

TURING

图灵计算机科学丛书

PEARSON

计算机科学概论 第11版

Computer Science: An Overview, Eleventh Edition



[美] J. Glenn Brookshear 著
刘艺 尚威海 马小会 毛倩倩 译



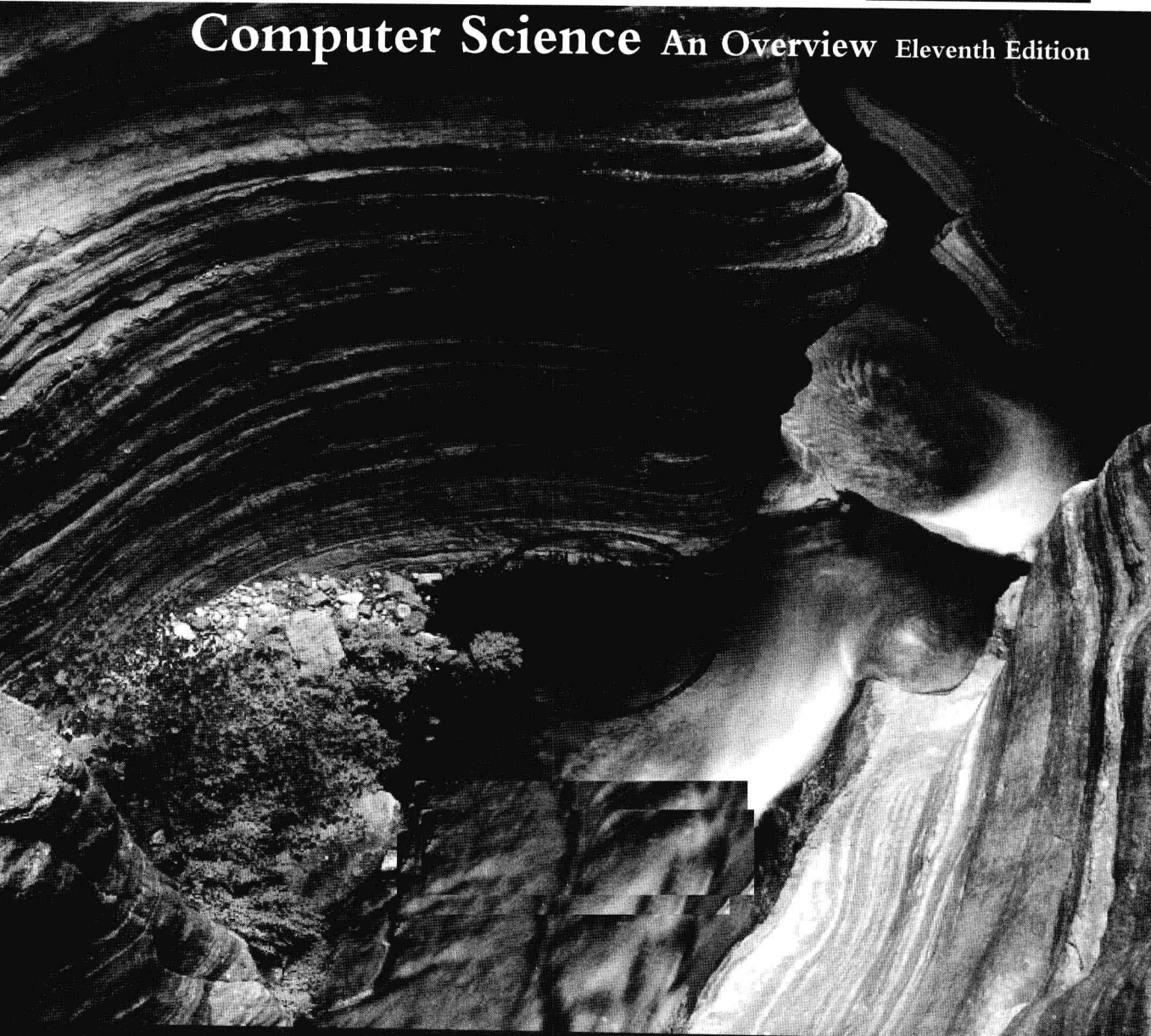
人民邮电出版社
POSTS & TELECOM PRESS

TURING

图灵计算机科学丛书

计算机科学概论 **第11版**

Computer Science An Overview Eleventh Edition



[美] J. Glenn Brookshear 著
刘艺 肖成海 马小会 毛倩倩 译

图书在版编目 (C I P) 数据

计算机科学概论 : 第11版 / (美) 布鲁克希尔
(Brookshear, J. G.) 著 ; 刘艺等译. — 北京 : 人民邮
电出版社, 2011.10

(图灵计算机科学丛书)

书名原文: Computer Science : An Overview ,
Eleventh Edition

ISBN 978-7-115-26196-0

I. ①计… II. ①布… ②刘… III. ①计算机科学
IV. ①TP3

中国版本图书馆CIP数据核字(2011)第174089号

内 容 提 要

本书是计算机科学概论课程的经典教材, 全书对计算机科学做了百科全书式的精彩阐述, 充分展现了计算机科学的历史背景、发展历程和新的技术趋势。本书首先介绍的是信息编码及计算机体系结构的基本原理(第1章和第2章), 进而讲述操作系统(第3章)和组网及因特网(第4章), 接着探讨了算法、程序设计语言及软件工程(第5章至第7章), 然后讨论数据抽象和数据库(第8章和第9章)方面的问题, 第10章通过图形学讲述计算机技术的一些主要应用, 第11章涉及人工智能, 第12章通过对计算理论的介绍来结束全书。本书在内容编排上由具体到抽象逐步推进, 很适合教学安排, 每一个主题自然而然地引导出下一个主题。此外, 书中还包含大量的图、表和示例, 有助于读者对知识的了解与把握。

