



DEITEL

C How to Program, Sixth Edition

# C 语言大学教程

—— (第六版) ——



### C 过程式程序设计

- 控制语句
- 程序开发
- 函数 ■ 数组 ■ 指针
- 字符串 ■ 输入/输出
- 文件 ■ 结构体 ■ 共用体
- “更上一层楼”练习题
- 位操作 ■ 枚举
- C99 ■ 数据结构 ■ 游戏编程
- GNU gdb与Visual C++调试器

### C++ 面向对象程序设计

- C++, 更好的C ■ 输入/输出
- 类 ■ 对象 ■ 重载 ■ 继承
- 多态 ■ 模板 ■ 异常处理

国外计算机科学教材系列

# C 语言大学教程

## (第六版)

C How To Program

Sixth Edition

[美] Paul Deitel 著  
~~Harvey Deitel~~

苏小红 李东 王甜甜 等译

电子工业出版社  
Publishing House of Electronics Industry  
北京·BEIJING



## 内 容 简 介

本书是全球畅销的 C 语言教程之一。全书系统地介绍了 4 种当今流行的程序设计方法——面向过程、基于对象、面向对象以及泛型编程,内容全面、生动、易懂,作者由浅入深地介绍了结构化编程及软件工程的基本概念,从简单的概念到最终的完整的语言描述,清晰、准确、透彻、详细地讲解了 C 语言,尤其注重程序设计思想和方法的介绍。

相对于上一版,这一版在内容方面新增加了 C 安全程序设计、“更上一层楼”练习集,更新了 C++ 和面向对象程序设计、基于 Allegro 的游戏编程、C99 标准介绍等内容。

“活代码”方法(即通过可执行的代码来讲解理论概念的方法)是本书的另一特色,它可以促进学生积极地参与到程序设计中来。被突出显示的各种程序设计提示,则是作者多年教学的经验总结。

本书不仅适合于初学者学习,作为高校计算机程序设计教学的教科书,也同样适用于有经验的程序员,作为软件开发人员的专业参考书。

Authorized translation from the English language edition, entitled C How to Program, Sixth Edition, ISBN: 9780136123569 by Paul J. Deitel and Harvey M. Deitel, published by Pearson Education, Inc., publishing as Prentice Hall, Copyright ©2010 Pearson Education, Inc.

All rights reserved. No part of this book may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording or by any information storage retrieval system, without permission from Pearson Education, Inc.

CHINESE SIMPLIFIED language edition published by PUBLISHING HOUSE OF ELECTRONICS INDUSTRY, Copyright ©2012.

本书简体中文版由 Pearson Education 培生教育出版集团授予电子工业出版社,未经许可,不得以任何方式复制或抄袭本书的任何部分。

本书简体中文版贴有 Pearson Education 培生教育出版集团激光防伪标签,无标签者不得销售。

版权贸易合同登记号 图字:01-2009-7066

### 图书在版编目(CIP)数据

C 语言大学教程:第 6 版/(美)戴特尔(Deitel,H.), (美)戴特尔(Deitel,P.)著;苏小红等译.

北京:电子工业出版社,2012.1

国外计算机科学教材系列

书名原文:C How to Program, Sixth Edition

ISBN 978-7-121-15221-4

I. ①C… II. ①戴… ②戴… ③苏… 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(2011)第 241382 号

策划编辑:许菊芳

责任编辑:许菊芳

印刷:北京市顺义兴华印刷厂

装订:三河市双峰印刷装订有限公司

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:47.25 字数:1437 千字

印次:2012 年 1 月第 1 次印刷

定价:89.00 元(含 DVD 光盘 1 张)

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。

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

服务热线:(010)88258888。

# 前 言

欢迎学习 C/C++ 程序设计语言。本书为学生、教师和软件开发专业人员而编写,介绍了计算机领域的最新技术。

本书作者 Deitel 博士提出的“活代码方法”(live-code approach)是本书的核心内容,即概念是通过完整的可执行的程序而非代码片段来介绍的。在每个程序例子的后面都给出了一个或多个运行的实例。所有的程序源代码都可以从 [www.deitel.com/books/cht6](http://www.deitel.com/books/cht6) 网站下载。

我们相信本书及其辅助材料将为读者学习 C 语言提供一个内容丰富、快乐有趣而又极富挑战性的学习体验。

在阅读本书时,若有问题,请发送电子邮件到: [deitel@deitel.com](mailto:deitel@deitel.com),我们将及时答复。欢迎访问 [www.deitel.com](http://www.deitel.com),及时了解本书及其 C/C++ 软件的更新以及 Deitel 公司出版和服务等方面的最新消息。

