C语言从入门到精通

（第2版）

陈锐 王慧博 耿国华 赵娟 编著

- 讲解由浅入深，融入算法思想
- 图示详尽细致，总结常犯错误
- 案例实用典型，贯穿调试技术

中国工信出版集团
電子工業出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
http://www.phei.com.cn

C 语言从入门到精通

（第 2 版）

陈　锐　王慧博　耿国华　赵　娟　编著
夏政伟　张　雷　赵志华　高　妮　邢金萍　高全力　参编

電子工業出版社
Publishing House of Electronics Industry
北京 • BEIJING

内 容 简 介

本书从最基本的 C 语言概念入手，由浅入深，综合典型的实例，引导初学者由浅入深地掌握 C 语言知识。全书共 17 章。其中，第 1~6 章是入门篇，包括 C 语言概述和 Visual Studio 2008 开发工具的介绍，数据类型，运算符与表达式，顺序结构、选择结构和循环结构程序设计等内容。第 7~13 章是提高篇，包括数组、函数、指针、预处理、结构体和联合体、链表与文件等内容。第 14~17 章是技术篇，包括常用算法、网络编程、综合应用案例——图书管理系统和 C 语言调试技术等内容。

本书内容全面，不仅涵盖了 C 语言的基本语法与程序设计知识，还较为深入地介绍了相关常用算法知识和实用案例、调试技术。本书在讲解知识的同时，指出了编程时一些应注意的地方和易犯的错误，给出了不少关于 C 语言方面的调试经验。全书案例丰富，讲解非常细致，语言通俗易懂，所有实例代码均在 Visual Studio 2008 开发环境下运行通过。本书配套资源有 PPT、源代码。技术支持 QQ 群：234500678。

本书适合作为计算机相关专业学习 C 语言的自学用书和计算机专业本、专科生的教材，也可作为相关人员的参考用书。

图书在版编目（CIP）数据

C 语言从入门到精通 / 陈锐等编著. —2 版. —北京：电子工业出版社，2016.9
ISBN 978-7-121-29893-6

Ⅰ. ①C… Ⅱ. ①陈… Ⅲ. ①C 语言－程序设计－高等学校－教材 Ⅳ. ①TP312.8

中国版本图书馆 CIP 数据核字（2016）第 218079 号

策划编辑：任欢欢
责任编辑：任欢欢
印　　刷：三河市良远印务有限公司
装　　订：三河市良远印务有限公司
出版发行：电子工业出版社
　　北京市海淀区万寿路 173 信箱　邮编：100036
开　　本：787×1092　1/16　印张：26.5　字数：695.4 千字
版　　次：2016 年 9 月第 1 版
印　　次：2016 年 9 月第 1 次印刷
定　　价：58.00 元

凡所购买电子工业出版社图书有缺损问题，请向购买书店调换。若书店售缺，请与本社发行部联系，联系及邮购电话：（010）88254888，88258888。

质量投诉请发邮件至 zlts@phei.com.cn，盗版侵权举报请发邮件至 dbqq@phei.com.cn。

本书咨询联系方式：192910558（QQ 群）。

PREFACE

第2版前言

C 语言是目前国内外使用最为广泛的程序设计语言之一。它具有功能丰富、表达能力强、使用方便灵活、执行效率高、可移植性好等优点，可用于几乎所有领域。C 语言既具有高级语言的特点，也具有汇编语言的功能，具有很强的系统处理能力，可以直接对硬件和外部接口进行控制。C 语言被广泛应用于系统软件和应用软件的开发。