本书适合用作高等院校计算机以及相关专业的本科生的教材。

图灵计算机科学丛书

计算机科学概论 (第11版)

-
- ◆ 著 [美] J. Glenn Brookshear
 - 译 刘 艺 肖成海 马小会 毛倩倩
 - 责任编辑 卢秀丽
 - 执行编辑 毛倩倩 李 静
 - ◆ 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: 26.5
 - 字数: 786千字 2011年10月第1版
 - 印数: 1-5 000册 2011年10月河北第1次印刷
 - 著作权合同登记号 图字: 01-2011-2960号
 - ISBN 978-7-115-26196-0
-

定价: 69.00元

读者服务热线: (010)51095186转604 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 *Computer Science: An Overview, Eleventh Edition*, 978-0-13-256903-3 by J. Glenn Brookshear, published by Pearson Education, Inc., publishing as Addison Wesley, Copyright © 2012, 2009, 2007, 2005, 2003 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 © 2011.

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。

前言

本书是计算机科学的入门教材。在力求保持学科广度的同时，还兼顾深度，以对所涉及的主题给出中肯的评价。

读者对象

本书面向计算机科学以及其他各个学科的学生。大多数计算机科学专业的学生在最初的学习中都有这样一个误解，认为计算机科学就是程序设计、网页浏览以及因特网文件共享，因为这基本上就是他们所看到的一切。实际上计算机科学远非如此。因此，在入门阶段，学生们需要了解他们主攻的这门学科所涉及内容的广度，这也正是本书的宗旨。本书力图使学生对计算机科学有一个总体的了解，希望在这个基础上，他们可以领会该领域今后其他课程的特点以及相互关系。事实上，本书采用的综述方式也是自然科学入门教程的常用模式。

其他学科的学生如果想融入这个技术化社会，也需要具备这些宽泛的知识背景。适用于他们的计算机科学课程提供的应该是对整个领域很实用的剖析，而不仅仅是培训学生如何上网和使用一些流行的软件。当然这种培训也有其适用的地方，但本书的目的不在于此，而是用作计算机科学的教科书。

因此，在写这本书时，我们力求保持其对非技术类学生的可读性。因此，先前的版本已经被很成功地用作教科书，读者囊括了从高中生到研究生的各个教育层次众多专业的学生。这一版仍将贯彻这一目标。

第 11 版新增的内容

第11版主要增加关于手持移动设备的内容，特别是智能手机。因此，这一版的内容经过了修改、扩充，目的便是呈现所述主题与智能手机技术之间的关系。具体主题包括：

- 智能手机硬件；
- 3G网络与4G网络的区别；
- 智能手机操作系统；
- 智能手机软件开发；
- 智能手机的人机交互界面。

以上新增内容主要出现在第3章（操作系统）和第4章（组网），第6章（编程语言）和第7章（软件工程）中也有提及。

这一版中的其他明显变动包括对以下内容的更新。

- 软件所有权和责任：这一版重写并更新了第7章（软件工程）中与此主题相关的内容。
- 训练人工神经网络：相关内容（见第11章，人工智能）已经实现了现代化。

最后，这一版更新了全书内容来反应当今的技术状况。这主要体现在第0章（绪论）、第1

章（数据存储）和第2章（数据操作）。

章节安排

本书主题由具体到抽象逐步推进——这是一种很利于教学的顺序，每一个主题自然而然地引导出下一个主题。首先介绍的是信息编码、数据存储及计算机体系结构的基本原理（第1章和第2章），进而是操作系统（第3章）和计算机网络（第4章），接着探讨了算法、程序设计语言及软件开发（第5章至第7章），然后探索如何更好地访问信息（第8章和第9章），第10章讲述计算机图形学技术的一些重要应用，第11章涉及人工智能，第12章通过对计算理论的介绍来结束全书。

本书编排顺序自然连贯，但各个章节具有很强的独立性，可以单独查阅，也可以根据不同学习顺序重新排列。事实上，本书通常作为各类课程的教材，内容选择的顺序是多种多样的。其中一种教法是先介绍第5章和第6章（算法和程序设计语言），然后按照需要返回到前面相应章节。我还知道有人是从第12章有关可计算性的内容开始的。这本书还曾作为深入不同领域项目的基础，用于“高级研讨班”的教科书。对于不需要了解太多技术的学生，教学中可以重点讲述第4章（组网及因特网）、第9章（数据库系统）、第10章（计算机图形学）和第11章（人工智能）。

每章开篇都用星号标出了选学章节。选学章节要么是讨论更专业的话题，要么是对传统内容作深入探究。此举仅仅是为那些想采取不同阅读顺序的人提供一点建议。当然，还有其他读法。尤其对于那些寻求快速阅读的读者，我建议采取下面的阅读顺序。

章 节	主 题
1.1~1.4	数据编码和存储基础
2.1~2.3	计算机体系结构和机器语言
3.1~3.3	操作系统
4.1~4.3	组网及因特网
5.1~5.4	算法和算法设计
6.1~6.4	程序设计语言
7.1~7.2	软件工程
8.1~8.3	数据抽象
9.1~9.2	数据库系统
10.1~10.2	计算机图形学
11.1~11.3	人工智能
12.1~12.2	计算理论

