

21世纪高等学校规划教材 | 计算机科学与技术



浙江省“十一五”重点教材建设项目



数据结构

——Java语言描述（第2版）

刘小晶 杜 选 主编
朱 蓉 杜卫锋 副主编



清华大学出版社

21世纪高等学校规划教材 | 计算

数据结构

——Java语言描述（第2版）

刘小晶 杜 选 主编
朱 蓉 杜卫锋 副主编

清华大学出版社

内 容 简 介

本书是浙江省“十一五”规划重点建设教材,内容涵盖了教育部计算机科学与技术教指委关于“高等学校计算机科学与技术本科专业规范”中制定的课程体系中的核心知识,并在紧扣考研大纲的前提下剔除了一些难度较大的内容。

本书采用 Java 语言作为描述算法的语言,共 9 章,可分成两大部分。第一部分主要介绍线性表、栈、队列、串、数组、树和图等基本数据结构的特点、存储方式、运算原理、实现方法以及它们在现实中的典型应用;第二部分主要讨论查找与排序这两种最常用操作的实现原理、方法及性能分析。

全书条理清楚、语言精练、重点突出,叙述循序渐进、深入浅出;表达通俗易懂,特别注重理论与实践相结合;强调算法实现方法的分析,并通过丰富、典型的实例来强化知识的实际应用。

本书可作为普通高等院校计算机科学与技术、软件工程、信息管理与信息系统、信息与计算科学、电子信息等专业的“数据结构”课程的教材,也可作为工程技术和自学数据结构人员的参考读物。

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:Java 语言描述/刘小晶,杜选主编.--2 版.--北京:清华大学出版社,2015

21 世纪高等学校规划教材·计算机科学与技术

ISBN 978-7-302-38944-6

I. ①数… II. ①刘… ②杜… III. ①数据结构—高等学校—教材 ②JAVA 语言—程序设计—高等学校—教材 IV. ①TP311.12 ②TP312

中国版本图书馆 CIP 数据核字(2015)第 005675 号

责任编辑:黄 芝 王冰飞

封面设计:傅瑞学

责任校对:时翠兰

责任印制:刘海龙

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦 A 座 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

课 件 下 载: <http://www.tup.com.cn>, 010-62795954

印 装 者:北京国马印刷厂

经 销:全国新华书店

开 本:185mm×260mm 印 张:25.25 字 数:615 千字

版 次:2011 年 2 月第 1 版 2015 年 4 月第 2 版 印 次:2015 年 4 月第 1 次印刷

印 数:1~2000

定 价:49.00 元

产品编号:061800-01

出版说明

随着我国改革开放的进一步深化,高等教育也得到了快速发展,各地高校紧密结合地方经济建设发展需要,科学运用市场调节机制,加大了使用信息科学等现代科学技术提升、改造传统学科专业的投入力度,通过教育改革合理调整和配置了教育资源,优化了传统学科专业,积极为地方经济建设输送人才,为我国经济社会的快速、健康和可持续发展以及高等教育自身的改革发展做出了巨大贡献。但是,高等教育质量还需要进一步提高以适应经济社会发展的需要,不少高校的专业设置和结构不尽合理,教师队伍整体素质亟待提高,人才培养模式、教学内容和方法需要进一步转变,学生的实践能力和创新精神亟待加强。

教育部一直十分重视高等教育质量工作。2007年1月,教育部下发了《关于实施高等学校本科教学质量与教学改革工程的意见》,计划实施“高等学校本科教学质量与教学改革工程”(简称“质量工程”),通过专业结构调整、课程教材建设、实践教学改革、教学团队建设等多项内容,进一步深化高等学校教学改革,提高人才培养的能力和水平,更好地满足经济社会发展对高素质人才的需要。在贯彻和落实教育部“质量工程”的过程中,各地高校发挥师资力量强、办学经验丰富、教学资源充裕等优势,对其特色专业及特色课程(群)加以规划、整理和总结,更新教学内容、改革课程体系,建设了一大批内容新、体系新、方法新、手段新的特色课程。在此基础上,经教育部相关教学指导委员会专家的指导和建议,清华大学出版社在多个领域精选各高校的特色课程,分别规划出版系列教材,以配合“质量工程”的实施,满足各高校教学质量和教学改革的需要。