## 新的和改动的特色

下面是本书提供的新内容。

- **“更上一层楼”练习集。**我们鼓励读者使用计算机和因特网来研究和解决实际问题,这些练习有助于增加读者对当今世界所面临重要问题的了解。希望读者用自己的观点来解决这些问题。
- **在 Windows 和 Linux 上测试了所有的代码。**我们使用 Visual C++ 2008 和 GNU GCC 4.3 测试了所有的程序(例子和练习)。例子程序和练习的答案也都在 Visual Studio 2010 Beta 上测试过。
- **新的设计。**本书的内部设计做了更新,以图形的形式来使内容层次清晰、重点突出,并增强了本书的教育功能。
- **改进的术语表。**为了便于查找,对于术语表中的每一项,都增加了节号来表示它所在的位置。
- **更新了 C++ 和面向对象程序设计的内容。**用我们刚刚出版的《C++ 大学教程》(第七版)的一些材料更新了本书第 15 章至第 24 章关于基于 C++ 的面向对象程序设计的内容。
- **给编程练习题增加标题。**本书给所有的编程练习题增加了标题,以帮助教师针对不同班级调整所布置的作业。
- **新的在线附录。**上一版本的第 15 章至第 17 章现在改成了可以从本书配套网站下载的附录 E 至附录 G(PDF 格式)。
- **新的调试器附录。**增加了介绍 Visual C++ 2008 和 GNU gdb 调试内容的附录。
- **表达式求值顺序。**增加了关于表达式求值顺序问题的忠告。
- **用 fgets 函数替换 gets 函数。**由于反对使用 gets 函数(来自 `<stdio.h>`)的呼声很高,所以用 fgets 函数替换了原书中使用的所有函数 gets。
- **增加了练习题。**本书增加了很多有关函数指针的练习题。还增加了旨在改进 Fibonacci 递归例子(尾递归)的 Fibonacci 课后大作业。
- **C 安全程序设计资源中心。**我们在 [www.deitel.com/SecureC/](http://www.deitel.com/SecureC/)上新建了 C 安全程序设计资源中心。还在本书第 7 章至第 8 章中增加了关于 C 安全程序设计的注意事项。
- **基于 Allegro 的游戏编程。**我们更新了有关基于 Allegro C 库函数的游戏编程的章节。特别地,增加了在 Visual C++ 2008 和 GNU GCC 4.3 平台上 Allegro 库的安装指南。
- **C99 标准的范围。**更新并加强了介绍 C99 的详细的附录,并邀请 ISO WG14(负责制定 C 程序设计语言标准的工作组)的召集人 John Benito 审阅这部分内容。每一个 C99 的概念都与本书中前面讲授过的章节相联系。由于微软公司一直不支持 C99,而大多数 C 语言课程又都采用微软公司的

Visual C++ 编译器,所以没有把 C99 的介绍贯穿在整个本书中。若想了解更多关于 C99 的信息,请访问 C 语言资源中心 [www.deitel.com/C/](http://www.deitel.com/C/) 中“C99 标准”专区。在那里,您将会看到关于 C99 特色的介绍、专家撰写的文章、C 标准和 C99 标准的差别、常见问题 FAQ、下载专区等。

- **C++ 风格的注释**//。本书尽早向喜欢采用 C++ 风格注释//的教师和学生介绍这方面的内容。虽然 Microsoft C 并不支持 C99,但它支持 C99 的注释,而 C99 的注释就是借鉴了 C++。
- **C 标准库函数**。1.8 节参考了 P. J. Plauger 的 Dinkumware 网站([www.dinkumware.com/manuals/default.aspx](http://www.dinkumware.com/manuals/default.aspx))。学生可以登录该网站查找 C 标准库函数的完整文档。

## 其他特色

本书的其他特色如下。

### 基于 Allegro C 游戏程序设计库的游戏程序设计

附录 E 介绍了 Allegro C 游戏程序设计库。尽管与其他如 DirectX 和 OpenGL 那些相对复杂的图形函数库相比还略显简单,但是这个库——最初是由 Climax 公司的游戏程序员 Shawn Hargreaves 开发的——不失为一个面向 C 语言的功能强大的游戏程序设计库。附录 E 中用 Allegro 的功能创作了一个简单的游戏 Pong。在创作过程中,演示了如何显示图形、播放声音、接受键盘输入以及创建定时事件等,在创作游戏程序时,您可能会用到这些技术。无论是学生还是教师,都会从 Allegro 找到挑战和成就感。作为提供超过 1000 个开源 Allegro 游戏的网站之一,Allegro 资源中心([www.deitel.com/allegro](http://www.deitel.com/allegro))包含有大量的网络资源。

### 排序:深入探究

排序就是基于一个或几个关键值将一组数据按照某种顺序排列。第 6 章从一个简单的算法开始介绍排序。附录 F 中给出对排序的深入探究。将从消耗的存储空间和处理器时间两方面来对若干个排序算法进行分析和比较。为此,将引入用于表示一个算法为解决问题需要付出代价大小的大 O 记号。通过例题和课后练习,附录 F 将讨论选择排序、插入排序、递归的归并排序、递归的选择排序、桶式排序和递归的快速排序。

## 网上的辅导材料

有大量的在线辅导材料帮助读者学习本书。本书的配套网站([www.pearsonhighered.com/deitel](http://www.pearsonhighered.com/deitel))包含了下列 PDF 格式的附录:

- 附录 E,基于 Allegro C 函数库的游戏程序设计
- 附录 F,排序:深入探究
- 附录 G,C99 导论
- 附录 H,Visual Studio 调试器的使用
- 附录 I,GNU 调试器的使用

## 章节依赖关系图

图 1 和图 2 给出了本书各章节的依赖关系,以帮助教师安排教学计划。本书更适合于计算机科学专业第一学期和第二学期课程以及中级的 C 和 C++ 程序设计课程。本书的 C++ 部分假设读者已经学习过 C 部分。

