

普通高校本科计算机专业特色教材精选·算法与程序设计

数据结构及其应用

滕国文 主编
李颖 李洪川 王洪君 谭振江 副主编



清华大学出版社

普通高校本科计算机专业特色教材精选·算法与程序设计

数据结构及其应用

滕国文 主编
李颖 李洪川 王洪君 谭振江 副主编



清华大学出版社
北京

内 容 简 介

本书系统地介绍各种常用的数据结构以及排序、查找的各种算法,阐述各种数据结构的逻辑特性、存储结构、运算及其实现的算法等。全书共 10 章,各章都配有习题,方便读者巩固所学知识。

本书的特色是:理论讲解与实践应用相结合。即在对每种数据结构基本概念、基本理论和基本方法详细讲解的基础上,给出应用该数据结构的具体实例,并给出解决问题的完整过程,以帮助学生系统地掌握该门课程的基本内容,并运用所学的数据结构知识解决实际问题。强调求解问题的思想方法,重点培养学生的实际分析问题和解决问题的能力、算法设计能力、数据结构的应用创新能力等。

本书是作者多年教学实践经验的总结,用标准 C 语言描述算法,内容丰富,概念清楚,论述充分,取材得当。

本书可作为高等院校计算机及相关专业数据结构课程的教材,也可供计算机应用软件开发人员和计算机编程爱好者参考。

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

数据结构及其应用/滕国文主编. —北京:清华大学出版社,2015

普通高校本科计算机专业特色教材精选·算法与程序设计

ISBN 978-7-302-39656-7

I. ①数… II. ①滕… III. ①数据结构—高等学校—教材 IV. ①TP311.12

中国版本图书馆 CIP 数据核字(2015)第 055533 号

责任编辑:袁勤勇 战晓雷

封面设计:傅瑞学

责任校对:李建庄

责任印制:何 芊

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 刷 者:三河市君旺印务有限公司

装 订 者:三河市新茂装订有限公司

经 销:全国新华书店

开 本:185mm×260mm 印 张:16

字 数:402千字

版 次:2015年4月第1版

印 次:2015年4月第1次印刷

印 数:1~2000

定 价:34.50元

出版说明

INTRODUCTION

在我国高等教育逐步实现大众化后，越来越多的高等学校将会面向国民经济发展的第一线，为行业、企业培养各级各类高级应用型专门人才。为此，教育部已经启动了“高等学校教学质量和教学改革工程”，强调要以信息技术为手段，深化教学改革和人才培养模式改革。如何根据社会的实际需要，根据各行各业的具体人才需求，培养具有特色显著的人才，是我们共同面临的重大问题。具体地说，培养具有一定专业特色的和特定能力强的计算机专业应用型人才则是计算机教育要解决的问题。

为了适应 21 世纪人才培养的需要，培养具有特色的计算机人才，急需一批适合各种人才培养特点的计算机专业教材。目前，一些高校在计算机专业教学和教材改革方面已经做了大量工作，许多教师在计算机专业教学和科研方面已经积累了许多宝贵经验。将他们的教研成果转化为教材的形式，向全国其他学校推广，对于深化我国高等学校的教学改革是一件十分有意义的事情。

清华大学出版社在经过大量调查研究的基础上，决定组织出版一套“普通高校本科计算机专业特色教材精选”。本套教材是针对当前高等教育改革的新形势，以社会对人才的需求为导向，主要以培养应用型计算机人才为目标，立足课程改革和教材创新，广泛吸纳全国各地的高等院校计算机优秀教师参与编写，从中精选出版确实反映计算机专业教学方向的特色教材，供普通高等院校计算机专业学生使用。

本套教材具有以下特点。

1. 编写目的明确

本套教材是在深入研究各地各学校办学特色的基础上，面向普通高校的计算机专业学生编写的。学生通过本套教材，主要学习计算机科学与技术专业的基本理论和基本知识，接受利用计算机解决实际问题的基本训练，培养研究和开发计算机系统，特别是应用系统的基本能力。