为了深入贯彻落实教育部《关于加强高等学校本科教学工作,提高教学质量的若干意见》精神,紧密配合教育部已经启动的“高等学校教学质量与教学改革工程精品课程建设工作”,在有关专家、教授的倡议和有关部门的大力支持下,我们组织并成立了“清华大学出版社教材编审委员会”(以下简称“编委会”),旨在配合教育部制定精品课程教材的出版规划,讨论并实施精品课程教材的编写与出版工作。“编委会”成员皆来自全国各类高等学校教学与科研第一线的骨干教师,其中许多教师为各校相关院、系主管教学的院长或系主任。

按照教育部的要求,“编委会”一致认为,精品课程的建设工作从开始就要坚持高标准、严要求,处于一个比较高的起点上。精品课程教材应该能够反映各高校教学改革与课程建设的需要,要有特色风格、有创新性(新体系、新内容、新手段、新思路,教材的内容体系有较高的科学创新、技术创新和理念创新的含量)、先进性(对原有的学科体系有实质性的改革和发展,顺应并符合21世纪教学发展的规律,代表并引领课程发展的趋势和方向)、示范性(教材所体现的课程体系具有较广泛的辐射性和示范性)和一定的前瞻性。教材由个人申报或各校推荐(通过所在高校的“编委会”成员推荐),经“编委会”认真评审,最后由清华大学出版

社审定出版。

目前,针对计算机类和电子信息类相关专业成立了两个“编委会”,即“清华大学出版社计算机教材编审委员会”和“清华大学出版社电子信息教材编审委员会”。推出的特色精品教材包括:

(1) 21世纪高等学校规划教材·计算机应用——高等学校各类专业,特别是非计算机专业的计算机应用类教材。

(2) 21世纪高等学校规划教材·计算机科学与技术——高等学校计算机相关专业的教材。

(3) 21世纪高等学校规划教材·电子信息——高等学校电子信息相关专业的教材。

(4) 21世纪高等学校规划教材·软件工程——高等学校软件工程相关专业的教材。

(5) 21世纪高等学校规划教材·信息管理与信息系统。

(6) 21世纪高等学校规划教材·财经管理与应用。

(7) 21世纪高等学校规划教材·电子商务。

(8) 21世纪高等学校规划教材·物联网。

清华大学出版社经过三十多年的努力,在教材尤其是计算机和电子信息类专业教材出版方面树立了权威品牌,为我国的高等教育事业做出了重要贡献。清华版教材形成了技术准确、内容严谨的独特风格,这种风格将延续并反映在特色精品教材的建设中。

清华大学出版社教材编审委员会

联系人:魏江江

E-mail: weijj@tup.tsinghua.edu.cn

第2版前言

本书第1版自2011年2月出版以来,已重印4次,深受广大读者的认可和好评,被数十所院校选为课程教材或教学参考书。

在第1版书的使用期间,编者对原书内容进行了多次勘校,对所发现的舛误进行了逐一的纠正。为了便于读者将理论与实践紧密结合,使学习者分析问题和解决问题的能力得到提高,于2013年2月出版了与之配套使用的《数据结构实例解析与实验指导》教辅书,详细地分析和解答了教材中的所有习题,并与考研内容相结合,增加了很多经典的例题,为每一章的知识应用都设计了课题实验题目及分析解答。

改版是在编者广泛征求意见的基础上,根据读者使用时反馈的意见和编者的新认识,以追求更加有利于读者学习为目标的前提下进行的,改版对原书中的内容做了以下4个方面的修改:

(1) 为了使本书的内容全部涵盖最新考研大纲的内容要求,增加了“外部排序”一章。

(2) 为了提高数据的封装性,原书中将类成员变量都定义为私有属性,但为了使算法描述更加接近C/C++的描述风格,增强算法的可读性和减少算法内容的篇幅,特将原书中描述数据结构的单独类中的私有成员变量全部改成了公有属性。

(3) 为了便于读者自学,在本书的最后增加了每章的习题参考答案。

(4) 在“二叉树遍历算法的应用举例”章节中提出了递归模型的概念,并运用于具体算法设计中。

参加第2版各章编写的是刘小晶、朱蓉、杜选、杜卫锋,全书由刘小晶策划和统稿。本书共分9章,内容简洁明了,问题分析细致,语言表述通俗易懂,配套教学资源丰富,适合作为普通高校计算机专业及相关专业的数据结构教材,也可供有兴趣的学习者参考和借鉴。书中所有算法(包括习题)都在Eclipse环境下测试通过,源程序代码和教学课件可在清华大学出版社网站(www.tup.tsinghua.edu.cn)下载。

