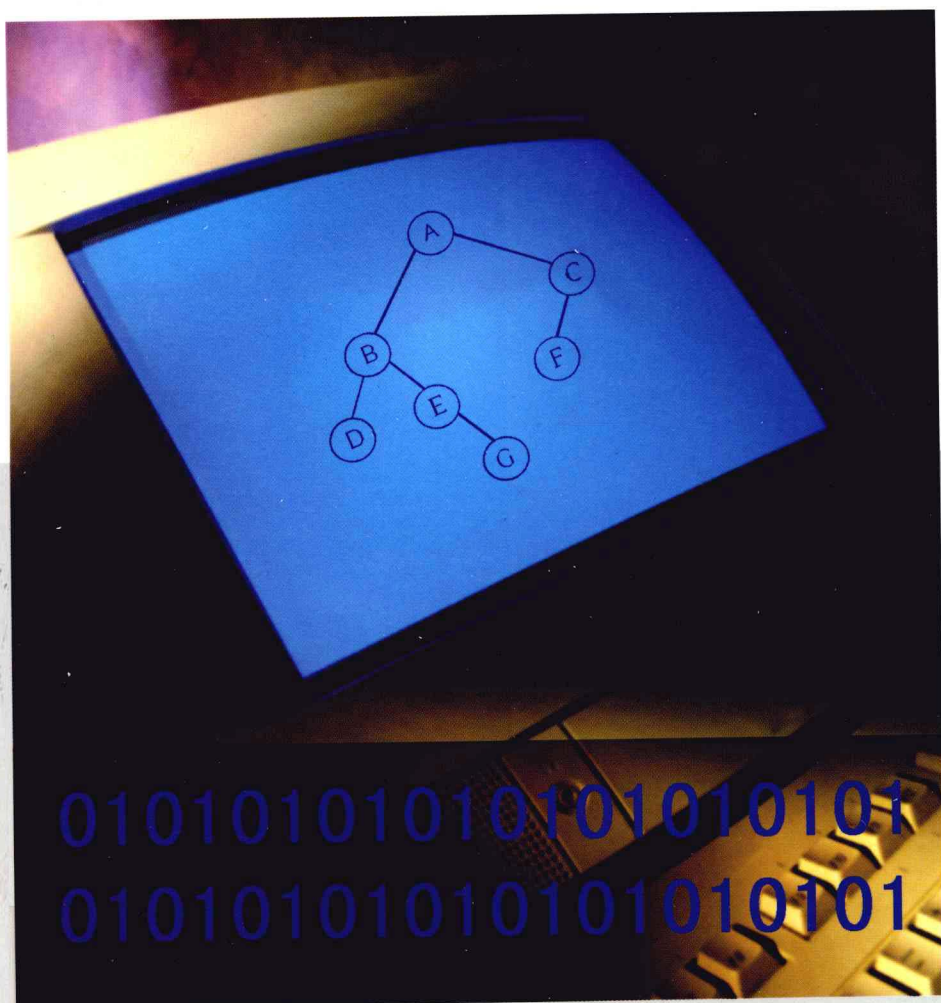


新世纪高等学校计算机系列教材

数据结构

杨薇薇 张胜利 殷贤亮 编



高等教育出版社
中山大学出版社

策划：湖北省计算机学会·诺亚教科文中心

新世纪高等学校计算机系列教材

数 据 结 构

杨薇薇 张胜利 殷贤亮 编

高等教育出版社·北京

中山大学出版社·广州

图书在版编目(CIP)数据

数据结构 / 杨薇薇, 张胜利, 殷贤亮编. —北京: 高等教育出版社. —广州: 中山大学出版社, 2004. 7
(新世纪高等学校计算机系列教材/湖北省计算机学会, 诺亚教科文中心 策划)
ISBN 7-306-02312-8

