



语言详解 (第6版)

Problem Solving and Program Design in C, Sixth Edition

[美] Jeri R. Hanly Elliot B. Koffman 著

潘蓉 郑海红 孟广兰 万波 译

- ◆ 提高解题能力，掌握编程艺术
- ◆ 应用软件工程思想和方法
- ◆ 养成良好的编程风格



人民邮电出版社
POSTS & TELECOM PRESS

TURING

图灵程序设计丛书

TP312C
H381-2.02

-41



语言详解 (第6版)

Problem Solving and Program Design in C, Sixth Edition

[美] Jeri R. Hanly Elliot B. Koffman 著
潘蓉 郑海红 孟广兰 万波 译

TP312C

H381-2.02

人民邮电出版社
北 京

图书在版编目 (CIP) 数据

C语言详解: 第6版 / (美) 汉利 (Hanly, J. R.),
(美) 科夫曼 (Koffman, E. B.) 著; 潘蓉等译. —北京:
人民邮电出版社, 2010.4

(图灵程序设计丛书)

书名原文: Problem Solving and Program Design
in C, Sixth Edition

ISBN 978-7-115-22016-5

I. ① C… II. ① 汉… ② 科… ③ 潘… III. ①
C语言—程序设计—教材 IV. ① TP312

中国版本图书馆 CIP 数据核字 (2009) 第 233531 号

内 容 提 要

本书是C语言的经典教材。它不仅深入浅出地讲述了实际开发中最需要的C语言基础知识, 还充分介绍了动态数据结构和多进程等高级内容, 而且同时阐述了程序设计思想。本书着眼于提高读者的问题解决能力和实际编程能力, 使其初步掌握当前软件行业公认的程序设计风格和编程实践。书中对指针的讲解通俗易懂, 降低了这一难点的学习门槛。此外, 本书配有各种层次的习题和示例, 而且每一章都有一个或多个来自实际生活的实例研究。

本书适合作为计算机科学等专业本科生C语言程序设计课程的教材, 对已有一定经验的技术人员也是不可多得的参考书。

图灵程序设计丛书

C语言详解 (第6版)

-
- ◆ 著 [美] Jeri R. Hanly Elliot B. Koffman
译 潘 蓉 郑海红 孟广兰 万 波
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 电子函件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
北京艺辉印刷有限公司印刷
 - ◆ 开本: 787×1092 1/16
印张: 36.75
字数: 1087千字 2010年4月第1版
印数: 1—3 000册 2010年4月北京第1次印刷

著作权合同登记号 图字: 01-2009-5719号

ISBN 978-7-115-22016-5

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

版 权 声 明

Authorized translation from the English language edition, entitled *Problem Solving and Program Design in C, Sixth Edition*, 9780321535429 by Jeri R. Hanly and Elliot B. Koffman, published by Pearson Education, Inc., publishing as Addison Wesley, Copyright © 2010 by 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 PEARSON EDUCATION ASIA LTD. and POSTS & TELECOM PRESS Copyright © 2010.

本书中文简体字版由Pearson Education Asia Ltd.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。

本书封面贴有Pearson Education（培生教育出版集团）激光防伪标签，无标签者不得销售。
版权所有，侵权必究。

译 者 序

随着面向对象编程和可视化集成开发工具的发展，很多曾经非常流行的编程语言的影响力正在下降，甚至逐渐消失，但有一种语言例外，它就是C语言！时光的流逝丝毫没有减少C的魅力，它的风采依然。

作者高屋建瓴，向初学者分析了选择计算机科学作为职业的理由，并对计算机科学中的主流方向进行了分类，阐述了各个方向的相关课程、需要具备的能力，以及未来就业的方向。这些内容对所有学习C语言的人都是必须了解的。

因为C语言所涉及的概念比较复杂，规则繁多，用法灵活，容易出错，所以很多初学者感到无从下手。初学者通常很难理解C语言的某些概念，虽然非常渴望学好C语言，但往往因为难学而望而却步。更为突出的是，很多学生虽然掌握了C语言的各个知识点，但在解决实际问题时，却仍然不知所措，而本书的宗旨正是强调如何利用C语言解决问题，相信读完本书之后，读者在解决问题方面一定会有豁然开朗的感觉。

