

国外经典计算机科学教材

Data Structures and Other Objects
Using Java (Second Edition)

数据结构 Java 语言描述 (第二版)

[美] Michael Main 著
孔芳 周丽琴 译



中国电力出版社
www.infopower.com.cn

国外经典计算机科学教材

Data Structures and Other Objects
Using Java (Second Edition)

数据结构

Java 语言描述

(第二版)

[美] Michael Main 著
孔芳 周丽琴 译



中国电力出版社

www.infopower.com.cn

Data Structures & Other Objects Using Java,second edition (ISBN 0-201-74093-1)

Michael Main

Copyright © 2003 Pearson Education, Inc.

Original English Language Edition Published by Addison Wesley, Inc.

All rights reserved.

Translation edition published by PEARSON EDUCATION ASIA LTD and CHINA ELECTRIC
POWER PRESS, Copyright © 2005.

本书翻译版由 Pearson Education 授权中国电力出版社独家出版、发行。
未经出版者书面许可，不得以任何方式复制或抄袭本书的任何部分。

本书封面贴有 Pearson Education 防伪标签，无标签者不得销售。
北京市版权局著作权合同登记号 图字：01-2003-3829 号

图书在版编目 (CIP) 数据

数据结构：Java 语言描述/ (美) 梅因 (Main,M.) 著；孔芳，周丽琴译。

—北京：中国电力出版社，2004

(国外经典计算机科学教材系列)

ISBN 7-5083-2198-7

I.数... II.①梅...②孔...③周... III.①数据结构

②JAVA 语言—程序设计 IV.①TP311.12②TP312

中国版本图书馆 CIP 数据核字 (2004) 第 097697 号

从 书 名：国外经典计算机科学教材系列

书 名：数据结构：Java语言描述

编 著：(美) Michael Main

翻 译：孔芳 周丽琴

责任编辑：陈维宁

出版发行：中国电力出版社

地址：北京市三里河路6号 邮政编码：100044

电话：(010) 88515918 传 真：(010) 88518169

印 刷：三河汇鑫印务有限公司

开本尺寸：185×233 印 张：42.5 字 数：1095千字

书 号：ISBN 7-5083-2198-7

版 次：2005年8月北京第1版 2005年8月第1次印刷

定 价：66.00元

版权所有 翻印必究

出版说明

新世纪的朝阳刚刚露出丝抹微红,如火如荼的全球信息化浪潮便汹涌而至,让人无时无刻不感受到新一轮产业革命的气息。如何在这场变革中占尽先机,既是对民族信息业的挑战,也是机遇。从而,作为民族信息产业发展基石的高等教育事业就被赋予了比以往更重的责任,对培养和造就我国 21 世纪的一代新人提出了更高的要求。但在计算机科学突飞猛进的同时,专业教材的发展却严重滞后,越来越成为人才培养的瓶颈。同时,以美国为代表的西方国家计算机科学教育经历了充分的发展,产生了一批有着巨大影响力的经典教材,因此,以批判、借鉴的态度有选择地引进这些国外经典计算机教材,将促进国内教学体系和国外接轨,大大推动我国计算机教育事业的发展。

中国电力出版社进入计算机图书市场已有近 6 个年头,通过坚持“高端、精品、经典”战略,致力于与国外著名出版机构合作,出版了大批博得计算机业界和教育界赞誉的作品。通过与信息技术教育界人士的广泛沟通,同时依托丰富的出版资源,中国电力出版社适时推出了“国外经典计算机科学教材”的出版计划。本次教材出版计划是和美国最大的计算机教育出版机构——Pearson 教育集团(Addison-Wesley、Prentice-Hall 等皆为其下属子公司)合作,依托其数十年积累的大批经典教材资源,确保了教材选题的权威经典。

为保证这套教材的含金量,并做到有的放矢,我们在国内组织了由中国科学院、北京大学等一流院校教师组成的专家指导委员会,对高校课程教学体系做了系统、详细的调查,听取了众多教育专家、行业专家的意见,对教育部的教育规划进行了认真研究,并深入了解国外大学实际教学选用的教材状况,对国外教材做了理性的分析,确立了依托国家教育计划、传播先进教学理念、为培养符合社会需要的高素质创新型人才服务,来作为本次“国外经典计算机科学教材”出版计划的宗旨。