由于数据结构知识的应用非常广泛,加之编者水平有限,书中定有疏漏和不足之处,敬请批评指正。

作者

2014年12月

数据结构是计算机科学与技术等计算机类专业的核心主干基础课程,也是计算机科学与技术等相关专业考研的必考科目。

数据结构一般开设在大学二年级,其目的是对前期学习的计算机技术进行总结与提高,为后续其他专业课程的学习提供基础。“数据结构”上承“计算机导论”、“程序设计语言”和“离散数学”;下启“算法分析与设计”,同时它又是“操作系统”、“软件工程”、“数据库原理”、“编译原理”、“计算机图形学”和“人工智能”等专业课程的必修先行课。此外,许多计算机应用软件都要运用“数据结构”课程中讲授的知识来编写程序,进行科学计算和模拟试验。

学习数据结构的过程,既是一个复杂程序设计的训练过程,也是一个构造优化算法的训练过程,其技能培养的重要程度与知识传授等同。数据结构的教学内容、重点和难点在于让学生理解和掌握算法的设计与分析,使学生通过该课程的学习具有较好的数据抽象能力、算法设计能力以及创新思维能力。

在计算机类专业的课程设置中,目前许多高校都将“Java 程序设计语言”这门课程当作第一门程序设计语言课程进行开设,这就需要在人才培养方案中能有贯穿于整个过程的对 Java 编程能力进行培养的课程安排。所以,如果数据结构能用 Java 语言进行描述,一则有利于提升数据结构的学习效果,二则能巩固 Java 语言方面的知识。而且相对于面向过程的程序设计语言,用面向对象的程序设计语言描述数据结构更为自然。Java 是相对较新同时也是当前使用极为广泛的面向对象的程序设计语言。与 C++ 相比,Java 是一种更安全、更具可移植性并且更容易使用的语言。因此,这使得它成为目前讨论和实现数据结构的一种优秀的可选语言。

下面是本教材的一些特色:

(1) 精简内容、强化基础、突出知识的应用性。针对一般本科院校学生的实际情况,把握“适用”与“够用”的尺度。做到把重点放在基础知识的介绍上,缩减了一些难度较大的内容。并强调其在实际问题中的应用性,充分体现了理论与应用背景的紧密结合。

(2) 理论叙述简洁明了,重点突出,应用实例丰富完整并做到内容由浅入深,循序渐进。各章节都从基本概念入手,逐步介绍其特点和基本操作的实现,然后通过应用实例来讲述如何运用所学的原理和方法来解决实际问题,最后附有小结和习题,便于学习总结和提高。这些内容做到“环环相扣,层层推进”,充分体现解析法的精髓,达到通俗易懂、由浅入深的效果,培养读者迁移知识的能力。

(3) 算法的描述既严谨,又充分利用 Java 的泛型方法来体现方法的通用性,支持高效的代码重用。

(4) 书中融入了新的教学理念,并配有大量的图表,便于学生直观地理解数据结构与算法。

(5) 教学、实验相互配合,内容全面,实现立体化教学资源建设。与教材一同配有 PPT

电子教案、习题解答,有“数据结构”精品课程网站。与本书配套的《数据结构实例解析与实验指导》预计在2011年完成,书中将结合数据结构课程中的重要知识点提供丰富的典型应用实例,进一步强化理论知识应用于实际问题的解决能力的培养。

本教材主要面向一般本科院校使用,因此,在紧扣考研大纲的前提下尽量剔除了一些难度较大的内容。教材的内容组织共分8章,其中:

第1章绪论,主要介绍了数据结构课程讨论的内容;数据结构的常用术语及基本概念;数据结构算法的描述和算法分析方法以及本书中所应用到的Java泛型方法等内容。

第2章线性表,主要介绍了线性表的抽象数据类型定义;线性表类型在顺序存储和链式存储两种存储结构下的实现方法以及线性表的应用等内容。

第3章栈与队列,主要介绍了栈与队列的抽象数据类型定义;栈与队列在顺序存储和链式存储结构下其基本操作的实现方法以及栈与队列的应用等内容。

第4章串与数组,主要介绍了串的基本概念;串的存储结构;串的基本操作实现;数组的定义、操作和存储结构以及矩阵的压缩存储等内容。

第5章树,主要介绍了树与二叉树的基本概念和存储结构;树、二叉树和森林的遍历;树、二叉树与森林之间的转换方法以及哈夫曼树与哈夫曼编码等内容。

第6章图,主要介绍了图的基本概念;图的邻接矩阵和邻接表两种最基本的存储结构;图的广度优先搜索和深度优先搜索两种最基本的遍历方法;有关最小生成树的克鲁斯卡尔(Kruskal)和普里姆(Prim)两种实现算法;拓扑排序和求关键路径等内容。