I. 数… II. ①杨… ②张… ③殷… III. 数据结构—高等学校—教材 IV. TP311.12

中国版本图书馆 CIP 数据核字(2004)第 057293 号

内 容 简 介

本书根据中国计算机学会、全国高等学校计算机教育研究会等联合推出的《CCC2002 教程》中数据结构课程大纲的基本要求, 较全面系统地介绍了各种典型数据结构的特点、原理与方法。全书共分 11 章, 主要内容包括线性表、栈和队列、数组、树、图、查找、排序、文件组织和若干供选用的实验等。并且, 书中对各种典型数据结构及其算法均用类 C 语言给出了完整而又准确的描述。

本书内容全面适用, 叙述通俗易懂, 例、习题丰富典型, 各种概念与算法描述准确。并且, 为便于教学与初学者自学使用, 在本书的附录中还给出了绝大部分习题的解题思路与参考答案, 供教学者参考。

本书适合新世纪各类高等学校计算机、通信、电子、自动化、信息安全等 IT 类企业, 以及管理类与财经类有关专业作为教材使用, 亦适合广大科技人员自学参考。

版权所有 盗印必究

数据结构

©杨薇薇 张胜利 殷贤亮 编

责任编辑: 里 引 唐 源

封面设计: 袁 作

责任校对: 诺 亚

责任技编: 潘 隆

出版发行: 高等教育出版社

(地址: 北京市西城区德外大街 4 号

邮编: 100011)

中山大学出版社

(地址: 广州市新港西路 135 号

邮编: 510275)

经 销: 广东新华发行集团公司

武汉诺亚信息传播有限公司 (电话: 027—87597347(带传真) 87554561 邮编: 430073)

印 刷: 华中理工大学印刷厂

开 本: 787mm×1092mm

1/16

印 张: 21.5

字 数: 530 千字

版 次: 2004 年 7 月第 1 版

印 次: 2004 年 7 月第 1 次印刷

印 数: 1—5 000 册

定 价: 29.00 元

《新世纪高等学校计算机系列教材》

编审指导委员会

主 任：卢正鼎(华中科技大学教授、博士生导师)

副 主 任：何炎祥(武汉大学教授、博士生导师)

编 委：(以姓氏笔画排序)

王元珍(华中科技大学教授、博士生导师)

叶俊民(华中师范大学副教授、博士)

李 兵(湖北大学副教授、博士)

李鸣山(武汉大学教授)

陈 珉(武汉大学教授、博士)

陈传波(华中科技大学教授、博士生导师)

陈建勋(武汉科技大学教授、博士)

陆际光(中南民族大学教授)

汪厚祥(海军工程大学教授、博士)

程元斌(江汉大学副教授)

程学先(湖北工业大学教授)

谭连生(华中师范大学教授、博士)

熊家军(空军雷达学院教授、博士)

戴光明(中国地质大学教授、博士)

执行编委：唐元瑜(华中科技大学副编审)

余健棠(华中科技大学编审)

《新世纪高等学校计算机系列教材》

总 序

21 世纪人类已跨入了信息时代,以计算机为核心的信息技术正在迅猛发展,并不断改变着人类社会的工作方式、生产方式、生活方式和学习方式。当今,各行各业的现代化都离不开计算机,各行各业的人们都在学习和使用计算机,而计算机科学技术及其教育本身也在日新月异地发展变化。为了顺应时代的潮流,满足新世纪高等学校计算机教育事业、教学改革和人才培养对高质量特色教材的需求,湖北省计算机学会及其教育与培训专业委员会和诺亚教科文中心等共同策划、组织并约请华中科技大学、武汉大学、华中师范大学、中国地质大学、中南民族大学、武汉科技大学、海军工程大学、空军雷达学院、湖北大学、湖北工业大学、江汉大学及其他高校长期奋斗在教学科研第一线,且具有丰富教学实践经验的部分优秀骨干教师共同编写了这套计算机系列教材。