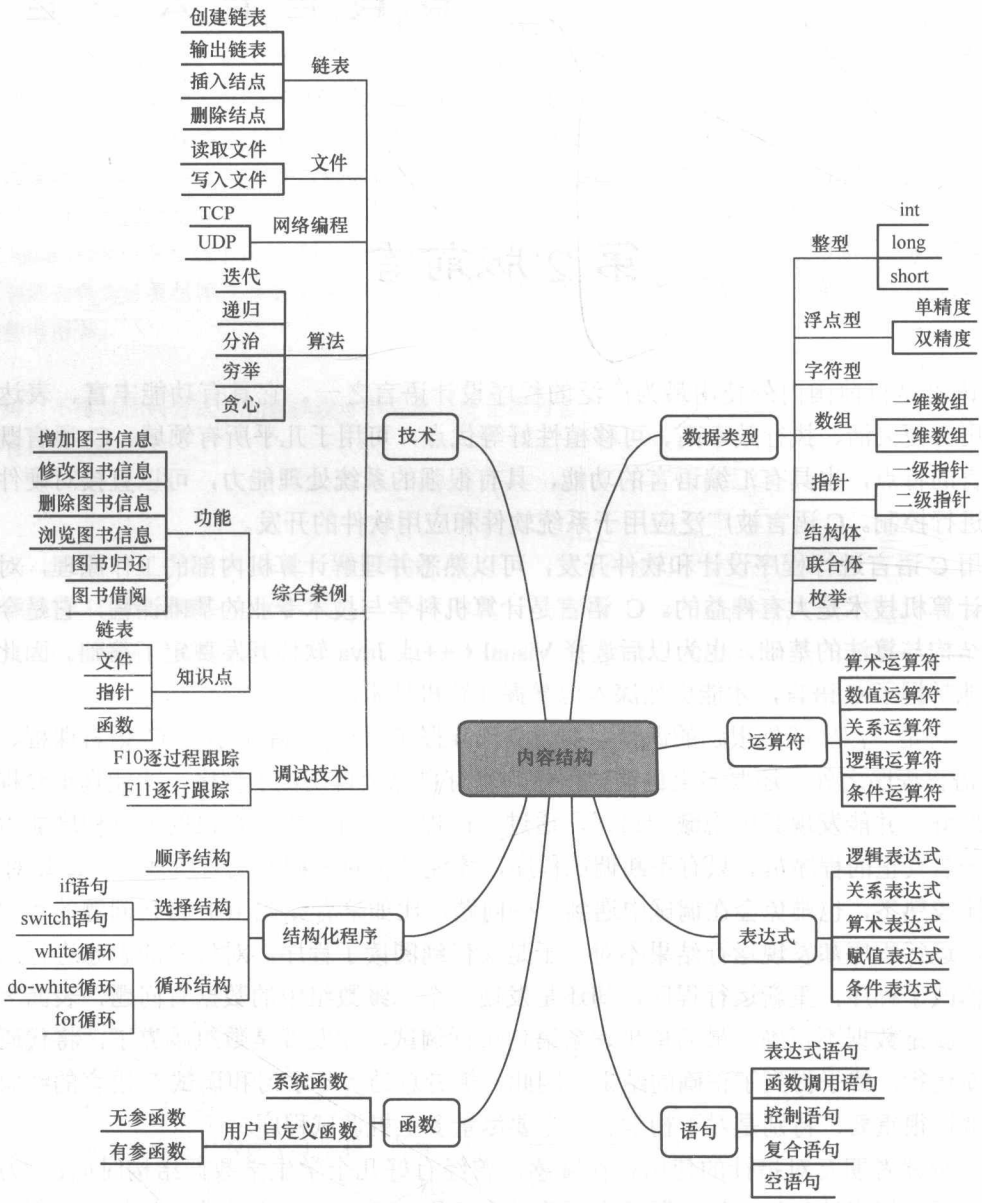
使用 C 语言进行程序设计和软件开发，可以熟悉并理解计算机内部的工作原理，对于深入学习计算机技术是大有裨益的。C 语言是计算机科学与技术专业的基础课程，它是今后学习数据结构与算法的基础，也为以后选择 Visual C++或 Java 软件开发奠定了基础。因此，只有熟练地掌握了 C 语言，才能更加深入地掌握计算机技术。