本书还具备许多其他书籍没有的特色。

- 在介绍新的概念之前，通常先引入一个贴切有趣的应用示例，这样做有助于调动初学者的兴趣，使初学者对概念能够顺利理解。
- 程序设计思想的介绍和C语言语法的讲述双管齐下，使学生先理解程序设计的宏观概念，然后在此基础上学习C语言的具体语法，有助于加深其对C语言的理解，增强其驾驭C语言的能力，在实际应用中做到举一反三，灵活应变。
- 结构图的采用使得本书图文并茂，例程代码清晰完整，为学生仔细研读程序提供了可能。
- 每章都有一个来自于具体应用的完整的实例研究，虽然由于国情不同，对某些应用的细节国内读者可能并不熟悉，但这些实例却能够使读者在贴近现实的环境中应用所学到的C语言知识，这也是本书的亮点之一。
- 各章均指出了常出现的错误，为初学者提供了宝贵的实践经验。
- 书中很多地方涉及程序风格，这一点是国内书籍涉及甚少的一方面，根据译者的教学经验，这一点相当重要。国内学生在这方面欠缺甚多，通常都认为程序风格不如编写程序重要，然而良好的程序风格将会使每一个从事程序设计的人受益无穷，对于学习程序设计的学生更显得尤为重要。

本书是为计算机专业低年级本科生以及众多其他学科的初学者或自学者编写的C语言课程教材，也可供那些已有一些C语言基础但还想进一步提高的读者参考。

参加本书翻译工作的人员有万波、潘蓉、郑海红等。此外特别感谢我的妻子孟广兰在英语语言方面提供的帮助。衷心感谢人民邮电出版社图灵公司的编辑所做的大量细致的工作。

由于时间仓促，译者水平有限，译文中难免还存在不少纰漏和欠妥之处，恳请广大读者不吝赐教和指正。

万 波

2009年9月 于西安电子科技大学

译者简介

- 潘 蓉** 西安电子科技大学计算机学院讲师、博士。研究方向为图形图像处理、数字水印技术等，主要教授面向对象程序设计和网络应用程序设计等课程，在核心期刊及会议上已发表10余篇论文。
- 郑海红** 西安电子科技大学计算机学院副教授、博士。研究方向为计算机图形图像处理，曾参与武器装备预研基金、陕西省自然科学基金等项目，并在核心期刊、国际会议上发表论文10余篇。承担的教学课程有离散数学、人工智能等。
- 孟广兰** 西安电子科技大学人文学院讲师。研究方向为外国语言学与应用语言学，对科技英语写作和翻译有较深入的研究。
- 万 波** 西安电子科技大学计算机学院副教授、博士。研究方向为计算机系统结构、图形图像处理，曾参与及主持多项武器装备预研基金、陕西省自然科学基金等项目。目前为计算机学院程序设计课程组组长，承担的教学课程有图形学、Java程序设计、C++程序设计、网络应用程序设计等，其中C++程序设计是陕西省精品课程。

前言

本书教给读者怎样用训练有素的方法来解决问题，并应用广为接受的软件工程方法将程序解决方案设计为内聚的、可读的、可复用的模块。本书将ANSI C的一个子集作为这些模块的实现工具进行介绍，ANSI C是一种标准化的、在行业中广泛使用的编程语言，功能强大，可移植性好。本书可以用于程序设计方法的第一门课程，因为它不要求计算机和程序设计的预备知识；此外，书中选择了大量实例研究和练习，教师完全可以用本书为计算机专业或其他专业的学生开设一门导论性的C语言程序设计课程。

本版的更新

本版中增加了新的第0章，这一章向刚入门的学生介绍了计算机领域的构成和各方面的发展机会。我们希望这一章引导学生展望未来世界，激励他们认真考虑选修计算机方向。另外，第1章中硬件方面的内容已经有了重大的更新，涵盖了自上一版以来的技术最新发展。最后，我还修订和增强了实例研究和章末项目中的许多地方，以便给读者一个全新的感受。

用C来教授程序开发

本书有两个目标，讲授程序设计以及讲授C，这在某些人看来可能是矛盾的。很多人认为应该先用其他更加友好的语言学习编程基础之后再学习C语言。C语言之所以被认为非常难，可以追溯到它的历史。C语言是作为编写UNIX操作系统的工具而设计的，它最初的使用者就是理解操作系统和底层机器复杂性的程序员，这些使用者认为在其程序中利用这些底层知识很自然。所以，许多讲授C语言的教材展示给学生的程序示例都需要理解机器概念就不足为奇了，尽管这些概念并没有出现在标准的导论性程序设计课程的大纲中。

本书之所以能既讲授程序开发的正确方法又讲授ANSI C语言，是因为我们选择前者作为最主要的目标。有人可能会担心这种选择会弱化对ANSI C的阐述。事实正相反，我们发现将程序设计的概念和这些概念在C中的实现相结合，能够清晰地展示C作为一种高级编程语言所具备的威力，而这一点在那些以介绍ANSI C所有内容为主的教科书中往往是很模糊的。虽然与讨论C语言特征相比我们更强调程序设计的方法，但本书仍然全面讲述了C的基本结构。

指针与本书的组织

书中C语言各主题出现的顺序是按照我们理解的初级编程者的需要确定的，而不是由C语言的结构决定的。读者会惊奇地发现没有标题为“指针”的章节。这是因为我们将C当作一种高级语言来讨论，而并不是忽视了指针在C语言中的重要作用。

