

高等院校信息技术规划教材

算法与数据结构

宁正元 王秀丽 编著

INFORMATION TECHNOLOGY
INFORMATION TECHNOLOGY
INFORMATION TECHNOLOGY

清华大学出版社



高等院校信息技术规划教材

算法与数据结构

宁正元 王秀丽 编著

清华大学出版社
北京

内 容 简 介

本书覆盖了《中国计算机科学与技术学科教程 2002》中关于核心课程“算法与数据结构”的所有知识单元和课程提纲,系统地介绍了各种常用的数据结构的有关知识和各种基本的检索排序算法。每章配有足量的例题、习题和上机实验题,并另有与之配套的《数据结构学习辅导》和电子教案(清华大学出版社出版),便于教师组织教学和学生自学。

全书以知识单元为基本构件,便于拆卸和重组,可以满足不同院校计算机科学与技术学科各专业的教学需求,也可作为从事计算机科学与技术工作的科技人员的参考用书。

版权所有,翻印必究。举报电话:010-62782989 13501256678 13801310933

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。

本书防伪标签采用特殊防伪技术,用户可通过在图案表面涂抹清水,图案消失,水干后图案复现;或将表面膜揭下,放在白纸上用彩笔涂抹,图案在白纸上再现的方法识别真伪。

图书在版编目(CIP)数据

算法与数据结构/宁正元,王秀丽编著. —北京:清华大学出版社,2006.1

(高等院校信息技术规划教材)

ISBN 7-302-11865-5

I. 算… II. ①宁…②王… III. ①算法分析—高等学校—教材②数据结构—高等学校—教材
IV. ①TP301.6②TP311.12

中国版本图书馆 CIP 数据核字(2005)第 109683 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

组稿编辑:张 龙

文稿编辑:王冰飞

印 刷 者:清华大学印刷厂

装 订 者:三河市李旗庄少明装订厂

发 行 者:新华书店总店北京发行所

开 本:185×260 印张:17.75 字数:414 千字

版 次:2006 年 1 月第 1 版 2006 年 1 月第 1 次印刷

书 号:ISBN 7-302-11865-5/TP·7710

印 数:1~3000

定 价:23.00 元

编委会名单

主 任： 李文忠

副主任： 王正洪 鲁宇红 焦金生

成 员： (按拼音排序)

常晋义 邓 凯 范新南 高佳琴 高玉寰

龚运新 顾建业 顾金海 林 罡 刘训非

马正华 沈孟涛 王继水 王 骏 王 晴

王志立 吴访升 肖 玉 杨长春 袁启昌

张旭翔 张 燕 赵明生 郑成增 周凤石

策划编辑： 张 龙 袁勤勇

序

preface

在科教兴国方针的指引下,我国高等教育进入了一个新的历史发展时期,招生规模 and 在校生数量都有了大幅度的增长。我们在进行着世界上规模最大的高等教育。与此同时,对于高等教育的研究和认识也在不断深化。高等学校要明确自己的办学方向和办学特色,这既是不断提高高等教育水平的必然要求,更是高校不断发展和壮大必须首先考虑的问题。

教育部领导明确提出要有相当部分的高校致力于培养应用型人才,此类院校在计算机教学中如何实现自己的培养目标,如何选择适用的应用型教材,已成为十分重要和迫切的任务。应用型人才的培养不能简单照搬研究型人才的培养方案,要在丰富的实践基础上认真总结,摸索新形势下的教学规律,在此基础上设计相关课程、改进教学方法,同时编写应用型教材。这一工作是非常艰巨的,也是非常有意义的。

在清华大学出版社的大力支持和配合下,于2003年成立了应用型教材编委会。编委会汇集了众多高校的实践经验,并经过集中讨论和专家评审,遴选了一批优秀教材,希望能够通过这套教材的出版和使用,促进应用型人才培养的实践发展,为建立新的人才培养模式作出贡献。

我们编写应用型教材的主要出发点是:

1. 适应教育部对高等教育的新要求,以及市场对应用型人才需求量的不断增加。
2. 计算机科学技术不断更新,发展速度加速,教材内容和教学方式将适时更新和改进。
3. 教育技术的发展,对教材建设提出了更高的要求,教材将呈现出纸介质出版物、电子课件以及网络学习环境等相互配合的立体化形态。