我们从 2002 年的下半年开始着手这套教材的策划工作,并多次组织了专家研讨会、座谈会等,分析现有教材的优点与不足,采其精华,并力争体现本套教材的质量和特色。

1. 深入理解国内的教学体系结构,并比照国外相同专业的课程设置,既具有现实的适用性,又立足发展眼光,具备一定的前瞻性。

2. 以计算机专业的核心课程为基础,同时配合专业教学计划,争取覆盖专业选修课程和专业任选课程。

3. 选取国外的最新教材版本,同时对照国内同专业课程的学时要求,对不适用的版本进行剔除,充分满足国内教学要求。

4. 根据专业对口和必须具备同课程教学经验的要求,严格挑选译者,并严把质量关,确保教材翻译的高质量。

5. 通过从原出版社网站下载勘误表及与原书作者进行沟通的方式,对原书中的错误一一做了修改。

6. 对教材出版的后期工作, 如审校、编辑、排版、印刷进行了严格的质量把关。

经过专家指导委员会的集体讨论, 并广泛听取广大高等院校师生的意见, 反复比较, 从数百种国外教材中遴选出数十种, 列入第一阶段的出版计划。这些教材的作者无一不是学富五车的大师, 如 Stallings, Date, Ullman, Aho, Bryant, Sedgewick 等, 他们的作品均是一版再版, 并被众多国外一流大学如 Stanford University, MIT, UC Bekerley, Carnegie Mellon Univeristy, University of Michigan 等采用为教材。拟订的第一阶段出版计划包括 30 种图书, 内容覆盖程序设计、数据结构、操作系统、计算机体系结构、数据库、编译原理、软件工程、图形学、通信与网络、离散数学等计算机专业核心基础课程, 基本满足国内计算机专业的教学要求。

此外, 为了帮助广大任课教师加深对本系列教材的理解, 减轻他们的备课难度, 我们从国外出版机构引进了大批的课程教学辅助资料, 并积极延请国内优秀教师, 根据其使用该系列教材中的教学经验, 着手编写更加适合国内应用状况的教辅材料。

由于我们对国内高校计算机教育存在认识深度上的不足, 在选题、翻译、编辑加工出版等方面的工作中还有许多有待提高之处, 恳请广大师生和读者提出批评和建议, 并期待有更多的人加入到我们的工作中来。我们的联系方式是:

电子邮件: csbook@cepp.com.cn

联系电话: 010-88515918-300

联系地址: 北京市西城区三里河路 6 号中国电力出版社

邮政编码: 100044

译 者 序

用计算机解决任何实际问题都离不开数据表示和数据处理,数据表示和数据处理的核心问题之一就是确定数据结构,并实现它,而这正是数据结构课程的基本内容。数据结构是计算机及其相关专业的一门重要的专业基础课,为计算机专业人员提供重要的专业基础知识和必要的技能训练。

本书系统地介绍了各种常用的数据结构,对基本概念、基本原理和基本方法做了深入浅出的介绍,同时对相关的算法设计也做了详细和通俗的讲解。本书的一大特点就是采用了非常流行的面向对象程序设计语言——Java 语言作为算法描述语言,将传统的数据结构内容与面向对象技术完整结合,使讲解更加自然贴切。

全书共有 14 章,分成三个层次不断深入。前 4 章首先对 Java 语言的基本知识作了简要复习;第 5 章站在数据结构的角度的角度介绍了几种 Java 特有的技术,包括如何构建基于 Java 的 Object 类型的通用集合类,如何使用 Java 接口和 API 类,如何构建和使用 Java 迭代器等;从第 6 章开始依次介绍了特定的数据结构(堆栈、队列、树、散列表和图)或相关的编程技术(递归、排序及继承)。

本书还根据需要添加了其他一些新的 Java 特性(如使用断言来检查前置条件),同时详细地介绍了继承、新的 Java API 类(例如 ArrayList、Vector 和 HashTable 的另一可选类 HashMap),以及 Javadoc 的新特性等。

另外,本书在章节的安排上还充分考虑到了读者现有的知识背景,各章节间相对独立,读者可以根据自身的知识结构有选择地跳过相关章节。

本书覆盖面广,结构清晰合理,是一本全新的有关数据结构的教材,可作为大专院校计算机专业与计算机应用专业学生的教材和参考书,对于计算机科学与工程领域的从业人员也是一本很好的参考书。

本书由苏州大学的孔芳、周丽琴主译,参与翻译的人员还有赵丰收、陈浩奎、陈华全、李明、祝青、谢君英、杨会珍和于旭升。本书的初排和代码录入工作由盛海燕、谢小花完成。在翻译过程中,译者尽可能地参考了最新的数据结构教材,并结合了译者的教学经验。但由于译者的知识局限,加之时间也比较仓促,译文中难免会出现一些疏漏,诚恳地希望广大读者给予批评与指正。