其他高级语言在输出参数和数组方面具有独立的语言结构，但C语言将这些概念包装在它的指针概念中，这明显增加了学习的难度。本书在涉及的地方分散地讨论指针，从而简化了学习过程，使得学生能够每次吸收一点指针用法的精髓。这种方法便于用传统的高级语言术语（输出参数、数组、数组下标和字符串）来表示基本概念，方便了没有汇编语言背景的学生掌握指针的多种用法。

因此，本书不是用1章而是用4章来讲解指针。第6章讨论了指针作为输出和输入/输出参数的用法，第8章和数组一起讨论，第9章讲述了字符串和指针数组，第14章在回顾了以前讨论的指针用法后描述了动态内存分配。此外，第2章和第12章讨论了文件指针。

软件工程概念

本书涉及了软件工程的许多方面，其中有一些是明确讨论的，而另外一些则是通过例子讲解的。良好的问题解决技巧和有效的软件开发之间的联系在第1章讨论问题解决的科学和艺术时就已经讲到了。第1章给出的5步软件开发方法适用于第一个实例研究，并贯彻于本书的其他实例研究中。重要的程序风格用特定方式突出显示，而且示例中使用的编码风格符合软件行业的规范要求。有几章还讨论了算法跟踪、程序调试和测试。

第3章通过介绍部分C库函数、无参数void函数以及接受输入参数并返回值的函数讲解了过程抽象。第4章和第5章包括更多的函数示例，第6章研究具有简单参数的函数，还讨论了使用指针表示输出和输入/输出参数，第7章介绍怎样把函数作为参数使用。

第6章、第8章和第11章的实例研究和示例程序介绍了数据抽象的概念以及数据类型和运算符封装的概念。第13章给出在个人库中规范过程抽象和数据抽象的C工具，这些个人库由单独的头文件和实现文件定义。第16章介绍C++实现的面向对象设计概念。

整本书中强调了可视函数接口的使用。直到第13章才提到使用全局变量的可能性，然后仔细地描述了使用全局变量的优点和危害。

教学辅助内容

我们使用以下教学辅助内容来增强本书作为学习工具的实用性。

- **小节末尾的练习。**大部分小节结尾都有自测练习，分为需要分析程序段的练习和简短的编程练习。部分自测练习的答案在<http://cssupport.pearsoncmg.com>上的“Hanly/Koffman”目录中可以找到。
- **示例和实例研究。**本书包含各种各样的编程示例。只要可能，示例就给出完整的程序或函数而不是未完成的程序片段。每一章都包含一个或多个具有实际意义的实例研究，而且都遵循软件开发方法来解决。大量实例研究让学生看到了计算的重要应用，包括数据库查询、用于记账和销售分析的商业应用、字处理以及诸如辐射等级监测和节约用水等有关环境的应用。
- **语法展示框。**语法展示框描述了语法和新的C特性的语义并给出了例子。
- **程序风格展示。**程序风格展示并讨论了良好的编程风格。
- **错误讨论和每章复习。**每章都有一小节讨论常见编程错误。每章后面的复习经常以表格形式给出这一章介绍的新的C语言结构。
- **章末练习。**每章的复习之后是附有答案的快速测试练习，同时还有复习题。
- **章末项目。**每章末尾是程序设计项目，这些项目给学生一个实践其在这一章中学到的知识的机会。

附录

由于本书只讨论了ANSI C的一个子集，附录对提升本书作为参考书的价值非常重要。整本书中数组引用都通过下标进行。附录A讨论了指针运算。附录B是ANSI C的标准库函数的表格，按字母排序。附录C中给出了所有C运算符的优先级和结合性的表格，而且给出了前面没有讨论过的运算符的解释。附录D给出了字符集表。附录E列出了所有的C保留字。ANSI C结构的参照表在附录F中给出。

补充

本书的所有读者都可以从http://cssupport.pearsoncmg.com/book.php?author_id=68得到以下补充材料^①。

- 源代码。
- 已知的勘误表。
- 奇数页自测练习的答案。

以下指导教师资源仅提供给采用本书授课的教师。访问www.pearsonhighered.com/irc或发邮件至computing@aw.com即可了解如何获得以下内容^②。