这套系列教材共 40 余种,主要是根据中国计算机学会教育委员会、全国高等学校计算机教育研究会等联合推出的《中国计算机科学与技术学科教程 2002》(简称《CCC2002 教程》)中的课程体系与课程大纲的要求,进行规划和组织编写的,主要供高等学校计算机科学与技术专业本科或研究生教学使用。对于本系列教材中部分供软件学院等使用的教材,大都是一些理论与实践并重,从内容到结构都是全新的特色教材,并且有些教材也可以适合其他专业选用。

当今,计算机科学技术突飞猛进地向前发展,计算机新技术和新产品不断涌现,高等教育事业和教学改革不断深化,国内教育逐步与国际教育接轨,社会对计算机专业人才的要求越来越高,等等。面对这些新形势,这套系列教材以培养学生具有较扎实的专业基础理论知识、实践能力、创新能力和较高的综合素质能力为目的,既注重知识的更新与合理的结构,又注意学习和汲取国内外优秀教材的优点与精华,并尽力反映国内外最新的教学科研成果及作者们宝贵的实践经验。

我相信,通过作者们的共同努力,定能将这套系列教材打造成为一套既具有时代特色,又非常适用的、高质量的系列教材,为我国高等教育事业的发展和高素质专业人才的培养作出应有的贡献。

《新世纪高等学校计算机系列教材》

编审指导委员会主任

卢正鼎

2003 年 7 月

前 言

数据结构是各类高等学校计算机类各个专业的核心课程,也是新世纪高等学校通信、电子、自动化,以及管理与财经类等有关专业的必修课程。数据结构为研究和解决非数值性问题提供了重要的理论和方法,是非数值性程序设计的基础。

本书详细介绍了基本的数据结构及其应用,讨论了各种结构的逻辑特性和它们在存储器中的物理表示,为每种结构定义了基本运算,对部分运算给出了相应的算法,并对算法的复杂度进行了适当的分析。本书的学习目的是:学会分析待处理数据的特性,以便为给定问题建模,选择适当的数据结构和设计出“好”的算法。

全书分为 11 章。具体安排是:第 1 章介绍数据结构、算法及算法分析的基本概念;第 2~5 章分别介绍线性表、堆栈、队列、串和数组几种典型的线性数据结构;第 6、7 章介绍树和图,即层次和网状的非线性数据结构;第 8、9 章分别介绍查找和排序的基本原理和方法;第 10 章围绕数据在外存储器上的组织,介绍常用的文件结构;第 11 章给出了 10 个实验,供任课教师选用。各章后面都配有一定数量的习题,并在附录中给出了绝大部分习题的参考答案或解题思路,供教学者参考。

本书采用类 C 语言(类似于 C 语言,但又不完全等同于 C 语言的一个子集)作为数据结构和算法的描述语言。考虑到 C 语言数组的特色,书中利用数组的动态分配实现顺序表、顺序栈和顺序队列的物理结构。为便于算法的描述,在书中还引入了 C++ 语言的引用调用参数,故使得函数的调用方式除原有的值调用外,又增添了地址调用。全书突出了抽象数据类型这一概念,为读者进一步学习面向对象的程序设计和使用 C++ 语言编程打下良好的基础。

作为系列教材之一,本书在内容的选取方面符合“IEEE-CS ACM Computing Curricula 2001”和“China Computing Curricula 2002”对数据结构部分知识点的要求;在概念的引入和文字的表述方面都遵循了既科学准确,又通俗易懂,便于学习理解这一原则;在例题和习题的选择方面注重面向应用、理论与实践相结合。因此,该书是一本较系统阐述数据结构的原理与方法,并且概念准确、通俗易懂和颇为适用的好教材。

本书是编者们在总结多年从事数据结构、C 语言研究和教学实践经验体会的基础上,参考了国内外相关资料而写成的。该书内容丰富、层次清晰、实用性很强,既适合新世纪各类高等院校计算机、电子、通信、自动化、电子信息、光信息、信息安全,以及管理与财经类等有关专业本科及专科作为教材使用,也适合广大科技人员和自学人员学习参考。本书授课学时可为 40~80 学时。当用于不同对象和讲授不同的学时时,建议讲授内容可按以下方法进行裁剪:在学时较少或专科教学的情况下选讲或不讲书中带“*”的章节;用于非计算机及专科类专业教学时甚至还可以不讲第 4 章和第 10 章的内容。