前 言

Java 为程序员提供了引人入胜的论坛。有了 Java 和 World Wide Web，程序员的作品可以通过动人的图片实现快速的全球分发，并在反馈、修改、软件重用以及功能完善方面具有非凡的能力。当然，Java 的快速发展也得益于它的其他品质：它能够实现整洁的、面向对象的设计；Java 的语法结构是大家熟悉的；Java 类库中包含优秀的现成特性集。但是，是 World Wide Web 的迷人之处使得有经验的程序员和刚进入计算机科学领域的新手都选择了 Java。

也就是说，令人满意的是，即使在课程中开始使用 Java 语言，第一学年的核心课程（ACM 课程中的 CS1 和 CS2）仍然没有变。就算课程变成了相应的面向对象方法，但已被证明的表示和使用数据结构的方法对数据结构（或 CS2）课程中刚起步的程序员来说仍然是至关重要的。

本书正是为演进中的 CS2 课程编写的，其中的程序设计采用 Java 来进行。本书的重点在于规格说明、设计、实现，以及通常在第二学期面向对象课程中讲述的基本数据类型的使用。本书广泛覆盖了一些重要的程序设计技术，如递归、查找和排序。还介绍了算法的大 O 时间分析，选读的附录中介绍了如何编写交互式 applet 来测试数据结构的实现，以及使用 Javadoc 来规定前置—后置条件契约。

本书假设学生已经修完了计算机科学导论和程序设计课程，但本书内容还是包含了在入门课程中没有完全覆盖的主题（例如 Java 的 Object 类型以及参数传递的精确描述）。前言的其余部分将从第二版增加的新内容开始，讨论本书将如何逐一介绍这些内容。

第二版：新增内容和附加的 Web 项目

第二版中介绍的基本数据结构的内容与第一版相同。具体地说，CS2 课程强调了规格说明、设计和分析与特定语言无关的数据结构的能力，以及在现代语言中实现新的数据结构和使用现有数据结构的能力。前 4 章将采用这样的方法，学生先复习 Java，然后继续学习，使用很容易应用于高级语言的数组和链表技术来实现大多数基本数据结构（包和序列类）。

第 5 章的内容有点偏离主题，提出了几种 Java 特有的技术，每种技术都与数据结构紧密相关：如何构建基于 Java 的 Object 类型的通用集合类；使用 Java 接口和 API 类；构建和使用 Java 迭代器。这些内容中的大部分都是第二版中新增的。尽管接下来的大部分内容与 Java 特性无关，但学生在这时学习这些内容是很自然的。

接下来的章节重点讨论了特定的数据结构（堆栈、队列、树、散列表和图）或程序设计技术（递归、排序、继承）。直到开始学习树之前才介绍递归，这样可以使用一些简单但很有意义的例子来介绍递归，这些例子不仅仅完成尾递归（tail recursion）。这样就避免了一些

学生在刚学习递归时总将它误认为是某种神奇的循环。在介绍完开始的这些简单例子之后（包括这一版的一个新例子），我们将学习包含两个递归调用的大型例子，其中涉及二叉树的常见状态。

本书特别强调了树，第 10 章给出了使用平衡树（堆项目和 B-树项目）数据结构两个重要例子。树和其他方面的附加项目也已经在线添加到了 <http://www.aw.com/cssupport> 中。

树和其他方面的新项目

全书根据需要添加了一些新的 Java 特性：使用断言来检查前置条件、对继承的更完整介绍、构建诸如 Othello 或 Connect Four 这样的双人策略游戏的例子、对新的 Java API 类（例如 Vector 和 HashTable 的另一可选类 ArrayList 和 HashMap）以及 Javadoc 的新特性的介绍，等等。这些 Java 新技术使用起来很方便，学生们需要及时更新知识——但设计、规格说明、编制文档、实现、使用和分析数据结构的方法对学生来说是具有持久影响的。

每种数据类型的五个步骤

本书的核心由一些知名的数据类型构成：集合、包（或多集合）、顺序表、堆栈、队列、表和图。此外，还有一些额外补充的数据类型，例如优先队列。有些类型使用多种方法实现，例如包首先通过将数据元素存储在数组中来实现，随后又使用二分查找树再次实现。每种数据类型都按照下列五个步骤的模式介绍。