## 教学方法

本书包含了丰富的程序实例。本书专注于展示优秀软件工程的基本原则,并强调程序的清晰性。

**活代码方法。**本书充满了“活代码”的程序实例,绝大多数的新概念都是通过一个完整的可运行的 C 程序来介绍的,程序后面会跟随着一个或多个测试用例来显示程序执行时的输入和输出。

**语法的阴影表示。**为了提高可读性,对源代码采用语法的阴影表示,这类似于绝大多数集成开发环境和代码编辑器用不同的颜色表示不同的语法代码。约定的语法阴影表示如下:

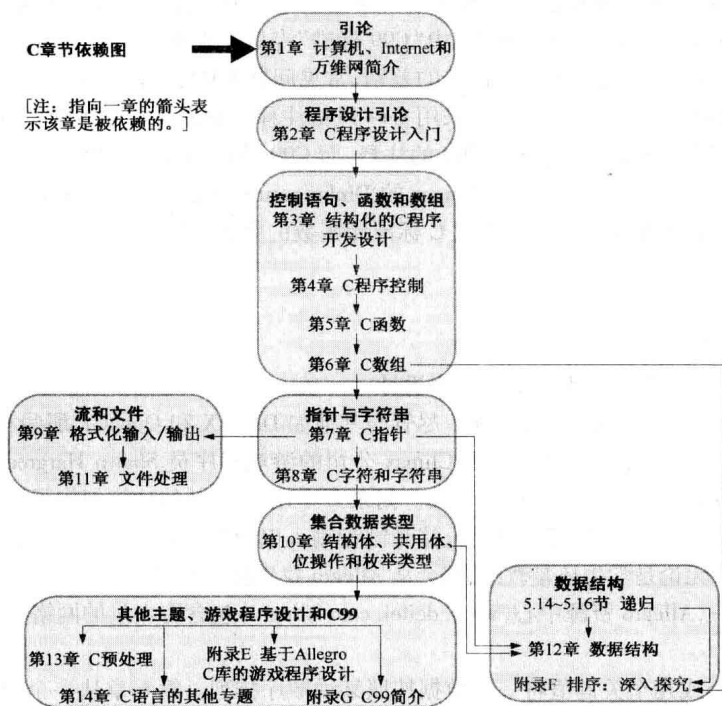


图 1 C 章节依赖图

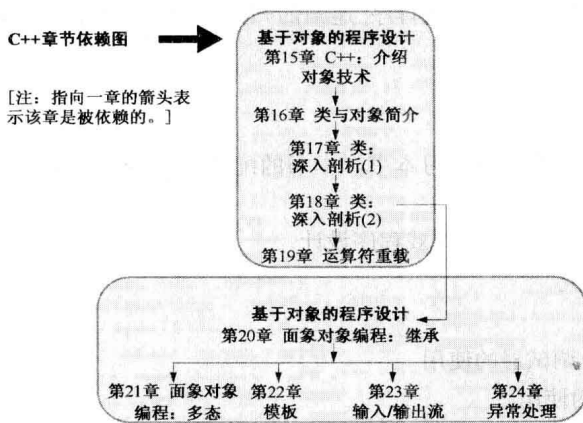


图 2 C++ 章节依赖图

注释以细字体表示

关键字以粗黑体表示

常量和字面文字以比注释稍微粗一点的字体表示

所有其他代码以黑体表示

代码的加亮突出显示。把程序中的关键代码放在一个灰色的矩形中。

用不同的字体强调关键信息。为了便于读者索引,用蓝色加粗的文本来编辑关键词和每一个定义所在的节号。用加粗的 Helvetica 字体来强调计算机屏幕上的模块(即文件菜单),用 Lucida 字体来显示 C 程序文本(例如 `int x = 5;`)<sup>①</sup>。

① 中文版中统一以宋体表示。——编者注

访问万维网。所有源代码例子都可以从如下网站下载：

[www.deitel.com/books/cht6/](http://www.deitel.com/books/cht6/)

**名人名言。**每一章都是以一些名人名言开始。我们希望您能够欣赏它们与该章内容之间的联系。

**目标。**名人名言后面就是每一章的学习目标。

**图例/图形。**书中包含了大量的流程图、表格、图形、UML 图、程序和程序输出。

**编程提示。**本书增加了编程提示,来帮助学生关注程序开发应注意的重要事项。这些提示及其应用是两位作者累积 70 年的编程和教学经验的体现。



#### 良好的编程习惯

这种提示可以提醒读者注意那些能够编写出更清晰、更易于理解和更可维护的的技术。



#### 常见的编程错误

指出这些常见的编程错误,能够减少读者犯同样错误的的可能性。



#### 错误预防提示

这些提示包含了发现软件错误并修正它们的建议。许多提示描述了防止软件错误进入 C 语言程序的措施。



#### 性能提示

性能提示突出了那些使程序运行得更快或者占用存储空间最小的可能性。



#### 可移植性提示

可移植性提示能帮助编写出可以运行于多种平台上的代码。



#### 软件工程视点

软件工程视点突出了影响软件系统(特别是大规模软件系统)的开发,在体系结构和设计两方面应注意的事项。

**摘要。**每一章的最后都提供了本章的摘要,采用逐节编排、快捷条目列表风格。

**术语表。**每一章中增加了一个按照字母表顺序排列的本章定义的重要术语列表。

**自测题及其答案。**为了便于自学,本书大量增加了自测题及其答案。

**练习题。**每章的最后都有内容充实的练习题,其中包括:

- 对重要术语和概念的简单回忆。
- 在代码例子中找出错误。
- 编写单一的 C 程序语句。
- 编写函数或类的一个小部分。
- 编写一个完整的 C 函数、类和程序。
- 主要的课程项目(大作业)。

教师可以用这些练习题来布置家庭作业、进行小测验、课程考试和学期项目(大作业)。[注:请不要写信给我们要求访问 PrenticeHall 出版社的教师资源中心。访问权限严格限制给采用本书进行教学的大学教师。教师只能通过 PrenticeHall 出版社的代表获得访问权限。没有为课程项目(大作业)提供答案。]若想得到更多的附加练习题和可选课程项目(大作业),请访问程序设计项目资源中心([www.deitel.com/ProgrammingProjects/](http://www.deitel.com/ProgrammingProjects/))<sup>①</sup>。

**索引。**本书增加了一个内容丰富的索引,它对于将本书作为参考书的研发人员特别有用。

<sup>①</sup> 国内教师申请本书教学资源的方法,请参考本书末尾的“教学支持说明”页。

## 学生资源

许多 C 和 C++ 开发工具是现成的。本书使用微软公司免费的 Visual C++ Express Edition(可以从网站 [www.microsoft.com/express/vc/](http://www.microsoft.com/express/vc/) 上免费下载)和已经在 Linux 系统上广泛使用的免费 GNU C++ (gcc. [gnu.org/install/binaries.html](http://gnu.org/install/binaries.html))。GNU C++ 也可以在 Mac OS X 和 Windows 系统上安装使用。如想更深入地学习 Visual C++ Express Edition,可以访问 [msdn.microsoft.com/vstudio/express/visualc](http://msdn.microsoft.com/vstudio/express/visualc)。如想更深入地学习 GNU C++,还可以访问 [gcc.gnu.org](http://gcc.gnu.org)。苹果公司已经将 GNU C++ 包含到 Xcode 开发工具中,Mac OS X 的用户可以从 [developer.apple.com/tools/xcode](http://developer.apple.com/tools/xcode) 下载 Xcode。