本书第 1、2、3、4、5、10 章和第 11 章的大部分内容 by 华中科技大学杨薇薇副教授编写,第 6、7 章和第 11 章的一部分由湖北工业大学张胜利副教授编写,第 8、9 章由华中科技大学殷贤亮副教授编写,附录由三人合写。全书由杨薇薇主编并修改定稿。在本书的编写过程中得到了湖北省计算机学会及其教育与培训专业委员会、华中科技大学、湖北工业大学,以及《新世

纪高等学校计算机系列教材》编审指导委员会等有关专家的大力支持与帮助,在此一并表示感谢。

由于编者的水平有限,书中不妥或错误之处在所难免,敬请广大读者批评指正。

编 者

2004 年 5 月

目 录

第 1 章 绪论	(1)
1.1 数据结构的产生和发展	(1)
1.2 什么是数据结构	(2)
1.3 基本术语	(3)
1.4 算法和算法的分析	(5)
1.4.1 算法	(5)
1.4.2 算法的描述	(5)
1.4.3 算法分析	(6)
本章小结	(10)
习题一	(10)
第 2 章 线性表	(12)
2.1 线性表及其基本运算操作	(12)
2.2 线性表的顺序存储结构——顺序表	(13)
2.2.1 顺序表	(13)
2.2.2 顺序表的存储定义和基本运算	(14)
2.3 线性表的链式存储结构——单链表	(21)
2.3.1 单链表	(21)
2.3.2 静态单链表	(26)
2.3.3 单向循环链表	(31)
2.4 线性表的链式存储结构——双向链表	(34)
2.4.1 双向链表	(34)
2.4.2 双向链表的基本运算及其实现	(35)
2.5 单链表的应用实例	(38)
2.5.1 多项式相加	(38)
2.5.2 集合的运算	(42)
本章小结	(44)
习题二	(44)
第 3 章 栈和队列	(49)
3.1 栈及其运算	(49)
3.2 栈的顺序存储结构	(51)
3.2.1 顺序栈及其运算	(51)
3.2.2 两个栈共享存储空间	(54)
3.3 栈的链式存储结构	(56)
3.4 栈的应用举例	(57)
3.4.1 算术表达式求值	(57)

3.4.2 数制的转换问题	(59)
3.4.3 括号匹配的检测	(60)
3.4.4 栈与递归	(61)
3.5 队列	(65)
3.5.1 队列及其运算	(65)
3.5.2 队列的链式存储结构及其运算	(66)
3.5.3 队列的顺序存储结构及其运算	(69)
* 3.5.4 其他队列	(73)
3.6 队列的应用	(74)
本章小结	(78)
习题三	(78)
第4章 数组及其应用	(82)
4.1 数组及其顺序存储结构	(82)
4.1.1 数组	(82)
4.1.2 数组的顺序存储结构	(83)
4.2 矩阵的压缩存储	(85)
4.2.1 特殊矩阵的压缩存储	(85)
4.2.2 稀疏矩阵的压缩存储	(87)
本章小结	(93)
习题四	(93)
第5章 字符串	(95)
5.1 串和串的主要运算	(95)
5.1.1 串的基本概念	(95)
5.1.2 串的主要运算及其函数	(96)
5.2 串的存储结构及其基本运算的实现	(98)
5.2.1 串的定长顺序存储表示	(98)
5.2.2 串的堆分配存储表示	(100)
5.2.3 串的块链存储表示	(103)
* 5.3 串的模式匹配及其算法	(105)
5.4 串的应用实例——文本编辑	(109)
本章小结	(110)
习题五	(111)
第6章 树与二叉树	(114)
6.1 树	(114)
6.1.1 树的基本概念	(114)
6.1.2 树的存储表示	(116)
6.2 二叉树	(119)
6.2.1 二叉树的定义及运算	(119)
6.2.2 二叉树的性质	(120)

