。通过本章学习，使读者对C语言程序有一定了解，为后继编写和调试C程序打下基础。

1.1 C语言出现的历史背景

C语言的产生和发展颇为有趣，和UNIX操作系统以及美国贝尔实验室有着密切的关系。

C语言的起源可以追溯到ALGOL 60。1960年出现的ALGOL 60对其后高级语言的发展起到了很好的推进作用，但是，它是一种面向问题的语言，过于抽象，难以描述系统，因此没有得到真正的推广。当然也不适宜用作为操作系统的开发工具。

1963年，剑桥大学将ALGOL 60语言发展成为CPL（Combined Programming Language）语言，但过大的规模使CPL语言难以实现。

1967年英国剑桥大学的Matin Richards对CPL语言进行了简化，于是出现了CPL的简化版BCPL（Basic Combined Programming Language）语言。

1970年，贝尔实验室的Ken Thompson为了寻求一种理想的操作系统开发工具，以BCPL语言为基础，对其进行了进一步简化，设计出了很简单且接近硬件的一种语言，并为它起了一个有趣的名字“B语言”。意思是将CPL语言煮干（Boil），提炼出它的精华，同时也考虑该语言源于BCPL语言，故取BCPL语言的第一个字母加以命名。并且他用B语言编写了第一个高级语言版本的UNIX操作系统，在DEC PDP-7型计算机上实现。

而在1972年，B语言也给“煮”了一下，美国贝尔实验室的Dennis M.Ritchie在保持B语言精练性、可操作硬件等优点的同时，对其对于简单的功能进行加强，在B语言的基础上设计出了一种全新的语言，并取了BCPL的第二个字母作为这种语言的名字，这就是C语言。

其后，Ken Thompson和Dennis M.Ritchie合作，使用C语言改写了UNIX操作系统90%以上的代码，形成了UNIX操作系统的第一个C语言版本，即UNIX第5版。此后,C语言经过多次改进，不过它主要还是用在贝尔实验室内部。直至1975年用C语言编写的UNIX 6公布后，C语言和UNIX操作系统引起了业内人士的广泛关注。不仅使UNIX在PDP系

列机以外的诸如 VAX、AT&T 等各种机型上得到开发，也使 C 语言独立于 UNIX 操作系统成为一种既用于系统开发又可用于编写应用软件的高级语言。在整个20世纪70年代，UNIX 操作系统和 C 语言相辅相成，获得迅速发展。

1978 年，以发表的 UNIX 7 中的 C 编译程序为基础，Brian W.Kernighan 和 Dennis M.Ritchie（合称 K&R）合著了影响深远的名著《The C Programming Language》，这本书中介绍的 C 语言成为后来广泛使用的 C 语言版本的基础，称它为标准 C。

以后相继出现了许多 C 语言版本。由于没有统一标准，使得这些 C 语言之间出现了一些不一致的地方。为了改变这种情况，1983 年美国国家标准化协会（ANSI）根据 C 语言问世以来的各种版本对 C 的发展和扩充，为 C 语言制订了第一个 ANSI 标准，称为 ANSI C。1987 年 ANSI 又公布了新的 C 语言标准，称为 87 ANSI C。这个标准在 1990 年被国际标准化组织（ISO）接受为 ISO C 的标准（ISO9899-1990），成为现行的 C 语言标准。目前流行的各个 C 编译系统均以此标准为基础。

本书的叙述基本上以 ANSI C 为基础。尽管这样，各种版本的 C 编译系统还是略有差异。诸如在微机上广为使用的 Microsoft C、Turbo C、Quick C、BORLAND C 等，虽然这些版本的基本部分是相同的，但在某些细节或库函数调用等方面还是有一些不同。因此，读者在使用具体的 C 语言编译系统时，还应参考相关手册以了解具体的规定。

1.2 C 程序的结构

用 C 语言编写的程序称为 C 语言源程序，简称为 C 程序。下面通过 C 程序的实例来认识和了解 C 程序的语法结构及其特点。

【例 1-1】 编写一个简单的 C 程序，输出显示如下字符串。

```
This is a C program.
```

源程序如下：

```
#include <stdio.h>
void main()      /*无参数的主函数*/
{                /*函数体开始*/
    printf("This is a C program.\n");
}                /*函数体结束*/
```

【程序说明】通过分析该程序，可得出 C 程序在语法结构上的一些特点：

（1）该例是结构最为简单的 C 程序实例，整个程序由一个模块组成，用 C 语言的一个主（main）函数实现该模块。

（2）主函数从语法结构上由{}前的“函数首部”以及{}对括的“函数体”两部分组成。

函数首部是函数的声明部分，对函数的属性诸如函数名、函数类型、形式参数等加以说明。在本例中，主函数按系统规定取名为 main，其是一个无参函数，且没有返回值，故其函数类型为空（void）类型。