C 语言是一门实践性很强的课程，要想真正掌握 C 语言，除了学习 C 语言课程、大量阅读 C 语言程序之外，还要多上机调试程序。只有自己亲自调试了程序，才能真正对程序代码深入理解，并能发现其中隐藏的错误，通过调试程序还可以提高自己的调试程序能力，从而成为一名真正的程序员。只有不断调试程序，才能真正对编程开发有所体会，即使对 C 语言已经比较熟悉，也难免会在调试中遇到一些问题，比如笔者完成 0-1 背包问题的程序代码编写后，运行程序却发现运行结果不对，于是又仔细阅读了程序，对程序的思想进行深入理解，并修改了程序，重新运行程序，却还是发现一个二维数组中的数据有问题，表面上看没有问题，就是数据不正确，最后单步逐条语句进行调试，才发现是数组越界了，将代码修改后，重新运行，终于得到了正确的结果。因此，笔者总结多年学习和调试 C 语言的经验，认为程序调试很重要，特别是对于初学者一定要尽量多上机调试程序。

有不少读者朋友对指针的使用存在疑惑，曾经有好几个学生学数据结构时问：“为什么有时候要用二级指针？”他们的疑惑在于为什么要用二级指针作为函数的参数，而不用一级指针。说明有不少读者对指针的理解还不是很透彻，比如在主函数中创建一个指针，同时想通过另外一个函数去创建一个链表，于是希望在主函数中能使用这个已经创建的链表，但是又不想用 return 去返回链表的头指针，那么此时就需要使用二级指针作为函数的参数，是不是这个道理呢？二级指针的作用就在于此。其实，笔者是想通过这些例子来说明对于许多读者来说，看书太少，上机实践更少，如果能够多看些算法方面的书，多上机调试程序，有任何疑惑就上机调试一下，那么疑惑就会尽释，否则心理总存在疑惑，对问题理解得不够

透彻。

本书共 17 章，其中第 1~6 章是入门篇，第 7~13 章是提高篇，第 14~17 章是技术篇。本书知识结构可用如下图所示的思维导图来表示。



本书由西北大学的陈锐、辽源职业技术学院的王慧博、西北大学的耿国华、南阳理工学院的赵娟编著，许昌学院的夏政伟、运城学院的张雷、山西水利职业技术学院的赵志华、西安财经学院的高妮、郑州职业技术学院的邢金萍、西北大学的高全力参编。

修订后本书特点

1. 内容全面，讲解详细

本书不仅介绍了数据类型、结构化程序设计、数组、指针、结构体等，还介绍了链表、

文件、常用算法、网络编程，接着给出了一个图书管理系统的综合案例，以加强 C 语言综合实践能力的培养，最后还提供了 C 语言调试技术，针对具体的例子进行讲解，这些常见的错误也是笔者多年来遇到的问题，希望有助于提高读者的 C 语言调试能力。改版后，书中去掉了图形界面设计、键盘与鼠标操作、位操作等不常用的技术。

2．语言通俗，图表丰富

本书用非常通俗的语言，针对 C 语言中每个知识点，进行了详细的讲解，以便读者能快速理解并掌握每个知识点。C 语言概念较多，知识点零碎，为便于读者学习，在概念出现之后都给出了相应的例子和图表进行详细讲解，每个知识点都配有案例，便于读者领会其含义。

3．层次清晰，结构合理

本书内容结构设计合理，按照章、节和小节进行划分，将每一小节作为一个知识点，这样的设计使重点更加突出，结构层次感强，使读者在学习中容易抓住重点，通过将概念与例子结合，使知识更容易理解与消化。