第一步：抽象地理解数据类型。在这一层次，学生在概念和图形上对数据类型及其操作进行理解。例如，学生可以可视化堆栈，理解元素的出栈和入栈操作。理解简单的应用程序，并手工去执行，如使用堆栈来反转单词的字母次序。

第二步：以 Java 类的方式编写数据类型的规格说明。在这一步中，学生将看到并学会如何为一个实现数据类型的 Java 类编写规格说明。使用 Javadoc 工具编写的规格说明包含构造函数的头部、公共方法和其他一些共有特性（例如受内存限制的约束）。每个方法的头部和前置-后置条件一起呈现，完整地指定了方法的行为。在这一层次，让学生意识到规格说明与任何选定的特定实现技术无关是非常重要的。实际上，可以为相同数据类型的几种不同实现而多次使用相同的规格说明。

第三步：使用数据类型。有了规格说明，学生可以编写小的应用程序或 applet 来演示数据类型的使用。这些应用程序仅仅建立在数据类型的规格说明之上，因为规格说明还没有进一步实现。

第四步：选择相应的数据结构，继续设计和实现数据类型。在很好地理解抽象的数据类型后，就可以选择相应的数据结构了，例如数组、节点的链表或节点构成的二叉树。对于许多数据类型，首先选择像数组这样的简单方法来设计和实现，然后再使用复杂的底层结构来重新设计和实现相同的数据类型。

由于使用了 Java 类，数据类型的实现将选中的数据结构（数组、其他对象的指针，等等）作为类的私有实例变量。在教学过程中，让学生清楚地理解把私有实例变量与数据类型的抽象概念相关联的规则是十分必要的。要求每个学生用英语句子清晰地写出这些规则，这称规则称为抽象数据类型的不变式。一旦编写好不变式，学生就可以继续实现各种方法。不

变式有助于编写正确的方法，其原因有两个：(a) 在方法开始运行时，每个方法（除了构造函数）都知道不变式为 true；(b) 在方法运行结束时，每个方法都负责确保不变式再次为 true。

第五步：分析实现。可以分析每个实现的正确性、灵活性，并且能够对操作进行时间分析（使用大 O 表示法）。当通过多个不同的方法实现同一个数据类型时，学生就有很好的机会能够进行这些分析。

本课程结束时，学生能学到什么？

在本课程结束时，学生能够彻底理解数据类型。知道如何使用数据类型以及如何通过多种方式来实现它们；明了选择不同实现的实际效果；能够通过大 O 分析推导算法效率，并能利用 ADT 不变式证明实现的正确性。

本书中的数据类型是 Java 类库的精简版本

本书的重要目标之一是增强学生在规格说明、设计和实现方面的体验，帮助学生提高程序推理的能力。但最重要的可能是展示了许多在各类情况下都能使用的类。这样学生就无需再从头开始编写所有的一切。我们告诉学生，可能某天他们正在考虑一个问题，突然之间，他们意识到大量工作都可以使用包（bag）、堆栈、队列等来实现。他们不再需要完成这其中的大量工作，而是可以直接利用本学期编写的包、堆栈或队列，甚至可以不做任何修改。或者，更有可能的是，他们将使用标准数据类型库中熟悉的数据类型，如推荐的 *Java* 类库。实际上，本书的一些数据类型是 JCL 的精简版本，所以，当学生开始真正学习 JCL 时，从如何使用类到构建类时需要考虑的因素，他们都会觉得非常熟悉。

其他基本主题

贯穿全书，我们还为“真实程序设计”的其他方面打下了基础，除了基本数据结构方面的内容之外，我们还讲述了下列主题。

面向对象程序设计。本书通过让学生深入理解 Java 类来为面向对象程序设计打下基础。前面讲述类的一些重要方面：方法的概念、私有和公共成员的区别、构造函数的用途，并展示复制和相等性测试。这些内容主要在第 2 章中介绍，在 CS1 课程中已经很好地理解了 Java 类的学生可以跳过这部分内容。

关于类的进一步内容在第一次使用动态数组时（第 3 章）介绍。这里介绍了使用更复杂复制方法的必要性。在讲述该 OOP 方面的知识时第一次使用了动态内存，这让学生对于如何将实例变量作为指针传给动态对象（如数组）有了具体的印象。

