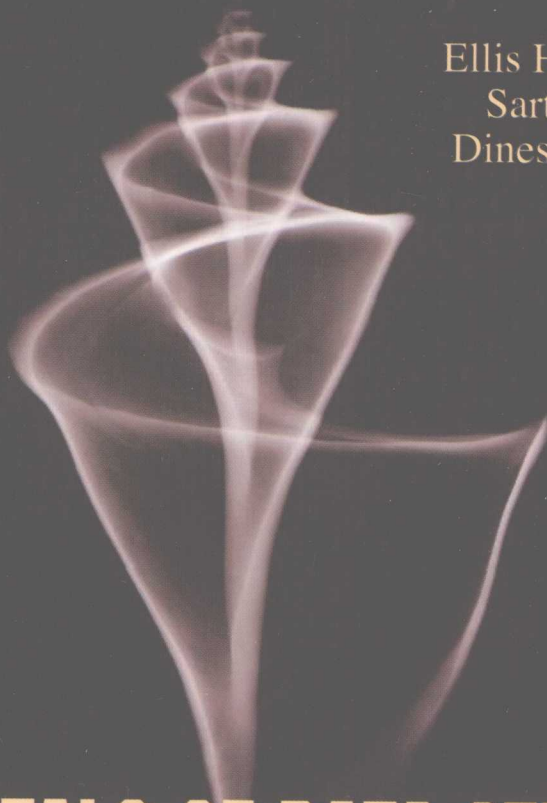


世界著名计算机教材精选

数据结构基础

(C++语言版) (第2版)

Ellis Horowitz
Sartaj Sahni 著
Dinesh Mehta
张 力 等译



FUNDAMENTALS OF DATA STRUCTURES
IN C++

Second Edition



清华大学出版社

世界著名计算机教材精选

Fundamentals of Data Structures in C++
Second Edition

数据结构基础 (C++ 语言版)

(第 2 版)

Ellis Horowitz

Sartaj Sahni

Dinesh Mehta 著

张 力 等 译

清华大学出版社
北 京

Original English language title: Fundamentals of Data Structures in C++, Second Edition by Ellis Horowitz, Sartaj Sahni, and Dinesh Mehta, Copyright © 2009
All Rights Reserved.

This edition has been authorized by Silicon Press for sale in the People's Republic of China only and not for export therefrom.

本书简体中文版由 Silicon Press 授权给清华大学出版社出版发行。

本书封面贴有清华大学出版社防伪标签, 无标签者不得销售。

版权所有, 侵权必究。侵权举报电话: 010-62782989 13701121933

图书在版编目(CIP)数据

数据结构基础(C++ 语言版)(第2版)/(美)霍罗维兹(Horowitz, E.), (美)萨尼(Sahni, S.), (美)梅塔(Mehta, D.)著; 张力等译. —北京: 清华大学出版社, 2009. 3
(世界著名计算机教材精选)

书名原文: Fundamentals of Data Structures in C++, 2E

ISBN 978-7-302-18703-5

I. 数… II. ①霍… ②萨… ③梅… ④张… III. ①数据结构—教材 ②C语言—程序设计—教材 IV. TP311.12 TP312

中国版本图书馆 CIP 数据核字(2008)第 155693 号

责任编辑: 龙啟铭

责任校对: 徐俊伟

责任印制: 何 芊

出版发行: 清华大学出版社

地 址: 北京清华大学学研大厦 A 座

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

邮 编: 100084

社 总 机: 010-62770175

邮 购: 010-62786544

投稿与读者服务: 010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈: 010-62772015, zhiliang@tup.tsinghua.edu.cn

印 刷 者: 北京国马印刷厂

装 订 者: 三河市李旗庄少明装订厂

经 销: 全国新华书店

开 本: 185×260

印 张: 31

字 数: 746 千字

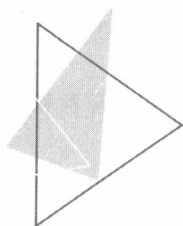
版 次: 2009 年 3 月第 1 版

印 次: 2009 年 3 月第 1 次印刷

印 数: 1~4000

定 价: 49.00 元

本书如存在文字不清、漏印、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: 010-62770177 转 3103 产品编号: 026389-01