4．实例丰富、典型，深入剖析讲解

本书案例丰富，每章的每一个知识点都有配套案例，全书大约 160 个案例，均有完整代码实现。本书在案例选择上非常有代表性，有的是考研题目，有的是程序员考试题目，这些题目很有技巧性，通过这些典型案例，可以提高读者程序设计的能力，有助于熟练理解编程思想，对于以后学习数据结构、算法等内容很有好处。书中针对每一个程序都进行了详细的讲解，特别是对于关键代码行，都进行了深入的分析，最后给出了运行结果。

5．实用性强，延伸知识

除了介绍基本的语法知识外，本书还介绍了链表、常用算法技术、网络编程等知识，特别是常用算法涉及迭代、递归、分治、贪心等算法思想。本书最后以一个综合案例和程序调试技术作为结束内容。在内容的讲解上也渗透了一些程序调试技术知识，通过学习本书，真正使读者能熟练掌握 C 语言，并且熟练调试 C 语言程序。通过这些知识点的介绍，使读者能进一步理解计算机相关技术的实现原理，也为今后的继续深入学习奠定了基础。

本书代码全部在 Visual Studio 2008 开发环境下运行通过。

适合的读者

- 没有任何基础，准备学习 C 语言的初学者
- 有一定 C 语言基础，想进一步掌握 C 语言的中高级读者
- 大中专院校学生
- 软件开发人员
- 计算机相关的科研工作者

应该感谢的人

在本书出版的过程中，许多热心的读者朋友提出了改进意见，特别是 puppypyb（网名）很认真地提供了具体的修改意见，感谢中国科学院大学的胡英鹏，中国科学技术大学的王启，华中科技大学的杨梨花，西安电子科技大学的杜坚，西安交通大学的郝昊天，华东师范大学的牛颖楠，南京航空航天大学的韩琦文，南京理工大学的邓裕彬，北京工业大学的潘

姝妤，电子科技大学的丁亮，上海海事大学的左伟康，福州大学的李川，湘潭大学的王乾，天津职业技术师范大学的董春妹，桂林电子科技大学的曹礼，郑州大学的张杨、张冬冬，成都理工大学的张良，西华师范大学的刘富腾，衡水学院的杨帅，重庆电子工程职业学院的冯博，湖南女子学院的李奇，湖北汽车工业学院的李兴海，黄淮学院的于景波，九江学院的樊美林，信阳师范学院的周亚林，云南大学的袁宏磊，广东技术师范学院的欧阳镇，江苏省扬州中学的张佑杰，浙江工业大学的陈文邦，电子科技大学的吕鑫垚，北京邮电大学世纪学院的昂超，兴义民族师范学院的鲜一峰，新乡学院的方湘豫，西安航空职业技术学院的王少波，赶集网的康钦谋，济南趣维网络科技有限公司的刘晓倩，中国航空计算研究所的王泉，中兴通讯公司的杨柯，华为科技有限公司的卢春俊，浪潮集团的郭鹏，大唐电信的张天广，三星电子的欧小哲，腾讯科技（北京）有限公司的杨凡，云南昆船设计研究院的夏翔。此外，还有很多网友也提出了宝贵建议，这里不再一一列举。

由于时间仓促，加上本人水平有限，书中难免会存在一些不足和错误，希望读者朋友通过邮箱 vc-study@163.com 联系，也可通过 QQ 群（234500678）沟通交流。祝愿各位读者在阅读本书的过程中有一个愉快的体验。

编　者

目录

CONTENTS

C 语言的前世今生

【学习目标】了解计算机语言的发展阶段、C 语言的诞生历史及在计算机语言中的地位，大致了解 C 语言的程序结构。

【知识点】（1）计算机及语言经历了哪些发展阶段？（2）什么是结构化程序设计？（3）C 语言具有哪些特点？

【重点】初步了解 C 语言的程序结构。

1.1 C 语言的起源

学习 C 语言，有必要了解一下有关 C 语言的背景知识，为什么会有计算机语言的出现？计算机语言经历了哪些发展阶段？在开始学习 C 语言之前，你也许会有许多这样或那样的疑问，下面我们就来了解一下计算机语言的来龙去脉，对计算机语言有一个初步的印象。