从概念上讲，OOP 最大的革新在于通过继承实现了软件重用。当然可以在数据结构课程的一开始就介绍继承（例如，将一个集合类作为包类的子孙来实现）。当然，过早介绍继承也会带来一定的麻烦，让学生一下子接受过多的新概念，使得他们没有精力去深刻理解基本数据结构。因此，我总是在课程的最后介绍继承，让学生认识这是一种新事物。但是也可以在理解了类之后立刻介绍继承（第 13.1 节和第 13.2 节）。出于这种想法，有些教师会提前讲解第 13 章，即在堆栈和队列之前讲解，这样就能从另一个类中派生出堆栈和队列。

另一个方法是把那些早就掌握了类的基本知识的学生分离出来，让他们去完成继承项目（如第 13.2 节的生态系统），而其余的学生则要首先学习类。

Java 对象。Java 的 Object 类型以其他所有（或几乎所有）的 Java 类型作为基础。八种基本类型不是 Java 对象，并且对于许多学生来说，CS1 课程的主要目标是使用这八种基本类型。正因为如此，最先介绍的数据结构是基本数值的集合，如整数包或 double 数值序列。

在介绍第一个集合数据结构之后不久，将在第 5 章介绍 Java 的 Object 类型，通过学习，学生意识到很容易就能将数据结构的一个实现（如整数包）转变成另一个实现（对象包）。对象包是最常见的包类型，并且通过使用封装类，甚至能够将基本值放入对象包中。绝大多数的子序列数据结构是将 Java 对象作为基本类型来实现的，当然也要提醒学生在必要的时候可以使用基本类型（如，一个应用程序使用 double 数值堆栈来计算算术表达式）。

迭代器。迭代器是 Java 类库的一个重要组成部分，通过使用它，程序员能够方便地遍历集合类中的元素。本文第 5 章介绍了 Iterator 接口。虽然在程序设计项目中提供了很好的使用迭代器的机会，如使用堆栈来实现二分查找树的迭代器（第 9 章），但是本书的其余部分没有直接使用迭代器。

递归。第一学期的课程经常让学生使用递归。但是，第一学期的许多例子都是尾递归，在尾递归中，方法的最后一步就是递归调用。这可能会误导学生，让他们错误地认为递归仅仅是一个循环。正是因为这样，我尽量避免在第二学期的课程中过早使用尾递归。

因此，在第二学期的课程中，我强调了递归的解决方案，而不是对尾递归的使用。在“递归思想”那一章连续提供了 4 个例子。学生对其中的两个例子——产生随机分形和走迷宫非常感兴趣。分形例子是一个图形 applet，而迷宫例子虽然是基于文本的，但富于创造精神的学生可以将它转变为图形 applet。这些递归例子（第 8 章）出现在树（第 9 章）之前，这是因为在递归树算法中递归变得非常重要。然而，如果教师想强调递归的话，可以将这一章的内容提前，甚至提到第 2 章之前。

如果课程时间足够，还可以学习高级树项目（第 10 章），在这一章中，我们将分析递归树算法，解释保证树平衡的重要性——能提高最差性能并能避免潜在的执行栈溢出。

查找和排序。第 11 章和第 12 章将介绍查找和排序算法的基本知识。该查找回顾了排序数组的二分查找，许多同学可能以前已经看到过了。在“查找”一章中还介绍了散列表，即使用链表代替开型寻址来实现 JCL 散列表的一个版本，这也是另一个散列表。“排序”一章介绍了简单的二次型排序方法，但是该章的绝大部分内容集中在快速算法：递归合并排序（最坏情况时间为 $O(n \log n)$ ）、Tony Hoare 的递归快速排序（平均时间为 $O(n \log n)$ ）、以及基于树的堆排序（最坏情况时间为 $O(n \log n)$ ）。

高级项目

本书提供了许多可选项目，这些项目可以使用更为高级的类来完成，或者可以由一些在大型类方面具有良好背景知识的学生来完成。这些高级项目包括：

- 任何数据结构的基于 applet 的交互式测试程序（列在附录 I 中）。

- 实现序列类的 `Iterator`（请查看第 5 章的程序设计项目）。
- 编写集合类的深度复制（`deep clone`）方法（请查看第 5 章的程序设计项目）。
- 编写一个应用程序的 `applet` 版本（如第 8.1 节的走迷宫或第 13.2 节的生态系统）。
- 使用堆栈构建二分查找树的迭代器（请查看第 9 章的程序设计项目）。
- 使用普通队列数组（第 7.4 节）或堆（第 10.1 节）实现优先队列。
- 使用 `B`-树实现集合类（第 10.2 节）。我特别重视这个项目，在实现这个项目的时候，应该给学生提供实现类的足够信息，这样，学生就不需要参阅其他书了。某些优秀的学生能够独立完成该项目。
- 一个继承项目，如第 13.2 节的生态系统。
- 第 14 章的图形类和相关的图形算法。优秀的学生也可以独立完成这些项目。