第7章排序,主要介绍了排序的基本概念;常用内部排序(插入排序、交换排序、选择排序、归并排序和基数排序)方法的实现及性能分析以及各种内部排序方法的比较等内容。

第8章查找,主要介绍了查找的基本概念;静态表查找(顺序查找、二分查找和分块查找)、动态表查找(二叉排序树、平衡二叉树、B树和红黑树)、哈希表查找的实现方法及性能分析等内容。

第1~3章及第5章由刘小晶副教授执笔,第4、第7~8章由杜选副教授执笔,第6章由杜卫锋博士执笔,书中的代码由邹益民博士负责设计和调试。刘小晶提出了本书的编写大纲及负责完成了本书的统稿工作。由朱蓉博士负责审阅了全稿,并提出了许多宝贵的修改意见。

本书在编写过程中,参阅了大量的参考资料,列于书目的参考文献中,在此谨向其作者表达衷心的感谢。

由于作者学识所限,书中定有不足之处,敬请读者批评指正,提出宝贵意见,联系方式:liuxiaojing99999@163.com。

目录

| | |
|--------------------------|----|
| 第 1 章 绪论 | 1 |
| 1.1 数据结构课程讨论的内容 | 1 |
| 1.1.1 求解问题举例..... | 1 |
| 1.1.2 本课程讨论的内容..... | 4 |
| 1.2 基本概念与术语 | 4 |
| 1.2.1 数据与数据结构..... | 5 |
| 1.2.2 数据类型..... | 9 |
| 1.2.3 抽象数据类型 | 10 |
| 1.3 算法和算法分析..... | 12 |
| 1.3.1 算法的基本概念 | 12 |
| 1.3.2 算法的描述 | 13 |
| 1.3.3 算法分析 | 14 |
| 1.3.4 算法设计举例 | 20 |
| 1.4 Java 提供的泛型方法 | 23 |
| 小结 | 25 |
| 习题 1 | 26 |
| 第 2 章 线性表 | 29 |
| 2.1 线性表及其基本操作..... | 29 |
| 2.1.1 线性表的基本概念 | 29 |
| 2.1.2 线性表的抽象数据类型描述 | 30 |
| 2.2 线性表的顺序存储及其实现..... | 31 |
| 2.2.1 线性表的顺序存储 | 31 |
| 2.2.2 顺序表上基本操作的实现 | 33 |
| 2.2.3 顺序表应用举例 | 36 |
| 2.3 线性表的链式存储及其实现..... | 37 |
| 2.3.1 单链表的表示 | 38 |
| 2.3.2 单链表上基本操作的实现 | 40 |
| 2.3.3 单链表应用举例 | 47 |
| 2.3.4 其他链表 | 50 |
| 2.4 顺序表与链表的比较..... | 55 |
| 2.5 线性表的应用举例..... | 56 |

| | |
|--------------------------------|------------|
| 小结 | 63 |
| 习题 2 | 63 |
| 第 3 章 栈与队列 | 66 |
| 3.1 栈 | 66 |
| 3.1.1 栈的概念 | 66 |
| 3.1.2 栈的抽象数据类型描述 | 67 |
| 3.1.3 顺序栈及其基本操作的实现 | 67 |
| 3.1.4 链栈及其基本操作的实现 | 70 |
| 3.1.5 栈的应用 | 73 |
| 3.2 队列 | 85 |
| 3.2.1 队列的概念 | 85 |
| 3.2.2 队列的抽象数据类型描述 | 86 |
| 3.2.3 顺序队列及其基本操作的实现 | 86 |
| 3.2.4 链队列及其基本操作的实现 | 90 |
| 3.2.5 队列的应用 | 92 |
| 3.2.6 优先级队列 | 94 |
| 3.3 栈与队列的比较 | 98 |
| 3.4 栈与队列的综合应用举例 | 100 |
| 小结 | 103 |
| 习题 3 | 104 |
| 第 4 章 串与数组 | 107 |
| 4.1 串概述 | 107 |
| 4.1.1 串的基本概念 | 107 |
| 4.1.2 串的抽象数据类型描述 | 108 |
| 4.2 串的存储结构 | 109 |
| 4.2.1 串的顺序存储结构 | 109 |
| 4.2.2 串的链式存储结构 | 109 |
| 4.3 顺序串的实现 | 110 |
| 4.3.1 顺序串的定义 | 110 |
| 4.3.2 串的基本操作实现 | 112 |
| 4.4 串的模式匹配操作 | 115 |
| 4.4.1 Brute-Force 模式匹配算法 | 115 |
| 4.4.2 KMP 模式匹配算法 | 117 |
| 4.5 串的应用举例 | 122 |
| 4.6 数组的概念及顺序存储结构 | 128 |
| 4.6.1 数组的基本概念 | 129 |
| 4.6.2 数组的抽象数据类型描述 | 129 |