译者序

数据结构是计算机科学与技术领域的专业基础课程。它在软件设计和开发中起到举足轻重的作用,几乎所有的计算机软件系统,如操作系统、数据库系统、信息检索系统、编译系统等都要用到数据结构的基本知识。数据结构的基本内容为计算机科学与技术学科的后续专业课程学习提供必要的知识和技能准备。

本书由 Ellis Horowitz, Sartaj Sahni 和 Dinesh P. Mehta 等三人合著。其中, Ellis Horowitz 是南加利福尼亚大学计算机科学系的教授, Sartaj Sahni 是南佛罗里达大学计算机与信息科学工程系的教授。这两位教授早在 1976 年就合作出版了数据结构的经典教材“Fundamentals of Data Structures”,该书在全世界发行超过 10 万册,并被翻译成包括中文在内的多种语言版本。此后,两位教授又在此基础上多次合作,先后出版了面向 Pascal、面向 C 和面向 C++ 的数据结构教材,并被国际上许多著名大学指定为数据结构教科书。本书的另一位作者 Dinesh P. Mehta 是美国科罗拉多矿业大学数学和计算机科学系的教授。曾教授数据结构、高级数据结构和并行计算机等多门计算机专业课程,具有丰富的计算机专业教学经验。本书的英文第一版于 1995 年出版,于 1997 年被翻译成中文,并成为一本经典的数据结构教材。

本书具有内容全面、论述深刻、实践性强等突出的特点。尤其在新版本中,除了对最基本的数据结构,如数组、队列、栈、链表、树、图和哈希表等,有深入全面的阐述外,还对一些较为复杂的高级数据结构,如优先队列、高效二叉查找树、多路查找树和数字查找结构等,也进行了深入浅出的详细讨论。另外,本书采用了当今广为使用的面向对象的 C++ 语言。通过应用 C++ 语言中的一些重要特征,如对象和类、继承、多态性、操作符重载、模板等,对各种数据结构及其算法进行描述,这就使得本书中介绍的许多算法可以直接应用到未来软件开发实践中。

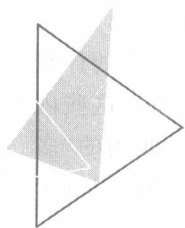
本书适合作为高等院校计算机专业本科生和研究生的数据结构课程教材。本书的内容按照由浅入深、从一般到具体的原则进行组织。第 1 章概略性地阐述了 C++ 语言、算法及其度量等基本概念;第 2 章详细介绍了数组,并讨论了抽象数据类型和 C++ 类;第 3 章全面介绍了栈和队列;第 4 章详述了链表;第 5 章着重讨论树的结构以及对树的各种操作;第 6 章深入探讨了图结构以及图的若干种经典算法;第 7 章介绍了各种经典的排序算法;第 8 章详述了散列,包括静态散列和动态散列;第 9 章详述了优先队列,包括左偏树、二项式堆、配对堆和区间堆等;第 10 章重点讨论高效二叉查找树,包括 AVL 树,红黑树和伸展树;第 11 章阐述多路查找树,包括 B-树和 B+ 树;第 12 章重点介绍了数字查找结构,包括二叉 Trie 树、Patricia 树和后缀树,并讨论了 Trie 树和因特网包转发等问题。