4. 突出应用,增强实训,根据不同的专业要求,加强针对性,使理论与实践紧密结合。

从上述各点出发,我们将努力建设一套全新的、有实用价值的应用型计算机教材。经过参编教师的努力,第一批教材已经面世。教材将滚动式地不断更新、修正、提高,逐渐树立起自己的品牌。希望使用本系列教材的广大师生不断反馈各类意见,以便改进、修订,逐步建设具有应用型特色的精品教材。

李文忠

2004年9月

前言

foreword

“算法与数据结构”是计算机科学与技术学科中的一门十分重要的专业基础课程和专业核心课程。中国计算机学会教育专业委员会和全国高等学校计算机教育研究会在《计算机学科教学计划 1993》中,把“算法与数据结构”课程列为计算机科学与技术学科公共要求的 9 个主科目之一。中国计算机科学与技术学科教程 2002 研究组在《中国计算机科学与技术学科教程 2002》中,把“算法与数据结构”课程列为计算机科学与技术学科 16 个专业核心课程之一。计算机科学与技术学科的各个领域,都要求科技工作者具备良好的算法与数据结构基础。

在前导课程“高级语言程序设计”中,我们已经学会了简单程序设计的基本理论和基本方法。“算法与数据结构”课程讨论程序数据的结构、组织和管理技术以及在此基础上的算法设计与分析技术,即学习较复杂程序设计的问题。“算法与数据结构”课程的教学要求是,学会分析研究计算机加工的数据对象的特性,以便在实际应用中选择适当的数据结构、存储结构和相应的算法,初步掌握算法的时间性能与空间性能的分析技巧,得到复杂程序设计的初步训练。本书在内容的组织上旨在阐明常用数据结构的内在逻辑关系,讨论各种结构的机内存储表示方法,通过实例说明各种结构在运算操作时的动态特性,并结合典型应用问题给出算法设计与分析的示例。这样,不仅为后续课程如“操作系统”、“编译原理”、“数据库原理”、“软件工程”、“人工智能”、“计算机图形学”、“计算机辅助设计”、“多媒体技术”等提供必要的知识准备,更重要的是可以进一步提高读者从事软件分析、设计、编程和数据组织的能力和水平。

作者在认真总结这二十多年来从事“数据结构”及其相关课程的教学经验和体会的基础上,仔细分析现阶段大学生的素质特点和学习现状,按照《中国计算机科学与技术学科教程 2002》中设定的

“算法与数据结构”课程教学大纲要求编写了这本教材。书中强调算法与数据结构之间密不可分的联系,体现程序设计方法中抽象、枚举、归纳三个原则的运用,反复再现算法的复杂性、效率、折衷、重用等重要概念,并增加了“数据结构的程序实现”一章内容,有利于读者在较深层次上掌握好“算法与数据结构”这一核心课程。

在内容的选取上较好地体现了突出应用的原则——在实例列举上力求做到典型和恰当,并在各章都附有相应的习题和上机实验题;以实例介绍各种数据结构的应用,为后续课程的衔接即应用各种数据结构奠定了良好的基础。

在内容的组织上遵循循序渐进的原则——由程序与算法导入,从数组和串开讲数据结构,与前导课程“高级语言程序设计”自然衔接;内容由浅入深,按常用数据结构、简单数据结构、树与二叉树、图与网、检索和排序的次序安排主要教学内容。

在内容的叙述上力求做到符合通俗易懂的原则——算法描述结构清晰、易读易理解,并对每一个算法都做了大量的注释。

全书内容共8章。第1章介绍算法与程序的基本概念、表示、设计和分析方法,是在前导课程“高级语言程序设计”基础上的进一步引申和总结;第2章由前导课程中的数据类型概念引出数据结构的有关基本概念,并介绍读者已在前导课程中熟知的数组结构和串结构的进一步内容;第3章深入浅出地介绍一些简单的线性结构(如顺序表、链表、栈、队列、广义表等结构)的定义、基本算法和典型应用;第4章和第5章介绍非线性结构中的树、二叉树、图与网的有关内容;第6章介绍各类数据结构在程序设计语言中的实现策略和大批量数据的组织策略,并结合实际问题讨论了数据结构在问题建模中的应用方法;第7章和第8章介绍在实际应用中广泛使用的检索和排序的基本算法,并进行了必要的算法分析。全书以知识单元为基本构件,各知识单元内容相对独立,具有可拆卸性、可重组性,便于不同院校根据各自的需要组织教学。