可以从如下网站下载本书的例子和附加资源:

[www.deitel.com/books/cht6/](http://www.deitel.com/books/cht6/)

若想下载附加资源和软件,请访问 C 资源中心:

[www.deitel.com/c/](http://www.deitel.com/c/)

下面两个网站上还有一些其他的 C 和 C++ 编译器可供免费下载:

[www.thefreecountry.com/developercity/ccompilers.shtml](http://www.thefreecountry.com/developercity/ccompilers.shtml)

[www.compilers.net/Dir/Compilers/CCpp.htm](http://www.compilers.net/Dir/Compilers/CCpp.htm)

## CourseSmart 网络书库

对于现在的学生和教师,需要花费时间与金钱的地方越来越多。培生出版公司(Pearson)通过 CourseSmart 提供在线的数字课本和课程辅导材料,以节省学生和教师的时间和金钱。在 CourseSmart 上,教师可以在线浏览课程辅导材料,学生只需本书印刷版一半的价钱就可以获得完全一样的高质量的数字版本。这样,学生不仅获得了印刷版教材的全部内容,而且还可以拥有搜索、记笔记和打印的工具。欲了解更多信息,请访问 [www.coursesmart.com](http://www.coursesmart.com)。

## 本书提供的软件

本书附有了 Microsoft Visual Studio 2008 Express Editions All-in-One 的 DVD 盘,其中包含了 Visual C++ 2008 Express Edition(以及其他开发工具)。也可以从下面的网站下载 Visual C++ Express Edition 的最新版本:

[www.microsoft.com/express/vc](http://www.microsoft.com/express/vc)

在微软公司的每个网址中,Express Edition 的宣传语都是“轻量级的,对于业余爱好者、初学者和学生是易用和易学的”。它们还适合专业培训课程以及还没有接触过 Visual Studio 2008 完整版本的专业人员。

除了第 9 章中的一个例子和附录 G 中的例子外,本书的所有例子都编译并运行于 Visual C++ 2008 和 Visual C++ 2010 Beta 上。所有例子也都编译并运行于 GNU GCC 4.3 上。绝大多数平台,包括 Linux、Mac OS X(通过 Xcode)和装有诸如 Cygwin([www.cygwin.com](http://www.cygwin.com))和 MinGW([www.mingw.org](http://www.mingw.org))等工具的 Windows,都有 GCC。

## 为教师提供的资源

下列资源只在培生教育(Pearson Education)的教师资源中心([www.pearsonhighered.com/irc](http://www.pearsonhighered.com/irc))上向教师提供:

- 每章后面的绝大多数练习题的答案手册。
- 多选题(几乎每节两个问题)的试题文件。
- 包含书中所有代码和图以及总结教材中关键点的快捷条目的定制的 PowerPoint 幻灯片。

如果您还不是一个已注册的教师,请联系培生公司的代表或者访问 [www.pearsonhighered.com/educator/relocator/](http://www.pearsonhighered.com/educator/relocator/)<sup>①</sup>。

---

<sup>①</sup> 国内教师申请教学资源的方法,请参见本书末尾的“教学支持说明”页。



## Deitel Buzz Online 免费的电子邮件新闻快报

电子邮件新闻快报 Deitel Buzz Online 将随时向您邮寄与本书相关的问题。它还包含有关于行业发展趋势和研发成果的评论、与我们已出版和即将出版的书籍中的免费资料和资源链接、产品发布计划、勘误表、挑战问题、奇闻轶事、我们举办的教师指导的培训课程信息等。欢迎访问如下网址来订阅：

[www.deitel.com/newsletter/subscribe.html](http://www.deitel.com/newsletter/subscribe.html)

## Deitel 的在线资源中心

[www.deitel.com](http://www.deitel.com) 网站提供了一百多个主题资源中心,这些主题有:程序设计语言、软件开发、Web 2.0、因特网商业以及开发源代码项目等,详细情况请查看本书前几页的资源中心列表或访问 [www.deitel.com/ResourceCenters.html](http://www.deitel.com/ResourceCenters.html)。我们还发现了许多特殊的在线资源,例如辅导材料、软件下载、文章、博客、视频、代码样本、书籍、电子书籍等,它们绝大多数是免费的。我们会在每周的 Deitel Buzz Online 新闻快报上宣布新的资源中心。在学习本书的过程中,可能对您有用的资源中心包括 C、C++、C++ Boost 库、C++ 游戏程序设计、Visual C++、UML、代码搜索引擎和代码网址、游戏程序设计和编程项目。

## 在 Twitter 和 Facebook 上紧跟 Deitel

若想及时接收到有关 Deitel 出版物、资源中心、培训课程、合作伙伴提供等信息,请在 Twitter 上紧跟我们:

@deitel

或者在 Facebook 上加入 Deitel & Associates 组中:

[www.deitel.com/deitelfan/](http://www.deitel.com/deitelfan/)

## 致谢

很高兴在此表达对那些名字没有出现在封面的人士的谢意!他们的辛勤工作、配合、友谊及理解,对于这本书的诞生是至关重要的。Deitel & Associates 公司的许多员工为这本书的编撰付出了大量的时间,其中特别要感谢 Abbey Deitel 和 Barbara Deitel。

我们还想感谢两位为这本书的出版作出重要贡献的荣誉实习合作伙伴——美国康奈尔大学运行研究和信息工程的负责人 Christine Cheb 和康奈尔大学计算机科学专业的学生 Matthew Pearson。

荣幸地能与能干敬业的培生出版社的专业出版团队共同工作。我们对培生出版社工程与计算机科学分社的编辑主管 Marcia Horton 和计算机科学的主编 Michael Hirsch 所付出的努力表示感谢。为招募本书的审稿队伍和管理审稿过程,Carole Snyder 做了大量的工作。Francesco Santalucia(一位独立艺术家)和培生出版社的 Kristine Carney 设计了本书的封面——我们提出想法,他们将其变为现实。Scott Disanno 和 Bob Engelhardt 为管理本书的出版做出了令人惊叹的工作。Erin Davis 和 Margaret Waples 积极地通过学术界和专业界的渠道在市场上推广这本书。