在本书中有几条贯穿始终的主线。主线之一是计算机科学是不断发展变化的。本书从历史发展的角度反复呈现各个主题，讨论其当前的状况，并指出研究方向。另一条主线是抽象的作用以及用抽象工具控制复杂性的方式。该主线在第0章引入，然后在操作系统、体系结构、组网、算法、程序设计语言、软件工程、数据组织和计算机图形学等内容中反复体现。

致教师

本教材所包含的内容很难在一个学期内讲授完，因此一定要果断地砍掉不适合自己的教学目标的那些主题，或者根据需要重新调整讲授顺序。你会发现，尽管本书有它固有的结构体系，但各个主题在很大程度上是相对独立的，完全可以根据需要作出选择。我写本书的目的是把它

作为一种课程的参考书，而非圈定课程的内容。我希望你把某些主题留作阅读作业，鼓励学生自己学习，而不在课堂讲授。如果我们认为所有的东西都一定要在课堂上讲，那就低估学生的能力了。我们应该教会他们独立学习。

关于本书从具体到抽象的组织结构，我觉得有必要多言几句。作为学者，我们总以为学生会欣赏我们对于学科的观点，这些观点通常是我们某一领域多年工作中形成的。但作为老师，我认为最好从学生的视角呈现教材。这就是为什么本书首先介绍数据的表示/存储、计算机体系结构、操作系统以及组网，因为这些都是学生们最容易产生共鸣的主题——他们很可能听说过JPEG、MP3这些术语，可能用CD和DVD刻录过资料，买过计算机配件，应用过某一操作系统，或者上过因特网。我发现，从这些主题开始讲授这门课程，学生可以为许多困惑他们多年的问题找到答案，并且把这门课看做是实践课程而不是纯理论的课程。由此出发就会很自然地过渡到较抽象的内容上，例如算法、算法结构、程序设计语言、软件开发方法、可计算性以及复杂性等，而这些内容就是我们本领域的人所认为的计算机科学的主要内容。正如我前面所说的，我并不是强求大家都按此顺序讲课，只是鼓励你们如此尝试一下。

我们都知道，学生能学到的东西要远远多于我们直接传授的内容，而且潜移默化传授的知识更容易被吸收。当要“传授”问题的解决方法时，就更是如此。学生不可能通过学习问题求解的方法变成问题的解决者，他们只有通过解决问题——还不仅仅是那些精心设计过的“教科书式的问题”，才能成为问题的解决者。因此我在本书中加入了大量的问题，并特意让其中一些问题模棱两可——意味着正确方法或正确答案可能不只一个。我建议你们采用并充分拓展这些问题。

另一类适合“潜移默化学习”的主题还有职业精神、伦理和社会责任感。我认为这种内容不应该独立成章，而是应该在有所涉及时讨论，而这正是本书的编排方法。你们会发现，3.5节、4.5节、7.8节、9.7节和11.7节分别在操作系统、组网、软件工程、数据库系统和人工智能的上下文中提及了安全、隐私、责任和社会意识的问题。此外，0.6节就通过总结一些比较著名的理论而引入这一主题——这些理论都企图把伦理上的决断建立在哲学的坚实基础上。你还会发现，每一章都包含了“社会问题”小节，这些问题将鼓励学生思考现实社会与教材内容的关系。

感谢你对本书感兴趣。无论你是否选用本书作为教材，我都希望你认同它是一部好的计算机科学教育文献。

教学特色

本书是多年教学经验的结晶，因此在辅助教学方面考虑较多。最主要的是提供了丰富的问题以加强学生的参与——这一版包含1000多个问题，分为“问题与练习”、“复习题”和“社会问题”。“问题与练习”列在每节末尾（除了第0章外），用于复习刚刚讨论过的内容、扩充以前讨论过的知识，或者提示以后会涉及的有关主题。这些问题的答案可以从图灵社区（www.ituring.com.cn）本书网页免费注册下载。

“复习题”列在每章的末尾（第0章除外）。它们是课后作业，内容覆盖整章，在书中不给出答案。

“社会问题”也列在每章的末尾，供思考讨论。许多问题可以用来开展课外研究，可要求学生提交简短的书面或口头报告。

在每章的末尾还设有“课外阅读”，它列出了与本章主题有关的参考资料。同时，前言以及正文中所列的网址也非常适合查找相关资料。

补充材料

本书的许多补充材料可以从配套网站www.pearsonhighered.com/brookshear上找到。以下内容面向所有读者。

- 每章的实践项目帮助加深理解本教材的主题，并可以帮助了解其他相关主题。
- 每章的“自测题”帮助读者复习本书中的内容。
- 介绍Java和C++基本原理的手册，它在教学顺序上与本书是兼容的。

除此之外，教师还可以登录Pearson Education的教师资源中心（www.pearsonhighered.com）网站申请获得下面的教辅资料。

- 包含“复习题”答案^①的教师指南。
- PowerPoint幻灯片讲稿。
- 测试题库。

你也许还想看一下我的个人网站www.mscs.mu.edu/~glennb，不是很正式（体现了我某一时的灵感和幽默），但你或许能找到些有用的信息。特别值得一提的是，你将在此找到本书的勘误。

致学生

我有一点点偏执（一些朋友说我可远不是一点点），所以写本书时，我很少接受他人的建议，其中许多人认为一些内容对于初学者过于高深。我相信即使学术界把它们归为“高级论题”，但只要与主题相关就是合适的。读者需要的是一本全面介绍计算机科学的教科书，而不是“缩水”的版本——只包括那些简化了的、被认为适合初学者的主题。因此我不回避任何主题，而是力求寻找更好的解释。我力图在一定深度上向读者展示计算机科学最真实的一面。就好比对待菜谱里的那些调味品一样，你可以有选择地略过本书的一些主题，但我全部呈现出来是为了在你想要的时候供你“品尝”，而且我也鼓励你们去尝试。