张力全面组织了本书的翻译工作。鲍奋业、徐云龙、张雨辰、张宇、卢亚辉等在本书的翻译和审稿中做了大量的工作。参加翻译的还有张兰、沙文杰和周莉等。

由于译者自身的知识局限及时间的仓促,译稿中难免存在错误和遗漏。谨向读者和原书作者致以歉意,并欢迎指正。

译者

2009 年 1 月 16 日



前 言

自从 1995 年本书的第 1 版发行以来, C++ 语言已经发生了许多明显的变化。基于这些变化, 我们在本书新版本中也进行了多处修改。例如, 将 C++ 的例外(exception)和模板(template)等概念添加到新版本中。也对标准模板库(STL)函数进行了介绍。随着 STL (以及像 Boost 和 LEAD 等其他函数库)的广泛使用, 产生了一些关于应该如何看待数据结构的有趣问题。毫无疑问, 从软件开发的角度看, 用 STL 来实现数据结构比程序员自己实现它要容易得多。根据这种观点, 人们有理由认为, 未来对数据结构的应用会像今天的组合式程序设计一样。也就是说, 在今后, 程序设计中的数据结构问题将交给程序语言和编译器来处理, 它将不再属于主程序程序设计人员需要考虑的技术范围。

然而, 从教学的角度来看, 数据结构显然是计算机科学领域的基本技术, 它不应该被简单化, 也不能为了让其更容易被理解而将一些难懂的内容从教材中删除。因此, 在本书新版本的编写过程中, 我们仍然认为, 让学生了解如何实现和分析高效的数据结构, 并掌握其中的本质内容是非常重要的。基于这种观点, 本书继续强调数据结构的底层实现技术细节, 并对各种数据结构操作算法的复杂性进行精确地分析。

除了增加一些与 C++ 语言有关的内容之外, 在新版本中我们还对数据结构的内容进行了一些扩充和修改。第 2 章(数组)增加了对“数组加倍”技术的讨论, 这种技术的特点是, 当向一个数组插入元素或从中删除元素时, 可以重新改变数组的大小, 而不会因此产生对数组操作复杂性不利的影响。关于这种技术的讨论, 被编排在有关动态数组的章节中。第 3 章(栈和队列)增加了有关队列抽象数据类型的数组实现的一些技术细节。第 4 章(链表)增加了对迭代器改进技术的讨论。在第 5 章(树)和第 6 章(图)中没有增加新的概念。在第 7 章(排序)中, 我们删除了对时间复杂度为 $O(1)$ 的归并方法的讨论。在第 8 章(散列)中, 专门增加了一节讨论“安全哈希函数”, 在动态散列一节中增加了一些新内容(特别将动态散列与数组加倍进行了比较)。我们把“布隆过滤器”的内容也移到了这一章中。第 9 章(优先队列)也有多处被修改。我们增加了关于“重量左偏树”(除了高度左偏树以外)和“配对堆”(作为斐波那契堆的一种替代)的讨论。我们删除了最小-最大堆, 并用更有效的对称“最小-最大堆”和“区间堆”替代。

我们对本书第 1 版中的第 10 章(查找结构)内容进行了重要扩充, 并将这一章的内容分成 3 章: 高效二叉查找树(第 10 章)、多路查找树(第 11 章)和数字查找结构(第 12 章)。在第 1 版中, “红-黑树”是从“234-树”中导出的(而 234-树的讨论是在 23-树之后)。在本版本中, 我们独立地给出了红-黑树定义, 这就使得教师不必先讲解 23-树或 234-树, 再讲解红-黑树(有关 23-树和 234-树的内容已被删除, 因为它们可以作为 B-树的一种特例)。在第 10 章

中还增加了“自顶向下的伸展树”一节(除了前一版的自底向上的伸展树之外)。第 11 章包含了对 B 树的讨论。因为 B+ 树在数据库中的应用,在这一章中我们增加了对它的详细讨论。最后,第 12 章中只是简单地讨论了数字查找树、二叉 Trie 树和 Patrica 树。这些内容在前一版本中是被重点介绍的。在这一版本中,我们选择了重点介绍更广泛使用的“多路 Trie 树”。关于 Trie 树的各种变化和实现也在本书中被进一步探讨。这一章中还包含了一节关于“后缀树”的内容,以及关于 Trie 树在 Internet 包转发中应用的内容。