## 本书的审稿人

我们向下列审稿人的辛勤工作表示感谢。在很紧的时间计划内,他们仔细审阅了书稿和程序,为改进本书的精确性和表述的完整性提出了无数的建议。

- Blue Pilot 咨询公司的 John Benito,他也是负责 C 程序设计语言标准的工作组 ISO WG14 的召集人
- Xiaolong Li,印地安纳州立大学
- Tom Rethard,得克萨斯州立大学 Arlington 分校

## 本书上一版的审稿人

- Alireza Fazelpour (Palm Beach 社区学院)
- Don Kostuch (独立咨询师)

- Ed James Beckham (Altera 公司)
- Gary Sibbitts (St. Louis 社区学院 Meramec 分校)
- Ian Barland (Radford 大学)
- Kevin Mark Jones (惠普公司)
- Mahesh Hariharan (微软公司)
- William Mike Miller (爱迪生设计集团有限公司)
- Benjamin Seyfarth (Southern Mississippi 大学)
- William Albrecht (South Florida 大学)
- William Smith (Tulsa 社区学院)

#### 本书上一版中 Allegro 的审稿人

- Shawn Hargreaves (微软 Xbox 软件设计工程师)
- Matthew Leverton (Allegro.cc 网站的创始人和网站管理人)
- Ryan Patterson (独立咨询师)
- Douglas Walls (太阳微系统公司 C 编译器部门的资深工程师)

#### 本书上一版中 C99 的审稿人

- Lawrence Jones (UGS 公司)
- Douglas Walls (太阳微系统公司 C 编译器部门的资深工程师)

好! 现在您已经拿到这本书了。C 语言是一种能够帮助您快速而有效地编写程序的功能强大的程序设计语言。C 语言已经很好地进入企业系统开发的领域,帮助各类组织构建商业攸关和任务攸关的信息系统。当您阅读这本书时,我们真诚地希望能得到您的旨在改进本书的意见、批评、更正和建议。请将信件邮寄到:

deitel@deitel.com

我们将会及时回复,同时也会在如下的网站上发布更改和澄清信息:

[www.deitel.com/books/cht6/](http://www.deitel.com/books/cht6/)

希望您享受与本书一起工作的快乐时光,就像我们编写它时那样!

Paul Deitel  
Harvey Deitel  
Maynard, Massachusetts  
2009 年 8 月

#### 作者简介

Paul J. Deitel, Deitel & Associates 公司的 CEO 与 CTO, MIT Sloan 管理学院信息技术专业毕业。在 Deitel & Associates 公司,他为包括 Cisco 公司、IBM 公司、太阳微系统公司、Dell 公司、Lucent 技术公司、Fidelity 公司、NASA 肯尼迪航天中心、国家 Severe Storm 实验室、白沙导弹射击场、Rogue Wave 软件公司、波音公司、SunGard 高等教育公司、Stratus 公司、剑桥技术伙伴、Open Environment 公司、One Wave 公司、Hyperion 软件公司、Adra Systems 公司、Entergy 公司、CableData Systems 公司、Nortel 网络公司、Puma 公司、iRobot 公司和 Invensys 公司等在内的众多工业客户讲授过 C、C++、Java、C#、Visual Basic 和因特网编程等课程。他拥有 Java 认证程序员和 Java 认证开发员证书,曾被太阳微系统公司评选为 Java 优胜者。他还为美国计算机协会波士顿分会讲授 Java 和 C++ 课程。他和他的合作者 Harvey M. Deitel 博士是世界上最畅销的程序设计语言教材的作者。

Harvey M. Deitel 博士, Deitel & Associates 有限公司的主席与首席战略官。在计算机领域拥有 48 年的学术界和工业界工作经历。Deitel 博士在 MIT 获得学士和硕士学位,在波士顿大学获得博士学位。他具有丰富的大学教学经历,包括在与儿子 Paul J. Deitel 创立 Deitel & Associates 有限公司之前就获得波士顿学

院的终生职位并担任该学院计算机科学系的主席。他与 Paul 合作编写了几十本书籍和多媒体音像包,并仍然在撰写更多的书籍。由于被翻译成日文、德文、俄文、繁体中文、简体中文、西班牙文、韩文、法文、波兰文、意大利文、葡萄牙文、希腊文、乌尔都文和土耳其文,他们的教材获得了国际赞誉与认可。Deitel 博士已经为著名大公司、学术机构、政府组织和军方做了数百次专业报告。

## 关于 Deitel & Associates 公司

Deitel & Associates 公司是一家国际知名的写作和团队培训公司,特别是在计算机程序设计语言、因特网和万维网软件技术、对象技术教育和 iPhone 应用开发等领域。在例如 C、C++、Visual C++、Java、Visual C#、Visual Basic、XML、Python、对象技术、因特网与万维网程序设计、iPhone 程序设计以及其他程序设计与软件开发相关课程等主要的程序设计语言及平台方面,公司面向世界各地的客户提供教师指导的课程。Deitel & Associates 公司的创始人是 Paul J. Deitel 先生和 Harvey M. Deitel 博士。公司的客户包括很多世界级的大公司、政府部门、军方单位和学术机构。经过与培生出版社 33 年的出版合作,Deitel & Associates 公司出版了先进的程序设计教材、专业书籍、交互式多媒体出版物 Cyber Classrooms、视频课程 LiveLessons(可登录 [www.safaribooksonline.com](http://www.safaribooksonline.com) 在线观看或从 [www.deitel.com/books/livelessons/](http://www.deitel.com/books/livelessons/) 订购 DVD),并为流行的课程管理系统提供电子化内容。

可以通过如下电子邮箱来与 Deitel & Associates 公司及其作者联系:

[deitel@deitel.com](mailto:deitel@deitel.com)

欲了解更多的关于 Deitel & Associates 有限公司及其出版物和它为世界各地客户提供的 Dive Into 系列团队培训计划,请访问:

[www.deitel.com/training/](http://www.deitel.com/training/)

并通过如下网址来订阅免费的 Deitel Buzz Online 电子邮件新闻快报:

[www.deitel.com/newsletter/subscribe.html](http://www.deitel.com/newsletter/subscribe.html)

个人欲购买 Deitel 书籍、LiveLessons DVD 和基于 Web 的训练课程,请访问 [www.deitel.com](http://www.deitel.com)。公司、政府组织、军方和学术机构的大批订购应与培生出版社直接联系。欲了解更多的信息,请访问:[www.prenhall.com/misctm/support.html#order](http://www.prenhall.com/misctm/support.html#order)。

# 目 录

|   |    |
|---|----|
| <b>第 1 章 计算机、Internet 和万维网导论</b> .....                              | 1  |
| 1.1 引言 .....  | 2  |
| 1.2 计算机:硬件和软件 .....   | 3  |
| 1.3 计算机组成 .....   | 3  |
| 1.4 个人计算、分布式计算和客户/服务器计算 .....                                       | 4  |
| 1.5 因特网和万维网 .....   | 4  |
| 1.6 机器语言、汇编语言和高级语言 .....  | 4  |
| 1.7 C 语言的发展历史 .....   | 5  |
| 1.8 C 语言标准库 .....   | 6  |
| 1.9 C++ 语言 .....  | 7  |
| 1.10 Java 语言 .....  | 7  |
| 1.11 著名的高级语言 Fortran, COBOL, Pascal 和 Ada .....                     | 7  |
| 1.12 BASIC, Visual Basic, Visual C++, Visual C# 语言和 .NET 编程平台 ..... | 8  |
| 1.13 软件发展的重要趋势:对象技术 .....   | 8  |
| 1.14 典型的 C 语言程序开发环境 .....   | 9  |
| 1.15 硬件的发展趋势 .....  | 11 |
| 1.16 关于 C 语言和本书的一些注解 .....  | 11 |
| 1.17 有价值的 Web 站点 .....  | 12 |
| <b>第 2 章 C 语言程序设计入门</b> .....                                       | 19 |
| 2.1 引言 .....  | 20 |
| 2.2 一个简单的 C 程序:打印一行文字 .....   | 20 |
| 2.3 另一个简单的 C 程序:两个整数求和 .....  | 23 |
| 2.4 存储单元的基本概念 .....   | 26 |
| 2.5 C 语言中的算术运算 .....  | 27 |
| 2.6 做出判断:相等和关系运算符 .....   | 30 |
| <b>第 3 章 结构化的 C 程序设计</b> .....                                      | 43 |
| 3.1 引言 .....  | 44 |
| 3.2 算法 .....  | 44 |
| 3.3 伪码 .....  | 44 |
| 3.4 控制结构 .....  | 44 |
| 3.5 if 选择语句 .....   | 46 |
| 3.6 if...else 选择语句 .....  | 47 |
| 3.7 while 循环语句 .....  | 49 |
| 3.8 算法设计案例 1:计数控制的循环 .....  | 50 |
| 3.9 自顶向下、逐步求精的算法设计案例 2:标记控制的循环 .....                                | 52 |
| 3.10 自顶向下、逐步求精的算法设计案例 3:嵌套的控制结构 .....                               | 57 |
| 3.11 赋值运算符 .....  | 60 |
| 3.12 增 1 和减 1 运算符 .....   | 60 |



|                                 |     |
|---------------------------------|-----|
| <b>第 4 章 C 程序控制</b> .....       | 77  |
| 4.1 引言 .....                    | 78  |
| 4.2 循环的基本原理 .....               | 78  |
| 4.3 计数控制的循环 .....               | 78  |
| 4.4 for 循环语句 .....              | 79  |
| 4.5 for 循环语句:注意事项 .....         | 81  |
| 4.6 应用 for 循环语句的例子 .....        | 82  |
| 4.7 switch 多重选择语句 .....         | 85  |
| 4.8 do...while 循环语句 .....       | 89  |
| 4.9 break 和 continue 语句 .....   | 90  |
| 4.10 逻辑运算符 .....                | 91  |
| 4.11 区分相等运算符(==)和赋值运算符(=) ..... | 93  |
| 4.12 结构化程序设计总结 .....            | 94  |
| <b>第 5 章 C 函数</b> .....         | 110 |
| 5.1 引言 .....                    | 111 |
| 5.2 C 语言中的程序模块 .....            | 111 |
| 5.3 数学库函数 .....                 | 112 |
| 5.4 函数 .....                    | 113 |
| 5.5 函数定义 .....                  | 113 |
| 5.6 函数原型 .....                  | 116 |
| 5.7 函数调用堆栈及活动记录 .....           | 118 |
| 5.8 头文件 .....                   | 118 |
| 5.9 按值或按引用调用函数 .....            | 119 |
| 5.10 随机数的生成 .....               | 119 |
| 5.11 程序案例:运气游戏 .....            | 123 |
| 5.12 存储类型 .....                 | 126 |
| 5.13 标识符作用域的规定 .....            | 128 |
| 5.14 递归 .....                   | 130 |
| 5.15 使用递归的例子:Fibonacci 数列 ..... | 133 |
| 5.16 递归与迭代 .....                | 135 |
| <b>第 6 章 C 数组</b> .....         | 154 |
| 6.1 引言 .....                    | 155 |
| 6.2 数组 .....                    | 155 |
| 6.3 数组的定义 .....                 | 156 |
| 6.4 数组的应用举例 .....               | 156 |
| 6.5 向函数传递数组 .....               | 166 |
| 6.6 数组元素的排序 .....               | 169 |
| 6.7 案例分析:采用数组来计算平均值、中值和众数 ..... | 171 |
| 6.8 数组元素的查找 .....               | 174 |
| 6.9 多下标数组 .....                 | 178 |
| <b>第 7 章 C 指针</b> .....         | 198 |
| 7.1 引言 .....                    | 199 |
| 7.2 指针变量的定义和初始化 .....           | 199 |
| 7.3 指针运算符 .....                 | 200 |