建议读者多动手,通过实际编程实现各种算法,并在深入分析的基础上改进算法或提出新的算法。《中国计算机科学与技术学科教程2002》中建议本课程的讲授课时为72学时,实验课时为32学时;各校可根据自己的实际情况作适当调整。

本书由宁正元和王秀丽共同编著,其中第1、6章由宁正元执笔,第2、3、4、5、7、8章由王秀丽执笔。林大辉和黄健承担了全书的文字录入、排版、校对和部分算法的上机调试;林大辉还针对本书制作了电子教案;陈国龙和康宝生在百忙中分别阅读了书稿的部分章节并提出了一些宝贵的修改意见;清华大学出版社对本书的出版给予了极大的支持和鼓励,作者在此一并表示真诚的感谢。

鉴于时间仓促和作者水平所限,书中一定还存在着疏漏之处,敬请同行专家和广大读者不吝赐教,我们将不胜感激。

作 者

2005年8月

目录

Contents

第 1 章	算法与程序	1
1.1	算法的基本概念	1
1.1.1	什么是算法	1
1.1.2	算法的基本特性	2
1.2	算法的表示	3
1.2.1	自然语言表示	3
1.2.2	流程图表示	3
1.2.3	N-S 图表示	4
1.2.4	伪代码表示	4
1.2.5	程序语言表示	5
1.3	算法的设计与评价	6
1.3.1	评价算法的标准	6
1.3.2	算法的环路复杂度	7
1.3.3	算法的时空效率	8
1.3.4	常见的算法设计方法	11
1.4	算法与程序	14
1.4.1	程序的基本概念	14
1.4.2	问题求解与实现策略	15
1.4.3	程序调试与查错策略	16
1.4.4	程序设计方法概述	18
习题		22
第 2 章	常用数据结构	23
2.1	数据类型与数据结构	23
2.1.1	数据、数据元素与数据类型	23
2.1.2	数据结构的基本概念	25
2.1.3	抽象数据类型	27



2.2 数组	29
2.2.1 数组及其运算	29
2.2.2 数组的顺序存储结构	30
2.2.3 特殊矩阵的压缩存储	32
2.3 串	34
2.3.1 串的基本概念	35
2.3.2 串的定长顺序存储及运算实现	36
2.3.3 模式匹配	40
2.3.4 串的堆式动态存储及运算实现	44
2.3.5 汉字串	47
习题	50
上机实验题	52
第3章 简单数据结构	53
3.1 顺序表	53
3.1.1 线性表的基本概念	53
3.1.2 线性表的顺序存储结构——顺序表	54
3.1.3 顺序表上的基本运算	55
3.2 链表	59
3.2.1 线性表的链式存储结构 链表	59
3.2.2 单链表上的基本运算	61
3.2.3 循环链表和双向链表	66
3.2.4 线性表应用举例——一元多项式相加问题	68
3.3 栈	70
3.3.1 栈的概念及运算	70
3.3.2 顺序栈及运算实现	71
3.3.3 链栈及运算实现	74
3.3.4 栈的应用举例——递归的实现	75
3.4 队列	78
3.4.1 队列的概念及其运算	78
3.4.2 顺序队列及运算实现	79
3.4.3 链队列及运算实现	82
3.4.4 队列的应用举例——I/O 缓冲区管理及其他	84
3.5 广义表	86
3.5.1 广义表的概念	86
3.5.2 广义表的存储结构及运算实现	87
3.5.3 广义表的应用举例——m 元多项式的表示	89
习题	91