在 URL 地址 http://www.mines.edu/~dmehta/FDS_CPP/ 中有本书中程序的代码、勘误表和一些从第 1 版中已删除章节的 pdf 文件。

本书的使用

对于准备使用本书作为教材,并在一学期内讲授该课程的教师,建议采用两种授课方案:中级程度和高级程度。对于计算机专业的学生,建议采用中级程度方案。本门课程可能是他们课程计划中的第 2 门或第 3 门课程。对大部分教师,包括本书作者,都可按照中级程度进行教学。以下教学大纲是与 ACM 推荐的课程计划中课程 C2(课程计划 78, CACM 3/79 和 CACM 8/85)相对应的。

教学计划——中级程度

周次	内 容	阅读章节
1	C++ 和算法介绍	第 1 章
2	C++ 和数组	第 2 章
3	数组(串)	完成第 1 次程序设计作业
4	栈和队列	第 3 章
5	链表(单链表和双向链表)	第 4 章
6	链表	完成第 2 次程序设计作业
7	树(基本概念和二叉树)	第 5 章
8	树(查找和堆)	
9	期中考试	
10	图(基本概念和表示)	第 6 章
11	图(最短路径、生成树、拓扑排序)	完成第 3 次程序设计作业
12	内部排序(插入、快速和归并)	第 7 章
13	内部排序(堆和基数)	完成第 4 次程序设计作业
14	散列	第 8 章
15	选择高级主题	第 9 章~第 12 章
16	选择高级主题	第 9 章~第 12 章

建议布置若干次程序设计作业,时间上有所间隔,并贯穿整个学期。第 1 次作业的目的是使学生熟悉计算机环境。第 2 次作业应该重点强调链表结构,相关内容将在第 4 章介绍。

在第4章的练习中,给出了一些程序设计的题目。中等程度的课程进度中不包括外部排序的内容,这样做的目的是留更多的时间讲授更重要的内容,即散列。散列的概念将在以后多门课程中应用,所以,在本门课程中详细讨论是非常重要的。讲授完散列后,建议从第9章~第12中选择一些高级主题进行讲授。

高级程度的教学内容适合于研究生一年级或本科生高年级。推荐的教学进度如下。

教学计划——高级程度

周次	内 容	阅读章节
1	C++ 和算法介绍	第1章
2	C++ 和数组	第2章
3	栈和队列	第3章 完成第1次程序设计作业
4	链表	第4章
5	树	第5章
6	树(继续)	完成第2次程序设计作业
7	期中考试	
8	图	第6章
9	图(继续)	完成第3次程序设计作业
10	内部排序	第7章
11	外部排序	第7章
12	散列	第8章
13	优先队列	第9章 完成第4次程序设计作业
14	高效二叉查找树(可选内容)	第10章
15	多路查找树(可选内容)	第11章
16	数字查找结构(可选内容)	第12章

高级程度教学计划中的程序设计作业和期中考试安排与中级程度完全相同。但总的教学进度更快。对于高级程度教学计划,安排了4周学习第9章~第12章内容,这就使得其他章的教学内容要相应地减少。

最后,将给出高级数据结构的教学课程计划。假定学生已经了解了一些基本概念,特别是关于链表、树和图的概念。在高级数据结构教学计划中,我们计划用4周的时间对基本概念进行深入的回顾,这在时间上应该是足够了。

高级数据结构

周次	内 容	阅读章节
1	复习树	第5章
2	复习图	第6章

续表

周次	内 容	阅读章节
3	内部排序	第 7 章
4	外部排序	
5	散列(复习基本概念,布隆过滤器和动态散列)	第 8 章 完成第 1 次程序设计作业
6	优先队列(左偏树、对称最小-最大堆和区间堆)	第 9 章
7	优先队列(分摊复杂性和二项式堆)	第 9 章
8	优先队列(斐波那契堆和配对堆)	完成第 2 次程序设计作业
9	高效二叉查找树(最优二叉查找树和 AVL 树)	第 10 章
10	期中考试	
11	高效二叉查找树(红-黑树和伸展树)	
12	多路查找树(B-树和 B+ 树)	第 11 章
13	数字查找结构(数字查找树、二叉 Trie 树和 Patricia 树)	第 12 章 完成第 3 次程序设计作业
14	多路 Trie	
15	后缀树	完成第 4 次程序设计作业
16	Trie 和因特网包转发	