6.2.3 二叉树的存储表示	(123)
6.3 二叉树的遍历	(126)
6.3.1 二叉树的先(根)序遍历	(127)
6.3.2 二叉树的中(根)序遍历	(129)
6.3.3 二叉树的后(根)序遍历	(130)
6.3.4 层次序遍历	(132)
6.3.5 二叉树遍历算法的应用	(133)
6.4 线索二叉树	(136)
6.4.1 线索化二叉树	(137)
6.4.2 线索二叉树的遍历	(141)
6.5 树、森林与二叉树的转换及遍历	(143)
6.5.1 树与二叉树的转换	(144)
6.5.2 森林与二叉树的转换	(145)
6.5.3 树、森林的遍历	(147)
6.6 哈夫曼树及其应用	(147)
6.6.1 哈夫曼树	(148)
6.6.2 哈夫曼算法	(148)
6.6.3 哈夫曼编码	(151)
本章小结	(156)
习题六	(157)
第 7 章 图	(159)
7.1 图的基本概念	(159)
7.1.1 图的定义及运算	(159)
7.1.2 图的基本术语	(160)
7.2 图的存储表示	(162)
7.2.1 邻接矩阵表示法	(162)
7.2.2 邻接表表示法	(164)
7.2.3 图的其他存储表示法	(166)
7.2.4 图的存储结构的建立	(168)
7.3 图的遍历	(171)
7.3.1 深度优先遍历	(172)
7.3.2 广度优先遍历	(175)
7.3.3 求图的连通分量	(177)
7.4 图的生成树和最小生成树	(177)
7.4.1 无向连通图的生成树	(177)
7.4.2 网络的最小生成树	(178)
7.5 有向无环图及其应用	(183)
7.5.1 AOV 网及其拓扑排序	(183)
7.5.2 AOE 网及其关键路径	(187)
* 7.6 最短路径问题	(192)

7.6.1 单源最短路径问题	(192)
7.6.2 每对顶点间的最短路径问题	(195)
本章小结	(198)
习题七	(199)
第8章 查找	(201)
8.1 查找的基本概念	(201)
8.2 线性顺序表的静态查找	(202)
8.2.1 顺序查找	(202)
8.2.2 线性顺序表的二分查找	(203)
8.2.3 分块查找	(206)
8.3 二叉排序树的动态查找	(207)
8.3.1 二叉排序树	(208)
8.3.2 平衡二叉树	(213)
* 8.4 B ⁻ 树和 B ⁺ 树	(219)
8.4.1 B ⁻ 树及其查找	(219)
8.4.2 B ⁻ 树的插入和删除	(220)
8.4.3 B ⁺ 树	(222)
8.5 哈希表及其查找	(223)
8.5.1 概述	(223)
8.5.2 哈希函数的构造方法	(224)
8.5.3 处理冲突的方法	(226)
8.5.4 哈希表的查找	(227)
本章小结	(229)
习题八	(230)
第9章 排序	(232)
9.1 排序的基本概念	(232)
9.2 简单排序	(233)
9.2.1 直接插入排序	(233)
9.2.2 冒泡排序	(234)
9.3 堆排序	(235)
9.4 归并排序	(240)
9.5 基数排序	(242)
9.6 快速排序	(244)
9.7 各种内部排序的比较	(246)
* 9.8 外部排序	(247)
9.8.1 2-路归并	(247)
9.8.2 多路替代选择归并排序	(249)
9.8.3 最佳归并排序	(250)
本章小结	(251)