1.1.1 计算机语言的发展阶段

自 1946 年世界上第一台电子计算机问世以来，短短的几十年，计算机技术得到飞速的发展，计算机已广泛应用于生产、生活的各个领域。计算机不光包括硬件，还包括软件，其中硬件是物质基础，软件是计算机的灵魂。没有软件，计算机只是不能工作的一堆废铁，而所有软件都是由计算机语言编写的。

计算机语言，也称为**程序设计语言**，就是完成一定任务的指令，人们通过一系列指令让计算机完成指定的任务。计算机语言的发展经历了机器语言、汇编语言、高级语言等阶段。

1. 机器语言——0 和 1 二进制字符串

尽管现在的计算机可以处理文字、声音、图像、视频等信息，但是不管是哪种类型的信息，最终表现在计算机中都是“0”和“1”组成的二进制数，因此在计算机刚刚诞生的时候，人们编写出一串串“0”和“1”组成的指令序列让计算机执行指定的任务，我们把这种由“0”和“1”组成的二进制指令称为**机器语言**。可以想象，使用机器语言编写程序是多么痛苦，这就要求程序员有很强的记忆力，需要记住那些难懂的“0”和“1”二进制字符串，这些“0”和“1”组成的二进制字符串还很容易混淆，错误不可避免。而且，由于每台计算

机的指令系统往往互不相同，所以，在一台计算机上执行的程序，要想在另一台计算机上执行，必须另外编写程序，造成了重复工作。但由于使用的是针对特定型号计算机的语言，不需要中间的翻译过程，故而运算效率是所有语言中最高的。这就是第一代计算机语言。

2. 汇编语言——符号语言

为了减轻使用机器语言编程给人们带来的困难，人们开始使用好记的助记符代替“0”和“1”表示的指令，用地址符号或标号代替地址码。比如，“ADD”代表相加，“SUB”表示相减，“MOV”代表数据传递。这种程序设计语言就称为**汇编语言**，即第二代计算机语言。这些指令很容易记忆并容易理解程序的含义，也减轻了维护的难度，然而计算机是不认识这些符号的，这就需要一个专门的程序，专门负责将这些符号翻译成二进制数表示的机器语言，这种翻译程序被称为**汇编程序**。

【说明】库函数是指使用命令#include 的扩展名为“.h”的文件，这种文件称为**头文件**。它是 C 语言提供的具有特定功能的函数，被保存在“.h”文件中，在使用时通过#include 命令把它包含进来。

汇编语言比机器语言更易于读写、易于调试和修改，同时也具有机器语言执行速度快，占内存空间少等优点，但汇编语言依赖于具体的机型，移植性不好。汇编语言主要应用于与系统硬件紧密相关的部分，比如操作系统的核心程序段、计算机外部设备的驱动程序、工业控制等。

3. 高级语言——接近自然语言

汇编语言虽然执行效率高，但是它并不接近人类自然语言，并且过于依赖于计算机硬件，这就要求人们开发一种效率更高的语言，这种语言接近自然语言，只需编写更少量的代码，而不依赖于计算机硬件，编写出的程序能在所有机器上都通用。经过努力，1954 年，FORTRAN 诞生了，这是第一个完全脱离机器硬件的高级语言。60 多年来，共有几百种高级语言出现，其中影响较大、使用较普遍的有 FORTRAN、ALGOL、COBOL、BASIC、LISP、Pascal、C、C++、PROLOG、Ada、Delphi、Java、Python 等。

这些语言有的应用在专业的领域，如 LISP 应用在人工智能领域，FORTRAN 应用在科学和工程计算领域。目前应用最为广泛的语言有 C、C++和 Java，我们使用的大多数软件都是用这些语言开发出来的。