|               |                                |            |
|---------------|--------------------------------|------------|
| 7.4           | 按引用向函数传递实参 .....               | 201        |
| 7.5           | 对指针使用 const 限定符 .....          | 204        |
| 7.6           | 使用按引用调用方法实现冒泡法排序 .....         | 209        |
| 7.7           | sizeof 运算符 .....               | 211        |
| 7.8           | 指针表达式和指针运算 .....               | 213        |
| 7.9           | 指针和数组的关系 .....                 | 215        |
| 7.10          | 指针数组 .....                     | 218        |
| 7.11          | 案例学习:扑克牌洗牌和发牌模拟 .....          | 218        |
| 7.12          | 函数指针 .....                     | 222        |
| <b>第 8 章</b>  | <b>C 字符和字符串 .....</b>          | <b>242</b> |
| 8.1           | 引言 .....                       | 243        |
| 8.2           | 字符串和字符基础 .....                 | 243        |
| 8.3           | 字符处理函数库 .....                  | 244        |
| 8.4           | 字符串转换函数 .....                  | 248        |
| 8.5           | 标准输入输出库函数 .....                | 251        |
| 8.6           | 字符串处理函数库中的字符串处理函数 .....        | 254        |
| 8.7           | 字符串处理函数库中的比较函数 .....           | 256        |
| 8.8           | 字符串处理函数库中的查找函数 .....           | 258        |
| 8.9           | 字符串处理函数库中的内存函数 .....           | 262        |
| 8.10          | 字符串处理函数库中的其他函数 .....           | 265        |
| <b>第 9 章</b>  | <b>格式化输入/输出 .....</b>          | <b>279</b> |
| 9.1           | 引言 .....                       | 280        |
| 9.2           | 流 .....                        | 280        |
| 9.3           | 用 printf 函数实现格式化输出 .....       | 280        |
| 9.4           | 打印整数 .....                     | 281        |
| 9.5           | 打印浮点数 .....                    | 282        |
| 9.6           | 打印字符串或字符 .....                 | 283        |
| 9.7           | 其他的转换说明符 .....                 | 284        |
| 9.8           | 带域宽和精度的打印 .....                | 285        |
| 9.9           | 在 printf 函数的格式控制字符串中使用标记 ..... | 286        |
| 9.10          | 打印文本和转义序列 .....                | 288        |
| 9.11          | 用 scanf 函数读取格式化的输入 .....       | 289        |
| <b>第 10 章</b> | <b>结构体、共用体、位操作和枚举类型 .....</b>  | <b>300</b> |
| 10.1          | 引言 .....                       | 301        |
| 10.2          | 结构体的定义 .....                   | 301        |
| 10.3          | 结构体的初始化 .....                  | 303        |
| 10.4          | 对结构体成员的访问 .....                | 303        |
| 10.5          | 在函数中使用结构体 .....                | 304        |
| 10.6          | typedef 的使用 .....              | 305        |
| 10.7          | 实例:高性能的洗牌与发牌模拟 .....           | 305        |
| 10.8          | 共用体 .....                      | 307        |
| 10.9          | 位运算符 .....                     | 309        |
| 10.10         | 位域 .....                       | 315        |
| 10.11         | 枚举常量 .....                     | 318        |

|  |     |
|--|-----|
| <b>第 11 章 文件处理</b> .....               | 328 |
| 11.1 引言 .....                          | 329 |
| 11.2 数据的层次结构 .....                     | 329 |
| 11.3 文件与流 .....                        | 330 |
| 11.4 顺序存取文件的创建 .....                   | 330 |
| 11.5 从顺序存取文件中读取数据 .....                | 334 |
| 11.6 随机存取文件 .....                      | 338 |
| 11.7 随机存取文件的创建 .....                   | 338 |
| 11.8 随机地向一个随机存取文件中写入数据 .....           | 340 |
| 11.9 从一个随机存取文件中读取数据 .....              | 342 |
| 11.10 案例学习:事务处理程序 .....                | 343 |
| <b>第 12 章 数据结构</b> .....               | 356 |
| 12.1 引言 .....                          | 357 |
| 12.2 自引用结构体 .....                      | 357 |
| 12.3 动态内存分配 .....                      | 358 |
| 12.4 链表 .....                          | 359 |
| 12.5 堆栈 .....                          | 365 |
| 12.6 队列 .....                          | 369 |
| 12.7 树 .....                           | 373 |
| <b>第 13 章 C 预处理</b> .....              | 388 |
| 13.1 引言 .....                          | 389 |
| 13.2 #include 预处理命令 .....              | 389 |
| 13.3 #define 预处理命令:符号常量 .....          | 389 |
| 13.4 #define 预处理命令:宏 .....             | 389 |
| 13.5 条件编译 .....                        | 391 |
| 13.6 #error 和#pragma 预处理命令 .....       | 392 |
| 13.7 #和##运算符 .....                     | 392 |
| 13.8 行号 .....                          | 392 |
| 13.9 预定义的符号常量 .....                    | 393 |
| 13.10 断言 .....                         | 393 |
| <b>第 14 章 C 语言的其他专题</b> .....          | 397 |
| 14.1 引言 .....                          | 398 |
| 14.2 I/O 的重定向 .....                    | 398 |
| 14.3 可变长的实参列表 .....                    | 398 |
| 14.4 使用命令行实参 .....                     | 400 |
| 14.5 由多个源文件组成的程序的编译问题 .....            | 401 |
| 14.6 使用 exit 和 atexit 终止程序 .....       | 402 |
| 14.7 volatile 类型限定符 .....              | 403 |
| 14.8 整型和浮点型常量的后缀 .....                 | 403 |
| 14.9 关于文件的深入讨论 .....                   | 404 |
| 14.10 信号处理 .....                       | 405 |
| 14.11 动态内存分配:函数 calloc 和 realloc ..... | 407 |
| 14.12 用 goto 实现无条件转移 .....             | 407 |

