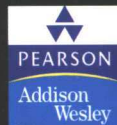




国外经典教材·计算机科学与技术



Algorithms in Java

Part 5 Graph Algorithms 3e

Java算法 (第3版, 第2卷)

——图算法

Robert Sedgewick 著
傅为 译



清华大学出版社

国外经典教材·计算机科学与技术

Java 算法（第 3 版，第 2 卷）

——图算法

（美） Robert Sedgewick 著
傅为 译

清华大学出版社

北京

内 容 简 介

本书深入介绍了图算法。书中分别对图属性和类型、图搜索、有向图、最小生成树、最短路径以及网络流的有关内容进行了透彻的讨论。书中不仅对基本内容做了全面的阐述，而且对经典算法也提供了详尽的分析，同时还涵盖了有关的高级主题。全书既强调了与实用有关的内容，在分析和理论研究上也很有深度。另外，对于书中提供的算法，读者可以放心实现和调试，并用这些算法来解决问题。

本书内容全面、论述清晰，适合于计算机科学和数学领域各个层次的人员使用。

Simplified Chinese edition copyright © 2004 by **PEARSON EDUCATION ASIA LIMITED and TSINGHUA UNIVERSITY PRESS.**

Original English language title from Proprietor's edition of the Work.

Original English language title: **Algorithms in Java, Part 5: Graph Algorithms, 3e** by Robert Sedgewick, Copyright © 2004

EISBN: 0-201-36121-3

All Rights Reserved.

Published by arrangement with the original publisher, Pearson Education, Inc., publishing as Pearson Education, Inc..

This edition is authorized for sale only in the People's Republic of China (excluding the Special Administrative Region of Hong Kong and Macao).

本书中文简体翻译版由 Pearson Education 授权给清华大学出版社在中国境内(不包括中国香港、澳门特别行政区)出版发行。

北京市版权局著作权合同登记号 图字: 01-2003-5385

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。

图书在版编目 (CIP) 数据

Java 算法——图算法. 第 2 卷: 第 3 版 / (美) 塞奇威克 (Sedgewick, R.) 著; 傅为译. —北京: 清华大学出版社, 2004

(国外经典教材·计算机科学与技术)

书名原文: **Algorithms in Java, Part 5: Graph Algorithms, 3e**

ISBN 7-302-08654-0

I. J... II. ①塞... ②傅... III. Java 语言—计算方法—教材 IV. TP312

中国版本图书馆 CIP 数据核字 (2004) 第 046167 号

出版者: 清华大学出版社

<http://www.tup.com.cn>

社总机: 010-62770175

地 址: 北京清华大学学研大厦

邮 编: 100084

客户服务: 010-62776969

文稿编辑: 车立红

封面设计: 久久度文化

印刷者: 北京市通州大中印刷厂

装订者: 北京市密云县京文制本装订厂

发行者: 新华书店总店北京发行所

开 本: 185×260 印张: 25.25 字数: 600 千字

版 次: 2004 年 6 月第 1 版 2004 年 6 月第 1 次印刷

书 号: ISBN 7-302-08654-0/TP·6207

印 数: 1~4000

定 价: 45.00 元

本书如存在文字不清、漏印以及缺页、倒页、脱页等印装质量问题, 请与清华大学出版社出版部联系调换。联系电话: (010)62770175-3103 或 (010)62795704。

出版说明

近年来，我国的高等教育特别是计算机学科教育，进行了一系列大的调整和改革，急需一批门类齐全、具有国际先进水平的计算机经典教材，以适应当前我国计算机科学的教學需要。通过使用国外先进的经典教材，可以了解并吸收国际先进的教学思想和教学方法，使我国的计算机科学教育能够跟上国际计算机教育发展的步伐，从而培育出更多具有国际水准的计算机专业人才，增强我国计算机产业的核心竞争力。为此，我们从国外知名的出版集团 Pearson 引进这套“国外经典教材·计算机科学与技术”教材。

作为全球最大的图书出版机构，Pearson 在高等教育领域有着不凡的表现，其下属的 Prentice Hall 和 Addison Wesley 出版社是全球计算机高等教育的龙头出版机构。清华大学出版社与 Pearson 出版集团长期保持着紧密友好的合作关系，这次引进的“国外经典教材·计算机科学与技术”教材大部分出自 Prentice Hall 和 Addison Wesley 两家出版社。为了组织该套教材的出版，我们在国内聘请了一批知名的专家和教授，成立了一个专门的教材编审委员会。

教材编审委员会的运作从教材的选题阶段即开始启动，各位委员根据国内外高等院校计算机科学及相关专业的现有课程体系，并结合各个专业的培养方向，从 Pearson 出版的计算机系列教材中精心挑选针对性强的题材，以保证该套教材的优秀性和领先性，避免出现“低质重复引进”或“高质消化不良”的现象。