高级语言的发展也经历了从早期语言到结构化程序设计语言，从面向过程到面向对象程序语言的过程。相应地，软件的开发也由最初的个体手工作坊式的封闭式生产，发展为产业化、流水线式的工业化生产。

（1）非结构化程序设计语言与结构化程序设计语言

20 世纪 60 年代中后期，软件越来越多，规模越来越大，而软件的生产基本上是各自为战，缺乏科学规范的系统规划与测试、评估标准，其后果是大批耗费巨资建立起来的软件系统，由于含有错误而无法使用，甚至带来巨大损失，软件给人的感觉是越来越不可靠，以致几乎没有不出错的软件。这一切，极大地震动了计算机界，史称“软件危机”。这一时期的计算机程序结构杂乱无章，令人难以理解，且容易出错。人们开始认识到大型程序的编制不同于编写小程序，它应该是一项新的技术，应该像处理工程一样处理软件研制的全过程。程序的设计应易于保证正确性，也便于验证正确性。于是，在 1969 年，结构化程序设计方法

被提出，1970 年，第一个结构化程序设计语言——Pascal 语言出现，标志着结构化程序设计时期的开始。

（2）面向过程的程序设计与面向对象的程序设计

20 世纪 80 年代初，在软件设计思想上，又产生了一次革命，其成果就是面向对象的程序设计。在此之前的高级语言，几乎都是面向过程（Procedure-Oriented）的，程序按照流水线方式执行，它是一种以过程为中心的编程思想，即在一个模块执行完成前，人们不能干别的事，也无法动态地改变程序的执行方向。这与人们日常处理事物的方式是不一致的，对人而言发生一件事就处理一件事，也就是说，使用面向过程的方法模拟人的行为是不合适的，而应使用面向具体的应用功能，也就是**对象**。其方法就是软件的集成化，如同硬件的集成电路一样，生产一些通用的、封装紧密的功能模块，称之为软件集成块，它与具体应用无关，但能相互组合，完成具体的应用功能，同时又能重复使用。

例如五子棋程序，面向过程的设计思路就是首先分析问题的执行步骤：1．开始游戏，2．黑子先走，3．绘制画面，4．判断输赢，5．轮到白子，6．绘制画面，7．判断输赢，8．返回步骤 2，9．输出最后结果。然后把每个步骤分别用函数实现，问题就解决了。

而面向对象的设计则是从另外的思路来解决问题。整个五子棋程序可以分为：1．黑白双方，这两方的行为是一模一样的，在面向对象程序设计中将黑方和白方称为两个同类对象；2．棋盘系统，负责绘制画面；3．规则系统，负责判定是否输赢、犯规等。第一类对象（玩家对象）负责接收用户输入（移动棋子），并告知第二类对象（棋盘系统）棋子布局的变化；棋盘系统将接收到的棋子变化显示在屏幕上；第三类对象（规则系统）对棋局的输赢、犯规进行判定。

不难看出，面向对象是以功能来划分问题的，而不是步骤。同样是绘制棋局，这样的行为在面向过程的设计中分散在了很多步骤中，很可能出现不同的绘制版本，因为通常设计人员会考虑到实际情况进行各种各样的简化。而面向对象的设计中，绘图只可能在棋盘对象中出现，从而保证了绘图的统一。这种功能上的统一保证了面向对象设计的可扩展性。例如，若要加入悔棋的功能，对于面向过程的设计，那么可能输入、判断和显示等一系列步骤都需要改动，甚至步骤之间的顺序都要进行大规模调整，不利于维护和扩展。对于面向对象的设计，因为棋盘系统保存了黑白双方的棋谱，所以只需将棋盘系统简单回退到上一步即可，而其他功能（如输入、规则判断）都无须改动，同时整个对象功能的调用顺序都没有变化。

面向对象的编程语言（Object-Oriented Language）是面向对象程序设计的语言，对象成为程序结构的基本单位，用面向对象的语言开发出的程序易于维护和扩展，符合人们处理问题的方式。C++、Java、Delphi 就属于面向对象程序设计语言。

