

TURING 图灵程序设计丛书

C 语言程序设计 现代方法 (第2版)

C Programming: A Modern Approach, Second Edition

[美] K. N. King 著
吕秀锋 黄倩 译

 人民邮电出版社
POSTS & TELECOM PRESS

清华大学出版社

语言程序设计

现代方法

清华大学出版社

清华大学出版社

清华大学出版社



TURING 图灵程序设计丛书

C 语言程序设计 现代方法 (第2版)

C Programming: A Modern Approach, Second Edition

[美] K. N. King 著

吕秀锋 黄倩 译

人民邮电出版社
北京

图书在版编目 (CIP) 数据

C 语言程序设计: 现代方法: 第 2 版 / (美) 金 (King, K. N.) 著; 吕秀锋, 黄倩译. —北京: 人民邮电出版社, 2010.4

(图灵程序设计丛书)

书名原文: C Programming: A Modern Approach, Second Edition

ISBN 978-7-115-21957-2

I. ① C… II. ①金…②吕…③黄… III. ① C 语言—程序设计 IV. ① TP312

中国版本图书馆 CIP 数据核字 (2009) 第 231901 号

Copyright © 2008, 1996 by W. W. Norton & Company, Inc.

All rights reserved. No portion of this book may be reproduced in any form or by any means without the prior written permission of the publisher.

本书中文版由 W. W. Norton 公司授权人民邮电出版社独家出版。未经出版者书面许可, 不得以任何方式复制或抄袭本书内容。

版权所有, 侵权必究。

内 容 提 要

时至今日, C 语言仍然是计算机领域的通用语言之一, 但今天的 C 语言已经和最初的时候大不相同了。本书最主要的一个目的就是通过一种“现代方法”来介绍 C 语言, 书中强调标准 C, 强调软件工程, 不再强调“手工优化”。这一版中紧密结合了 C99 标准, 并与 C89 标准进行对照, 补充了 C99 中的最新特性。本书分为 C 语言的基础特性、C 语言的高级特性、C 语言标准库和参考资料 4 个部分。每章末尾都有一个“问与答”小节给出一系列与该章内容相关的问题及答案, 此外还包含适量的习题。

本书是 C 程序员的理想参考书, 在国外也被众多大学采用为 C 语言课程的教材。

图灵程序设计丛书

C 语言程序设计: 现代方法 (第 2 版)

-
- ◆ 著 [美] K. N. King
 - 译 吕秀锋 黄倩
 - 责任编辑 杨海玲
 - ◆ 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 电子邮件 315@ptpress.com.cn
 - 网址 <http://www.ptpress.com.cn>
 - 三河市海波印务有限公司印刷
 - ◆ 开本: 787×1092 1/16
 - 印张: 38.5
 - 字数: 1130 千字 2010 年 4 月第 1 版
 - 印数: 1—3 000 册 2010 年 4 月河北第 1 次印刷
 - 著作权合同登记号 图字: 01-2009-0531 号
 - ISBN 978-7-115-21957-2
-

定价: 79.00 元

读者服务热线: (010)51095186 印装质量热线: (010)67129223

反盗版热线: (010)67171154

前 言

在计算领域中，把显而易见的转变为有实用价值的，这一过程是“挫折”一词的生动体现。


自本书第1版出版以来，基于C的语言大量兴起（其中最杰出的代表是Java和C#），已有的C++和Perl等相关语言也取得了更大的成就。尽管如此，C语言仍然像当年一样流行，仍然悄无声息地掌控着世界上的许多软件。跟1996年一样，C语言仍然是计算机领域里的通用语言。

但即便是C语言也必须随着时间而发展。C99标准的发布催生了对本书新版的需求，而且，第1版涉及的DOS和16位处理器也已经趋于过时。本版对内容进行了全面更新，并在其他许多方面进行了改进。

本版新增内容

下面列出了本版的新特色和所做的改进。

- **完整地覆盖了C89标准和C99标准。**本版和第1版最大的差别就在于覆盖了C99标准。我的目标是覆盖C89和C99之间的每一个重要差别，包括C99新增的所有语言特性和库函数。C99中的每一处改变都会清楚地标出来，或者在小节标题中加上“C99”字样，或者在讨论比较简短时在正文的左边空白处用一个专门的图标表示。我这样做有两个目的：一是提醒读者注意C99中的改变，二是让那些对C99不感兴趣或没有C99编译器的读者知道哪些内容可以跳过。C99新增的许多内容可能只有特定的读者会感兴趣，但有些新特性几乎对所有的C程序员都有用。
- **提供了对所有C89和C99库函数的快速参考。**第1版中附录D介绍了C89的所有标准库函数，本版的附录D给出了C89和C99的所有库函数。
- **扩展了GCC的内容。**自本书第1版出版以来，GCC（最初是GNU C Compiler的简称，现在指GNU Compiler Collection）得到了广泛应用。GCC有很多优点，包括高性能、低成本（不用花钱）以及在众多软硬件平台之间的可移植性等。由于认识到GCC日渐重要，我在本版中介绍了更多与GCC相关的信息，包括如何使用GCC以及常见的GCC错误消息和警告。
- **增加了对抽象数据类型的讨论。**在第1版中，第19章重点讨论了C++。这部分内容现在看起来似乎作用不大，因为本书的读者可能已经学过C++、Java或者C#了。在本版中，对C++的介绍替换为讨论如何在C中建立抽象数据类型。
- **扩展了国际化特性的内容。**本版第25章更加详尽地讨论了C语言的国际化特性。重点扩展了Unicode字符集及编码。
- **针对CPU和操作系统做了更新。**当我编写本书第1版时，许多读者用的还是16位机和DOS操作系统，但现在情况不同了。在本版中，我把讨论的重点放在32位机和64位机上。尽管本版也介绍了Windows和Mac OS操作系统中影响C程序员的方面，但是针对Linux和其他UNIX版本的出现，我们的讨论更侧重于后一个系列的操作系统。