上机实验题	94
第4章 树与二叉树	95
4.1 树的基本概念	95
4.1.1 树的定义及表示	95
4.1.2 树的常用术语及运算	96
4.2 二叉树	98
4.2.1 二叉树的概念	98
4.2.2 二叉树的性质	99
4.2.3 二叉树的存储结构	101
4.2.4 二叉树的简单运算实现	103
4.3 二叉树的遍历	104
4.3.1 遍历二叉树的递归算法	104
4.3.2 遍历二叉树的非递归算法	106
4.3.3 遍历序列与二叉树的复原	110
4.3.4 基于遍历的几种二叉树运算的实现和应用举例	112
4.4 线索二叉树	114
4.4.1 线索二叉树的概念	114
4.4.2 线索二叉树的构造算法	115
4.4.3 线索二叉树上的运算实现	116
4.5 树和森林	118
4.5.1 树和森林的存储结构	118
4.5.2 树和森林与二叉树之间的转换	120
4.5.3 树和森林的遍历	122
4.5.4 树的应用举例——判定树	123
4.6 哈夫曼树	124
4.6.1 哈夫曼树的概念及其构造算法	124
4.6.2 哈夫曼树的应用——哈夫曼编码	126
习题	128
上机实验题	131
第5章 图与网	132
5.1 图与网的基本概念	132
5.1.1 图与网的定义	132
5.1.2 图的相关术语	133
5.2 图与网的存储结构	135
5.2.1 邻接矩阵	135



5.2.2	邻接表与逆邻接表	136
5.2.3	邻接多重表	138
5.3	图的遍历	139
5.3.1	深度优先搜索遍历	140
5.3.2	广度优先搜索遍历	141
5.3.3	图的遍历应用举例——图的连通性与生成树	142
5.4	无向连通网的最小生成树	144
5.4.1	最小生成树的概念	144
5.4.2	Prim 算法	144
5.4.3	Kruskal 算法	147
5.5	有向网的最短路径	148
5.5.1	单源最短路径	148
5.5.2	所有顶点对之间的最短路径	150
5.6	有向无环图及其应用	152
5.6.1	有向无环图的概念	152
5.6.2	AOV 网与拓扑排序	153
5.6.3	AOE 网与关键路径	158
习题		163
上机实验题		165
第 6 章	数据结构的程序实现	166
6.1	基本的实现策略	166
6.1.1	简单数据结构的程序实现	166
6.1.2	构造型数据结构的程序实现	167
6.1.3	数据结构的链式实现	167
6.1.4	数据结构的数组实现	167
6.2	动态结构的静态实现	167
6.2.1	静态链表	168
6.2.2	二叉树的静态二叉链表表示法	169
6.2.3	树和森林的双亲表示法	169
6.2.4	哈夫曼算法的静态实现	170
6.3	大批量数据的组织策略	174
6.3.1	文件的组织	175
6.3.2	数据库技术	181
6.4	数据结构在问题建模中的应用	184
6.4.1	Josephus 问题	184
6.4.2	教务管理与二分图	187
6.4.3	学籍管理系统中的数据组织	190

习题	195
上机实验题	195
第 7 章 检索及基本算法	196
7.1 检索的概念	196
7.2 线性表的检索	198
7.2.1 顺序检索	198
7.2.2 二分法检索	199
7.2.3 黄金分割点检索	202
7.2.4 精算点检索	204
7.2.5 分块检索	206
7.3 树表的检索	208
7.3.1 二叉检索树	208
7.3.2 二叉检索树的平衡性调整	215
7.3.3 B 树和 B ⁺ 树	219
7.4 哈希检索	221
7.4.1 哈希检索与哈希表	222
7.4.2 哈希函数的构造方法	222
7.4.3 地址冲突的消解策略	225
7.4.4 哈希表的检索算法及性能分析	226
习题	228
上机实验题	231
第 8 章 排序及基本算法	232
8.1 排序的基本概念	232
8.2 插入排序	233
8.2.1 直接插入排序	234
8.2.2 希尔排序	235
8.2.3 其他插入排序简介	238
8.3 交换排序	241
8.3.1 冒泡排序	241
8.3.2 快速排序	243
8.4 选择排序	246
8.4.1 直接选择排序	246
8.4.2 树形选择排序	247
8.4.3 堆排序	248
8.5 归并排序	252