相应于计算机语言，也出现了各种各样的开发工具，开发工具的作用就是将计算机语言翻译成计算机能识别的二进制编码，以便执行相应的操作。例如 VC++是用于 C 程序的开发工具，VB 是 Basic 语言的开发工具，Eclipse 是 Java 开发工具。

1.1.2　C 语言的产生

C 语言的发展颇为有趣，它的原型是 ALGOL 60 语言（一种算法，适合于数值计算）。

1963 年，剑桥大学在 ALGOL 60 语言的基础上创建了 CPL（Combined Programming Language），这种语言更接近于硬件。

1967 年，剑桥大学的 Matin Richards 对 CPL 语言进行了简化，于是产生了 BCPL 语言。

1970 年，美国贝尔实验室的 Ken Thompson 将 BCPL 进行了修改，设计出了简洁的 B 语言，并用 B 语言写了第一个 Unix 操作系统。

1973 年，贝尔实验室的 Dennis M. Ritchie（图 1.1）在 B 语言的基础上设计出了一种新的语言——C 语言。C 语言既保持了 BCPL 语言和 B 语言的精练、接近硬件，又克服了它们的过于简单、数据无类型的缺点。

1978 年，Brian W. Kernighan（图 1.2）和 Dennis M. Ritchie 出版了影响深远的经典著作《The C Programming Language》，成为 C 语言最权威的教材。从此，C 语言一直成为目前世界上最为流行的高级程序设计语言，经久不衰。

计算机科学家 Dennis M. Ritchie 对 C 语言和其他编程语言以及 Unix 等操作系统的发展做出了巨大贡献。Dennis M. Ritchie 先后在哈佛大学学习物理学和应用数学毕业，并获得博士学位，1967 年进入贝尔实验室，是朗讯技术公司系统软件研究部门的领导人。因为研究发展了通用的操作系统理论，特别是实现了 Unix 操作系统，所以，1983 年，他与 Ken Thompson 一起获得了计算机领域的最高奖项——图灵奖。Dennis M. Ritchie 还被誉为“C 语言之父”。

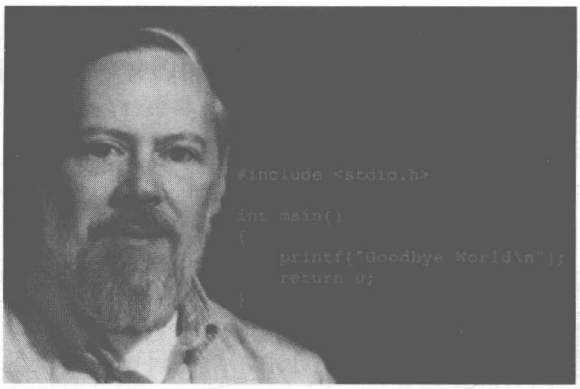


图 1.1　计算机科学家 Dennis M. Ritchie

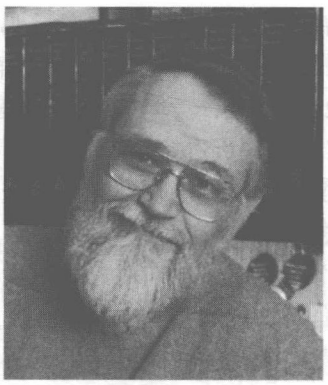


图 1.2　计算机科学家 Brian W. Kernighan

由于大部分应用软件最终都需要和操作系统打交道，所以用来开发应用软件的语言，绝大部分也需要利用 C 语言完成和操作系统的通信。这个世界上绝大部分流行的编程语言，都选择了用 C 语言来实现其编译器或解释器，以及底层的运行时库。也许 C 语言还有缺憾，比如说指针造成的不安全性，但任何东西都不可能是完美的，至少今天它已无可取代。