2. 理论知识与实践训练相结合

根据计算学科的三个学科形态及其关系,本套教材力求突出学科的理论与实践紧密结合的特征,结合实例讲解理论,使理论来源于实践,又进一步指导实践。学生通过实践深化对理论的理解,更重要的是使学生学会理论方法的实际运用。在编写教材时突出实用性,并做到通俗易懂,易教易学,使学生不仅知其然,知其所以然,还要会其如何然。

3. 注意培养学生的动手能力

每种教材都增加了能力训练部分的内容,学生通过学习和练习,能比较熟练地应用计算机知识解决实际问题。既注重培养学生分析问题的能力,也注重培养学生解决问题的能力,以适应新经济时代对人才的需要,满足就业要求。

4. 注重教材的立体化配套

大多数教材都将陆续配套教师用课件、习题及其解答提示,学生上机实验指导等辅助教学资源,有些教材还提供能用于网上下载的文件,以方便教学。

由于各地区各学校的培养目标、教学要求和办学特色均有所不同,所以对特色教学的理解也不尽一致,我们恳切希望大家在使用教材的过程中,及时地给我们提出批评和改进意见,以便我们做好教材的修订改版工作,使其日趋完善。

我们相信经过大家的共同努力,这套教材一定能成为特色鲜明、质量上乘的优秀教材。同时,我们也希望通过本套教材的编写出版,为“高等学校教学质量和教学改革工程”做出贡献。

清华大学出版社

前言

PREFACE

数据结构是计算机软件和计算机应用专业的核心课程之一，是计算机程序设计的理论基础，也是学习操作系统、数据库原理、编译原理等课程的先导课。

数据结构课程的教学目标是：要求学生学会分析数据对象特征，掌握数据组织方法和计算机的表示方法，以便为应用所涉及的数据选择适当的逻辑结构、存储结构以及相应算法，初步掌握算法时间和空间分析的技巧，培养较强的程序设计技能。

数据结构的学习过程是进行复杂程序设计的训练过程。技能培养的重要程度不亚于知识传授，学生不仅要理解授课内容，还应培养应用知识解答复杂问题的能力，形成良好的算法设计思想、方法技巧与风格，进行构造性思维，强化程序抽象能力和数据抽象能力。因此，学习数据结构，仅学习理论知识是不够的，必须经过大量的实践，在实践中体会构造性思维的方法，掌握数据组织与程序设计的技术。

本书共 10 章，第 1 章介绍数据结构的基本概念和算法描述以及算法分析等内容；第 2 章~第 8 章分别介绍线性表、栈、队列、串、数组和广义表、树与二叉树和图等基本数据结构；第 9 章介绍在计算机中广泛使用的各种排序方法，并对各种排序算法的优劣进行分析和比较；第 10 章介绍顺序表、索引表和散列表上的查找算法。

本书由吉林师范大学滕国文教授任主编，吉林师范大学李颖、王洪君、谭振江和吉林信息工程学校李洪川任副主编。具体分工是：滕国文编写第 1 章和第 2 章，李洪川编写第 3 章和第 4 章，李颖编写第 5 章和第 6 章，王洪君编写第 7 章和第 8 章，谭振江编写第 9 章和第 10 章。参加编写、程序调试和绘制图形的还有滕泰、张静、高阳、姜丽丽、温毓铭、高巨、李东林、周伟、夏凤琴等人。谭振江教授审阅了书稿，并提出修改意见，最后由滕国文教授统稿、整理后定稿。