对于实行 4 学期制的学校,以下给出了两个可供参考的两学期教学计划。该计划假设学生在先修的高级程序设计课程中对算法分析和数据结构的基本概念已经有所了解。

第 1 学期

周次	内 容	阅读章节
1	算法和数组回顾	第 1 章和第 2 章
2	栈和队列	第 3 章
3	链表(栈、队列和多项式)	第 4 章
4	链表	
5	树(遍历和集合表示)	第 5 章 完成第 1 次程序设计作业
6	树(堆和查找)	期中考试
7	图(遍历和元素)	第 6 章
8	图(最小生成树)	
9	图(最短路径)	完成第 2 次程序设计作业
10	图(活动网络)	

第 2 学期

周次	内 容	阅读章节
1	内部排序(插入、快速、下界和归并排序)	第 7 章
2	排序(堆、基数、链式排序和表排序)	
3	外部排序	第 7 章
4	散列	第 8 章
5	期中考试	完成第 1 次程序设计作业
6	优先队列(左偏树、对称最小-最大堆和区间堆)	第 9 章
7	优先队列(斐波那契堆)	
8	高效二叉查找树(AVL 或红-黑树,伸展树)	第 10 章
9	多路查找树(B-树和 B+树)	完成第 2 次程序设计作业 第 11 章
10	数字查找结构(Tries 和后缀树)	第 12 章

关于第 1 版的说明

为什么要用C++语言来描述数据结构?原因有多种。其中最重要的原因是:授课教师纷纷采用C++语言作为讲授数据结构这门课程的首选语言。这并不会令人感到惊讶,因为C++语言具有清晰地定义抽象数据类型(类)的层次结构的能力,它是一种描述数据结构规范及其操作的理想语言。这种语言具备了面向对象编程方法所必须拥有的所有重要方面,包括信息隐藏、数据抽象和继承。

C++ 编译器及其特点:我们采用ANSI C++标准描述本书中的所有程序。关于这些标准的说明可以查阅 Ellis 和 Stroustrup 编写的 *The Annotated C++ Reference Manual* 一书。本书中给出的所有程序都通过了编译并进行了测试。我们在安装了 DOS/Windows 3.1 操作系统的 Intel/386 计算机上运行 Borland C/C++ 3.1 编译器,并对程序进行了编译。我们也在安装了 SUNOS 4. x 操作系统的 SUN Sparcstation 计算机上通过 GNU C++ 编译器对程序进行了编译。C++ 语言一直在不断进化,像模板和嵌套类这些新特性在几年前是没有的。根据我们的经验,有许多商业C++编译器(特别是旧版本)不能够很好地支持某些新的C++特性,在这种情况下,你可以删除或修改与模板相关的语句。

C++ 指导:本书假设读者熟悉C语言,但不要求一定要掌握C++语言。在前4章中,我们提供了一些C++指导,大致说明这种语言的相关特点。第1.4节描述了C++语言中不同于C且与类和继承无关的一些特性。第2.1节介绍了C++语言的类。第3.1节介绍模板函数和类。第3.4节简单地描述了继承,以及如何使用它将两种数据结构进行关联。第4.11节详细说明了动态类型和虚拟函数。

面向对象的程序设计:不能仅仅将C++语言作为对本书中数据结构进行编程的工具来使用,我们更加强调采用与优异的面向对象程序设计方法相一致的方法来使用C++语言的特性,并且要阐明在这种情况下使用这种语言的原因。这也是我们的职责所在。例如,第4章描述了链式数据结构的设计过程,同时也说明了如何通过使用模板,在一个迭代器类的相