我还要指出的是，在任何与技术有关的课程中，当前学到的详细信息未必就适合以后的需要。这个领域是发展变化的——这正是使人兴奋的方面。本书将从现实及历史的角度展现本学科的内容。有了这些背景知识，你们就会和技术一起成长。我希望你们现在就开始行动起来，不要局限于课本的内容，而要进行大胆探索。要学会学习。

感谢你们的信任，选择了我的这本书。作为作者，我有责任创作出值得一读的作品。我希望你们觉得我已经尽到了这份责任。

致谢

首先我要感谢那些支持本书（阅读并使用本书前几个版本）的人们，我感到很荣幸。

David T. Smith（宾夕法尼亚州印第安纳大学）和Dennis Brylow（马凯特大学）对这一版的制作给予了很大帮助。David的贡献主要集中在第0章、第1章、第2章、第7章和第11章，Dennis的贡献主要集中在第3章、第4章、第6章和第10章。如果没有他们的工作，这一版将不会出现。我衷心地感谢他们。

^① 也可在图灵社区（www.ituring.com.cn）本书网页免费注册下载。——编者注

我在第10版的前言中已经提到过，要特别感谢Ed Angel、John Carpinelli、Chris Fox、Jim Kurose、Gary Nutt、Greg Riccardi和Patrick Henry Winston在第10版中所做的贡献。他们的努力成果在第11版被保留了下来。

其他对这一版和之前版本作出贡献的人包括J. M. Adams、C. M. Allen、D. C. S. Allison、R. Ashmore、B. Auernheimer、P. Bankston、M. Barnard、P. Bender、K. Bowyer、P. W. Brashear、C. M. Brown、H. M. Brown、B. Calloni、M. Clancy、R. T. Close、D. H. Cooley、L. D. Cornell、M. J. Crowley、F. Deek、M. Dickerson、M. J. Duncan、S. Ezekiel、S. Fox、N. E. Gibbs、J. D. Harris、D. Hascom、L. Heath、P. B. Henderson、L. Hunt、M. Hutchenreuther、L. A. Jehn、K. K. Kolberg、K. Korb、G. Krenz、J. Liu、T. J. Long、C. May、J. J. McConnell、W. McCown、S. J. Merrill、K. Messersmith、J. C. Moyer、M. Murphy、J. P. Myers, Jr., D. S. Noonan、W. W. Oblitey、S. Olariu、G. Rice、N. Rickert、C. Riedesel、J. B. Rogers、G. Saito、W. Savitch、R. Schlafly、J. C. Schlimmer、S. Sells、G. Sheppard、Z. Shen、J. C. Simms、M. C. Slattey、J. Slimick、J. A. Slomka、D. Smith、J. Solderitsch、R. Steigerwald、L. Steinberg、C. A. Struble、C. L. Struble、W. J. Taffe、J. Talburt、P. Tonellato、P. Tromovitch、E. D. Winter、E. Wright、M. Ziegler，还有一位匿名的朋友。我向他们中的每一位致以最真诚的谢意。

如前所述，本书的配套网站上有Java和C++手册来讲述这两种语言的基础知识，与本书的内容相得益彰。它们是由Diane Christie撰写的，在此表示感谢。另外，还要感谢Roger Eastman，他是本书配套网站上每章实践项目的出题人。

我同时要感谢为本项目作出贡献的Addison-Wesley的员工。他们不仅是很好的合作伙伴，而且是很好的朋友。如果你们打算写一本教材，可以考虑交给Addison-Wesley出版。

我还要感谢我的夫人Earlene和我的女儿Cheryl，感谢她们这么多年对我的鼓励。当然Cheryl已经长大，几年以前已经离家开始独自生活。Earlene还陪在我身边。我是一个幸运的人。1998年12月11日的早晨，我突发心脏病，是她及时把我送到了医院，让我逃过了一劫。（对于年轻一代的你们，我有必要解释一下，躲过心脏病的一劫有点像你们又获准延期提交课后作业。）

最后，我要感谢我的父母，本书即是给他们的献礼。我用下面一句赞美的话作为结束，就不说是他们哪一位说的了：“我们儿子的书真的非常好，人人都应该阅读。”

J. G. B.

目 录

第 0 章 绪论	1	1.7.2 截断误差	39
0.1 算法的作用	1	*1.8 数据压缩	41
0.2 计算机器的由来	3	1.8.1 通用的数据压缩技术	41
0.3 算法的科学	7	1.8.2 图像压缩	43
0.4 抽象	8	1.8.3 音频和视频压缩	44
0.5 学习大纲	8	*1.9 通信差错	45
0.6 社会影响	9	1.9.1 奇偶校验位	45
社会问题	11	1.9.2 纠错编码	46
课外阅读	12	复习题	47
第 1 章 数据存储	13	社会问题	50
1.1 位和位存储	13	课外阅读	51
1.1.1 布尔运算	13	第 2 章 数据操控	52
1.1.2 门和触发器	14	2.1 计算机体系结构	52
1.1.3 十六进制记数法	17	2.1.1 CPU 基础知识	52
1.2 主存储器	18	2.1.2 存储程序概念	53
1.2.1 存储器结构	18	2.2 机器语言	54
1.2.2 存储器容量的度量	19	2.2.1 指令系统	54
1.3 海量存储器	20	2.2.2 一种演示用的机器语言	56
1.3.1 磁学系统	20	2.3 程序执行	58
1.3.2 光学系统	22	2.3.1 程序执行的一个例子	60
1.3.3 闪存驱动器	23	2.3.2 程序与数据	62
1.3.4 文件存储及检索	24	*2.4 算术/逻辑指令	63
1.4 用位模式表示信息	25	2.4.1 逻辑运算	63
1.4.1 文本的表示	25	2.4.2 循环移位及移位运算	65
1.4.2 数值的表示	26	2.4.3 算术运算	66
1.4.3 图像的表示	27	*2.5 与其他设备通信	67
1.4.4 声音的表示	28	2.5.1 控制器的作用	67
*1.5 二进制系统	29	2.5.2 直接内存存取	68
1.5.1 二进制记数法	29	2.5.3 握手	69
1.5.2 二进制加法	31	2.5.4 流行的通信媒介	69
1.5.3 二进制中的小数	32	2.5.5 通信速率	70
*1.6 整数存储	33	*2.6 其他体系结构	70
1.6.1 二进制补码记数法	33	2.6.1 流水线	70
1.6.2 余码记数法	36	2.6.2 多处理器计算机	71
*1.7 小数的存储	37	复习题	72
1.7.1 浮点记数法	37	社会问题	77