在本书的编写过程中，作者参阅并借鉴了国内外诸多同行的文章和著作，在此向他们致以谢意。

由于作者知识水平有限，时间仓促，本书难免有不足之处，恳请专家和读者批评指正。

编者

2015年3月

目 录

CONTENTS

第 1 章 绪论	1
1.1 问题求解过程	1
1.2 数据结构的基本概念	3
1.2.1 实例	3
1.2.2 基本概念与术语	5
1.3 算法	9
1.3.1 算法的基本概念	9
1.3.2 算法的基本特性	9
1.4 算法描述	10
1.4.1 基本控制结构的描述	10
1.4.2 C 语言算法描述规定	12
1.5 算法分析	15
1.5.1 算法的评价标准	15
1.5.2 算法的时间复杂度	16
1.5.3 算法的空间复杂度	17
小结	18
习题	18
第 2 章 线性表	21
2.1 基本概念	21
2.1.1 线性表的定义	21
2.1.2 线性表的逻辑结构	22
2.1.3 线性表的基本运算	22
2.2 线性表的顺序存储结构	23
2.2.1 顺序表	23
2.2.2 顺序表的基本运算	25

2.3	线性表的链式存储结构	29
2.3.1	链表	29
2.3.2	单链表	30
2.3.3	单链表的基本运算	32
2.3.4	循环链表	36
2.3.5	双链表	38
2.4	顺序表与链表的比较	39
2.5	线性表的应用	40
2.5.1	顺序表运算的综合实例	40
2.5.2	一元多项式的表示及相加	42
	小结	47
	习题	48
第3章	栈	53
3.1	基本概念	53
3.1.1	栈的概念	53
3.1.2	栈的基本运算	54
3.2	栈的顺序存储结构	54
3.2.1	顺序栈	54
3.2.2	顺序栈的基本运算	55
3.3	栈的链式存储结构	56
3.3.1	链式栈	56
3.3.2	链式栈的基本运算	57
3.4	栈的应用	58
3.4.1	数制转换问题	58
3.4.2	西文编辑器	61
3.4.3	括号匹配问题	62
	小结	65
	习题	65
第4章	队列	69
4.1	基本概念	69
4.1.1	队列的概念	69
4.1.2	队列的基本运算	70
4.2	队列的顺序存储结构	70
4.2.1	顺序队列	70
4.2.2	顺序队列的基本运算	71
4.2.3	循环队列	72

4.2.4 循环队列的基本运算	74
4.3 队列的链式存储结构	75
4.3.1 链队列	75
4.3.2 链队列的基本运算	75
4.4 队列的应用	77
4.4.1 舞伴问题	77
4.4.2 看病排队候诊问题	79
小结	81
习题	81
第5章 串	85
5.1 基本概念	85
5.1.1 串的定义	85
5.1.2 串的术语	85
5.1.3 串的基本运算	86
5.2 串的存储结构	87
5.2.1 串的顺序存储结构	87
5.2.2 串的链式存储结构	91
5.2.3 串的堆式存储	92
5.3 串的应用	93
5.3.1 KMP 算法实现串的模式匹配	93
5.3.2 最长公共子串	95
小结	97
习题	97
第6章 数组和广义表	99
6.1 数组的基本概念	99
6.1.1 数组的定义	99
6.1.2 数组的相关运算	100
6.2 数组的顺序存储结构	100
6.3 矩阵的压缩存储	101
6.3.1 特殊矩阵	101
6.3.2 稀疏矩阵	103
6.4 广义表	104
6.4.1 广义表的基本概念	104
6.4.2 广义表的表头和表尾	105
6.5 数组的应用	106
6.5.1 魔方阵	106

6.5.2 稀疏矩阵的加法运算	109
小结	115
习题	115
第7章 树和二叉树	119
7.1 树的基本概念	119
7.1.1 树的定义	119
7.1.2 树的基本术语	119
7.1.3 树的基本运算	121
7.2 二叉树的基本概念	121
7.2.1 二叉树的定义	121
7.2.2 二叉树的性质	122
7.3 二叉树的存储结构	123
7.3.1 顺序存储结构	123
7.3.2 链式存储结构	124
7.4 二叉树的遍历	125
7.4.1 先序遍历	126
7.4.2 中序遍历	126
7.4.3 后序遍历	126
7.4.4 层次遍历	127
7.5 线索二叉树	128
7.5.1 线索二叉树的概念	128
7.5.2 中序线索二叉树的构造算法	129
7.5.3 线索二叉树的遍历	130
7.6 哈夫曼树及其应用	131
7.6.1 哈夫曼树的定义	131
7.6.2 构造哈夫曼树	132
7.6.3 哈夫曼树的应用	134
7.7 树、森林与二叉树的关系	135
7.7.1 树的存储结构	135
7.7.2 森林与二叉树的转换	137
7.7.3 树与森林的遍历	138
7.8 二叉树的应用	139
7.8.1 由遍历确定二叉树	139
7.8.2 电文的编码和译码	142
小结	147
习题	147