为了保证出版质量，我们为这套教材配备了一批经验丰富的编辑、排版、校对人员，制定了更加严格的出版流程。本套教材的译者，全部来自于对应专业的高校教师或拥有相关经验的 IT 专家。每本教材的责编在翻译伊始，就定期不间断地与该书的译者进行交流与反馈。为了尽可能地保留与发扬教材原著的精华，在经过翻译、排版和传统的三审三校之后，我们还请编审委员或相关的专家教授对文稿进行审读，以最大程度地弥补和修正在前面一系列加工过程中对教材造成的误差和瑕疵。

由于时间紧迫和受全体制作人员自身能力所限，该套教材在出版过程中很可能还存在一些遗憾，欢迎广大师生来电来信批评指正。同时，也欢迎读者朋友积极向我们推荐各类优秀的国外计算机教材，共同为我国高等院校计算机教育事业贡献力量。

清华大学出版社

2004.03.20

国外经典教材·计算机科学与技术

编 审 委 员 会

主任委员：

孙家广 清华大学教授

副主任委员：

周立柱 清华大学教授

委员（按姓氏笔画排序）：

| | |
|-----|---------------|
| 王成山 | 天津大学教授 |
| 王 珊 | 中国人民大学教授 |
| 冯少荣 | 厦门大学教授 |
| 冯全源 | 西南交通大学教授 |
| 刘乐善 | 华中科技大学教授 |
| 刘腾红 | 中南财经政法大学教授 |
| 吉根林 | 南京师范大学教授 |
| 孙吉贵 | 吉林大学教授 |
| 阮秋琦 | 北京交通大学教授 |
| 何 晨 | 上海交通大学教授 |
| 吴百锋 | 复旦大学教授 |
| 李 彤 | 云南大学教授 |
| 沈钧毅 | 西安交通大学教授 |
| 邵志清 | 华东理工大学教授 |
| 陈 纯 | 浙江大学教授 |
| 陈 钟 | 北京大学教授 |
| 陈道蓄 | 南京大学教授 |
| 周伯生 | 北京航空航天大学教授 |
| 孟祥旭 | 山东大学教授 |
| 姚淑珍 | 北京航空航天大学教授 |
| 徐佩霞 | 中国科学技术大学教授 |
| 徐晓飞 | 哈尔滨工业大学教授 |
| 秦小麟 | 南京航空航天大学教授 |
| 钱培德 | 苏州大学教授 |
| 曹元大 | 北京理工大学教授 |
| 龚声蓉 | 苏州大学教授 |
| 谢希仁 | 中国人民解放军理工大学教授 |

前 言

图和图算法在当今的计算应用中颇为常见。对于在实际中出现的图处理问题，本书描述了一些已知的最重要的解决方法。由于需要相关知识的人日渐增多，这本书的主要目的就是让他们了解这些方法及其所蕴藏的基本原则。全书由最基本的原则展开，并从基本概念开始介绍，逐步过渡到经典方法，最后对仍在开发中的最新技术加以讨论。在对算法和应用的描述中，我们提供了精心挑选的示例、详尽的图表以及完备的补充说明。

算 法

为研究当前所使用的最为重要的计算机算法，计划共出版 3 卷，本书是其中的第 2 卷。第 1 卷（第 I ~ IV 部分）所涵盖的是基础知识（第 I 部分）、数据结构（第 II 部分）、排序算法（第 III 部分）以及查找算法（第 IV 部分）；这一卷（第 V 部分）则讨论图与图算法；而未出版的第 3 卷（第 VI ~ VIII 部分）将介绍串（第 VI 部分）、计算几何（第 VII 部分）以及高级算法和应用（第 VIII 部分）。

在学习计算机科学课程之初，即学生已经掌握了基本的编程技巧，熟悉计算机系统，但是尚未选修计算机科学或计算机应用高级领域中的专业课程时，将这些书作为教材是很有用的。这些书也可用于自学，对从事计算机系统或应用程序开发的人来说，将这些书用作参考书也是相当有用的，书中包含了实用算法的实现，并对这些算法的性能特性提供了详尽的信息。该系列图书覆盖面非常之广，因此适于作为这一领域的入门读物。

多年以来，《Java 算法》一书已由世界各地的学生和程序员广泛使用，而以上这 3 卷书加在一起则构成了这本书的第 3 版。在这一版本中，我完全重写了有关内容，并且增加了数千个新练习、数百个新图表以及数十个新程序，而且对所有的图表和程序做了详尽的注释说明。在此不仅涵盖了新的主题，而且还对许多经典算法提供了更为充分的解释。全书强调了抽象数据类型，从而使得有关程序的应用面更广，而且与当今的面向对象编程环境也更为相关。对于已经阅读过本书以前版本的人来说，会从这一版中发现相当多的新内容；而对于所有读者而言，都能从中得到极为丰富的学习资料，可以更好地理解基本概念。