- **更多的练习题和编程题。**本书第1版包括311道习题，本版有将近500（准确地说是498）道习题，分为两组：练习题与编程题。
- **练习题和编程题的答案。**本书第1版的读者反馈最多的问题就是希望我提供习题的答案。针对读者的这一需求，我将大约三分之一的练习题和编程题的答案放到了网上，见 knking.com/books/c2。这一特色对于那些没有选修相应大学课程但却需要检验自己工作的读者来说是非常有用的。提供了答案的练习题和编程题都用  图标标记（“W”表示“此题在网站上答案”）。
- **有密码保护的教师网站。**我为本版书建了一个新的教师资源网站（通过 knking.com/books/c2 访问），给出了其余练习题和编程题的答案以及大部分章节的PowerPoint讲义。教师可以通过 cbook@knking.com 与我联系。请使用您学校的邮件地址并给出一个可以访问贵系网站的链接，以便我核实您的身份。

此外，我在本版中对全书的文字和解释说明做了改进。这些改变所需的工作量很大，过程很辛苦：每句话都检查过并（在必要的时候）重新写过。

尽管本版改动很大，我仍然尽可能多地保持了原有的章节编号。尽管只有一章（最后一章）内容是全新的，但其他许多章都有新增的内容，少数原有章节的顺序也有所变动。本版删去了一个附录（C语言语法），但又新增了一个比较C99和C89的附录。

目标

本版的目标与第1版一致。

- **清晰易读，并尽可能带有趣味性。**对普通读者来说，许多C语言的书籍都过于简洁。甚至某些C语言书籍不是编写得一塌糊涂，就是平淡无趣。我试图对C语言进行清晰、全面的讲解，并用适当的幽默来激发读者的阅读兴趣。
- **适用于广泛的读者群。**我假设本书的读者都至少有一点点编程经验，但不需要掌握某种具体的编程语言。我尽量减少“行话”并定义用到的每一个术语。同时，为了鼓励初学者，我还尝试将某些高级内容从基本主题中分离出来。
- **有权威性，但不是学究气十足。**为了避免对应该包含哪些内容、不应该包含哪些内容的武断决定，我尽量涵盖了所有C语言的特性和库函数。同时，为了避免给读者造成负担，我还忽略了一些不必要的细节。
- **具备简单易学的组织结构。**根据多年教授C语言的经验，我强调循序渐进地展示C语言特性的重要性。针对有一定难度的主题，我采用了螺旋式的介绍方法。也就是说，对于较难的主题先进行简要介绍，然后在后续章节中再进行一次或多次介绍，每次逐渐增加一些细节内容。本书的进度是经过深思熟虑的。每章都按照循序渐进的方式进行组织，并且前后内容由浅入深，相互呼应。对于大多数学生来说，这种循序渐进的方法是最合适的：既能避免产生厌倦，又能防止“信息超载”。
- **深入探讨语言特性。**我的目标不是仅描述语言的每个特性，并展示应用该特性的几个简单示例，而是尝试深入讲解每一个特性，并且探讨如何将其应用到实际问题中。
- **强调编码风格。**对每位C程序员来说，采用一种统一的编码风格是非常重要的。但是，与指定某种风格相比，我更愿意给出多种编码风格，让读者根据自己的喜好做出选择。因为了解多种编码风格对阅读别人的程序是很有帮助的（程序员经常要花费大量时间阅读别人的程序）。

- 避免依赖任何特定的计算机、编译器或操作系统。C语言可以应用在许多平台上，所以我试图避免编写的程序依赖于任何特定的计算机、编译器或操作系统。所有程序都经过精心设计，可以移植到多种平台上去。
- 用图示的方法阐明关键概念。我在书中加入了尽可能多的图，因为我认为图对于理解C语言的许多方面都是至关重要的。特别地，我尽可能地通过图来显示不同计算阶段的数据状态，以此来动态地展示算法。