| | |
|-----------------------------|------------|
| 4.6.3 数组的顺序存储结构····· | 130 |
| 4.7 特殊矩阵的压缩存储····· | 131 |
| 4.7.1 对称矩阵的压缩存储····· | 131 |
| 4.7.2 三角矩阵的压缩存储····· | 132 |
| 4.7.3 对角矩阵的压缩存储····· | 132 |
| 4.8 稀疏矩阵的压缩存储····· | 133 |
| 4.8.1 稀疏矩阵的三元组表存储····· | 133 |
| 4.8.2 稀疏矩阵的十字链表存储····· | 139 |
| 4.9 数组的应用举例····· | 141 |
| 小结····· | 142 |
| 习题 4····· | 143 |
| 第 5 章 树与二叉树····· | 146 |
| 5.1 树的基本概念····· | 146 |
| 5.2 二叉树概述····· | 149 |
| 5.2.1 二叉树的基本概念····· | 149 |
| 5.2.2 二叉树的性质····· | 151 |
| 5.2.3 二叉树的存储结构····· | 153 |
| 5.3 二叉树的遍历····· | 156 |
| 5.3.1 二叉树的遍历方法及其实现····· | 156 |
| 5.3.2 二叉树遍历算法的应用举例····· | 163 |
| 5.3.3 建立二叉树····· | 168 |
| 5.4 哈夫曼树及哈夫曼编码····· | 173 |
| 5.4.1 哈夫曼树的基本概念····· | 173 |
| 5.4.2 哈夫曼树和哈夫曼编码的构造方法····· | 174 |
| 5.4.3 构造哈夫曼树和哈夫曼编码类的描述····· | 177 |
| 5.5 树与森林····· | 179 |
| 5.5.1 树、森林与二叉树之间的转换····· | 179 |
| 5.5.2 树的存储结构····· | 182 |
| 5.5.3 树和森林的遍历····· | 185 |
| 小结····· | 187 |
| 习题 5····· | 188 |
| 第 6 章 图····· | 190 |
| 6.1 图概述····· | 190 |
| 6.1.1 图的基本概念····· | 190 |
| 6.1.2 图的抽象数据类型描述····· | 194 |
| 6.2 图的存储结构····· | 194 |
| 6.2.1 邻接矩阵····· | 195 |

| | | |
|--------------|-----------------|------------|
| 6.2.2 | 邻接表 | 200 |
| 6.3 | 图的遍历 | 205 |
| 6.3.1 | 广度优先搜索 | 206 |
| 6.3.2 | 深度优先搜索 | 208 |
| 6.3.3 | 图的遍历方法的应用举例 | 209 |
| 6.4 | 最小生成树 | 215 |
| 6.4.1 | 最小生成树的基本概念 | 215 |
| 6.4.2 | 克鲁斯卡尔算法 | 216 |
| 6.4.3 | 普里姆算法 | 217 |
| 6.5 | 最短路径 | 220 |
| 6.5.1 | 求某个顶点到其余顶点的最短路径 | 221 |
| 6.5.2 | 求每一对顶点之间的最短路径 | 223 |
| 6.6 | 拓扑排序 | 227 |
| 6.6.1 | 拓扑排序的基本概念 | 228 |
| 6.6.2 | 拓扑排序的实现 | 228 |
| 6.7 | 关键路径 | 230 |
| | 小结 | 234 |
| | 习题 6 | 235 |
| 第 7 章 | 内排序 | 238 |
| 7.1 | 排序的基本概念 | 238 |
| 7.2 | 插入排序 | 241 |
| 7.2.1 | 直接插入排序 | 241 |
| 7.2.2 | 希尔排序 | 244 |
| 7.3 | 交换排序 | 245 |
| 7.3.1 | 冒泡排序 | 245 |
| 7.3.2 | 快速排序 | 247 |
| 7.4 | 选择排序 | 250 |
| 7.4.1 | 直接选择排序 | 250 |
| 7.4.2 | 树形选择排序 | 251 |
| 7.4.3 | 堆排序 | 255 |
| 7.5 | 归并排序 | 260 |
| 7.6 | 基数排序 | 263 |
| 7.6.1 | 多关键字排序 | 263 |
| 7.6.2 | 链式基数排序 | 263 |
| | 小结 | 266 |
| | 习题 7 | 267 |