这套书不仅适合程序员和计算机科学专业的学生阅读。每一个使用计算机的人都希望它能运行得更快，或者可以解决更大规模的问题。我们所考虑的算法代表了近 50 年发展起来的知识体系，该体系是在各种各样的应用中有效地使用计算机的基础。从物理学中的多体仿真问题到分子生物学中的基因序列问题，在此所描述的基本方法在科学研究中已日显重要；另外，对于从数据库系统到 Internet 搜索引擎等当今的软件系统，这些基本方法

也已经成为其基本的组成部分。随着计算机应用的覆盖面越来越广，基本算法的影响也日益显著，特别是本书所介绍的基本图算法，作用更为突出。广大学生以及专业人士可能会参与完成各种计算机应用，随着这些应用中相关需求的增长，本书的目标就是要提供一个有效的资源，从而使他们充分了解并明智地使用图算法。

本书范围

《Java 算法》（第 3 版）的“第 V 部分：图算法篇”共包括 6 章，分别介绍图的属性和类型、图搜索、有向图、最小生成树、最短路径以及网。其目的是为了使得读者能够了解尽可能多的基本图算法，并对其基本属性有所理解。

如果你曾经学过有关算法设计和分析基本原则的课程，并且有利用诸如 Java, C++ 或 C 等高级语言编程的经验，那么对于在此介绍的内容，就会充分领略到它的价值。当然，《Java 算法》（第 3 版）的第 I ~ IV 部分已经为此做了充分的准备。本书假设你已经对数组、链表以及 ADT（Abstract Data Type，抽象数据类型）设计等有基本的了解，而且使用过优先队列、符号表以及并查 ADT，所有这些在第 I ~ IV 部分中都有详细的描述（而且在另外一些有关算法和数据结构的介绍性文字中也有说明）。

图和图算法的基本属性由最基本的原则建立，但要充分理解，则往往需要拥有博大精深的数学背景。尽管在此对高级数学概念的讨论很简短，而且是概括性和描述性的，但与第 I ~ IV 部分所介绍的内容相同，要想对图算法有更深入的认识，自然应该有更高的数学水平。不过，数学水平各不相同的读者都可从此书中获益。这种说法可做如下考虑：相对于并非任何人都能理解的一些高级算法，每个人都应该理解并使用的基本图算法只是略有差异。在此的主要意图是结合贯穿于全书的其他方法来讨论重要的算法，而不是对所有数学知识做全面的介绍。不过，好的数学基础往往要求严格的行事方式，而这通常可使我们得到好的程序，因此我尽量在理论家所崇尚的形式规范性和实践家所需要的内容丰富性之间进行权衡，同时也不损害严格性。

教学使用

在本书的讲授方式上有很大的灵活性，这取决于教师的偏好，同时也依赖于学生所做的准备。可把本书用作面向初学者的数据结构课程，因为它阐述了足够的基本内容；也可把本书用作面向高水平学生的算法分析与设计课程，因为它不仅足够详细，而且涵盖了高级内容。有些教师可能希望强调与实现和实用有关的内容，而另外一些教师则可能希望把重点放在分析和理论概念上。

可将本书与第 I ~ IV 部分结合起来，作为一门更为全面的课程讲授。这样，教师就可以完全用一种一致的风格来介绍基础知识、数据结构、排序、查找和图算法等全部内容。

书中的练习（几乎全都是在这一版中新增加的）可分为多种类型。有一些是为了检查

对正文中内容的理解，只要求读者完成某个示例，或者应用正文中所描述的概念。另外一些则涉及实现和整理算法，或者进行实验研究，从而对不同算法加以比较以了解其属性。还有一些练习则相当于知识储备，是对一些重要信息所做的相当详细的说明，而这些信息本身不适于放在正文里。阅读这些练习并加以思考，会使每个读者都有意想不到的收获。

实用算法

任何人若希望更为有效地使用计算机，都可以将这本书作为参考，或用于自学。有编程经验的人可以从书中找到有关一些特定主题的信息。一般地，你可以抽取书中的各章独立地阅读。不过，有些情况下，某一章中的算法可能会用到前一章中所介绍的方法。

本书的定位是对很可能会在实际中使用的算法加以研究。本书对所讨论的工具（即算法）提供了详尽的信息，读者可以放心地实现和调试，并用这些算法来解决问题，或在应用中利用它们来提供有关功能。在此对所讨论的方法提供了完整的实现，同时，针对书中一系列一致的示例程序的操作做了描述。