第 8 章 图	151
8.1 图的基本概念	151
8.1.1 图的定义	151
8.1.2 图的基本术语	152
8.2 图的存储结构	153
8.2.1 邻接矩阵	153
8.2.2 邻接表	154
8.3 图的遍历	158
8.3.1 深度优先搜索算法	158
8.3.2 广度优先搜索算法	159
8.4 最小生成树	160
8.4.1 Prim 算法	161
8.4.2 Kruskal 算法	164
8.5 最短路径	166
8.6 拓扑排序	168
8.7 图的应用	171
8.7.1 地铁建设问题	171
8.7.2 安排教学计划	175
小结	178
习题	178
第 9 章 排序	183
9.1 基本概念	183
9.2 插入排序	184
9.2.1 直接插入排序	185
9.2.2 希尔排序	187
9.3 交换排序	188
9.3.1 冒泡排序	188
9.3.2 快速排序	190
9.4 选择排序	193
9.4.1 简单选择排序	193
9.4.2 堆排序	194
9.5 归并排序	197
9.6 各种排序方法的比较	199
9.7 排序的应用	201
9.7.1 排序的综合实例	201
9.7.2 希尔排序应用实例	204

小结·····	206
习题·····	207
第 10 章 查找 ·····	211
10.1 基本概念·····	211
10.2 静态查找表·····	212
10.2.1 顺序查找·····	212
10.2.2 折半查找·····	214
10.3 动态查找表·····	218
10.3.1 二叉排序树·····	218
10.3.2 平衡二叉树·····	223
10.3.3 B-树·····	225
10.4 哈希表·····	229
10.4.1 哈希表的概念·····	229
10.4.2 哈希函数的构建·····	230
10.4.3 处理冲突·····	232
10.4.4 哈希表的查找及其分析·····	234
10.5 查找的应用·····	235
10.5.1 查找方法综合实例·····	235
10.5.2 二分查找应用实例·····	238
小结·····	241
习题·····	241
参考文献 ·····	244

计算机是对各种数据进行处理机器。在进行数据处理时,实际需要处理的数据元素一般有很多,而这些大量的数据元素都需要存放在计算机中,因此,在计算机中如何组织数据,如何处理数据,从而如何更好地利用数据是计算机科学的基本研究内容。掌握数据在计算机中的各种组织和处理方法是深入学习计算机技术的基础,从程序设计的观点看,信息在计算机中的表示就是数据结构研究的问题;信息在计算机中处理就是算法研究的问题。因此,学习算法和数据结构的基础知识是了解计算机工作基本原理、掌握程序设计基本技术的必经之路,其目的是提高数据处理的效率。所谓提高数据的处理效率,主要包括两个方面:一是提高数据处理的效率,二是节省在数据处理过程中所占用的计算机存储空间。

本章主要内容包括数据结构内涵及其研究的问题,数据结构中的基本概念和术语,数据结构所使用的描述工具,算法的概念、特性以及性能分析等。

1.1 问题求解过程

用计算机解决实际问题,就是在计算机中建立一个解决问题的模型。在这个模型中包括以下 3 个部分:计算机内部的数据,它表示了需要被处理的实际对象,包括其内在的性质和关系;处理这些数据的程序;模拟对象领域中的求解过程。通过解读计算机程序的运行结果,可以得到实际问题的解。下面给出用计算机求解问题的一般步骤。