加操作中重用(reusable)它。我们强调这种设计过程,是因为其中用到的概念在任何采用结点和指针的结构中都普遍适用。在整本书中,我们始终强调抽象数据类型和信息隐藏的概念,并将这种概念融入到每一种数据结构的描述之中。对于既不熟悉数据结构,又不掌握面向对象编程的学生,教师要想在讲解数据结构的同时,阐明一种数据结构以及它与其他数据结构之间通过继承建立的关联关系是很困难的。另一方面,对于一本用C++语言描述的数据结构书籍来说,如果它不能够清晰地展示出这种基于继承的面向对象编程模式的威力,那么它也不能够算是完整的。在本书中,我们的做法是:首先仅仅介绍各种数据结构的概念,不讨论怎样将这种数据结构与面向对象编程相结合,以免将问题复杂化。然后,当介绍完若干种类型的数据结构的概念之后,在单独的一个章节中来讨论(或者在练习中探讨)怎样在面向对象的框架中实现这些数据结构。这样做可以将选择权交给授课教师,由教师根据学生熟悉面向对象编程方法的程度来决定是否讨论继承的应用问题。

最后,利用C++语言的动态类型可以定义结点指针,用于指向不同的结点结构。这一特性在Pascal和C语言中是没有的。在第4.12节讨论异构链表时,使用了这种特性。对于那些使用C语言或者Pascal语言讲授数据结构的教师,将会发现本书在算法和计算时间分析方面进行了深入讨论。另外,我们在本书的章节组织方面也尽量使本书的风格与较早出版的相关书籍保持一致。

致谢(第1版)

我们向在本书出版过程中给予我们帮助的所有人致以真诚的谢意。感谢在田纳西大学和田纳西航空学院学习高级数据结构课程的学生们,他们发现了原稿中的一些疏忽之处。我们尤其感谢乔治·布拉斯特(George Blust)和埃捷尤蒂·戴维(Anjanjyoti Das)对本书作出的评价意见。我们也要向凯廷·道斯(Ketan Doshi)(Remedy集团)、德诺克·伊利(Derek Ealy)、麦瑞·路彼兹(Mario Lopez)教授(丹佛大学)、苏艾·塞蒂(Sudhir Shetty)(Arrowsmith 科技有限公司)和露丝·唐(Rose Tsang)(明尼苏达大学)表示感谢,他们对原稿进行了修订,并提出了一些有价值的建议。我们还要感谢英·苏(Ying Shu)(南加里佛尼亚大学)在GNU C++编译器上对程序进行的测试工作。我们特别感谢巴布诺(Barbara)和阿特·富利德门(Art Friedman),他们作为本书最早的出版人,在本书的孕育之初给予了我们诚挚的支持。我们要特别感谢裴尼·霍尔(Penny Hull)副主席,她热情的帮助加快了本书出版的进程。

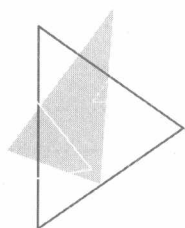
致谢(第2版)

我们感谢纳瑞·哥汗尼(Narain Gehani)博士,他对原稿的改进提出了一些建议。

埃利斯·霍罗维兹(Ellis Horowitz)

萨尔塔·萨尼(Sartaj Sahni)

获尼斯·梅塔(Dinesh Mehta)



目 录

第 1 章 基本概念	1
1.1 概述:系统生命周期	1
1.2 面向对象的程序设计	3
1.2.1 算法分解与面向对象分解	3
1.2.2 基本定义和面向对象编程的概念	3
1.2.3 程序设计语言的演化和C++ 语言历史	4
1.3 数据抽象和封装	4
1.4 C++ 语言基础	8
1.4.1 C++ 程序的组织结构	8
1.4.2 C++ 语言的作用域	9
1.4.3 C++ 的语句与操作符	10
1.4.4 C++ 语言中的数据声明	10
1.4.5 C++ 语言的注释	11
1.4.6 C++ 语言的输入输出	11
1.4.7 C++ 语言的函数	13
1.4.8 C++ 语言的参数传递	13
1.4.9 C++ 语言的函数名重载	14
1.4.10 内联函数	14
1.4.11 C++ 语言的动态内存分配	15
1.4.12 例外	15
1.5 算法规范	17
1.5.1 概述	17
1.5.2 递归算法	20
1.6 标准模板库	24
1.7 性能分析和度量	27
1.7.1 性能分析	28
1.7.2 性能度量	45
1.7.3 测试数据的生成	50
1.8 参考文献和推荐读物	54