由于我们采用了实际代码，而不是编写伪代码，因此这些程序很快就可以在实际中使用。通过访问本书的主页可以得到程序的代码清单。您可以用许多方法使用这些工作程序，从而帮助你研究算法。阅读它们以检查你对算法细节的了解，或用一种方法来处理实例化、边界条件和在编程中可能遇到的其他情况。运行这些程序，看看算法在实际中的表现，以根据经验研究性能，并根据书中提供的表检查结果，或试一下你自己所做的修改。

实际上，由算法的一个实际应用已经得到了本书中的数百个图表。许多算法正是通过这些图表所提供的视觉维度直观地发现和得到的。

本书将详细讨论这些算法的特性以及它们可能在哪些情况下是有用的。在此可建立算法分析与理论计算机科学之间的联系。在适当的情况下，都将给出经验性的结果以及分析结果，以说明为什么某些算法更为适用。如果有意义，还会对所讨论的实际算法与纯理论结果之间的关系加以描述。对于算法和实现的性能特性的特定信息，全书将对其进行综合性和概要性的讨论。

编程语言

书中所有实现所用的编程语言均为 Java。程序中使用了大量的标准 Java 习惯用法，而且对于每个构造，正文中都做了简洁的描述。

Mike Schidlowsky 和本人基于 ADT 建立了一种 Java 编程的风格，并认为这是一个将算法和数据结构表示为实际程序的有效方法。我们在实现的优雅性、简洁性、有效性和可移植性方面做了很大的努力。程序风格会尽可能保持一致，因此类似的程序看上去也是相似的。

本书的目标是以尽可能简单明了的方式来展示算法。对于许多算法而言，尽管所用的

语言不同，但存在着相似性。作为一个突出的例子，Dijkstra 算法就是 Dijkstra 算法，无论采用 Algol-60, Basic, Fortran, Smalltalk, Ada, Pascal, C, C++, Modula-3, PostScript, Java, Python, 还是任何一种其他的编程语言（这样的语言可谓不计其数）来编写，也不管所在的是何种环境，均可以证实为有效的图处理方法。一方面，采用这些语言（以及其他多种语言）来实现算法会获得一些经验（本书的 C 和 C++ 版本已经面世），代码会受到这些经验的影响；另一方面，对于这其中的一些语言，其属性会受其设计人员的经验所左右，而这些经验又来自于他对本书所讨论的部分算法和数据结构的使用。最后，我们认为本书所提供的代码不仅准确地定义了算法，而且在实际工作中也相当有用。

致 谢

对于本书以前的版本，很多人都提供了有益的反馈。特别要指出的是，普林斯顿大学和布朗大学的数千名学生多年以来都在阅读本书粗糙的初稿。特别要感谢 Trina Avery 和 Tom Freeman 对于第一版的出版所提供的帮助；还要感谢 Janet Incerpi，是她的创造性和聪明才智使我们有了最初的（同时也是原始的）数字计算机化排版硬件和软件，由此才产生了第一版；感谢 Marc Brown，他参与了算法可视化研究，而本书中很多的图表正来源于此；感谢 Dave Hanson 和 Andrew Appel，对于我有关编程语言的问题，他们总是乐于作答；感谢 Kevin Wayne，他耐心地回答了我所提出的关于网的基本问题。Kevin 力劝我在本书中加入网的单纯形算法（simplex algorithm），而我直到看到 Dagstuhl 的 Ulrich Lauther 所做的介绍，才采纳了他的建议，认为这样做是可能的，本书第 22 章的实现所基于的正是 Ulrich Lauther 的思想。我还要感谢许多读者，他们为各个版本提供了详尽的评论，这其中包括：Guy Almes, Jon Bentley, Marc Brown, Jay Gischer, Allan Heydon, Kennedy Lemke, Udi Manber, Michael Quinn, Dana Richards, John Reif, M. Rosenfeld, Stephen Seidman 和 William Ward。

为了完成此第 3 版，我很荣幸地与 Addison-Wesley 的 Peter Gordon 和 Helen Goldstein 共事，他们耐心地指导了整个项目的进行。我也很有幸能与 Addison-Wesley 的许多专业人员一起工作。这个项目的性质使得这本书对于其中许多人来说都是不同寻常的挑战，对于他们的忍耐力我致以诚挚的谢意。

在写这本书的过程中，我得到了三位良师益友，在此要特别向他们表示感谢。首先，Steve Summit 从技术角度对各版本的初稿都做了认真仔细的检查，并向我提出了数千条详细的意见，特别是在程序方面。Steve 很清楚我的目标是要提供优雅、高效而有用的实现，他的意见不仅有助于我对实现的一致性加以度量，而且还帮助我对其中一部分做了重大改进。其次，Lyn Dupré 也对初稿提出了数千条的详细意见，对我来说它们真是无价之宝，它们不仅帮助我纠正和避免了许多语法错误，而且更重要的是，使我发现了一种一致而连贯的编写风格，由此才使我能够将如此繁多的技术资料合理地整理在一起。第三，Chris Van Wyk 和我通过电子邮件进行了很多交流，这使我备受鼓舞。他耐心地告诉我遵守面向对象编程的基本规则的重要性，并帮助我养成了一种编码的风格，这种风格展示了清晰准确的