课外阅读	77	社会问题	131
第3章 操作系统	79	课外阅读	132
3.1 操作系统的历史	79	第5章 算法	134
3.2 操作系统的体系结构	82	5.1 算法的概念	134
3.2.1 软件概述	82	5.1.1 概览	134
3.2.2 操作系统组件	84	5.1.2 算法的正式定义	135
3.2.3 系统启动	86	5.1.3 算法的抽象本质	136
3.3 协调机器的活动	88	5.2 算法的表示	136
3.3.1 进程的概念	88	5.2.1 原语	137
3.3.2 进程管理	88	5.2.2 伪代码	139
*3.4 处理进程间的竞争	90	5.3 算法的发现	142
3.4.1 信号量	90	5.3.1 问题求解的艺术	142
3.4.2 死锁	91	5.3.2 入门	144
3.5 安全性	93	5.4 迭代结构	146
3.5.1 来自机器外部的攻击	93	5.4.1 顺序搜索法	147
3.5.2 来自机器内部的攻击	94	5.4.2 循环控制	148
复习题	95	5.4.3 插入排序算法	151
社会问题	98	5.5 递归结构	154
课外阅读	98	5.5.1 二分搜索算法	154
第4章 组网及因特网	99	5.5.2 递归控制	159
4.1 网络基础	99	5.6 有效性和正确性	160
4.1.1 网络分类	99	5.6.1 算法有效性	160
4.1.2 协议	100	5.6.2 软件验证	163
4.1.3 网络互连	102	复习题	167
4.1.4 进程间通信的方法	104	社会问题	171
4.1.5 分布式系统	105	课外阅读	171
4.2 因特网	106	第6章 程序设计语言	172
4.2.1 因特网体系结构	106	6.1 历史回顾	172
4.2.2 因特网编址	108	6.1.1 早期程序设计语言	172
4.2.3 因特网应用	109	6.1.2 独立并超越机器	174
4.3 万维网	113	6.1.3 程序设计范型	175
4.3.1 万维网实现	113	6.2 传统的程序设计概念	179
4.3.2 HTML	114	6.2.1 变量和数据类型	180
4.3.3 XML	117	6.2.2 数据结构	181
4.3.4 客户端和服务端的活动	118	6.2.3 常量和字面量	182
*4.4 因特网协议	119	6.2.4 赋值语句	183
4.4.1 因特网软件的分层方法	119	6.2.5 控制语句	184
4.4.2 TCP/IP协议簇	122	6.2.6 注释	187
4.5 安全性	123	6.3 过程单元	188
4.5.1 入侵的形式	124	6.3.1 过程	188
4.5.2 防护和对策	125	6.3.2 参数	189
4.5.3 加密	126	6.3.3 函数	192
4.5.4 网络安全的法律途径	128	6.4 语言实现	193
复习题	130	6.4.1 翻译过程	193

6.4.2 软件开发包	198	8.2.2 静态结构与动态结构	247
6.5 面向对象程序设计	199	8.2.3 指针	248
6.5.1 类和对象	199	8.3 数据结构的实现	248
6.5.2 构造器	202	8.3.1 数组的存储	248
6.5.3 附加特性	202	8.3.2 列表的存储	251
*6.6 程序设计中的并发活动	204	8.3.3 栈和队列的存储	254
*6.7 说明性程序设计	206	8.3.4 二叉树的存储	255
6.7.1 逻辑推演	206	8.3.5 数据结构的操作	257
6.7.2 Prolog	208	8.4 一个简短案例	259
复习题	210	8.5 定制的数据类型	263
社会问题	213	8.5.1 用户自定义数据类型	263
课外阅读	214	8.5.2 抽象数据类型	264
第7章 软件工程	215	*8.6 类和对象	266
7.1 软件工程学科	215	*8.7 机器语言中的指针	267
7.2 软件生命周期	217	复习题	269
7.2.1 周期是个整体	217	社会问题	273
7.2.2 传统的开发阶段	218	课外阅读	274
7.3 软件工程方法	220	第9章 数据库系统	275
7.4 模块化	221	9.1 数据库基础	275
7.4.1 模块式实现	222	9.1.1 数据库系统的重要性	275
7.4.2 耦合	224	9.1.2 模式的作用	276
7.4.3 内聚	225	9.1.3 数据库管理系统	277
7.4.4 信息隐藏	225	9.1.4 数据库模型	278
7.4.5 构件	226	9.2 关系模型	279
7.5 行业工具	227	9.2.1 关系设计中的问题	279
7.5.1 较老的工具	227	9.2.2 关系运算	282
7.5.2 统一建模语言	228	9.2.3 SQL	285
7.5.3 设计模式	232	*9.3 面向对象数据库	287
7.6 质量保证	233	*9.4 维护数据库的完整性	289
7.6.1 质量保证的范围	233	9.4.1 提交/回滚协议	289
7.6.2 软件测试	234	9.4.2 锁定	290
7.7 文档编制	235	*9.5 传统的文件结构	291
7.8 人机界面	236	9.5.1 顺序文件	291
7.9 软件所有权和责任	238	9.5.2 索引文件	294
复习题	240	9.5.3 散列文件	294
社会问题	242	9.6 数据挖掘	297
课外阅读	243	9.7 数据库技术的社会影响	299
第8章 数据抽象	244	复习题	300
8.1 数据结构基础	244	社会问题	303
8.1.1 数组	244	课外阅读	304
8.1.2 列表、栈和队列	245	第10章 计算机图形学	305
8.1.3 树	245	10.1 计算机图形学的范围	305
8.2 相关概念	247	10.2 3D图形概述	307
8.2.1 抽象	247	10.3 建模	308