第 2 章 数组	55
2.1 抽象数据类型和 C++ 类	55
2.1.1 C++ 类介绍	55
2.1.2 C++ 语言中的数据抽象和封装	56
2.1.3 声明类对象和调用成员函数	56
2.1.4 特别类操作	57
2.1.5 其他主题	60
2.1.6 ADT 和 C++ 类	60
2.2 将数组作为一种抽象数据类型	62
2.3 多项式抽象数据类型	64
2.3.1 多项式的表示	65
2.3.2 多项式的加法	67
2.4 稀疏矩阵	71
2.4.1 绪论	71
2.4.2 稀疏矩阵的表示法	72
2.4.3 转置一个矩阵	73
2.4.4 矩阵的乘法	76
2.5 多维数组的表示	81
2.6 字符串抽象数据类型	84
2.6.1 一种简单的字符串模式匹配算法	85
2.6.2 字符串模式匹配: Knuth-Morris-Pratt 算法	85
2.7 参考文献和推荐读物	89
2.8 附加习题	89
第 3 章 栈和队列	94
3.1 C++ 模板	94
3.1.1 模板函数	94
3.1.2 用模板表示容器类	96
3.2 栈的抽象数据类型	99
3.3 队列抽象数据类型	103
3.4 C++ 中的子类型和继承	109
3.5 一个迷宫问题	112
3.6 计算表达式	116
3.6.1 表达式	116
3.6.2 后缀表达式	118
3.6.3 中缀表达式转换成后缀表达式	119
3.7 附加习题	122
第 4 章 链表	124
4.1 单链表和链	124

4.2	用C++语言表示链表	126
4.2.1	在C++语言中定义结点	126
4.2.2	用C++语言设计一个链表类	127
4.2.3	C++语言中的指针操作	130
4.2.4	链表的控制操作	131
4.3	链的模板类	134
4.3.1	用模板实现链表	134
4.3.2	链表迭代器	135
4.3.3	链表操作	138
4.3.4	类的重用	140
4.4	循环链表	141
4.5	可用空间链表	143
4.6	链式栈和链式队列	144
4.7	多项式	146
4.7.1	多项式的表示	146
4.7.2	多项式相加	147
4.7.3	用循环链表表示多项式	150
4.8	等价类	152
4.9	稀疏矩阵	157
4.9.1	稀疏矩阵的表示	157
4.9.2	稀疏矩阵的输入	159
4.9.3	删除稀疏矩阵	160
4.10	双向链表	162
4.11	广义表	165
4.11.1	广义表的表示	165
4.11.2	表的递归算法	168
4.11.3	引用计数、共享和递归表	171
第5章	树	176
5.1	概述	176
5.1.1	术语表	176
5.1.2	树的表示	178
5.2	二叉树	180
5.2.1	抽象数据类型	180
5.2.2	二叉树的性质	181
5.2.3	二叉树的表示	183
5.3	二叉树的遍历和迭代程序	185
5.3.1	概述	185
5.3.2	中序遍历	185