8.5.1 归并相邻两个有序序列	253
8.5.2 二路归并排序的递归算法	253
8.5.3 二路归并排序的非递归算法	254
8.6 基数排序	255
8.6.1 多关键字排序	255
8.6.2 链式基数排序	256
8.7 各种内部排序方法的比较和选择	259
8.8 外部排序简介	261
习题	263
上机实验题	265
参考文献	266

算法与程序

我们开篇讲算法与程序,是因为算法与程序既是计算机科学与技术学科的基础,又是计算机科学与技术学科永恒的主题。程序设计是算法与数据结构课程的基础,也是算法与数据结构课程的归宿。算法与数据结构以高级语言程序设计和离散结构作为先导课程,又服务于程序设计与软件开发;它主要研究的是复杂程序(或算法)中的数据的结构、组织和管理技术,以及复杂程序(或算法)的设计问题。

本章主要介绍算法的基本概念、特性、表示、评价和设计方法,并简要介绍程序的概念、特性、与算法的关系以及对实际问题的求解方法。

1.1 算法的基本概念

算法是计算学科的核心概念。研究算法能使我们深刻地理解问题的本质并进而找到可能的求解技术。算法设计的优劣决定着程序以至软件系统的性能。无论用计算机解决哪一方面的问题,都必须设法用数学方法来描述或模拟这些实际问题,把对实际问题的可行解决方案归结为计算机能够执行的若干个步骤,然后再把这些步骤用一组计算机指令进行描述形成所谓的程序,最后交给计算机去执行。

1.1.1 什么是算法

早在公元前 300 年左右出现的著名的欧几里德算法,就描述了求解两个整数的最大公因子的解题步骤。要求解的问题描述为:“给定两个正整数 m 和 n ,求它们的最大公因子,即能同时整除 m 和 n 的最大整数”。欧几里德当时给出的算法如下:

- ① 以 n 除 m ,并令所得余数为 r (必有 $r < n$)。
- ② 若 $r=0$,输出结果 n ,算法结束;否则继续步骤③。
- ③ 令 $m=n$ 和 $n=r$,返回步骤①继续进行。

由此,可以得出这样的结论,算法就是求解问题的方法和步骤。这里的方法和步骤是一组严格定义了运算顺序的规则;每一个规则都是有效的,且是明确的;按此顺序算法将在有限次数下终止。

有关算法(Algorithm)一词的定义不少,但其内涵基本上是一致的。最为著名的定

义是计算机科学家 D. E. Kunth 在其巨著《计算机程序的艺术》(Art of Computer Program)第一卷中所做的有关描述。其非形式化的定义是:

一个算法,就是一个有穷规则的集合,其中之规则定义了一个解决某一特定类型问题的运算序列。

算法的形式化定义如下所述:

算法是一个四元组,即 (Q, I, Ω, F) 。其中:

① Q 是一个包含子集 I 和 Ω 的集合,它表示计算的状态;

② I 表示计算的输入集合;

③ Ω 表示计算的输出集合;

④ F 表示计算的规则,它是由 Q 至它自身的函数,且具有自反性,即对任何一个元素 $q \in Q$,有 $F(q) = q$ 。

一个算法是对于任何的输入元素 x ,都在有穷步骤内终止的一个计算方法。在算法的形式化定义中,对任何一个元素 $x \in I$, x 均满足性质:

$$x_0 = x, x_{k+1} = F(x_k), (k \geq 0)$$

该性质表示任何一个输入元素 x 均为一个计算序列,即 $x_0, x_1, x_2, \dots, x_k$; 该序列在第 k 步结束计算。

1.1.2 算法的基本特性

为了读者能更准确地理解和掌握算法定义的内涵,下面给出算法的 5 个重要特性。

① 输入(Input):一个算法应该有 0 个或多个输入。对于数值计算类问题,算法一定会有相应的输入数据;而对于非数值计算类问题,算法则不一定有输入数据,如制表画线等。