| | |
|-------------------|-----|
| 第 8 章 外排序 | 269 |
| 8.1 外排序概述 | 269 |
| 8.2 磁盘排序 | 271 |
| 8.2.1 磁盘信息的存取 | 271 |
| 8.2.2 多路平衡归并 | 272 |
| 8.2.3 置换—选择排序 | 273 |
| 8.2.4 最优归并树 | 277 |
| 小结 | 280 |
| 习题 8 | 282 |
| 第 9 章 查找 | 285 |
| 9.1 查找的基本概念 | 285 |
| 9.2 静态表查找 | 286 |
| 9.2.1 顺序查找 | 286 |
| 9.2.2 二分查找 | 287 |
| 9.2.3 分块查找 | 289 |
| 9.3 动态表查找 | 292 |
| 9.3.1 二叉排序树 | 292 |
| 9.3.2 平衡二叉树 | 299 |
| 9.3.3 B-树和 B+树 | 302 |
| 9.3.4 红黑树简介 | 311 |
| 9.4 哈希表查找 | 313 |
| 9.4.1 哈希表的定义 | 313 |
| 9.4.2 常用的哈希函数 | 314 |
| 9.4.3 处理冲突的方法 | 317 |
| 9.4.4 哈希表的查找和性能分析 | 322 |
| 小结 | 323 |
| 习题 9 | 324 |
| 附录 A 习题参考答案 | 327 |
| 参考文献 | 390 |

第 1 章

绪论

随着计算机和信息技术的飞速发展,计算机应用远远超出了单纯进行数值计算的范畴,计算机技术已渗透到了国民经济的各行各业和人们日常生活的方方面面。由于现实世界中产生的大量数据只有经过计算机存储才能进行后续处理与利用,因此,数据的表示是计算机处理问题的基础。从表面上看,程序设计和编写的主要目标是为了实现某类数据的处理,而实质上,更为重要的是要考虑如何在计算机中组织这些数据,以支持后续高效的处理过程。因此,深入研究各种数据的逻辑结构及其在计算机中的表示和实现,科学、合理地指导程序设计,这正是学习“数据结构”这门课程的目的。

本章主要知识点:

- 本课程讨论的内容;
- 基本概念和术语;
- 算法的真正含义及其描述;
- 算法性能分析方法;
- 时间复杂度的估算;
- 空间复杂度的估算;
- Java 中的一些泛型特性。

1.1 数据结构课程讨论的内容

Pascal 语言之父、结构化程序设计的先驱、著名的瑞士科学家沃思(Niklaus Wirth)教授曾出版过一本著名的书籍《算法+数据结构=程序》,他认为:程序是计算机指令的组合,用来控制计算机的工作流程,完成一定的逻辑功能以实现某种任务;算法是程序的逻辑抽象,是解决某类客观问题的策略;数据结构是现实世界中的数据及其之间关系的反映,它可以从逻辑结构和存储(物理)结构两个层面进行刻画,其中客观事物自身所具有的结构特点,称为逻辑结构;而具有这种逻辑结构的数据在计算机存储器中的组织形式则称为存储结构。可见,这个等式反映了算法、数据结构对于程序设计的重要性。通过学习本课程,读者将体会到在解决一个现实问题时应该如何采用一种合适的方法(算法),又应该如何编写一个高效率程序,并通过计算机实现问题的求解。

1.1.1 求解问题举例

人们利用计算机的目的是为了能快速解决实际的应用问题,因而利用计算机实现问题

的求解,就需要完成一个从问题到程序的实现过程,此过程的主要步骤归纳如下。

(1) 确定问题求解的数学模型(或逻辑结构):对问题进行深入分析,确定处理的数据对象是什么,再考虑根据处理对象的逻辑关系给出其数学模型。

(2) 确定存储结构:根据数据对象的逻辑结构及其所需完成的功能,选择一种合适的组织形式将数据对象映射到计算机的存储器中。

(3) 设计算法:讨论要解决问题的策略,即算法的具体步骤。

(4) 编程并测试结果:根据算法编写程序并上机测试,直至得到问题的最终解。

由此可见,程序设计的本质在于解决两个主要问题:一是根据实际问题选择一种好的数据结构,二是设计一个好的算法,后者的好坏在很大程度上取决于前者。

例如:

问题一: N 个数的选择问题。假设有 N 个数,要求找出 N 个数中第 k 大的那个数。

要解决这个问题,首先要考虑的就是这 N 个数的取值范围是什么,也就是说,只有明确了这 N 个数在计算机中是如何组织的,才能决定采用何种求解问题的策略。

解决此问题的一种算法是将 N 个数读入一个数组中,再通过冒泡排序方法对 N 个数以递减顺序进行排序,最后在已排序数组中的第 k 个元素就是所要求的解。这种策略实施的前提是内存中有足够的空间能够容纳这 N 个数。然而,若空间不够,又该如何处理呢?