- Solutions Manual。
- Test Bank。
- PowerPoint®幻灯片。

致谢

许多人参与了本书的出版过程。我们感谢南平原学院的Charlotte Young在编写第0章时提供的帮助，WaveRules公司的Jeff Warsaw对本书的第15章作出了重要贡献。加利福尼亚理工大学喷气推动实验室的Joan C. Horvath提供了几个程序设计练习。加利福尼亚大学戴维斯分校的Nelson Max对本书提出了大量的建议。Jeri要感谢马里兰Loyola学院的同事James R. Glenn、Dawn J. Lawrie和Roberta E. Sabin的帮助，他们提供了几个程序设计项目。我们还要感谢那些在过去几年中帮助验证编程示例，并提供练习答案的天普大学、怀俄明大学和霍华德大学的几位学生，他们是Mark Thoney、Lynne Doherty、Andrew Wrobel、Steve Babiak、Donna Chrupcala、Masoud Kermani、Thayne Routh和Paul Onakoya。

在此过程中很高兴和Addison-Wesley团队一起合作。主编Michael Hirsch和Stephanie Sellinger一起在原稿修订的各个阶段提供了指导和鼓励，Heather McNally负责了本书的生产，而Erin Davis负责了市场运作。

Jeri R. Hanly
Elliot B. Koffman

① 源代码和奇数页自测练习的答案亦可从图灵网站（www.turingbook.com）本书页面免费注册下载。已知勘误已在本书中更正。——编者注

② 在图灵网站注册并通过教师验证的教师也可以从图灵网站获得这些教辅资料。

目 录

第0章 计算机科学的职业之路..... 1	快速测试..... 25
引言..... 1	快速测试答案..... 26
0.1 为什么选择计算机科学领域..... 1	复习题..... 26
0.1.1 将计算机科学作为专业的理由..... 1	第2章 C概述..... 27
0.1.2 计算机科学家的特质..... 2	2.1 C语言要素..... 27
0.2 大学经历：可选的计算机学科和专业..... 3	2.1.1 预处理指令..... 27
0.2.1 计算机科学..... 3	2.1.2 预处理指令的语法展示..... 28
0.2.2 计算机工程..... 4	2.1.3 main函数..... 29
0.2.3 信息系统..... 4	2.1.4 保留字..... 29
0.2.4 信息技术..... 4	2.1.5 标准标识符..... 30
0.2.5 软件工程..... 5	2.1.6 用户定义的标识符..... 30
0.2.6 交叉学科..... 5	2.1.7 大写字母和小写字母..... 31
0.3 就业机会..... 6	2.1.8 程序风格之选择标识符名称..... 31
0.3.1 美国和全世界的需求..... 6	2.2 变量声明和数据类型..... 31
0.3.2 非代表性群体的需求..... 6	2.2.1 变量声明..... 31
0.3.3 新职业持续出现..... 7	2.2.2 数据类型..... 32
第1章 计算机与程序设计概述..... 8	2.3 可执行语句..... 33
1.1 电子计算机的过去和现在..... 8	2.3.1 内存中的程序..... 34
1.2 计算机硬件..... 10	2.3.2 赋值语句..... 34
1.2.1 存储器..... 10	2.3.3 输入/输出操作和函数..... 35
1.2.2 中央处理器..... 13	2.3.4 printf函数..... 36
1.2.3 输入/输出设备..... 13	2.3.5 scanf函数..... 38
1.2.4 计算机网络..... 14	2.3.6 return语句..... 39
1.3 计算机软件..... 16	2.4 C程序的一般形式..... 40
1.3.1 操作系统..... 16	2.4.1 程序风格之程序中的空格..... 40
1.3.2 应用软件..... 17	2.4.2 程序的注释..... 40
1.3.3 计算机语言..... 17	2.4.3 程序风格之使用注释..... 41
1.3.4 执行程序..... 20	2.5 算术表达式..... 42
1.4 软件开发方法..... 21	2.5.1 运算符/和%..... 42
1.4.1 软件开发方法..... 21	2.5.2 表达式的数据类型..... 43
1.4.2 注意：失败是编程过程的一部分..... 23	2.5.3 混合类型赋值语句..... 44
1.5 应用软件开发方法..... 23	2.5.4 强制类型转换..... 44
实例研究 英里到公里的转换..... 23	2.5.5 具有多个运算符的表达式..... 45
复习..... 25	2.5.6 用C语言编写数学公式..... 47
	实例研究 超市硬币处理机..... 48