Java 语言版本

本书的所有源代码，包括对诸如 `ArrayList` 和 `TreeMap` 等新 Java 集合类的引用，都已在 Java2（1.4 版）下测试通过。有关 Sun 公司的所有 Java 产品信息，请访问 <http://java.sun.com/products/index.html>。

主题次序的灵活性

本书在编写上留给教师一定的回旋余地来调整教学内容的次序，以满足特定知识背景的学生需求，或者对选中的主题提前强调。下一页的图给出了各章节间的相互依赖关系。两个方框间的连线表明，上面的方框应该在下面的方框之前介绍。

下面对内容的次序给出一些建议：

典型课程。首先对于第 1~9 章，如果学生以前已经具有 Java 类的知识，则可跳过第 2 章。大多数章节只需要第一周时间介绍，但第 4 章（链表）、第 8 章（递归）或第 9 章（树）可能需要更多的时间。通常，第 1~9 章的内容需要花 13 周的时间，包括考试时间以及介绍链表和树所需的额外时间。剩余的几周可用于介绍第 10 章的树项目或二分查找（第 11.1 节）以及排序（第 12 章）。

重点学习面向对象程序设计。如果学生在其他课程中已经学习过查找和排序的内容，那么就有一定时间重点学习面向对象程序设计了。前三章需要详细介绍，然后介绍继承类（第 13.1 节）。到此为止，学生可以完成第 13.2 节中介绍的生态系统这一有趣的 OOP 项目了。然后介绍基本数据结构（第 4~7 章），其中队列以继承类的形式实现（第 13.3 节）。最后完成递归（第 8 章）和树（第 9 章）的内容时，侧重于递归方法。

加速课程。将前三章作为第一周的自学内容，然后从第 4 章（链表）开始讲授。这将在学期末空出两到三周的时间，以便学生可以在查找、排序以及其他高级主题（章节关系列表中的阴影部分）方面花更多的时间。