解决此问题的另一种算法是先将 N 个数的前 k 个数读入一个数组,并以递减顺序进行排序,再将剩下的数逐个读入,将它与数组中的第 k 个数进行比较,如果它不大于第 k 个数,则忽略;否则就将其插入到数组中的适当位置,使数组仍然保持有序,同时数组中的最后一个数被挤出数组。当剩下的所有数都处理完毕时,数组中位于第 k 个位置上的数就是所要求的解。

这两种算法用程序实现其编码都很简单,但哪种算法更好呢?当 n 和 k 都充分大(大于 10^6)时,这两种方法的程序在合理时间内都不能结束,它们需要计算机处理若干天后才能计算完毕,这是不切实际的,所以它们都不能被认为是好的算法。在第 7 章将给出解决这个问题的更好算法。

问题二: 生产订单的自动查询问题。在一个订单管理系统中,有一个实体“生产订单”,实体中包括若干个订单记录,它们按照每个生产订单的订单号递增顺序排列,如表 1.1 所示。

表 1.1 生产订单表

| 生产订单号 | 行 | 物料编码 | 物料名称 | 开工时间 | 完工时间 | 计量单位 | 生产数 | 制单 |
|----------|---|-------|------|-----------|-----------|------|-----|----|
| 00000010 | 1 | 12100 | 长针 | 2008-9-13 | 2008-9-14 | 根 | 60 | 张三 |
| 00000010 | 2 | 12100 | 长针 | 2008-9-14 | 2008-9-15 | 根 | 390 | 张三 |
| 00000011 | 1 | 10000 | 电子挂 | 2008-9-18 | 2008-9-20 | 个 | 60 | 李四 |
| 00000011 | 2 | 10000 | 电子挂 | 2008-9-19 | 2008-9-22 | 个 | 390 | 李四 |
| 00000012 | 1 | 12400 | 盘面 | 2008-9-13 | 2008-9-14 | 个 | 60 | 李四 |
| 00000012 | 2 | 12400 | 盘面 | 2008-9-14 | 2008-9-15 | 个 | 390 | 李四 |
| 00000013 | 1 | 12000 | 钟盘 | 2008-9-15 | 2008-9-19 | 个 | 390 | 张三 |
| 00000013 | 2 | 12000 | 钟盘 | 2008-9-14 | 2008-9-18 | 个 | 60 | 张三 |

在研究如何查找满足条件的订单时,首先必须考虑生产订单表在计算机中是如何组织的,每一订单包括哪些信息项,各订单之间又是按什么顺序存放的,是按订单号递增的顺序存放,还是按物料名称的顺序存放,或是按制单人的姓氏顺序存放,等等,然后才能根据特定的查找要求去确定某种查找方法。

例如,现在要求查找“张三”制作的所有订单。若在生产订单表中有该人制作的订单,则给出相关的所有订单;否则就指出没有该人制作的订单。要解决这个问题,首先应将生产订单表中的数据映射到计算机的存储器中,形成生产订单表的存储结构,设计的查找算法是否可行就取决于这个存储结构。一种算法是将表中数据顺序地存储到计算机中,查找时从表的第一个订单开始到最后一个订单为止,依次去核对制单人的姓名是否与“张三”相同,若找到,则可获得该人制作的一个订单;若找遍整个表均无此制单人,则说明没有该人制作的订单。这种算法在订单数不多的情况下是可行的,但是当订单数很多时是不实用的。另一种算法是将生产订单表顺序地存放,再按制单人的姓氏建立一张索引表,存储结构如图 1.1 所示。查找时首先在索引表中核对姓氏,然后根据索引表中的地址到生产订单表中查对姓名。注意这时已经不需要查找其他不同姓氏的名字了。相比之下,第二种查找算法比第一种算法更为高效。在第 8 章将给出更多的实现查找操作的算法。