算法，并利用了面向对象编程语言的优势。在本书的 C++ 版本中所开发的方法极大地影响了此处的 Java 代码，并且肯定会对用这两种语言（以及 C）编写的待出版的书籍产生影响。能够有机会向 Steve, Lyn 和 Chris 学习，我确实无比感激，他们的建议对于本书的完成至关重要。

我在这里所写的许多内容都受益于 Don Knuth 的授课和著作，他是我在斯坦福大学的导师。虽然 Don 对于这本书没有直接的影响，但在本书中仍然可以感受到他的存在，因为正是他为算法研究奠定了坚实的科学基础，从而使得像本书这样的工作才有可能完成。我的朋友兼同事 Philippe Flajolet 对于这本书也起到了类似的影响，算法分析得以发展为一个成熟的研究领域，他在其中也堪称主力。

非常感谢普林斯顿大学、布朗大学以及法国国立计算机与自动化研究所（Institut National de Recherche en Informatique et Automatique, INRIA）提供的支持，在这些地方我完成了本书的大部分工作；还要感谢美国国防部防御分析研究所（Institute for Defense Analyses）以及施乐的帕洛阿尔托研究中心（Xerox Palo Alto Research Center），在对它们进行访问期间，我也做了本书的一些工作。这本书的某些部分离不开国家自然科学基金（National Science Foundation）和海军研究中心（Office of Naval Research）的慷慨支持。最后，还要感谢 Bill Bowen, Aaron Lemonick 和 Neil Rudenstine，他们为普林斯顿大学建立一个良好的学术环境做了许多工作，正是在此环境下，我才能够承担众多其他事务的同时完成本书的编写。

Robert Sedgewick

Marly-le-Roi, 法国, 1983

普林斯顿, 新泽西州, 1990, 1992

詹姆斯镇, 罗得岛, 1997, 2001

普林斯顿, 新泽西州, 1998, 2003

Java 顾问所撰写的前言

在过去的十几年中，Java 已经成为各种应用程序所选择的语言。但是，对于常见编程问题的解决方案，Java 开发人员发现他们在重复参考如 Sedgewick 的 C 算法这样的参考书。很长时间以来，我们都需要一本关于 Java 的类似的参考书，这一系列书满足了这一要求。

我们把示例程序编写为在各种环境中使用的实用方法。为了达到此目的，我们不使用 Java 软件包机制。为了把重点放在手边的算法上（并且为了揭示许多基本的库类的算法基础），我们没有使用标准的 Java 库，以支持更基本的类型。适当的错误检查和其他防范实践都将极大地提高代码的数量，并把读者的注意力从核心算法上转移开。当在较大的应用中使用程序时，开发人员应该引入此类代码。

尽管我们所提供的算法是独立于语言的，我们也特别关注了 Java 特定的性能问题。因

为 Java 环境在不断发展，因此程序的执行速度将会同本地编译的代码一样快，但此类优化不会改变算法相对的性能。我们把时间作为此类比较的一个有用的参考。

感谢 Mike Zamansky，感谢他所提供的指导以及为计算机科学教学所做的贡献；感谢 Daniel Chaskes，Jason Sanders 和 James Percy，感谢他们所给予的坚定的支持；感谢我的家人。非常高兴能够用计算机科学的经典算法讲述 Java。感谢 Bob 为我提供了编写本书的机会。

Michael Schidlowsky
Oakland Gardens, 纽约, 2003

关于练习的说明

对练习进行分类是一件相当困难的工作，因为对读者来说，其知识水平和经验往往各不相同。不过，提供一些指导还是有益的，因此许多练习带有如下 4 种标记中的一种，以此帮助你确定如何对其进行处理。