1. 问题定义

这个阶段的任务是弄清已知信息和所要解决的问题是什么。完整地理解和描述问题是解决问题的关键。要做到这一点,必须注意以下一些问题:对未经加工的原始表达中所用的术语是否都明白其准确含义?题目提供了哪些已知信息?还可以得到哪些潜在的信息?题目中做了哪些假定?题目要求得到什么结果?等等。针对每个具体问题,必须认真审查问题的

有关描述,深入分析,以加深对问题的准确理解。

2. 建立模型

用计算机解决实际问题必须建立合适的数学模型,因为在现实问题面前,计算机是无能为力的。针对一个实际问题建立数学模型,可以考虑这样两个基本问题:最适合于此问题的数学模型是什么?是否有已经解决了的类似问题可以借鉴?

将实际问题经过多次抽象,建立计算机能存储、处理的数据模型。建立模型是最关键且较困难的一步,涉及4个世界和3级抽象。4个世界分别是现实世界(客观世界)、信息世界(概念世界)、数据世界、计算机世界。3级抽象分别是:现实世界到信息世界的抽象,建立信息模型或概念模型;信息世界到数据世界的抽象,将信息转化为数据模型;数据世界到计算机世界的抽象,将数据模型转化为存储模型并在计算机中实现。

3. 算法设计

算法设计是指设计求解某一特定类型问题的一系列步骤,并且这些步骤是可以通过计算机的基本操作来实现的。算法设计要同时结合数据结构的设计,简单地说,数据结构的设计就是选取存储方式,因为不同的数据结构设计将导致算法的差异很大。算法的设计与模型的选择更是密切相关的,但同一模型仍然可以有不同的算法,而且它们的有效性可能有相当大的差距。

算法设计方法也称算法设计技术或算法设计策略,是设计算法的一般性方法,可用于解决不同计算领域的多种问题。虽然设计算法,尤其是设计出好的算法是一件非常困难的工作,但是设计算法也不是没有方法可循,人们经过几十年来的工作,总结和积累了许多行之有效的方法,了解和掌握这些方法会给我们解决问题提供一些思路。常用的算法设计方法包括求值法、累加法、累乘法、递推法、递归法、枚举法、分治法、贪心法、回溯法和动态规划法等。这些算法设计方法构成了一组强有力的工具,可用于大量实际问题求解。

4. 算法表示

对于复杂的问题,确定算法后可以选择一种算法描述方法来准确表示算法。算法的描述方式很多,如传统流程图、盒图、PAD图、伪码和高级语言等。其中高级语言是最理想的描述算法的方法,因此,本书选择C语言来表示算法。

5. 算法分析

算法分析的目的主要有两个:首先是为了针对算法的某些特定输入,估算该算法所需的内存空间和运行时间;其次是为了建立衡量算法优劣的标准,用以比较同一类问题的不同算法。一般来说,一个好的算法首先应该是比同类算法的时间效率高,算法的时间效率用时间复杂度来度量。

6. 算法实现

算法实现就是指编码,也就是平常所说的编程序,即将算法设计“转译”成某种计算机语言的表述形式,才能够在计算机上执行。编码的目的是使用选定的程序设计语言,把算法描述翻译成为用该语言编写的源程序(或源代码)。源程序应该正确可靠、简明清晰,而且具有较高的效率。

在把算法转变为程序的过程中,虽然现代编译器提供了代码优化功能,但是,仍然需要一些技巧,例如,在循环之外计算循环中的不变式、合并公共子表达式等。

7. 程序调试

程序调试也称算法测试,其任务首先是发现和排除在前几个阶段中产生的错误,经测试通过的程序才可投入运行,在运行过程中还可能发现隐含的错误和问题,因此,还必须在使用中不断地维护和完善。

算法测试的实质是对算法应完成任务的实验证实,同时确定算法的使用范围。测试方法一般有两种:白盒测试,对算法的各个分支进行测试;黑盒测试,检验对给定的输入是否有指定的输出。

8. 结果整理文档编制

结果整理时,要对计算结果进行分析,看其是否符合实际问题的要求,如果符合,问题得到解决,可以结束;如果不符合,说明前面的步骤一定存在问题,必须返回,从头开始逐步检查,找出错误并重新设计,这个循环过程也可能重复多次。