现代方法到底指的是什么

本书最重要的目标之一就是“通过一种‘现代方法’来介绍C语言”。我试图通过以下这些途径来实现这一目标。

- 正确看待C语言。我没有把C语言看成是唯一值得学习的编程语言，而是把它作为众多有用语言中的一种进行介绍。我在书中提到了最适合用C语言编程的应用类型。此外，我还展示了如何扬长避短地使用C语言。
- 强调C语言的标准版本。我尽可能少地关注C89标准之前的C语言，只是零星地提到了经典(K&R) C语言(Brian Kernighan和Dennis Ritchie所著的*The C Programming Language*第1版中所描述的1978版C语言)。附录C列出了C89和经典C之间的主要差异。
- 揭穿神话。现今的编译器常常与过去的C语言基本假设不一致，我很乐于揭穿C语言的某些神话，并挑战一些存在了很久的C语言信条(例如，指针的算术运算一定比数组下标操作快)。我重新审查了C语言的旧惯例，保留了那些仍然有帮助的惯例。
- 强调软件工程。我把C语言视为一种成熟的软件工程工具，着眼于如何运用C语言来处理大规模程序设计过程中产生的问题。本书强调程序要易读、可维护、可靠且容易移植，尤其重视信息隐藏。
- 推迟介绍C语言的底层特性。虽然这些特性对于那些用C语言编写的系统来说非常有用，但现在它们已经不那么适用了，因为C语言的应用比以前广泛得多。本书没有像其他许多C语言书籍那样把这部分内容放在前面介绍，而是推迟到第20章再进行讲述。
- 不再强调“手工优化”。许多书籍指导读者编写一些技巧性较强的代码，以获得程序效率的些许提高。如今优化的C语言编译器随处可见，这些编程技巧往往已经不必要了；事实上，它们反而会降低程序的运行效率。

“问与答”部分

每章的末尾都有一个“问与答”部分，汇集了与本章内容相关的问题及其答案。“问与答”部分包括以下一些内容。

- 常见问题。我尽力回答了某些频繁出现在我的课堂里、其他书籍中及与C语言相关的新闻组里的问题。
- 对一些难以理解的问题的进一步讨论和澄清。虽然具有多种编程语言经验的读者会满足于简明扼要的说明和少量的示例，但是缺乏经验的读者却需要更多的内容以帮助理解。
- 非主流的问题。某些问题所引出的并不是所有读者都感兴趣的技术问题。
- 某些对普通读者来说过于超前或深奥的内容。这类问题都用星号(*)进行了标记。好学且有一定编程经验的读者也许希望立刻深入研究这些问题，而另外一些读者则需要

首次阅读时跳过这部分内容。提示：这类问题往往引用后续章节的内容。

- C语言编译器之间的常见差异。我讨论了某些特定编译器所提供的一些频繁使用的（非标准）特性。

“问与答”部分中的某些问题与对应章中的具体内容直接相关，我们用一个专门的图标 **Q&A** 来标记这些具体内容，以提示读者有附加信息可用。

其他特色

除了“问与答”部分，我还加入了许多有用的特色，其中很多都用简单而独特的图标做了标记。

- 警告（△）警示读者一些常见的缺陷。C语言以其陷阱多而出名，要记录所有的陷阱非常困难。我试着挑选出了一些最常见或最重要的缺陷供大家参考。
- 交叉引用（▶前言）提供一种类似超文本的能力来定位信息。多数引用指向后续章节中的内容，也有一些引用指向先前的内容供读者回顾。
- 惯用法是C语言程序中常见的代码模式。它被标记出来以便于速查参考。
- 可移植性技巧给出了编写不依赖于特定计算机、编译器或操作系统的程序所需的提示。
- 附加说明包含一些严格来讲并不属于C语言的内容，但每位熟练的C程序员都应该知道。（下面的“源代码”给出了附加说明的示例。）
- 附录提供有价值的参考资料信息。

程序

选择程序示例并不是件轻松的工作。如果程序过于简洁和做作，那么读者将无法体会如何将这特性应用于现实世界。另一方面，如果程序过于真实，那么它的要点将很容易被忽略在过多的细节中。我采取了折中方案。在首次介绍时，先通过小而简单的示例使概念清晰，然后再逐步建立完整的程序。我没有使用过长的程序，因为根据我个人的经验，教师没有时间介绍这些内容，而学生也不会有耐心去阅读。但是，我并没有忽视编写大规模程序时会出现的问题，相关内容在第15章和第19章中进行了详细的介绍。

我克制住了自己重新编写程序以利用C99特性的冲动，因为并不是每个读者都安装了C99编译器或愿意使用C99。但是，在有些程序中我仍然使用了C99的<stdbool.h>头文件，因为它定义了bool、true和false宏。如果你的编译器不支持<stdbool.h>，就需要自己定义这些宏。

本版的程序有些小的改变。现在main函数的格式在大多数情况下为int main (void) {...}。这一改变既反映了业界的惯例，又能够与C99兼容，C99要求每个函数都有一个显式的返回类型。

源 代 码

本书中所有程序的源代码都可以从knking.com/books/c2下载^①。有关本书的更新、校正和最新消息也可以从这一网站获得。

^① 本书源代码也可以在图灵网站（www.turingbook.com）本书网页免费注册下载。——编者注

读者

本书是为大学本科阶段的C语言课程编写的教材。具有其他高级语言或汇编语言的编程经验会很有帮助，不过这些经验对于会用计算机的读者（我以前的一位编辑称他们为“熟练的初学者”）来说并不是必需的。

因为本书内容齐备、自成一体，并且既可用于学习又可作为参考，所以它非常适合作为其他一些课程的辅助读物，如数据结构、编译器设计、操作系统、计算机图形学、嵌入式系统及其他要用C语言进行项目设计的课程。“问与答”部分以及对实际问题的强调，使得本书对于培训班学员和自学C语言的人来说也很有吸引力。