1.2　为什么要选择 C 语言

目前，世界上流行的计算机编程语言有许多，如 C、C++、Java、Pascal、BASIC、FORTRAN。我们为什么要选择 C 语言而不选择其他语言呢，也就是学习 C 语言有哪些好处，C 语言具有哪些特点及如何学好 C 语言？下面就为读者详细道来。

1.2.1　选择 C 语言的好处

C、C++、Java、BASIC 都是目前最为流行的编程语言，现在几乎每一位程序员，无论是 Java 程序员、C++程序员，还是 Web 程序员，他们都有过学习 C 语言的经历，并且他们

学习的第一门语言往往是 C 语言。对于初学者来说，C 语言是一个非常不错的选择。因为选择 C 语言有如下好处：

1．C 语言是 C++、Java、C#的基础。C++、Java、C#是目前最为流行的语言，这些语言都是建立在 C 语言基础之上的，当读者学了 C 语言之后，再学习 C++、Java 或 C#会变得非常容易。

2．C 语言涵盖了计算机编程语言中的所有知识，对于今后学习其他编程语言和理解计算机系统的工作方式有很大的帮助。

3．使用 C 语言编写的程序运行效率高。C 语言是一门高级语言，使用它编写的程序仅仅比汇编语言编写的程序运行效率低 10%~20%。

4．C 语言可以直接访问内存，并可以直接对硬件进行操作。读者无须使用汇编语言，使用 C 语言就可以完成相同的功能。同时，C 语言要比汇编语言更加容易学习和使用。

5．过去许多系统都是使用 C 语言实现的，如 Windows 和 Linux 操作系统大部分代码是用 C 语言编写的，它们都可以得到重新利用。

6．学好 C 语言，就为今后继续学习数据结构和算法打好了基础。因为目前绝大部分的数据结构和算法教材都是以 C 语言描述的。

7．使用 C 语言既可以编写系统软件也可以编写应用软件。现在的游戏软件、嵌入式开发和手机开发都选择 C 语言作为开发语言。

8．使用 C 语言编写的程序有很好的移植性。在一种计算机系统如 IBM PC 上编写的 C 语言程序，可以直接在其他系统如 DEC VAX 系统上运行。

既然 C 语言具有这么多的优点，相信读者也坚定了学习 C 语言的决心。接下来，就让我们来了解一下 C 语言的特点。

1.2.2　C 语言的特点

C 语言自从 1973 年诞生于贝尔实验室以来，经历了 40 多年的历史，仍然经久不衰，拥有广大的用户。从历年来的编程语言排行来看，C 语言和 Java 语言一直稳居前 1~2 位。这与 C 语言自身的特点分不开。

1．C 语言程序结构紧凑、简洁、规整，表达式简练、灵活——容易理解与学习

C 语言程序结构简洁、紧凑、规整，容易阅读与理解。表达式简练，去除了一些不必要的成分，书写简单，使用起来比较灵活。因此，C 语言更加容易掌握。

2．C 语言的数据类型丰富——可以描述各种复杂的数据结构

C 语言的数据类型包括整型、实型、字符型、数组类型、指针类型、结构体类型、联合体类型等。其中，结构体类型和联合体类型是用户自定义的类型，根据具体需要可以自己定义类型。C 语言所提供的数据类型可以描述各种复杂的数据结构。

3．C 语言的运算符丰富——实现复杂的运算比较方便

C 语言包含了 34 种运算符，丰富的运算符与丰富的数据类型结合，构成了多样的表达式。灵活的运算符可以很容易实现比较复杂的运算。

4．C 语言是一种结构化的语言——拥有 3 种控制语句、函数作为程序的模块单元

C 语言是一种结构化的程序设计语言，适用于大型的模块化程序设计。它拥有 3 种控制语句，函数是 C 语言程序的模块单元，每个函数各自独立。C 语言的源程序可以分为多个源