2.6 在程序输出中格式化数值	51	3.5.1 带输入参数的void函数	84
2.6.1 格式化int型值	51	3.5.2 带输入参数和单一结果的函数	85
2.6.2 格式化double型值	52	3.5.3 程序风格之函数接口注释	87
2.6.3 程序风格之消除前导空白	52	3.5.4 带多个参数的函数	87
2.7 交互模式、批处理模式和数据文件	53	3.5.5 参数表一致性	88
2.7.1 输入重定向	53	3.5.6 函数数据区	89
2.7.2 程序风格之回显与提示符	54	3.5.7 使用驱动测试函数	89
2.7.3 输出重定向	54	3.6 常见编程错误	90
2.7.4 程序控制的输入和输出文件	54	复习	90
2.8 常见编程错误	56	快速测试	91
2.8.1 语法错误	56	快速测试答案	92
2.8.2 运行时错误	58	复习题	92
2.8.3 未检测到的错误	58	程序设计项目	92
2.8.4 逻辑错误	60		
复习	60	第4章 选择结构: if 语句和 switch	
快速测试	61	语句	95
快速测试答案	62	4.1 控制结构	95
复习题	62	4.2 条件	95
程序设计项目	63	4.2.1 关系运算符和判等运算符	96
第3章 函数的自顶向下设计	66	4.2.2 逻辑运算符	96
3.1 利用已有信息编程	66	4.2.3 运算符优先级	97
实例研究 求圆的面积和周长	67	4.2.4 短路求值	99
实例研究 计算一批垫圈的重量	69	4.2.5 用C语言表示条件	99
3.2 库函数	72	4.2.6 比较字符	100
3.2.1 预定义函数和代码复用	72	4.2.7 逻辑赋值	100
3.2.2 使用灰色区分新结构	73	4.2.8 条件取反	101
3.2.3 C库函数	73	4.3 if语句	103
3.2.4 本节目标	75	4.3.1 有两个选项的if语句	103
3.3 自顶向下设计和结构图	76	4.3.2 只有一个选项的if语句	103
实例研究 画简单的图形	76	4.3.3 比较一个选项的if语句和两个选项的if语句	104
3.4 无参函数	77	4.3.4 程序风格之if语句的格式	105
3.4.1 函数原型	78	4.4 具有复合语句的if语句	105
3.4.2 函数定义	78	4.4.1 程序风格之编写具有复合语句的if语句	106
3.4.3 程序中函数的位置	80	4.4.2 跟踪if语句	106
3.4.4 程序风格之在含有函数的程序中使用注释	81	4.5 算法中的决策步骤	108
3.4.5 函数子程序和主函数的执行顺序	81	实例研究 水费问题	108
3.4.6 使用函数子程序的优点	82	4.5.1 程序风格之在函数中使用一致的名称	113
3.4.7 显示用户指令	82	4.5.2 程序风格之内聚函数	113
3.5 带输入参数的函数	83	4.5.3 程序风格之使用常量宏来增强	

可读性和易维护性	114	5.10.1 使用调试器程序	165
4.6 解决更多问题	114	5.10.2 不使用调试器进行调试	165
4.6.1 结构图中的数据流信息	114	5.10.3 循环次数差1的错误	165
4.6.2 使用函数子程序修改程序	115	5.10.4 测试	166
实例研究 具有节约需求的水费问题	115	5.11 常见编程错误	166
4.7 嵌套if语句和多选项决策	116	复习	168
4.7.1 比较嵌套if和if序列	117	快速测试	170
4.7.2 嵌套if的多选项决策形式	117	快速测试答案	170
4.7.3 多选项决策中条件的顺序	118	复习题	171
4.7.4 程序风格之确认变量的值	120	程序设计项目	172
4.7.5 具有多个变量的嵌套if语句	120		
4.8 switch语句	123	第6章 模块化编程	176
4.9 常见编程错误	126	6.1 带简单输出参数的函数	176
复习	127	6.2 带输入输出参数函数的多次调用	181
快速测试	128	6.3 名字的作用域	185
快速测试答案	130	6.4 输出形参作为实参变量	186
复习题	130	6.5 由多个函数构成的程序	189
程序设计项目	131	实例研究 简分数的算术运算	189
第5章 重复和循环语句	134	6.6 程序系统的调试与测试	195
5.1 程序中的重复	134	6.7 常见编程错误	197
5.2 计数循环和while语句	135	复习	197
5.3 在循环中计算和或者乘积	137	快速测试	198
5.3.1 程序风格之编写通用循环	139	快速测试答案	198
5.3.2 计算一列数的乘积	139	复习题	199
5.3.3 复合赋值运算符	140	程序设计项目	200
5.4 for语句	141	第7章 简单数据类型	205
5.4.1 程序风格之格式化for语句	141	7.1 数值类型的表示与转换	205
5.4.2 自增和自减运算符	142	7.1.1 数值类型的差异	205
5.4.3 步长非1的自增和自减	143	7.1.2 数值的不精确性	207
5.4.4 显示值构成的表格	145	7.1.3 数据类型的自动转换	207
5.5 条件循环	146	7.1.4 数据类型的显式转换	208
5.6 循环设计	150	7.2 字符类型的表示与转换	209
5.6.1 标记控制循环	150	7.3 枚举类型	210
5.6.2 使用for语句实现标记循环	152	7.4 迭代近似	215
5.6.3 文件结束控制循环	152	实例研究 求根的二分方法	216
5.6.4 错误数据上的无限循环	154	7.5 常见编程错误	221
5.7 嵌套循环	154	复习	221
5.8 do-while语句和标志控制循环	157	快速测试	222
5.9 问题求解示例	160	快速测试答案	223
实例研究 太阳能加热房的采暖面积	160	复习题	223
5.10 如何调试和测试程序	164	程序设计项目	224