组织结构

本书分为4个部分。

- **C语言的基本特性。**第1章~第10章包含的C语言内容足以帮助读者编写出使用数组和函数的单文件程序。
- **C语言的高级特性。**第11章~第20章建立在前面各章内容的基础上，内容有一定的难度，深入介绍了指针、字符串、预处理器、结构、联合、枚举以及C语言的底层特性。此外，第15章和第19章提供了程序设计方面的指导。
- **C语言标准库。**第21章~第27章集中介绍C语言库——与编译器相关联的庞大函数集合。其中一部分内容适合讲解，但大部分材料更适合作为参考。
- **参考资料。**附录A给出了C语言运算符的完整列表。附录B描述了C99和C89之间的主要差别。附录C讨论了C89和经典C之间的差异。附录D按字母顺序列出了C89和C99标准库中的全部函数，并为每个函数给出了详尽的说明。附录E列出了ASCII字符集。还有一个带注解的参考文献列表为读者指明了其他的信息来源。

全面讲授C语言的课程应该按顺序覆盖前20章的内容，并根据需要增加第21章~第27章中的一些内容（其中讨论了文件输入/输出的第22章最为重要）。短期课程可以忽略以下内容而不失连贯性：8.3节（变长数组）、9.6节（递归）、12.4节（指针和 multidimensional 数组）、14.5节（其他指令）、17.7节（指向函数的指针）、17.8节（受限指针）、17.9节（灵活数组成员）、18.6节（内联函数）、第19章（程序设计）、20.2节（结构中的位域）和20.3节（其他底层技术）。

练习题和编程题

作为一本教材，拥有多样化的精选习题显然是非常必要的。本版既有练习题（不需要写出完整程序的简短习题）又有编程题（需要编写或修改完整程序的习题）。

有些练习题的答案不是显而易见的（有人称其为“刁钻问题”）。因为C语言程序经常包含这类代码的大量案例，所以我认为有必要提供一些这样的练习，并用星号（*）进行了标注。对于有星号的习题一定要小心：要么格外小心，认真考虑，要么干脆绕开它。

反馈

为了使本书准确，我付出了极大的努力。然而，任何这种篇幅的书籍都不可避免地会有一

些错误。如果读者发现了错误，请通过cbook@knking.com这个电子邮箱与我联系。我也同样期望听到读者的其他反馈，比如，你觉得哪些内容特别有用，哪些内容没什么用，希望添加哪些内容等。

致谢

首先，我要感谢本书的编辑，Norton出版社的Fred McFarland和Aaron Javasicas。本书的编辑工作最初由Fred负责，随后Aaron加入并付出了极大努力使本书得以完成。同时，还要感谢副主编Kim Yi、文字编辑Mary Kelly、生产经理Roy Tedoff和编辑助理Carly Fraser。

以下同事对本版的部分或全部书稿进行了审阅，在此致以诚挚的谢意：Markus Bussmann（多伦多大学）、Jim Clarke（多伦多大学）、Karen Reid（多伦多大学）和Peter Seebach（comp.lang.c.moderated新闻组的主持人）。其中需要特别提到的是Jim和Peter，他们的详细审阅使本版少了许多错误。再次感谢第1版书稿的审稿人（按姓氏字母排序）：Susan Anderson-Freed、Manuel E. Bermudez、Lisa J. Brown、Steven C. Cater、Patrick Harrison、Brian Harvey、Henry H. Leitner、Darrell Long、Arthur B. Maccabe、Carolyn Rosner和Patrick Terry。

我收到了第1版读者反馈的许多有用的意见，感谢每一位花时间提意见的读者。佐治亚州立大学的学生和同事也向我反馈了不少有价值的意见。Ed Bullwinkel和他的妻子Nancy阅读了手稿的很多内容，在此我也要感谢他们。我还要特别感谢我的系主任Yi Pan，他非常支持我的这项工作。

感谢我的妻子Susan Cole一如既往地支持着我。还有我们的猫咪Dennis、Pounce和Tex，在完成本书的过程中，它们一直陪伴着我。有时，Pounce和Tex的争吵使我在深夜写作时仍能保持清醒。

最后，我还要感谢已故的Alan J. Perlis^①。他的警句出现在本书每一章的开始。20世纪70年代中期我在耶鲁大学求学期间，曾有幸在Alan的指导下进行过短暂的学习。我想如果他知道自己的警句出现在一本C语言书中一定会非常高兴。