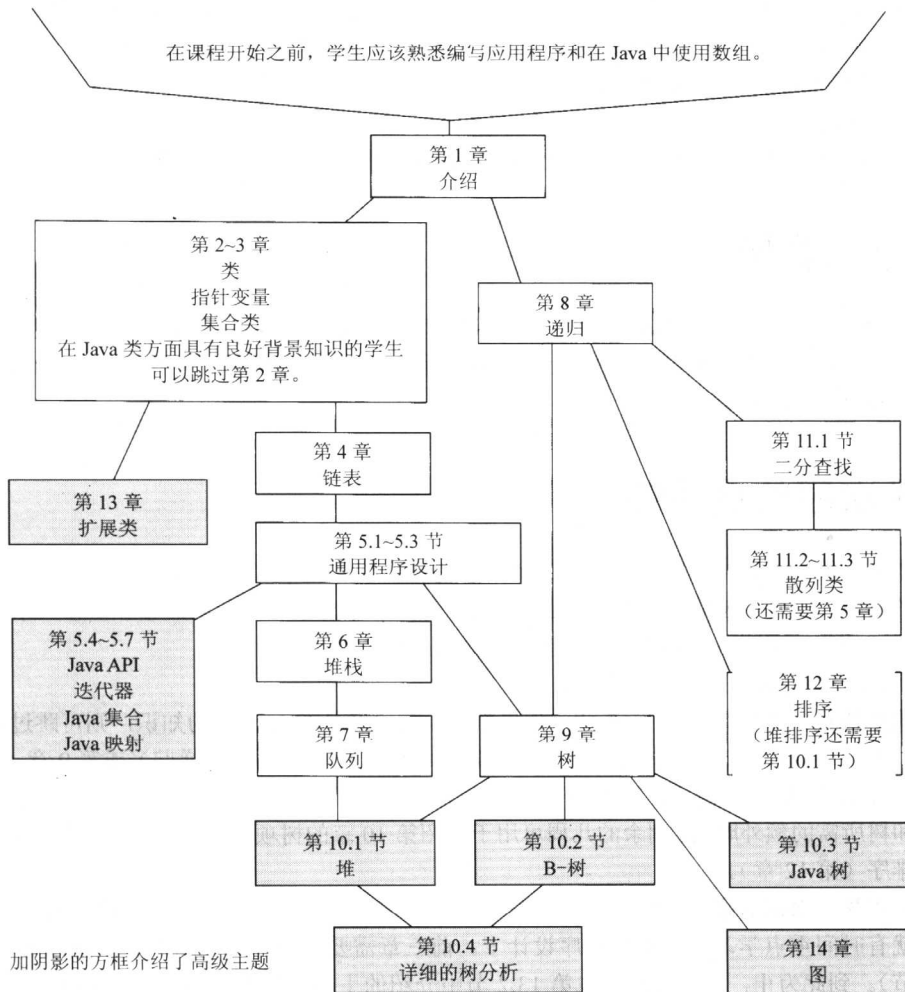
我们也在讲课时省略讲堆栈和列队的时间，以便进一步加快速度（让学生自学有关章

节)。

提前讲授递归/提前讲授排序。课程开始的前一到三周介绍递归思想。然后首先阅读第1章和第8章的内容，也可以补充一些额外的递归项目。

如果提前介绍递归，那么在介绍集合类前可以继续介绍二分查找（第11.1节）和大多数排序算法（第12章）。

在课程开始之前，学生应该熟悉编写应用程序和在Java中使用数组。



加阴影的方框介绍了高级主题

章节关系图

通过 Internet 进行补充

第二版已经
在线张贴了
许多新的项
目

所有的 Java 源代码均可以通过访问 www.aw.com/cssupport 获得。教师参考资料指南，包括示例作业、考试题、答案、教学大纲、PowerPoint 演示文稿，以及教学记录等，都可以在线获得。详细信息请联系 Addison-Wesley 出版公司的代表。

致谢

本书是我与 Walter Savitch 合作的结果，他是我的同事、朋友，一直以来都满腔热情地支持着我。在教学过程中，位于玻尔得的科罗拉多大学的学生给了我很多启示和乐趣，特别是 Mooses 和参加每周五下午课程的学生，他们学习的内容远远超出了数据结构。在这一版的修改过程中，本书由 Philip Barry、Arthur Crummer、Herber Dershem、Greg Dobbins、Zoran Duric、Dan Grecu、Scott Grissom、Bob Holloway、Rod Howell、Danny Krizanc、Ran Libeskind-Hadas、Meiliu Lu、Catherine Matthews、Robert Moll、Rebert Pastel、Don Slater、Ryan Stansifer 和 Deborah Trytten 进行了广泛的审阅。John Wegis 对本书的所有代码做了出色的技术检查。J.D. Baker 负责本书的文字录入工作，并在 Java applet 方面提供了帮助。非常感谢这些同事的出色评论和鼓励。

Addison-Wesley 的编辑和员工给予了极大的支持。Susan Hartman Sullivan, Michael Hirsch、Marilyn Lloyd 和 Beth Paquin 每天都给予了建议、鼓励和支持。漂亮的分形封面设计出于艺术家 Noel Griffin、设计师 Richard Hannus(hannusdesign.com 公司)和 Joyce Cosentino Wells 之手。非常感谢 Argosy 出版社的 Sally Boylan 和 Daniel Rausch 在最后阶段给予的极大灵活性。

在第二版的编写过程中，同事 John Gillett 的建议和友谊是非常珍贵的。此外，还要感谢我的朋友和同事，他们每天都给我支持和鼓励：David Benson、Zachary Bergen、Cathy Bishop、Lynne Conklin、Mary Dollarhide、Paul Eisenbrey、Stace Johnson、Rich Lorentz、Rick Lowell、Jim Martin、Tandy McKnight、Melanie Tem、Karl Winklmann、the 2002Boot Campers、Janet、Tim 和 Hannah。

Michael Main(main@colorado.edu)
Boulder, Colorado

目 录

译者序	
出版说明	
前 言	
第 1 章 软件开发周期	1
1.1 规格说明、设计和实现	3
1.2 运行时间分析	15
1.3 测试和调试	22
本章小结	29
自测练习参考答案	29
第 2 章 Java 类和信息隐藏	32
2.1 类及其成员	33
2.2 使用类	43
2.3 包	49
2.4 参数、equals 方法和 clone	53
本章小结	73
自测练习参考答案	73
程序设计项目	76
第 3 章 集合类	84
3.1 Java 数组简介	85
3.2 整数包 ADT	90
3.3 程序设计项目：序列 ADT	115
3.4 程序设计项目：多项式	125
本章小结	127
自测练习参考答案	128
程序设计项目	131
第 4 章 链表	136
4.1 链表基础	137
4.2 操作节点的方法	139
4.3 操作整个链表	149
4.4 使用链表的包 ADT	167
4.5 程序设计项目：带链表的序列 ADT	180
4.6 复杂链表	185
本章小结	188
自测练习参考答案	189