10.3.1 单个物体的建模	308	11.6 机器人学	356
10.3.2 整个场景的建模	313	11.7 后果的思考	358
10.4 渲染	314	复习题	359
10.4.1 光-表面交互	314	社会问题	363
10.4.2 裁剪、扫描转换和隐藏面的消除	316	课外阅读	364
10.4.3 着色	319	第 12 章 计算理论	365
10.4.4 渲染-流水线硬件	320	12.1 函数及其计算	365
*10.5 处理全局照明	321	12.2 图灵机	367
10.5.1 光线跟踪	321	12.2.1 图灵机的原理	367
10.5.2 辐射度	323	12.2.2 丘奇-图灵论题	369
10.6 动画	323	12.3 通用程序设计语言	370
10.6.1 动画基础	323	12.3.1 Bare Bones语言	370
10.6.2 运动学和动力学	325	12.3.2 用Bare Bones语言编程	372
10.6.3 动画制作过程	326	12.3.3 Bare Bones的通用性	373
复习题	326	12.4 一个不可计算的函数	375
社会问题	328	12.4.1 停机问题	375
课外阅读	329	12.4.2 停机问题的不可解性	376
第 11 章 人工智能	330	12.5 问题的复杂性	379
11.1 智能与机器	330	12.5.1 问题复杂性的度量	379
11.1.1 智能体	330	12.5.2 多项式问题与非多项式问题	382
11.1.2 研究方法	332	12.5.3 NP问题	383
11.1.3 图灵测试	332	*12.6 公钥密码学	386
11.2 感知	333	12.6.1 模表示法	386
11.2.1 理解图像	333	12.6.2 RSA公钥加密系统	387
11.2.2 语言处理	335	复习题	389
11.3 推理	338	社会问题	392
11.3.1 产生式系统	338	课外阅读	392
11.3.2 搜索树	340	附录 A ASCII 码	394
11.3.3 启发式搜索	342	附录 B 处理二进制补码表示的电路	395
11.4 其他研究领域	346	附录 C 一种简单的机器语言	397
11.4.1 知识的表达和处理	346	附录 D 高级编程语言	399
11.4.2 学习	347	附录 E 迭代结构与递归结构的等价性	401
11.4.3 遗传算法	349	索引	403
11.5 人工神经网络	349	问题与练习答案 (图灵社区网站下载)	
11.5.1 基本特性	350		
11.5.2 训练人工神经网络	351		
11.5.3 联想记忆	353		

绪论

在开篇的这一章，我们探讨计算机科学所涉及的领域，介绍其历史背景，然后为我们的深入学习奠定基础。

本章内容

0.1 算法的作用

0.2 计算机器的由来

0.3 算法的科学

0.4 抽象

0.5 学习大纲

0.6 社会影响

社会问题

课外阅读

计算机科学这门学科，是要为计算机设计、计算机程序设计、信息处理、问题的算法解决方案和算法过程本身等主题建立科学的基础。计算机科学既是当今计算机应用的支柱，又是今后计算基础设施的基础。

本书将详细介绍计算机科学，探索广阔的主题，包括构成大学计算机科学课程的大部分主题。我们要领略这个领域的博大精深和变化发展。因此，除了这些主题本身，我们还关注它们的历史发展、现今的研究动态以及今后的前景。我们的目标是让人们以学以致用态度来对待计算机科学——既帮助那些要在此领域继续深入学习的人，也促成其他领域的人在技术不断进步的社会崭露头角。

0.1 算法的作用

首先让我们了解一下计算机科学最基础的概念——“算法”。一般来讲，**算法**（algorithm）是完成一项任务所遵循的一系列步骤。（在第5章，我们将给出比较精确的定义。）例如，有关于烹饪的算法（称为菜谱），有在陌生城市准确定位的算法（通常称为道路指南），有使用洗衣机的算法（通常标示在洗衣机的内盖上或者贴在自助洗衣店的墙上），有演奏音乐的算法（以乐谱的形式表示），还有魔术表演的算法（见图0-1）。

在一台机器（如计算机）执行一项任务之前，必须先找到完成这项任务的算法，并且用与该机器兼容的形式表示出来。某一个算法的表示称作一个**程序**（program）。为了人们读写方便，计算机程序通常打印在纸上或者显示在计算机屏幕上。为了便于机器识别，程序需要采取一种与该机器技术兼容的形式进行编码。开发程序、采取与机器兼容的形式进行编码并将其输入到机器中的过程，称作**程序设计**（programming）。程序及其所表示的算法总称为**软件**（software），而机器设备本身则称为**硬件**（hardware）。