^① Alan J. Perlis（1922—1990）是计算机科学先驱，1966年首届图灵奖得主。——编者注

目 录

第1章 C语言概述	1	练习题	23
1.1 C语言的历史	1	编程题	24
1.1.1 起源	1	第3章 格式化输入/输出	26
1.1.2 标准化	1	3.1 printf函数	26
1.1.3 基于C的语言	2	3.1.1 转换说明	27
1.2 C语言的优缺点	3	程序 用printf函数格式化数	28
1.2.1 C语言的优点	3	3.1.2 转义序列	28
1.2.2 C语言的缺点	3	3.2 scanf函数	29
1.2.3 高效地使用C语言	4	3.2.1 scanf函数的工作方法	30
问与答	5	3.2.2 格式串中的普通字符	31
第2章 C语言基本概念	7	3.2.3 易混淆的printf函数和scanf函数	32
2.1 编写一个简单的C程序	7	程序 分数相加	32
程序 显示双关语	7	问与答	33
2.1.1 编译和链接	8	练习题	34
2.1.2 集成开发环境	8	编程题	35
2.2 简单程序的一般形式	9	第4章 表达式	36
2.2.1 指令	9	4.1 算术运算符	36
2.2.2 函数	9	程序 计算通用产品代码的校验位	38
2.2.3 语句	10	4.2 赋值运算符	39
2.2.4 显示字符串	10	4.2.1 简单赋值	40
2.3 注释	11	4.2.2 左值	41
2.4 变量和赋值	12	4.2.3 复合赋值	41
2.4.1 类型	12	4.3 自增运算符和自减运算符	42
2.4.2 声明	13	4.4 表达式求值	43
2.4.3 赋值	13	4.5 表达式语句	45
2.4.4 显示变量的值	14	问与答	45
程序 计算箱子的空间重量	14	练习题	47
2.4.5 初始化	15	编程题	49
2.4.6 显示表达式的值	16	第5章 选择语句	50
2.5 读入输入	16	5.1 逻辑表达式	50
程序 计算箱子的空间重量(改进版)	16	5.1.1 关系运算符	50
2.6 定义常量的名字	17	5.1.2 判等运算符	51
程序 华氏温度转换为摄氏温度	17	5.1.3 逻辑运算符	51
2.7 标识符	18	5.2 if语句	52
2.8 C程序的书写规范	19		
问与答	21		

5.2.1 复合语句	53	7.3 字符类型	94
5.2.2 else子句	53	7.3.1 字符操作	95
5.2.3 级联式if语句	54	7.3.2 有符号字符和无符号字符	95
程序 计算股票经纪人的佣金	55	7.3.3 算术类型	96
5.2.4 “悬空else”的问题	56	7.3.4 转义序列	96
5.2.5 条件表达式	57	7.3.5 字符处理函数	97
5.2.6 C89中的布尔值	58	7.3.6 用scanf和printf读/写字符	98
5.2.7 C99中的布尔值 C99	58	7.3.7 用getchar和putchar读/写字符	98
5.3 switch语句	59	程序 确定消息的长度	99
程序 显示法定格式的日期	61	7.4 类型转换	100
问与答	62	7.4.1 常用算术转换	101
练习题	65	7.4.2 赋值过程中的转换	102
编程题	67	7.4.3 C99中的隐式转换 C99	103
第6章 循环	69	7.4.4 强制类型转换	103
6.1 while语句	69	7.5 类型定义	105
程序 显示平方表	71	7.5.1 类型定义的优点	105
程序 数列求和	71	7.5.2 类型定义和可移植性	105
6.2 do语句	72	7.6 sizeof运算符	106
程序 计算整数的位数	73	问与答	107
6.3 for语句	73	练习题	109
6.3.1 for语句的惯用法	74	编程题	110
6.3.2 在for语句中省略表达式	75	第8章 数组	113
6.3.3 C99中的for语句 C99	75	8.1 一维数组	113
6.3.4 逗号运算符	76	8.1.1 数组下标	113
程序 显示平方表(改进版)	77	程序 数列反向	115
6.4 退出循环	78	8.1.2 数组初始化	115
6.4.1 break语句	78	8.1.3 指定初始化合式 C99	116
6.4.2 continue语句	78	程序 检查数中重复出现的数字	116
6.4.3 goto语句	79	8.1.4 对数组使用sizeof运算符	117
程序 账簿结算	80	程序 计算利息	118
6.5 空语句	81	8.2 多维数组	119
问与答	83	8.2.1 多维数组初始化	120
练习题	84	8.2.2 常量数组	121
编程题	85	程序 发牌	121
第7章 基本类型	88	8.3 C99中的变长数组 C99	122
7.1 整数类型	88	问与答	123
7.1.1 C99中的整数类型 C99	90	练习题	124
7.1.2 整数常量	90	编程题	125
7.1.3 C99中的整数常量 C99	91	第9章 函数	129
7.1.4 整数溢出	91	9.1 函数的定义和调用	129
7.1.5 读/写整数	91	程序 计算平均值	129
程序 数列求和(改进版)	92	程序 显示倒数	130
7.2 浮点类型	93	程序 显示双关语(改进版)	131
7.2.1 浮点常量	94	9.1.1 函数定义	132
7.2.2 读/写浮点数	94		