5.3.3	先序遍历	187
5.3.4	后序遍历	187
5.3.5	迭代中序遍历	188
5.3.6	层次遍历	190
5.3.7	不使用栈的遍历	191
5.4	补充的二叉树操作	193
5.4.1	复制二叉树	193
5.4.2	检测相等	193
5.4.3	满足性问题	194
5.5	线索二叉树	196
5.5.1	线索	196
5.5.2	线索二叉树的中序遍历	197
5.5.3	在线索二叉树上插入一个结点	198
5.6	堆	200
5.6.1	优先队列	200
5.6.2	大顶堆的定义	201
5.6.3	大顶堆的插入	202
5.6.4	大顶堆的删除	203
5.7	二叉查找树	205
5.7.1	定义	205
5.7.2	二叉查找树的查找	206
5.7.3	二叉查找树的插入	208
5.7.4	二叉查找树的删除	209
5.7.5	二叉树的连接和分裂	210
5.7.6	二叉查找树的高度	212
5.8	选择树	213
5.8.1	概述	213
5.8.2	胜者树	213
5.8.3	败者树	214
5.9	森林	215
5.9.1	将森林转换成二叉树	215
5.9.2	森林的遍历	216
5.10	离散集合表示	217
5.10.1	概述	217
5.10.2	并和查找操作	217
5.10.3	等价类的应用	223
5.11	二叉树计数	225
5.11.1	不同的二叉树	225
5.11.2	栈排列	226

5.11.3 矩阵相乘.....	227
5.11.4 不同二叉树的个数.....	228
5.12 参考文献和推荐读物.....	229
第6章 图	230
6.1 图的抽象数据类型	230
6.1.1 概述.....	230
6.1.2 定义.....	231
6.1.3 图的表示.....	234
6.2 图的基本操作	240
6.2.1 深度优先搜索.....	240
6.2.2 广度优先搜索.....	241
6.2.3 连通分量.....	242
6.2.4 生成树.....	243
6.2.5 重连通分量.....	244
6.3 最小代价生成树	248
6.3.1 克鲁斯卡尔算法.....	248
6.3.2 普里姆算法.....	251
6.3.3 索林算法.....	252
6.4 最短路径和传递闭包	253
6.4.1 单源/多目标: 非负权值	253
6.4.2 单源/多目标: 任意权值	257
6.4.3 多源最短路径.....	259
6.4.4 传递闭包.....	260
6.5 活动网络	263
6.5.1 顶点表示活动的网络(AOV 网)	263
6.5.2 用边表示活动的网络(AOE 网)	267
6.5.3 事件最早发生时间的计算.....	269
6.5.4 事件最迟发生时间的计算.....	269
6.6 参考文献和推荐读物	272
6.7 附加习题	273
第7章 排序	275
7.1 目的	275
7.2 插入排序	278
7.3 快速排序	280
7.4 排序算法能够多快	283
7.5 归并排序	284
7.5.1 归并.....	284
7.5.2 迭代归并.....	285

7.5.3 递归归并	287
7.6 堆排序	289
7.7 多关键字排序	292
7.8 链和列表排序	295
7.9 内部排序总结	302
7.10 外部排序	306
7.10.1 概述	306
7.10.2 k 路归并	308
7.10.3 并行操作的缓存处理	309
7.10.4 顺串产生	313
7.10.5 顺串的最佳归并	315
7.11 参考文献和推荐读物	318
第 8 章 散列	319
8.1 绪论	319
8.2 静态散列	319
8.2.1 哈希表	319
8.2.2 哈希函数	320
8.2.3 安全哈希函数	323
8.2.4 溢出处理	325
8.2.5 溢出技术的理论分析	329
8.3 动态散列	331
8.3.1 动态散列的目的	331
8.3.2 使用目录的动态散列	332
8.3.3 不使用目录的动态散列	334
8.4 布隆过滤器	335
8.4.1 勘误文件的应用	335
8.4.2 设计布隆过滤器	337
8.5 参考文献和推荐读物	338
第 9 章 优先队列	340
9.1 单端和双端优先队列	340
9.2 左偏树	342
9.2.1 高度左偏树	342
9.2.2 重量左偏树	346
9.3 二项式堆	349
9.3.1 代价分摊	349
9.3.2 二项式堆的定义	350
9.3.3 二项式堆的插入操作	351
9.3.4 合并两个二项式堆	352