第8章 数组	229
8.1 声明和引用数组	229
8.2 数组下标	232
8.3 使用循环顺序存取	233
8.3.1 使用数组进行统计计算	234
8.3.2 程序风格之使用循环控制变量 作为数组下标	236
8.4 将数组元素作为函数参数使用	236
8.5 数组参数	238
8.5.1 形参数组	238
8.5.2 和形参数组对应的实参	239
8.5.3 数组作为输入参数	240
8.5.4 返回数组结果	241
8.5.5 部分填充数组	243
8.5.6 栈	244
8.6 数组搜索和排序	247
8.6.1 数组搜索	247
8.6.2 数组排序	248
8.7 多维数组	250
8.7.1 多维数组的初始化	252
8.7.2 多维数组	252
8.8 数组处理示例	253
实例研究 医院收入汇总	253
8.9 常见编程错误	259
复习	260
快速测试	261
快速测试答案	262
复习题	262
程序设计项目	263
第9章 字符串	268
9.1 字符串基础	268
9.1.1 声明并初始化字符串变量	268
9.1.2 字符串数组	269
9.1.3 使用printf和scanf进行 输入/输出	269
9.2 字符串库函数: 赋值和子串	272
9.2.1 字符串赋值	272
9.2.2 子串	274
9.3 较长的字符串: 拼接和整行输入	278
9.3.1 拼接	278
9.3.2 字符和字符串的区别	279
9.3.3 扫描一个完整的行	279
9.4 字符串比较	281
9.5 指针数组	283
9.6 字符操作	287
9.6.1 字符输入/输出	287
9.6.2 字符分析和转换	288
9.7 字符串到数值和数值到字符串的转换	290
9.8 字符串处理示例	294
实例研究 文本编辑器	295
9.9 常见编程错误	301
复习	302
快速测试	304
快速测试答案	304
复习题	305
程序设计项目	306
第10章 递归	309
10.1 递归的本质	309
10.2 跟踪递归函数	312
10.2.1 带返回值的递归函数的 跟踪	312
10.2.2 void类型递归函数的跟踪	313
10.2.3 参数与局部变量栈	315
10.2.4 C中参数栈的实现	316
10.2.5 何时、如何跟踪递归函数	317
10.3 递归数学函数	318
10.4 带数组和字符串参数的递归函数	322
实例研究 找出字符串中的大写字母	322
实例研究 递归选择排序	324
10.5 用递归解决问题	326
实例研究 集合运算	326
10.6 递归的经典实例研究: 汉诺塔	331
10.7 常见编程错误	335
复习	336
快速测试	336
快速测试答案	337
复习题	337
程序设计项目	338
第11章 结构与联合类型	341
11.1 用户自定义结构类型	341
11.1.1 结构类型定义	341
11.1.2 操作结构化数据对象的单个	

成员	343	13.1.2 数据抽象	399
11.1.3 运算符优先级回顾	343	13.1.3 信息隐藏	399
11.1.4 操作结构整体	344	13.1.4 可复用代码	399
11.1.5 程序风格之类型命名约定	344	13.2 个人库: 头文件	400
11.2 结构类型数据作为输入/输出参数	345	13.2.1 头文件	401
11.3 返回值为结构类型的函数	349	13.2.2 头文件设计的忠告	402
11.4 用结构类型解决问题	351	13.3 个人库: 实现文件	403
实例研究 用于复数的用户自定义类型	351	13.4 存储类型	405
11.5 并行数组和结构的数组	356	13.4.1 全局变量	406
11.5.1 并行数组	356	13.4.2 static和register存储 类型	407
11.5.2 结构数组的声明	356	13.5 库中包含函数的修改	408
实例研究 通用计量转换	357	13.6 条件编译	410
11.6 联合类型 (可选)	363	13.7 函数main的参数	413
11.7 常见编程错误	367	13.8 定义带参数的宏	415
复习	368	13.8.1 在宏体中使用括号	416
快速测试	369	13.8.2 在两行以上扩展宏	417
快速测试答案	369	13.9 常见编程错误	418
复习题	370	复习	418
程序设计项目	371	快速测试	420
第 12 章 文本文件和二进制文件的 处理	375	快速测试答案	420
12.1 输入/输出文件: 回顾与进一步研究	375	复习题	421
12.1.1 键盘和屏幕作为文本流	375	程序设计项目	421
12.1.2 换行与EOF	376	第 14 章 动态数据结构	424
12.1.3 转义序列	376	14.1 指针	424
12.1.4 用printf实现格式化输出	376	14.1.1 作为函数参数的指针	425
12.1.5 文件指针变量	377	14.1.2 表示数组和字符串的指针	425
12.1.6 获取文件指针参数的函数	378	14.1.3 指向结构的指针	426
12.1.7 关闭文件	379	14.1.4 指针使用的小结	426
12.2 二进制文件	381	14.2 动态内存分配	427
12.3 数据库查询	385	14.2.1 访问动态分配的结构成员	428
实例研究 数据库查询	386	14.2.2 使用calloc动态分配数组	428
12.4 常见编程错误	393	14.2.3 返回堆单元	429
复习	393	14.3 链表	430
快速测试	394	14.3.1 带指针成员的结构	431
快速测试答案	395	14.3.2 连接结点	431
复习题	395	14.3.3 链表的优点	433
程序设计项目	395	14.4 链表运算符	434
第 13 章 大型程序设计	398	14.4.1 遍历链表	434
13.1 使用抽象处理复杂问题	398	14.4.2 获得输入表	436
13.1.1 过程抽象	398	14.4.3 在表中查询目标	437
		14.4.4 避免跟踪NULL指针	438