9.1.2 函数调用	133	第 12 章 指针和数组	184
程序 判定素数	134	12.1 指针的算术运算	184
9.2 函数声明	135	12.1.1 指针加上整数	185
9.3 实际参数	136	12.1.2 指针减去整数	185
9.3.1 实际参数的转换	137	12.1.3 两个指针相减	186
9.3.2 数组型实际参数	138	12.1.4 指针比较	186
9.3.3 变长数组形式参数 C99	140	12.1.5 指向复合常量的指针 C99	186
9.3.4 在数组参数声明中使用 static C99	141	12.2 指针用于数组处理	186
9.3.5 复合字面量 C99	141	12.3 用数组名作为指针	188
9.4 return 语句	142	程序 数列反向 (改进版)	189
9.5 程序终止	143	12.3.1 数组型实际参数 (改进版)	189
9.6 递归	144	12.3.2 用指针作为数组名	191
程序 快速排序	146	12.4 指针和多维数组	191
问与答	147	12.4.1 处理多维数组的元素	191
练习题	150	12.4.2 处理多维数组的行	192
编程题	153	12.4.3 处理多维数组的列	192
第 10 章 程序结构	155	12.4.4 用多维数组名作为指针	192
10.1 局部变量	155	12.5 C99 中的指针和变长数组 C99	193
10.1.1 静态局部变量	156	问与答	194
10.1.2 形式参数	156	练习题	195
10.2 外部变量	156	编程题	197
10.2.1 示例: 用外部变量实现栈	156	第 13 章 字符串	198
10.2.2 外部变量的利与弊	157	13.1 字符串字面量	198
程序 猜数	158	13.1.1 字符串字面量中的转义序列	198
10.3 程序块	161	13.1.2 延续字符串字面量	199
10.4 作用域	162	13.1.3 如何存储字符串字面量	199
10.5 构建 C 程序	163	13.1.4 字符串字面量的操作	200
程序 给一手牌分类	163	13.1.5 字符串字面量与字符常量	200
问与答	169	13.2 字符串变量	200
练习题	169	13.2.1 初始化字符串变量	201
编程题	170	13.2.2 字符数组与字符指针	202
第 11 章 指针	172	13.3 字符串的读和写	203
11.1 指针变量	172	13.3.1 用 printf 函数和 puts 函数 写字符串	203
11.2 取地址运算符和间接寻址运算符	173	13.3.2 用 scanf 函数和 gets 函数读 字符串	203
11.2.1 取地址运算符	173	13.3.3 逐个字符读字符串	204
11.2.2 间接寻址运算符	174	13.4 访问字符串中的字符	205
11.3 指针赋值	174	13.5 使用 C 语言的字符串库	206
11.4 指针作为参数	176	13.5.1 strcpy 函数	207
程序 找出数组中的最大元素和最 小元素	177	13.5.2 strlen 函数	208
11.5 指针作为返回值	179	13.5.3 strcat 函数	208
问与答	180	13.5.4 strcmp 函数	209
练习题	181	程序 显示一个月的提醒列表	209
编程题	182	13.6 字符串惯用法	211

13.6.1 搜索字符串的结尾	211	15.2.6 保护头文件	253
13.6.2 复制字符串	213	15.2.7 头文件中的#error指令	254
13.7 字符串数组	214	15.3 把程序划分成多个文件	254
程序 核对行星的名字	217	程序 文本格式化	255
问与答	218	15.4 构建多文件程序	260
练习题	220	15.4.1 makefile	260
编程题	222	15.4.2 链接期间的错误	262
第 14 章 预处理器	225	15.4.3 重新构建程序	262
14.1 预处理器的工作原理	225	15.4.4 在程序外定义宏	264
14.2 预处理指令	227	问与答	264
14.3 宏定义	227	练习题	265
14.3.1 简单的宏	227	编程题	266
14.3.2 带参数的宏	229	第 16 章 结构、联合和枚举	267
14.3.3 #运算符	231	16.1 结构变量	267
14.3.4 ##运算符	231	16.1.1 结构变量的声明	267
14.3.5 宏的通用属性	232	16.1.2 结构变量的初始化	269
14.3.6 宏定义中的圆括号	233	16.1.3 指定初始化 C99	269
14.3.7 创建较长的宏	233	16.1.4 对结构的操作	270
14.3.8 预定义宏	234	16.2 结构类型	270
14.3.9 C99中新增的预定义宏 C99	235	16.2.1 结构标记的声明	271
14.3.10 空的宏参数 C99	236	16.2.2 结构类型的定义	272
14.3.11 参数个数可变的宏 C99	236	16.2.3 结构作为参数和返回值	272
14.3.12 __func__标识符 C99	237	16.2.4 复合字面量 C99	273
14.4 条件编译	237	16.3 嵌套的数组和结构	274
14.4.1 #if指令和#endif指令	238	16.3.1 嵌套的结构	274
14.4.2 defined运算符	238	16.3.2 结构数组	274
14.4.3 #ifdef指令和#ifndef指令	239	16.3.3 结构数组的初始化	275
14.4.4 #elif指令和#else指令	239	程序 维护零件数据库	275
14.4.5 使用条件编译	240	16.4 联合	281
14.5 其他指令	240	16.4.1 用联合来节省空间	282
14.5.1 #error指令	240	16.4.2 用联合来构造混合的数据结构	284
14.5.2 #line指令	241	16.4.3 为联合添加“标记字段”	284
14.5.3 #pragma指令	242	16.5 枚举	285
14.5.4 _Pragma运算符 C99	242	16.5.1 枚举标记和类型名	286
问与答	243	16.5.2 枚举作为整数	286
练习题	245	16.5.3 用枚举声明“标记字段”	286
第 15 章 编写大型程序	248	问与答	287
15.1 源文件	248	练习题	289
15.2 头文件	249	编程题	293
15.2.1 #include指令	249	第 17 章 指针的高级应用	294
15.2.2 共享宏定义和类型定义	250	17.1 动态存储分配	294
15.2.3 共享函数原型	251	17.1.1 内存分配函数	294
15.2.4 共享变量声明	252	17.1.2 空指针	295
15.2.5 嵌套包含	253	17.2 动态分配字符串	296