用于检查对内容的理解的练习标记了一个空心三角形，如下所示：

△ 18.34 请考虑图

3-7 1-4 7-8 0-5 5-2 3-8 2-9 0-6 4-9 2-6 6-4。

画出其 DFS 树，并根据此树找到图的桥及边连通分量。

更常见的情况是，这种练习直接与正文中的例子相关。它们的难度不大，但完成这些练习可能会使你了解到某个事实或掌握某个概念，而在阅读正文时可能理解不了它们。

用于为内容增加新的信息或需要思考的信息的练习，则标记了一个空心圆，如下所示：

○ 19.106 请编写一个程序，对给定 DAG 可能的拓扑排序结果进行计数。

这种练习可促使你思考与正文内容相关的重要概念，或者解答在阅读正文时可能会出现的问题。即便是没有时间来完成，你也会发现看一下这些练习也是大有裨益的。

有一定难度的练习则标记了一个黑点，如下所示：

● 20.73 如果一个图过大，在内存中一次只能存放其中的 V 条边，请描述如何找到这种图的最小生成树 (MST)。

这种练习可能需要花费相当多的时间才能完成，这取决于你的经验。一般情况下，效率最高的方法是分多次来完成的。

有些练习相当难（相对于大多数其他练习而言），这些练习将标记两个黑点，如下所

示：

- 20.37 请设计一个合理的生成器，用以生成有 V 个顶点和 E 条边的随机图，从而对于 Dijkstra 算法，使其基于堆的 PFS 实现的运行时间是超线性的。

这些练习类似于在研究文献中所解决的问题，不过本书中的内容可以使你做好准备，并愿意尝试解决这些问题（而且很可能会成功）。

这些标记力图做到与你的编程能力和数学能力无关。对于需要在编程或数学分析方面有一定基础的练习，则是不言而喻的。建议所有读者都通过实现算法来检验自己对这些算法的理解。对于一个实际工作的程序员或正在上编程课的学生来说，完成以下的练习可能很简单，但对于近期末编程的人来说则可能需要做大量的工作：

- 17.74 请编写一个程序，在平面上生成 V 个随机点，再建立一个网，其中用边（双向）将给定距离 d 内的各对顶点连接起来（请参见程序 3.20），将每条边的权值置为它所连接的两个顶点之间的距离。请确定如何设置 d 从而使边的数目为 E 。

同样的道理，对于算法属性的知识，建议所有读者都要对其分析基础进行研读。如下的练习对于一位科学家或正在上离散数学课程的学生来说可能很简单，但对于最近未从事数学分析的人来讲，则需要做大量的工作：

- 19.5 对于有 V 个顶点及 E 条边的无向图，有多少个与之对应的有向图？

这里的练习确实太多，你不可能全部阅读并消化它们；我所希望的是，通过做足够的练习，可以促使你对感兴趣的主题有更宽泛的理解，而不仅仅是掌握通过阅读正文所得到的知识。

谨献给 Adam, Andrew, Brett, Robbie

特别要献给 Linda

目 录

| | |
|-------------------------------|------------|
| 第 17 章 图的属性和类型 | 1 |
| 17.1 术语 | 3 |
| 17.2 图的 ADT | 11 |
| 17.3 邻接矩阵表示 | 17 |
| 17.4 邻接表表示 | 22 |
| 17.5 变化、扩展和开销 | 26 |
| 17.6 图生成器 | 33 |
| 17.7 简单路径、欧拉路和哈密顿路径 | 43 |
| 17.8 图处理问题 | 55 |
| 第 18 章 图搜索 | 63 |
| 18.1 探索迷宫 | 63 |
| 18.2 深度优先搜索 | 68 |
| 18.3 图搜索 ADT 方法 | 72 |
| 18.4 DFS 森林的属性 | 76 |
| 18.5 DFS 算法 | 82 |
| 18.6 可分离性和重连通性 | 88 |
| 18.7 广度优先搜索 | 95 |
| 18.8 广义图搜索 | 103 |
| 18.9 图算法分析 | 110 |
| 第 19 章 有向图和无环有向图 | 117 |
| 19.1 术语和游戏规则 | 119 |
| 19.2 有向图中的 DFS 剖析 | 128 |
| 19.3 可达性和传递闭包 | 136 |
| 19.4 等价关系和偏序 | 146 |
| 19.5 无环有向图 | 148 |
| 19.6 拓扑排序 | 153 |
| 19.7 DAG 中的可达性 | 162 |
| 19.8 有向图中的强分量 | 165 |
| 19.9 再述传递闭包 | 173 |
| 19.10 展望 | 177 |

| | |
|----------------------------|------------|
| 第 20 章 最小生成树 | 181 |
| 20.1 表示 | 183 |
| 20.2 MST 算法的基本原理 | 191 |
| 20.3 Prim 算法和优先级优先搜索 | 197 |
| 20.4 Kruskal 算法 | 206 |
| 20.5 Boruvka 算法 | 211 |
| 20.6 比较与改进 | 215 |
| 20.7 欧几里得 MST | 221 |
| 第 21 章 最短路径 | 225 |
| 21.1 基本原则 | 231 |
| 21.2 Dijkstra 算法 | 238 |
| 21.3 全源最短路径 | 246 |
| 21.4 无环网中的最短路径 | 253 |
| 21.5 欧几里得网 | 260 |
| 21.6 归约 | 265 |
| 21.7 负权值 | 277 |
| 21.8 展望 | 293 |
| 第 22 章 网络流 | 295 |
| 22.1 流网络 | 300 |
| 22.2 扩充路径最大流算法 | 309 |
| 22.3 预流-压入最大流算法 | 328 |
| 22.4 最大流归约 | 339 |
| 22.5 最小成本流 | 354 |
| 22.6 网络单纯形算法 | 362 |
| 22.7 最小成本流归约 | 377 |
| 22.8 展望 | 384 |