② 输出(Output):一个算法应该有一个或多个输出。设计算法的目的是为了解决给定的问题,给定问题的解是算法最终的输出结果。没有输出结果的算法是没有任何意义的。

③ 确定性(Definiteness):算法的每一个步骤必须要有确切的定义。也就是说,算法中所有有待执行的动作必须严格地规定和描述,不能模棱两可含混不清,不能存在歧义性(也称二义性,即有不同的解释)。如欧几里德算法中步骤①明确规定是“以 n 除 m ”,而不能是“以 n 除 m 或以 m 除 n ”这类有两种可能做法的规定。

④ 有穷性(Finiteness):一个算法在执行有穷步骤之后必然会结束。也就是说,算法是由有序的操作序列组成的,它所包含的计算步骤是有限的。如在欧几里德算法中,由于在步骤①之后必有 $r < n$,步骤③中令 $n=r$ 后下一次执行步骤①时已使 n 的值减少,这种正整数的递降序列使得计算过程最后必然会终止。因此,不论给定 m 和 n 的初始正整数值有多大,步骤①的执行都是有穷次的。

⑤ 有效性(Effectiveness):组成算法的每一个操作都应该是在特定环境下允许使用的、可以执行的,并能在有限时间内完成,最后得出确定的结果。只要算法中有一个操作是不可执行的,整个算法就不具有有效性。

1.2 算法的表示

算法是问题求解方法及过程的精确描述,描述算法的工具直接影响着算法的质量。算法的表示方法很多,下面介绍几种常用的算法表示方法。

1.2.1 自然语言表示

自然语言即人们日常使用的语言,如汉语、英语、日语、法语、德语等。使用自然语言描述算法,人们比较容易接受和理解。如前面的欧几里德算法就是用自然语言描述的。然而,自然语言也具有许多缺点,在使用自然语言描述算法时一定要注意以下几点:

- ① 自然语言存在着歧义性,容易导致算法的不确定性。
- ② 自然语言容易冗长,使得描述不够简洁。
- ③ 自然语言的表示形式是顺序的,描述分支选择和转移时不够直观。
- ④ 自然语言与计算机程序设计语言的差别较大,不易转换为程序。

1.2.2 流程图表示

流程图是描述算法的图形工具,它采用如下所示的一组图形符号来表示算法。

- 起止框:表示算法的开始或结束;只有一个入口或一个出口。
- ▭ 输入输出框:表示算法中数据信息的输入和输出;有一个入口和一个出口。
- ◇ 判断框:表示条件判断;有一个入口和多个出口。
- ▭ 处理框:表示算法中的一个(或一组)运算或处理;有一个入口和一个出口。
- 流程线:表示算法中各步骤之间的次序关系。
- 连接点:表示算法中的连接位置,主要用于同一算法在不同页描述时的接续等情况。
- ▭ 注释框:用于对算法中某些操作的注释说明。

欧几里德算法的流程图描述如图 1-1 所示。同自然语言相比,用流程图描述算法直观,可以一目了然;算法步骤间用流程线连接,次序关系清楚,容易理解;可以很方便地表示顺序、选择和循环结构,不依赖于任何计算机和计算机程序设计语言,有利于不同环境下的程序设计。但是,流程图也存在一些缺点,诸如:

- ① 不易于表示算法的层次结构。

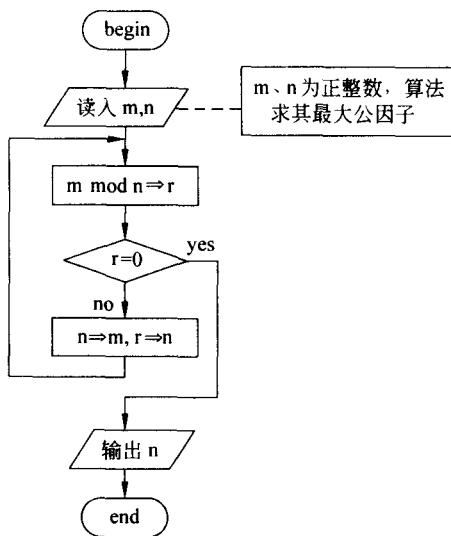


图 1-1 欧几里德算法的流程图表示