|   |     |
|---|-----|
| <b>第 15 章 C++ :介绍对象技术</b> .....                 | 413 |
| 15.1 简介 .....                                   | 414 |
| 15.2 C++ .....                                  | 414 |
| 15.3 简单程序:两个整数相加 .....                          | 414 |
| 15.4 C++ 标准库 .....                              | 416 |
| 15.5 头文件 .....                                  | 416 |
| 15.6 内联函数 .....                                 | 417 |
| 15.7 引用和引用形参 .....                              | 419 |
| 15.8 空形参列表 .....                                | 423 |
| 15.9 默认实参 .....                                 | 423 |
| 15.10 一元作用域运算符 .....                            | 424 |
| 15.11 函数重载 .....                                | 425 |
| 15.12 函数模板 .....                                | 427 |
| 15.13 对象技术与 UML 简介 .....                        | 429 |
| 15.14 本章小结 .....                                | 431 |
| <b>第 16 章 类与对象简介</b> .....                      | 437 |
| 16.1 简介 .....                                   | 438 |
| 16.2 类、对象、成员函数以及数据成员 .....                      | 438 |
| 16.3 定义一个具有成员函数的类 .....                         | 439 |
| 16.4 定义一个有参成员函数 .....                           | 441 |
| 16.5 数据成员、set 函数与 get 函数 .....                  | 443 |
| 16.6 用构造函数初始化对象 .....                           | 447 |
| 16.7 将类放在单独的文件中以增强重用性 .....                     | 450 |
| 16.8 将接口与实现分离 .....                             | 452 |
| 16.9 用 set 函数验证数据 .....                         | 456 |
| 16.10 本章小结 .....                                | 460 |
| <b>第 17 章 类:深入剖析(1)</b> .....                   | 466 |
| 17.1 简介 .....                                   | 467 |
| 17.2 Time 类的案例学习 .....                          | 467 |
| 17.3 类域和访问类的成员 .....                            | 472 |
| 17.4 将接口与实现分离 .....                             | 473 |
| 17.5 访问函数和工具函数 .....                            | 473 |
| 17.6 Time 类的案例学习:具有默认实参的构造函数 .....              | 475 |
| 17.7 析构函数 .....                                 | 479 |
| 17.8 构造函数与析构函数的调用时间 .....                       | 479 |
| 17.9 Time 类的案例学习:隐蔽陷阱——返回 private 数据成员的引用 ..... | 482 |
| 17.10 默认按成员赋值 .....                             | 483 |
| 17.11 本章小结 .....                                | 485 |
| <b>第 18 章 类:深入剖析(2)</b> .....                   | 490 |
| 18.1 简介 .....                                   | 491 |
| 18.2 const 对象和 const 成员函数 .....                 | 491 |
| 18.3 组合:对象作为类的成员 .....                          | 497 |
| 18.4 友元函数和友元类 .....                             | 501 |
| 18.5 使用 this 指针 .....                           | 503 |



|               |   |            |
|---------------|---|------------|
| 18.6          | static 类成员 .....  | 507        |
| 18.7          | 数据抽象和信息隐藏 .....   | 510        |
| 18.8          | 本章小结 .....  | 511        |
| <b>第 19 章</b> | <b>运算符重载 .....</b>  | <b>517</b> |
| 19.1          | 简介 .....  | 518        |
| 19.2          | 运算符重载基础 .....   | 518        |
| 19.3          | 运算符重载的限制 .....  | 519        |
| 19.4          | 作为类的成员和作为全局函数的运算符函数的比较 .....  | 520        |
| 19.5          | 重载流插入和流读取运算符 .....  | 521        |
| 19.6          | 重载一元运算符 .....   | 523        |
| 19.7          | 重载二元运算符 .....   | 524        |
| 19.8          | 动态内存管理 .....  | 524        |
| 19.9          | 案例学习: Array 类 .....   | 525        |
| 19.10         | 类型转换 .....  | 534        |
| 19.11         | 创建 String 类 .....   | 534        |
| 19.12         | 重载 ++ 和 -- .....  | 535        |
| 19.13         | 案例学习: Date 类 .....  | 536        |
| 19.14         | 标准库类 string .....   | 540        |
| 19.15         | explicit 构造函数 .....   | 542        |
| 19.16         | 代理类 .....   | 544        |
| 19.17         | 本章小结 .....  | 546        |
| <b>第 20 章</b> | <b>面向对象编程: 继承 .....</b>   | <b>556</b> |
| 20.1          | 简介 .....  | 557        |
| 20.2          | 基类和派生类 .....  | 557        |
| 20.3          | protected 成员 .....  | 559        |
| 20.4          | 基类和派生类的关系 .....   | 559        |
| 20.5          | 派生类的构造函数与析构函数 .....   | 580        |
| 20.6          | public、protected 和 private 继承 .....   | 586        |
| 20.7          | 关于继承的软件工程 .....   | 586        |
| 20.8          | 本章小结 .....  | 587        |
| <b>第 21 章</b> | <b>面向对象编程: 多态 .....</b>   | <b>591</b> |
| 21.1          | 简介 .....  | 592        |
| 21.2          | 多态的实例 .....   | 592        |
| 21.3          | 继承层次结构中对象间的关系 .....   | 593        |
| 21.4          | 类型域和 switch 语句 .....  | 605        |
| 21.5          | 抽象类和纯虚函数 .....  | 605        |
| 21.6          | 案例学习: 利用多态的工资系统 .....   | 607        |
| 21.7          | (选讲) 多态、虚函数以及动态联编 .....   | 618        |
| 21.8          | 案例学习: 利用多态的和运行时类型信息的工资系统, 通过向下类型转换、dynamic_cast、typeid 以及 type_info 使用运行时类型信息 ..... | 621        |
| 21.9          | 虚析构函数 .....   | 624        |
| 21.10         | 本章小结 .....  | 624        |
| <b>第 22 章</b> | <b>模板 .....</b>   | <b>629</b> |
| 22.1          | 简介 .....  | 630        |