第 17 章 图的属性和类型

现实中许多计算应用都不仅仅只涉及到元素集，还会涉及到元素与元素之间的连接集。由这些连接所表示的关系立即地会引出大量问题：有没有办法沿着这些连接由一个元素达到另一个元素？由一个给定的元素可以到达多少个其他元素？何为由该元素到达另一个元素的最佳路径？

为了对这些情况建立模型，我们使用了一种称为图（graph）的抽象对象。本章将详细分析图的基本属性，这为学习各种有助于回答前面所提出的问题的算法打下了基础。这些算法充分利用了我们在第 I ~ IV 部分所讨论的计算工具。另外它们还可为解决重要应用中存在的问题奠定基础，如果没有好的算法技术，我们甚至对这些应用的解决方案无从考虑。

图论是组合数学中的一个主要分支，数百年来，对此已经做了深入的研究。已经证实图有许多重要而有用的属性，但仍有大量难题未被攻破。尽管有许多内容需要学习，但在本书中我们将只是从有关图的众多知识中抽取出需要理解的部分，并使用大量有用而基本的算法。

与我们所研究的许多其他问题领域类似，对于图的算法研究也相对较新。尽管有几个基本算法很古老，但大多数有趣的算法都是在最近几十年发现的。即使是使用最简单的图算法，也会得到很有用的计算机程序，而我们将要分析的“非凡”算法当属目前已知最优也最有趣的一类。

为了阐明涉及到图处理的应用的多样性，我们首先考虑几个例子，由此对这个领域中的算法展开研究，该领域中的内容相当丰富。

地图 一个计划去旅行的人可能需要回答诸如这样的一些问题，“从普林斯顿到圣何塞，哪条路线花费最少？”。如果一个人更关注时间而不是花费，那么可能需要了解“从普林斯顿到圣何塞，哪一条路线最快？”。为了回答这些问题，我们要对元素（城镇）之间有关连接（旅行路线）的信息进行处理。

超文本 在浏览 Web 时，我们会遇到一些文档，其中包括有对其他文档的引用（链接），通过单击这些链接可以由一个文档转至另一个文档。整个 Web 即为一个图，这里的元素就是文档，而连接即为链接。搜索引擎可帮助我们在 Web 上查找信息，而图处理算法就是搜索引擎的基本组成部分。

电路 电路中包括诸如晶体管、电阻器和电容器等元件，它们通过导线杂乱地连接在一起。我们用计算机来控制生成电路的机器，并检查电路是否完成了所需的功能。我们需要回答一些简单问题，如“这是一条最短路径吗？”，以及诸如“能否将此电路做在芯片上而不出现任何线路交叉”等复杂问题。在这种情况下，对于第一个问题的答案仅取决于连接（导线）的属性，而第二个问题的答案则需要有关导线、导线所连接元素以及芯片的

物理约束等更详细的信息。

调度 制造过程中需要完成许多任务，而且还存在一组约束，其中指定某些任务在另外一些任务完成前不得开始。我们将约束表示为任务（元素）之间的连接，这样就面临着一个经典的调度问题，即如何合理调度任务，从而既满足给定约束，同时又在最少的时间内完成整个过程。

事务 某电话公司维护一个电话呼叫通信的数据库。这里的连接表示电话呼叫。对于其实际互连结构的本质我们很感兴趣，因为我们需要布线并设置交换机从而高效地处理通信。另外还有一个例子，某财政机构要跟踪市场上的购/销订单状况。这里的连接表示两个客户之间的现金转移。了解这种情况下的连接结构的本质可以加强对市场实际情况的了解。

匹配 学生们要在诸如社交俱乐部、大学或医学院等挑剔的机构中应聘职位。这里的元素对应为学生和机构；连接则对应为应聘关系。我们希望找出将感兴趣的学生与现有职位相匹配的方法。

网络 计算机网络由互联的站点组成，这些站点可发送、转发和接收不同类型的消息。我们所关心的不仅仅是知道从每个站点到各个其他站点有可能得到某个消息，而且随着网络的调整，还关注每两个站点之间的互连性的维护。例如，我们可能希望对一个给定网络进行检查，从而确保其中不存在某一部分站点或连接处于太过“要害”的地位，以至于它们一旦出现问题将导致其余站点无法连接。