程序设计项目	194
第 5 章 通用程序设计	197
5.1 Java 的 Object 类型	198
5.2 对象包	202
5.3 对象结点	216
5.4 接口和 API (选读)	219
5.5 Iterator 接口 (选读)	226
5.6 带有 Iterator 的通用包类 (选读)	229
5.7 Java 的 Collection 和 Map 接口的介绍 (选读)	232
本章小结	239
自测练习参考答案	240
程序设计项目	243
第 6 章 堆栈	246
6.1 堆栈简介	246
6.2 堆栈的应用	250
6.3 堆栈 ADT 的实现	262
6.4 更复杂的堆栈应用	271
本章小结	278
自测练习参考答案	279
程序设计项目	280
第 7 章 队列	283
7.1 队列简介	283
7.2 队列的应用	287
7.3 队列 ADT 的实现	301
7.4 优先队列	314
本章小结	317
自测练习参考答案	318
程序设计项目	319
第 8 章 递归思想	322
8.1 递归方法	322
8.2 递归的研究: 分形和迷宫	330
8.3 推导递归	346
本章小结	352
自测练习参考答案	352
程序设计项目	354
第 9 章 树	359
9.1 树的简介	360
9.2 树的表示方法	364

9.3 二叉树节点类.....	367
9.4 树的遍历.....	383
9.5 二叉搜索树.....	394
本章小结.....	407
自测练习参考答案.....	407
程序设计项目.....	410
第 10 章 树项目	414
10.1 堆.....	415
10.2 B-树.....	419
10.3 Java 对树的支持.....	438
10.4 树、日志和时间复杂度分析.....	443
本章小结.....	447
自测练习参考答案.....	448
程序设计项目.....	450
第 11 章 查找	451
11.1 顺序查找和二分查找.....	452
11.2 开型寻址散列.....	460
11.3 使用 Java 的 Hashtable 类.....	473
11.4 链式散列.....	474
11.5 散列的时间分析.....	477
本章小节.....	479
自测练习参考答案.....	480
程序设计项目.....	482
第 12 章 排序	483
12.1 二次排序算法.....	484
12.2 递归排序算法.....	494
12.3 一个使用堆的 $O(n\log n)$ 算法.....	508
本章小结.....	516
自测练习参考答案.....	516
程序设计项目.....	517
第 13 章 使用扩展类实现软件重用	519
13.1 扩展类.....	519
13.2 模拟生态系统.....	527
13.3 抽象类和 game 类.....	545
本章小结.....	556
进阶阅读.....	556
自测练习参考答案.....	556
程序设计项目.....	559

第 14 章 图.....	561
14.1 图的定义.....	561
14.2 图的实现.....	567
14.3 图的遍历.....	578
14.4 路径算法.....	586
本章小结.....	595
自测练习参考答案.....	596
程序设计项目.....	597
附录 A Java 的基本类型和算术溢出.....	598
char 类型的专有特性.....	599
附录 B Java 的输入和输出.....	601
EasyReader 类.....	601
FormatWriter 类.....	601
附录 C 抛出和捕获 Java 异常.....	618
如何抛出异常.....	618
RuntimeException 和 Error 类.....	619
捕获异常.....	620
throws 语句.....	621
进阶信息.....	621
附录 D ArrayList、Vector、Hashtable 和 HashMap 类.....	622
java.util.Vector 和 java.util.ArrayList 类.....	622
java.util.Hashtable 和 java.util.HashMap 类.....	624
附录 E 用于链表中的节点的类.....	626
附录 F 一个用于包对象的类.....	634
附录 G 深入 BIG-O 表示法.....	640
BIG-O 的常规定义.....	640
BIG-O 表达式表示什么.....	641
附录 H Javadoc.....	642
怎样使用 Javadoc 把你的作品提供给其他程序员.....	643
怎样编写 Javadoc 文档注释.....	643
描述整个类的 Javadoc 文档.....	644
用于单独的公共方法的 Javadoc.....	644
控制 html 链接和字体.....	645
运行 Javadoc.....	646
附录 I 用于交互式测试的 Applet.....	650
简单交互式 applet 的 6 个部分.....	652
怎样编译和运行 applet.....	659
超出 init 方法.....	660