编制文档的目的是让别人理解自己编写的算法。首先要把代码编写清楚,代码本身就是文档,同时还有代码的注释。另外还包括算法的流程图,自顶向下各研制阶段的相关记录,算法的正确性证明(论述),算法测试过程、结果,对输入输出的要求及格式的详细描述等。

1.2 数据结构的基本概念

数据结构是计算机专业的核心课程之一,也是计算机专业知识的基石。下面通过几个实际例子引出数据结构这门课程研究的主要内容,然后给出数据结构的基本概念和术语。

1.2.1 实例

【例 1-1】 已知学生某科考试成绩,进行如下成绩分析:

- (1) 求平均成绩;
- (2) 求高于平均分的学生数;
- (3) 求各分数段的学生人数。

问题分析:

这里需要多次使用到学生的考试成绩,所以需要成绩保存起来,以便多次使用。因为考试成绩就是一组整型值或实型值,所以,可以利用各种高级语言都支持的数组来存储。

数组就是一种数据结构。

【例 1-2】 学生信息表,如表 1-1 所示。

该表中,每一列称为一个属性,表示学生的一个特征;每一行称为一个记录,描述一个学生的有关信息,由学号、姓名、性别、年龄和入学成绩等属性组成。在数据结构中,将记录称为数据元素、结点或顶点,属性称为数据项。学生信息之间的关系可以看成是一个接一个排列的一对一关系,这种定义了元素之间的完全顺序关系的数据结构称为线性数据结构,简称线性表。

表 1-1 学生信息表

学号	姓名	性别	年龄	入学成绩
201441020101	王津柳	女	18	521.5
201441020102	张建莹	女	18	523
201441020103	孙天宇	男	19	525.5
201441020104	李玥	女	19	528
⋮	⋮	⋮	⋮	⋮

【例 1-3】 计算机系统中的文件管理问题。

计算机操作系统的文件管理采用多级目录结构,如图 1-1 所示。

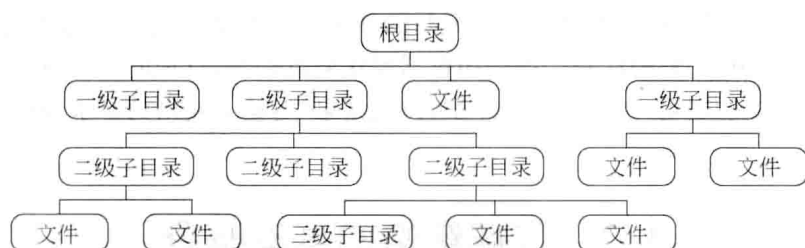


图 1-1 文件系统目录结构

这种多级目录层次结构形成一棵倒立的树。树中的目录或文件抽象成数据元素,也称为结点,结点之间的关系是一对多关系。通常,称此类具有一对多关系的数据结构为树形结构,简称树结构或树。

【例 1-4】 制定教学计划。以计算机科学与技术专业为例,部分教学计划课程安排如表 1-2 所示。

表 1-2 计算机专业部分教学计划表

课程代号	课程名称	先修课程	课程代号	课程名称	先修课程
C1	高等数学	无	C6	编译原理	C4, C5
C2	程序设计基础	无	C7	操作系统	C4, C9
C3	离散数学	C1, C2	C8	普通物理	C1
C4	数据结构	C3, C5	C9	计算机原理	C8
C5	C 语言	C2			

在教学计划表中,每个学期开设的课程是有先后顺序的,有的课程是基础课,不需要先修其他课程,如“高等数学”和“程序设计基础”;另一些课程则必须在先学完某些先修课程之后才能开设,如在学习“数据结构”课程之前,必须先学完“C 语言”和“离散数学”课程。一门课程可以有多门先修课程,也可以有多门后续课程,如图 1-2 所示。这种元素之间存在多对多关系的数据结构称为图状结构。图状结构是一种最复杂的数据结构。