程序结构 编译器通过构建图来表示大型软件系统中模块间的关系。这里的元素即为组成系统的各个类或模块；连接要么与一个类中的方法调用另一个方法的可能性相关（静态分析），要么与系统正在运行时的实际调用有关（动态分析）。我们需要对此图进行分析，从而确定如何最佳地为程序分配资源以做到最为高效。

从以上例子中可以看出，在许多应用中，图都是最合适的抽象工具，同时在使用图时也可能会遇到很多计算问题。这些问题即为本书的重点。以上应用付诸现实时，所涉及的数据量着实非常大，而高效的算法则会影响到一个解决方案是否真正可行。

第 I 部分已经对图进行了简要介绍，实际上，第 I 部分中首先详细分析的算法（即第 1 章中的并查算法）就是图算法的基本例子。在第 3 章中还用图说明了二维数组和链表的应用，在第 5 章则用图说明了递归程序和基本数据结构的关系。所有链式数据结构都可表示为图，而我们所熟知的一些用于处理树和其他链式结构的算法都是图算法的特例。本章的目的就是为理解图算法提供一个背景环境，从第 I 部分的简单算法到第 18~22 章的复杂算法都包括在图算法范畴之内。

毫无例外，我们很想知道解决一个特定问题时哪种算法最为高效。研究图算法的性能特点具有很大的挑战性，其原因在于：

- 算法的开销不仅取决于元素集的属性，还依赖于连接集的很多属性（以及由连接所包含的图的全局属性）。
- 对于我们可能遇到的各种图类型，要为其建立准确的模型是相当困难的。

对于图算法，我们通常采用最坏情况的性能下限，即使在多数情况下，其实际性能表现得可能并非如此低下。幸运的是，正如我们将要看到的，不少算法都是最优的，其中仅涉及到少量不必要的工作。对于给定规模的所有图，其他算法消耗的资源是相同的。我们

可以准确预测出这些算法在特定情况下是如何完成的。如果无法做出这样准确的预测，那么对于实际应用中可能遇到的不同类型的图，就需要特别注意其属性，而且必须就这些属性对算法的性能可能会有何种影响做出评判。

我们将从图的基本定义和图的属性展开讨论，在此将介绍用于描述图的标准术语。随后，我们定义了基本 ADT（抽象数据类型）接口，图算法的研究即利用此接口展开，另外还介绍了两种最重要的数据结构来表示图，即邻接矩阵表示和邻接表表示，此外还介绍了实现基本 ADT 操作的各种方法。接下来，我们将考虑一些能够生成随机图的客户程序，它们可用于检查算法，并有助于学习图的属性。所有这些内容都为引入图处理算法奠定了基础，对于在图中查找路径，这些算法可以解决与之相关的三个经典问题，这说明了尽管图问题看上去可能差别不大，但其难度可能会大相径庭。本章最后，将回顾书中分析过的最重要的图处理问题，并根据其解决难度的不同放到了不同位置。

17.1 术 语

与图相关的术语相当多。大多数术语都有简明的定义，为了便于参考，我们将在这里集中介绍这些术语。在介绍第 I 部分中的基本算法时，我们已经用过其中一些概念；而另外一些只有在介绍第 18~22 章的相关高级算法时才会涉及到。

定义 17.1 图由一个顶点集和一个边集组成，其中边将一对不同顶点连接在一起（而且至多只能有一条连接了某一对顶点的边）。

在一个有 V 个顶点的图中，用 0 到 $V-1$ 作为各顶点的名字。采用这种命名机制的主要原因是，我们可以利用数组索引快速地访问对应于各顶点的信息。在 17.6 节中，我们将讨论一个程序，该程序使用一个符号表在 V 个任意顶点的名字与从 0 到 $V-1$ 的 V 个整数之间建立起一对一映射。有了这个程序，我们就可以将索引用作顶点名（为标记方便），而不失一般性。有时我们假设顶点集是隐含定义的，即用边集来定义图，并且仅考虑至少在某条边中所包括的顶点。为了避免诸如“有如下边集且包括 10 个顶点的图”这种罗嗦的提法，当顶点个数可以从上下文中很明显地看出来时，我们不会明确地指出顶点个数。在此约定，一个给定图的顶点数总表示为 V ，而边数则表示为 E 。

我们将图的这一定义（在第 5 章曾第一次提到过）作为标准，不过需要说明，在技术上它存在两点简化。首先，在此不允许出现多重边的情况（数学家有时将这种边称为平行边，并将可以包括这种边的图称为多重图）。其次，此定义不允许出现顶点连至顶点自身的边；这种边称为自环。无平行边或自环的图有时称为简单图。

我们在正式定义中使用的是简单图，其原因是这更易于表示其基本属性，而且在许多应用中都不需要平行边和自环。例如，根据给定的顶点数，我们可以确定一个简单图中的边数。

属性 17.1 有 V 个顶点的图至多有 $V(V-1)/2$ 条边。