效果：表演者从一副普通的扑克牌中抽取若干张牌，充分洗牌后将牌正面朝下展开在桌面上。然后，表演者会根据观众的要求相应地翻出红牌或者黑牌。

秘诀：

步骤 1 从一副普通扑克牌中抽取10张红牌和10张黑牌。把它们根据颜色分为两摞，正面朝上放在桌面上。

步骤 2 告诉观众你已经选取了若干张红牌和黑牌。

步骤 3 拿起红牌，装作整理成一摞的样子，用左手正面朝下拿好牌，同时用右手的拇指和食指挤压这摞牌的两端，把牌面向下推，使得每张牌呈现向下的弧形。然后，把这摞红牌扣在桌子上并宣布：“这是其中的红牌。”

步骤 4 拿起黑牌，模仿步骤3的方法，但要使这些牌呈现向上的弧形。然后，把牌扣在桌子上并宣布：“这是其中的黑牌。”

步骤 5 把黑牌放回桌面后，立即用双手把红牌和黑牌混在一起（仍然正面朝下），平铺在桌面上。告诉大家你已经洗好了牌。

步骤 6 只要桌面上还有扣着的牌，可以重复下面的步骤：

6.1 请观众要一张红牌或黑牌；

6.2 如果所要的牌为红色，而且桌面上倒扣有凹形的牌，就翻开其中的一张并告诉大家“这是一张红牌”；

6.3 如果所要的牌为黑色，而且桌面上倒扣有凸形的牌，就翻开其中的一张并告诉大家“这是一张黑牌”；

6.4 否则，就告诉大家桌面上没有所要求颜色的牌了，然后翻开桌面上所有的牌，以证实你的断言。

图0-1 一个魔术的算法

算法的研究起源于数学学科。事实也的确如此，它是数学家的重要活动，远远早于当今计算机的出现。它的目标是找出一组指令，描述如何解决某一特定类型的所有问题。求解两个多位数商的长除算法是早期研究中一个最著名的例子。另一个例子是古希腊数学家欧几里得发现的欧几里得算法——求两个正整数的最大公约数（见图0-2）。

描述：本算法假定输入是两个正整数，目的是要计算这两个数的最大公约数。

过程：

步骤 1 将这两个数中较大的一个和较小的一个分别赋予 M 和 N 。

步骤 2 用 M 除以 N ，余数设为 R 。

步骤 3 如果 R 不为0，那么将 N 的值赋予 M ，并将 R 的值赋予 N ，然后回到步骤2；否则最大公约数就是 N 当前被赋予的值。

图0-2 求两个正整数的最大公约数的欧几里得算法

一旦我们找到了执行一个任务的算法，那么在执行该任务时，就不再需要了解该算法所依据的原理——任务的完成演变成了遵照指令操作的过程。（不需要了解算法的工作原理，我们就可以根据长除算法求商，或者根据欧几里得算法求得最大公约数。）在某种意义上，解决这个问题的智能被编码到算法中。

我们能够设计出那些执行有用任务的机器，是因为我们有上述能力通过算法来捕获和传达智能（至少是智能行为）。因此，机器的智能级别受限于算法所传达的智能。只有存在执行某一项任务的算法时，我们才可以制造出执行这一任务的机器，换言之，如果我们找不到一个解决某问题的算法，那么这个问题的解决就超出了机器的能力范围。

20世纪30年代，库尔特·哥德尔（Kurt Gödel）发表了不完备性定理的论文，它使确定算法能力的局限性成为数学的一个研究课题。这个定理的主旨就是，在任何一个包括传统意义的算术系统的数学理论内，总有一些命题的真伪无法通过算法的手段来确定。简言之，对于我们算术系统的任何全面研究都超越了算法活动的能力。

这一认识动摇了数学的基础，于是关于算法能力的研究随之而来，它开创了今天计算机科学这门学科。的确，正是算法的研究构成了计算机科学的核心。

0.2 计算机器的由来

今天的计算机有着庞大久远的世系渊源，其中较早的一个计算设备是算盘。历史告诉我们，它很可能源于中国古代且曾被用于早期希腊和罗马文明。算盘本身非常简单，一个矩形框里固定着一组小棍，而每个小棍上又各串有一组珠子（见图0-3）。在小棍上，珠子上下移动的位置就表示所存储的值。正是这些珠子的位置代表了这台“计算机”所表示和存储的数据。这台机器是依靠人的操作来控制算法执行的。因此，算盘自身只算得上一个数据存储系统，它必须在人的配合下才成为一台完整的计算机器。