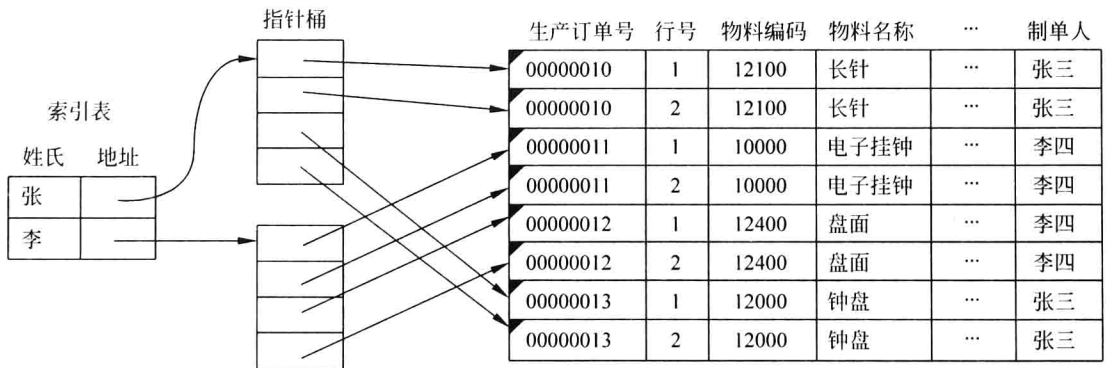


图 1.1 生产订单表的索引顺序存储结构示意图

问题三：城市网络的铺设问题。假设需为城市的各小区之间铺设网络线路(任意两个小区之间都铺设), n 个小区只需铺设 $n-1$ 条线路,就能使 n 个小区网络相通。然而,由于地理环境不同等因素导致了各条线路所需投资不同,问题是采用怎样的设计方案才能使总投资成本最低?

根据各小区之间的地理位置关系及各线路所需的投资情况,确定这个问题的数学模型可以用一个图来描述。假设该图是如图 1.2(a)所示,其中顶点表示城市,顶点之间的连线及其上面的数值表示可以铺设的线路及所需经费。求解该问题的算法为:在可以铺设的 n 条线路中选取 $n-1$ 条,使得这条线路既能连通 n 个小区,又使总投资最省。实际上,这是一个“求图的最小生成树”问题,图 1.2(a)对应的最小生成树可以用图 1.2(b)来描述。有关最小生成树的算法将在第 6 章中进行详细讨论。

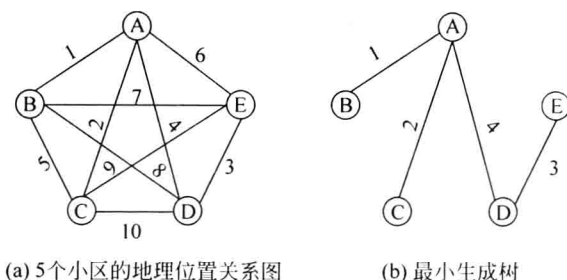


图 1.2 城市网络的铺设问题

1.1.2 本课程讨论的内容

数据结构与数学、计算机硬件和软件有着十分密切的关系。它是介于数学、计算机硬件和软件之间的一门计算机类和电子信息类相关专业的核心课程之一,是高级程序设计语言、编译原理、操作系统、数据库、人工智能等课程的基础。同时,数据结构技术也广泛应用于信息科学、系统工程、应用数学以及各种工程技术领域。

从上节讨论的几个求解问题可知,用计算机来解决一个具体问题,总是围绕以下 3 个主要步骤进行:

- (1) 抽象出所求解问题中需要处理的数据对象的逻辑结构(数学模型)。
- (2) 根据所求解问题需要完成的功能特性实现对抽象数据的存储结构表述。
- (3) 确定为求解问题而需要进行的操作或运算。

事实上,只有这三者的结合,才能清晰地刻画出数据结构的本质特性。因此,通常在本课程中讨论数据结构,既要讨论在解决问题时各种可能遇到典型的逻辑结构,又要讨论这些逻辑结构在计算机中的存储映射(存储结构),此外,还要讨论数据结构的相关操作及其实现。与此同时,为了构造出好的数据结构并加以实现,还必须考虑其实现方法的性能评价。因此,数据结构课程的内容体系可如表 1.2 所示进行归纳。

表 1.2 数据结构课程的内容体系

| 过程 \ 方面 | 数据表示 | 数据处理 |
|---------|------------------|------|
| 抽象 | 逻辑结构 | 基本运算 |
| 实现 | 存储结构 | 算法 |
| 评价 | 不同数据结构的比较和算法性能分析 | |

1.2 基本概念与术语

在本节中将对一些常用的基本概念与术语给出详细的解释,这些基本概念与术语将贯穿数据结构学习的整个过程。