17.2.1	使用malloc函数为字符串 分配内存	296	18.4	声明符	334
17.2.2	在字符串函数中使用动态 存储分配	296	18.4.1	解释复杂声明	335
17.2.3	动态分配字符串的数组	297	18.4.2	使用类型定义来简化声明	336
	程序 显示一个月的提醒列表 (改进版)	297	18.5	初始化式	336
17.3	动态分配数组	299	18.6	内联函数 ^{C99}	337
17.3.1	使用malloc函数为数组分配 存储空间	299	18.6.1	内联定义	338
17.3.2	calloc函数	300	18.6.2	对内联函数的限制	339
17.3.3	realloc函数	300	18.6.3	在GCC中使用内联函数	339
17.4	释放存储空间	301		问与答	339
17.4.1	free函数	301		练习题	342
17.4.2	“悬空指针”问题	301	第 19 章 程序设计		345
17.5	链表	302	19.1	模块	345
17.5.1	声明结点类型	302	19.1.1	内聚性与耦合性	347
17.5.2	创建结点	303	19.1.2	模块的类型	347
17.5.3	->运算符	303	19.2	信息隐藏	347
17.5.4	在链表的开始处插入结点	304	19.3	抽象数据类型	350
17.5.5	搜索链表	306	19.3.1	封装	351
17.5.6	从链表中删除结点	307	19.3.2	不完整类型	351
17.5.7	有序链表	308	19.4	栈抽象数据类型	352
	程序 维护零件数据库(改进版)	309	19.4.1	为栈抽象数据类型定义接口	352
17.6	指向指针的指针	313	19.4.2	用定长数组实现栈抽象数据 类型	353
17.7	指向函数的指针	314	19.4.3	改变栈抽象数据类型中数据 项的类型	354
17.7.1	函数指针作为参数	314	19.4.4	用动态数组实现栈抽象数据 类型	355
17.7.2	qsort函数	314	19.4.5	用链表实现栈抽象数据类型	357
17.7.3	函数指针的其他用途	316	19.5	抽象数据类型的设计问题	359
	程序 列三角函数表	317	19.5.1	命名惯例	359
17.8	受限指针 ^{C99}	318	19.5.2	错误处理	359
17.9	灵活数组成员 ^{C99}	319	19.5.3	通用抽象数据类型	359
	问与答	320	19.5.4	新语言中的抽象数据类型	360
	练习题	323		问与答	360
	编程题	325		练习题	361
第 18 章 声明		327		编程题	362
18.1	声明的语法	327	第 20 章 底层程序设计		363
18.2	存储类型	328	20.1	位运算符	363
18.2.1	变量的性质	328	20.1.1	移位运算符	363
18.2.2	auto存储类型	329	20.1.2	按位求反运算符、按位与运 算符、按位异或运算符和按 位或运算符	364
18.2.3	static存储类型	329	20.1.3	用位运算符访问位	365
18.2.4	extern存储类型	330	20.1.4	用位运算符访问位域	366
18.2.5	register存储类型	331		程序 XOR加密	366
18.2.6	函数的存储类型	332	20.2	结构中的位域	367
18.2.7	小结	332			
18.3	类型限定符	333			