14.5 用链表表示栈	438
14.6 用链表表示队列	441
14.7 有序表	445
实例研究 维护一个整数有序表	445
14.8 二叉树	453
14.8.1 二叉查找树	454
14.8.2 搜索二叉查找树	454
14.8.3 建立二叉查找树	455
14.8.4 显示二叉查找树	458
14.9 常见编程错误	460
复习	461
快速测试	462
快速测试答案	462
复习题	463
程序设计项目	464
第 15 章 使用进程和线程的多进程	466
15.1 多任务	466
15.1.1 线性程序设计和并行程序设计	466
15.1.2 分时多任务	467
15.1.3 抢占式多任务	467
15.1.4 时间片和并行	468
15.1.5 并发程序设计	469
15.2 进程	469
15.2.1 进程创建	469
15.2.2 进程等待	471
15.2.3 从进程中执行另一个程序	472
15.3 进程间通信和管道	473
15.3.1 管道	473
15.3.2 管道的用法	474
15.3.3 使用标准输入的进程间通信	475
15.3.4 父进程和子进程间通信举例	476
15.4 线程	478
15.4.1 线程的创建	479
15.4.2 线程的同步	480
15.4.3 互斥锁	482
15.4.4 死锁	483
15.5 线程举例	486
实例研究 生产者/消费者模型	486
15.6 常见编程错误	495
复习	495
C语言结构的复习	496
快速测试	497
快速测试答案	497
复习题	497
程序设计项目	498
第 16 章 关于 C++	499
16.1 C++控制结构、输入/输出以及函数	499
16.1.1 使用名字空间std	499
16.1.2 C++标准输入/输出	502
16.1.3 引用参数	502
16.1.4 输出格式化	504
16.2 C++对面向对象编程的支持	505
16.2.1 头文件complex.h	506
16.2.2 类名和构造函数	507
16.2.3 成员函数和运算符	507
16.2.4 实现文件complex.cpp	508
16.2.5 数据成员	511
16.2.6 输入/输出运算符重载	511
复习	513
快速测试	515
快速测试答案	515
复习题	515
程序设计项目	516
附录 A 关于指针的更多信息	517
附录 B ANSI C 标准库	521
附录 C C 运算符	533
附录 D 字符集	537
附录 E ANSI C 保留字	539
附录 F ANSI C 结构参照表	540
词汇表	542
索引	548

第 0 章

计算机科学的职业之路

引言

为了选择一门课程并最终选定希望从事的职业，需要思考很多重要的问题。为什么要选择这一领域？我们对此是否擅长？在完成学业时能否就业？是否喜欢这样的工作？本章将就这些问题为那些想在计算机科学或相关领域获得学位的人们给出一些建议。

0.1 为什么选择计算机科学领域

0.1.1 将计算机科学作为专业的理由

我们所做的每一件事情几乎都受计算机的影响，如今被称为“千禧世代”的新一代大学生并不会对此感到惊讶。他们的成长伴随着计算机、互联网、即时通信和电玩，他们推崇新技术，并期望它能做各种奇妙的事情。

千禧世代 1982年以后出生的人，据说很自信，爱好社交并具有团队精神，对成功充满了自豪感，喜欢利用分析技巧作出决策，并坚决追求自我安全、稳定和平衡。

但是，上一代人并不乐意接受技术，他们在解决问题时并不总是首先想到技术。很多在职的人抵制技术带来的改变，他们通常让年轻员工接管技术问题，并依靠他们对导致重要结果的事情作出选择。