函数体是函数的执行部分，它一般由若干语句组成，是函数功能的算法描述。在本例中，main 函数的函数体只包含一条语句，即通过调用库函数 printf 实现输出显示功能。

（3）为提高程序的可读性，应该给程序加上必要的注释。

C 语言程序中，注释由“/*”开始，由“*/”结束。注释不是程序的执行部分，而是帮助阅读程序的说明信息，合理使用注释是一种良好的编程习惯。

【例 1-2】 编程计算两个整数的平均值。

源程序如下：

```
#include <stdio.h>
void main()                                    /*一个求平均值的 C 程序*/
{
    int a,b;                                   /*定义两个整型变量*/
    float ave;                                 /*定义实型变量*/
    printf("Input  two integers: \n");         /*输入提示信息*/
    scanf("%d,%d",&a,&b);                      /*输入 a,b 之值*/
    ave=(a+b)/2.0;                             /*求平均值*/
    printf("Average=%.2f\n",ave);              /*输出平均值*/
}
```

【程序说明】本例的程序结构同例 1-1，均由一个模块构成 C 程序，并由 main 函数实现该模块。但相比于例 1-1，本例中的 main 函数具有了处理数据的功能。它通过调用库函数 scanf 接收两个整数之值（即程序中变量 a、b 之值），求出它们的平均值 ave 之后，再调用库函数 printf 输出该平均值。

程序运行结果：

```
Input  two integers:
18,35↙
Average=26.50
```

【例 1-3】 已知 x=2，求当 n=1,2,…,10 时，x^n 的值。

源程序如下：

```
#include <stdio.h>
int  power(x,n)
int x,n;
{
    int i,p=1 ;
    for(i=1;i<=n;i++)
      p=p*x;
    return p; /*返回 p 值*/
}
void main()
{
    int j;
    printf("n\tpower(2,n)\n");
    for(j=1;j<=10;j++)
    printf( "%d\t%d\n",j,power(2,j));
}
```

【程序说明】从程序结构上，本程序包含了两个用户自定义函数：主函数 main 和被调函数 power。

（1）由于求 x^n 的功能需多次使用，因此将求解 x^n 独立编写为一个函数，即程序中的 power(x,n)函数。该函数使用两个形式参数 x 和 n 来接收调用者发送的实际参数，求出 x^n 的值后，使用 return 语句加以返回。其中形式参数 x 和 n 均是整型数据，函数返回值 x^n 也是整型数值。因此，power 函数的首部描述如下：

```
int  power(x,n)
int x,n;
```

（2）main 函数是调用 power 函数的主调函数，在 main 函数的第 6 行调用 power 函数，在调用时将实际参数 2 和 j 的值分别传送给 power 函数中的形式参数 x 和 n，经过执行 power 函数求出 x^n 的值，使用 return 语句将求出的值带回到 main 函数调用处，并由 printf 函数实施打印。

程序运行结果（␣表示空格，后同）：

```
n␣␣␣␣␣␣␣␣power(2,n)
1               2
2               4
3               8
4               16
5               32
6               64
7               128
8               256
9               512
10              1024
```

通过以上 3 个具体例子的分析，可以概括出 C 程序的结构特点。

1．C 语言是“函数式”语言，一个 C 程序由一个或多个函数构成

C 程序中总是包含一个或多个用户所编写的函数。例 1-1 和例 1-2 的 C 程序均由一个 main 函数构成，而例 1-3 的程序则由 mian 和 power 两个函数通过一定的调用规则而构成。在 C 语言中，函数是组成 C 程序模块的唯一、基本的单位，模块之间的调用关系是通过 C 函数间的调用关系来实现的。

C 程序中除可调用用户自行编制的函数外，还可调用系统提供的库函数（如上述例题中的 scanf 函数和 printf 函数就是系统提供的两个 I/O 库函数，有关库函数的详细介绍参见第 3 章）。

程序中的全部工作是由各个函数分别完成的，编写 C 程序就是编写一个个的 C 函数。C 语言的这种特点使得容易实现程序的模块化。

2．每个 C 程序必须有一个且只能有一个 main 函数

main 函数用以标识程序中的主函数，指明程序执行的入口。一个 C 程序总是从 main 开始执行，而与 main 函数在整个程序中的位置无关。可以将 main 函数放在程序的最前面，