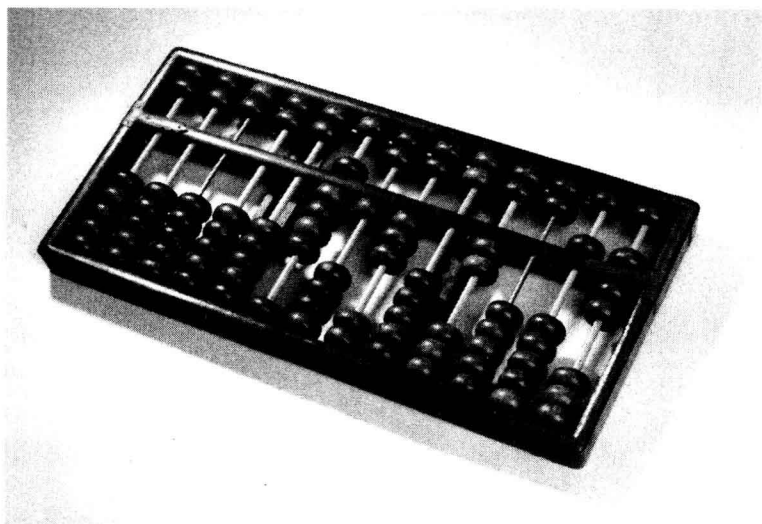


图0-3 算盘（Wayne Chandler拍摄）

从中世纪到近代，人们开始探求更复杂的计算机器。后来，一些发明人开始基于齿轮技术设计计算机器。采用这种技术的发明家有法国的布莱斯·帕斯卡尔（Blaise Pascal, 1623—1662）、德国的戈特弗里德·威尔赫尔姆·莱布尼茨（Gottfried Wilhelm Leibniz, 1646—1716）和英国的查尔斯·巴贝奇（Charles Babbage, 1792—1871）等。这些机器利用齿轮的位置来表示数据，要在规定齿轮初始位置的基础上机械地输入数据。帕斯卡尔和莱布尼茨的机器所计算的结果是通过观察齿轮的最终位置得到的。而巴贝奇构想了这样一种机器，可以把计算的结果打印在纸上，从而消除可能出现的誊写错误。

就执行算法的能力而言，我们可以看到这些机器在灵活性上的进步。帕斯卡尔的机器只是为了执行加法而设计，因此必须将正确的步骤序列嵌入到机器结构本身。同样，莱布尼茨的机器也把算法嵌入在其体系结构中，但它提供了多种算术运算供操作员选择。巴贝奇的差分机仅造了一个演示模型，可以被修改以执行各种计算，但他设计的分析机（因一直没有得到任何基金支持而未生产出来）则能够在纸卡片上读取以洞孔形式表示的指令。所以，巴贝奇的分析机是可编程的。事实上，奥古斯塔·艾达·拜伦（Augusta Ada Byron）通常被称为世界上第一位程序员，她曾发表过一篇论文，阐述巴贝奇的分析机如何编程并实现各种各样的计算。

巴贝奇的差分机

查尔斯·巴贝奇设计的这台机器的确是现代计算机设计的先驱。如果能用比较经济的技术制造出这台机器，如果当时商业和政府数据处理需求达到今天的规模，那么巴贝奇的思想可能在19世纪就引发了计算机革命。事实上，在他有生之年只是造出了差分机的演示模型。该机器通过“逐次差分”的计算来决定数字值。我们来研究一下计算整数平方的问题，帮助我们加深对这一技术的理解。首先我们从基础知识开始，0的平方是0，1的平方是1，2的平方是4，3的平方是9。据此，可以按照下面的方法得到4的平方（见下图）。首先我们来计算一下已知平方之间的差： $1^2 - 0^2 = 1$ ， $2^2 - 1^2 = 3$ ， $3^2 - 2^2 = 5$ 。然后，我们计算这些结果的差： $3 - 1 = 2$ ， $5 - 3 = 2$ 。注意看，这些差都是2。假设这个规律能够成立（数学上可以证明它是成立的），那么我们可以得出这样的结论： $(4^2 - 3^2)$ 和 $(3^2 - 2^2)$ 之间的差也一定是2。因为 $(4^2 - 3^2)$ 比 $(3^2 - 2^2)$ 大2，所以 $4^2 - 3^2 = 7$ ， $4^2 = 3^2 + 7 = 16$ 。现在，我们已经知道了4的平方，那么就可以依据 1^2 、 2^2 、 3^2 和 4^2 的值继续计算5的平方。（虽然更深入地讨论逐次差分已经超出了本书范围，但是学过微积分的学生可能已观察到，前面的例子是基于这样的事实： $y = x^2$ 的二阶导数是一条直线。）

x	x^2	一阶差分	二阶差分
0	0		
1	1	1	2
2	4	3	2
3	9	5	2
4	16	7	2
5			2

奥古斯塔·艾达·拜伦

洛夫莱斯伯爵夫人奥古斯塔·艾达·拜伦是计算界关注的焦点人物。艾达·拜伦的一生近乎悲惨，去世时还不到37岁（1815—1852）。她体弱多病，身处限制妇女从业的社会，还是新教教徒。尽管对广泛的科学感兴趣，但她还是专注于数学研究。1833年，目睹了查尔斯·巴贝奇的差分机样机演示后，她就被这台机器迷住了。她把一篇讨论巴贝奇分析机设计的论文从法文翻译为英文，算是她最早对计算机科学所做的贡献。巴贝奇还鼓励她在翻译中增加一个附录，介绍该机器的应用，并提供了例子说明该机器如何进行编程以实现各种各样的任务。巴贝奇对艾达·拜伦的工作十分热情，这是因为他希望论文的出版可以帮助他得到资金援助，以建造他的分析机。（作为拜伦勋爵的女儿，艾达·拜伦具有名人的地位，在生意场上也有一定的关系。）巴贝奇最终也没有得到资金援助，但是艾达·拜伦的附录保存了下来。人们认为该附录包含了第一批计算机程序的例子。关于巴贝奇对艾达的工作影响有多大，研究计算机历史的学者们一直争论不休。有些历史学家认为巴贝奇作出了重大贡献，但另外一些人则认为巴贝奇并没有帮到艾达，从很大程度上来看反而是一种阻碍。无论如何，奥古斯塔·艾达·拜伦都被认为是世界上第一位计算机程序员，美国国防部为了纪念这位伟大的女性，用她的名字命名了一种程序设计语言（Ada）。