不同时代的人之间的这种差异为那些聪明且专注的学生选择计算机科学或相关领域作为专业创造了很好的环境。计算机行业是经济增长最快的行业之一，而且在将来有望继续快速增长。一个公司要想有竞争力，必须不断雇用训练有素的专业人员，不仅能在现阶段生产出高质量的产品，也可作为将来具有创造性的科学和工程进步作好准备。

从事计算机行业的人有广泛的领域选择空间，其中有很多有趣并具挑战性的问题需要解决。获得计算机科学学位的人，除了首选从事商务和通信领域的工作外，还能解决生活中方方面面的问题。快速浏览技术文章会发现这么一些领域：为全国选举或州选举而开发的电子投票系统；利用无线设备信号统计车辆和行人的行进时间，为合理控制红绿灯或管理施工街区作最优决策；通过超级计算机运行的“虚拟地震”，研究如何利用真实地理位置和破坏场景的3D模型的预警系统。

计算机专家在医学领域从事的一些研究包括：通过模拟大脑的刺激模式来了解人脑是如何工作的，这主要是为了帮助那些患有孤独症或类似妄想型精神分裂症的人；为身体受伤的人定制大量的辅助设备，包括可编程的机器人假肢和数字“眼”等；从移植的起搏器上搜集信息，以便在出现危险时及时作出决策；通过实时分析人脸，开发能够识别人类感情状态的计算机系统；开发仅凭人的手势驱动计算机来操纵实际对象的人机界面。

安全和执法领域对计算机专业人员提出了很多挑战：美国政府对网络世界的正常行为进行研究分析，希望开发出一种能够识别恐怖组织从事互联网活动的技术；声音生物测定技术的发展，使我们能够利用计算机软件分析语音，以确定说话人的身份、真实性和感情状态；反恶意软件的电子保

护问题备受国家经济和安全部门的关注。

当今世界最具挑战性的一些问题将由各个领域的专家组成的团队来解决。显然，这些团队中包括具有创新精神并且知道如何最好地使用技术的计算机专家。不久的将来，在下列领域会出现很多创新：人类基因组项目、环境监控、艾滋病疫苗研究、清洁燃料、在有潜在危险的地区利用机器人跟踪天气变化、为预测自然灾害用超级计算机模拟地球的结构和功能。想使世界产生积极变化吗？你可以选择学习计算机。

0.1.2 计算机科学家的特质

人的个性和性格特征通常会影响到其选择研究领域，并最终影响他所从事的工作。不同领域要求人们具备不同的能力。成功的计算机科学专业的学生肯定具有很多共同的特点。阅读下列内容，看看和你是否相像。

首先，喜欢接受解决难题的挑战。计算机科学不仅仅是使用现有的计算机硬件或编程语言，而是要找到问题的解决方案。解决问题需要具有创新精神和非传统思维，还必须愿意尝试“公认”解决方案以外的事物。

喜欢从事技术工作并愿意终身学习。喜欢谜题并坚持不懈地寻找解决方案。当潜心寻求答案时，可能都没注意到时间已飞快流逝。既喜欢在现实世界建造事物，也喜欢在虚拟世界建造事物。明白如何定制一个特别的物体使它适应特定的环境。喜欢做大项目并乐于看到它们完成，喜欢做对人类有用且对人类生活有积极影响的事。

在工作中要想成功，还必须有良好的沟通能力。对技术人员和非技术人员都能很好地解释自己的计划和方案，在技术环境中能简明扼要地进行书面表达。大多数项目都包含很多人，所以与团队很好相处是非常重要的。如果想成为一名经理或经营自己的公司，能与不同性格的人一起工作是非常重要的。

因领导开发IBM System/360操作系统而闻名于世的弗雷德里克·布鲁克斯（Frederick P. Brooks），在1970年写了一本《人月神话》，尽管在他写完这本书以后，计算机世界已经发生了很大的改变，但是这本书在今天仍有很大的参考价值。他描述了“工艺的乐趣”：第一，因为制造自己设计的东西而快乐；第二，因为制造了对他人有用或受他人尊敬的东西而快乐；第三，因为用迷宫一样复杂的对象构建系统并使之正常运行而快乐；第四，因为工作非重复性的本质可以不断学习而充满乐趣；最后，因为使用容易驾驭的工具工作而快乐。程序员可以按照自己的想象进行创造，而且能够生产出可测试、易修改和改写的产品。难道雕刻家或土木工程师不喜欢这样的易操作性吗？

IBM System/360是1964年IBM推出的大型计算机系列，它是在体系结构和实现之间有明显界限的首个计算机系列，IBM因此发布不同价位的一系列兼容设计，这种设计被很多人认为是计算机历史上最成功的设计之一，并影响着未来的计算机设计。



图0-1 IBM在1964年引入的商用大型机系列System/360（IBM公司友情提供，未经授权不得使用）