20.3 其他底层技术.....	369	22.3.8 C99对...scanf转换说明的 改变 C99.....	400
20.3.1 定义依赖机器的类型.....	369	22.3.9 scanf示例.....	400
20.3.2 用联合提供数据的多个视角.....	370	22.3.10 检测文件末尾和错误条件.....	401
20.3.3 将指针作为地址使用.....	371	22.4 字符的输入/输出.....	403
程序 查看内存单元.....	371	22.4.1 输出函数.....	403
20.3.4 volatile类型限定符.....	373	22.4.2 输入函数.....	403
问与答.....	374	程序 复制文件.....	404
练习题.....	374	22.5 行的输入/输出.....	405
编程题.....	376	22.5.1 输出函数.....	405
第 21 章 标准库	377	22.5.2 输入函数.....	406
21.1 标准库的使用.....	377	22.6 块的输入/输出.....	406
21.1.1 对标准库中所用名字的限制.....	377	22.7 文件定位.....	407
21.1.2 使用宏隐藏的函数.....	378	程序 修改零件记录文件.....	409
21.2 C89标准库概述.....	378	22.8 字符串的输入/输出.....	409
21.3 C99标准库更新.....	380	22.8.1 输出函数.....	410
21.4 <stddef.h>: 常用定义.....	381	22.8.2 输入函数.....	410
21.5 <stdbool.h>: 布尔类型和值 C99.....	381	问与答.....	411
问与答.....	382	练习题.....	414
练习题.....	382	编程题.....	416
编程题.....	383	第 23 章 库对数值和字符数据的支持	419
第 22 章 输入/输出	384	23.1 <float.h>: 浮点类型的特性.....	419
22.1 流.....	384	23.2 <limits.h>: 整数类型的大小.....	421
22.1.1 文件指针.....	385	23.3 <math.h>: 数学计算 (C89).....	422
22.1.2 标准流和重定向.....	385	23.3.1 错误.....	422
22.1.3 文本文件与二进制文件.....	385	23.3.2 三角函数.....	422
22.2 文件操作.....	386	23.3.3 双曲函数.....	423
22.2.1 打开文件.....	386	23.3.4 指数函数和对数函数.....	423
22.2.2 模式.....	387	23.3.5 幂函数.....	424
22.2.3 关闭文件.....	388	23.3.6 就近取整函数、绝对值函数 和取余函数.....	424
22.2.4 为打开的流附加文件.....	389	23.4 <math.h>: 数学计算 (C99) C99.....	425
22.2.5 从命令行获取文件名.....	389	23.4.1 IEEE浮点标准.....	425
程序 检查文件是否可以打开.....	389	23.4.2 类型.....	426
22.2.6 临时文件.....	390	23.4.3 宏.....	426
22.2.7 文件缓冲.....	391	23.4.4 错误.....	426
22.2.8 其他文件操作.....	392	23.4.5 函数.....	427
22.3 格式化的输入/输出.....	393	23.4.6 分类宏.....	427
22.3.1 ...printf函数.....	393	23.4.7 三角函数.....	428
22.3.2 ...printf转换说明.....	393	23.4.8 双曲函数.....	428
22.3.3 C99对...printf转换说明的 修改 C99.....	395	23.4.9 指数函数和对数函数.....	429
22.3.4 ...printf转换说明示例.....	396	23.4.10 幂函数和绝对值函数.....	430
22.3.5 ...scanf函数.....	398	23.4.11 误差函数和伽玛函数.....	430
22.3.6 ...scanf格式串.....	398	23.4.12 就近取整函数.....	431
22.3.7 ...scanf转换说明.....	398	23.4.13 取余函数.....	432
		23.4.14 操作函数.....	432

23.4.15	最大值函数、最小值函数 和正差函数	433	25.3.2	双字符	466
23.4.16	浮点乘加	433	25.3.3	<iso646.h>: 拼写替换	467
23.4.17	比较宏	434	25.4	通用字符名 C99	467
23.5	<ctype.h>: 字符处理	434	25.5	<wchar.h>: 扩展的多字节和宽字符 实用工具 C99	467
23.5.1	字符分类函数	435	25.5.1	流倾向	468
程序	测试字符分类函数	436	25.5.2	格式化宽字符输入/输出 函数	468
23.5.2	字符大小写映射函数	437	25.5.3	宽字符输入/输出函数	470
程序	测试大小写映射函数	437	25.5.4	通用的宽字符串实用工具	471
23.6	<string.h>: 字符串处理	437	25.5.5	宽字符时间转换函数	474
23.6.1	复制函数	437	25.5.6	扩展的多字节/宽字符转换 实用工具	474
23.6.2	拼接函数	438	25.6	<wctype.h>: 宽字符分类和映射实 用工具 C99	477
23.6.3	比较函数	439	25.6.1	宽字符分类函数	477
23.6.4	搜索函数	440	25.6.2	可扩展的宽字符分类函数	478
23.6.5	其他函数	442	25.6.3	宽字符大小写映射函数	478
问与答		442	25.6.4	可扩展的宽字符大小写映射 函数	478
练习题		443	问与答		479
编程题		444	练习题		479
第 24 章	错误处理	446	编程题		480
24.1	<assert.h>: 诊断	446	第 26 章	其他库函数	482
24.2	<errno.h>: 错误	447	26.1	<stdarg.h>: 可变参数	482
24.3	<signal.h>: 信号处理	448	26.1.1	调用带有可变参数列表的 函数	483
24.3.1	信号宏	449	26.1.2	v...printf 函数	484
24.3.2	signal 函数	449	26.1.3	v...scanf 函数 C99	485
24.3.3	预定义的信号处理函数	450	26.2	<stdlib.h>: 通用的实用工具	485
24.3.4	raise 函数	450	26.2.1	数值转换函数	485
程序	测试信号	451	程序	测试数值转换函数	486
24.4	<setjmp.h>: 非局部跳转	452	26.2.2	伪随机序列生成函数	488
程序	测试 setjmp 和 longjmp	452	程序	测试伪随机序列生成函数	488
问与答		453	26.2.3	与环境的通信	489
练习题		454	26.2.4	搜索和排序实用工具	490
第 25 章	国际化特性	456	程序	确定航空里程	491
25.1	<locale.h>: 本地化	456	26.2.5	整数算术运算函数	492
25.1.1	类别	456	26.3	<time.h>: 日期和时间	492
25.1.2	setlocale 函数	457	26.3.1	时间处理函数	493
25.1.3	localeconv 函数	458	26.3.2	时间转换函数	495
25.2	多字节字符和宽字符	461	程序	显示日期和时间	498
25.2.1	多字节字符	461	问与答		500
25.2.2	宽字符	461	练习题		501
25.2.3	统一码和通用字符集	462			
25.2.4	统一码编码	462			
25.2.5	多字节/宽字符转换函数	463			
25.2.6	多字节/宽字符串转换函数	465			
25.3	双字符和三字符	465			
25.3.1	三字符	465			