习题九.....	(251)
第 10 章 文件	(253)
10.1 文件的概念.....	(253)
10.2 顺序文件.....	(255)
10.3 索引文件.....	(258)
10.4 ISAM 文件	(260)
10.5 VSAM 文件	(262)
10.6 直接存取文件.....	(264)
10.7 多关键字文件.....	(265)
10.7.1 多重表文件	(265)
10.7.2 倒排文件	(266)
本章小结.....	(267)
习题十.....	(267)
第 11 章 实验	(271)
实验一 线性表和数组的应用.....	(271)
实验二 链表的应用与操作.....	(271)
实验三 数组和栈的应用.....	(272)
实验四 字符串的存储和运算.....	(273)
实验五 二叉排序树的建立与遍历.....	(274)
实验六 哈夫曼编/译码器的设计	(274)
实验七 图的存储操作与应用.....	(275)
实验八 构造 Hash 表	(276)
实验九 排序的应用.....	(277)
实验十 多关键字排序的应用.....	(277)
附件 实验报告书写规范.....	(278)
附录 习题参考答案	(280)
参考文献	(331)

第 1 章

绪 论

计算机是处理信息的机器。数据结构不仅研究这些信息的数学性质,也研究如何在计算机中有效地存储和处理这些信息。当今,随着计算机信息量的增加、信息范围的拓宽和信息结构复杂化的加深,为了编写出高质量的程序,还必须分析这些信息的特征以及它们之间存在的关系。程序设计的实质就是为确定的问题选择一种适当的数据结构并设计一个好的算法。因此,数据结构在程序设计中是至关重要的。

本章介绍数据结构的基本概念与基本术语。要求读者牢固掌握这些概念和术语,为后续章节的学习与理解打下基础。

1.1 数据结构的产生和发展

数据结构是随着电子计算机的产生和发展而发展起来的一门计算机学科。最近 20 年来,电子计算机技术飞速发展,这不仅体现在计算机本身运算速度不断提高、信息存储量日益扩大,而且更重要的是其应用范围的扩展。

早期的电子计算机主要用于科学计算,所处理的对象是纯数值性的信息。虽然这类问题的解题算法比较复杂,但数据量较少,结构也简单。因此,早期的计算机科学以研究程序及所描述的算法为中心。

目前,计算机已广泛地应用于情报检索、事务管理、系统工程等领域。与此相应,计算机加工处理的对象也从简单的纯数值性信息发展到数、字符、字符串、表、文件、图像、声音等各种复杂的、具有一定结构的数据。人们称前者为数值问题,而称后者为非数值问题。

非数值问题要求用复杂的数据结构来描述系统的状态,其主要的运算是实现对数据结构的访问和修改。要设计出效率高、可靠性强的非数值程序,则要求程序设计人员不但要掌握一般的程序设计技巧,而且还必须研究计算机程序加工的对象,即研究各种数据的特性以及数据之间的关系,这就促进了数据结构这一学科的发展。然而,数据必须在计算机中进行处理,因此,不仅要考虑数据本身的数学性质,还必须考虑数据在计算机内的存储方式和相应的运算,从而扩大了数据结构研究的范围。随着数据库系统、情报检索系统的不断发展,在数据结构的技术中又增加了文件结构,特别是增加了大型文件的组织和 B^- 、 B^+ 树的知识,使得数据结构逐步成为了一门比较完整的学科。

数据结构是计算机类各专业及其他很多相关专业本、专科学生必修的课程,是研究计算机程序设计的重要理论和技术的专业基础课程。

1.2 什么是数据结构

为了要知道什么是数据结构,下面先举一个简单的例子予以说明,然后再给出其明确的定义。

例 1.1 假定有一个职工通信录,记录了某单位全体职工的姓名和相应的住址。现要求设计一个算法:当给定某一位职工的姓名时,计算机能够查出该职工的住址,如果根本就没有这个人,则计算机也能报告“查无此人”。

这种任务称为“查找”。从题意可以发现,这个算法的设计完全依赖于通信录中职工姓名和相应住址等信息在计算机内的存储方式(或方法)。

一种方法是,通信录中职工的姓名是随意排列的,其次序没有任何规律。那么,当给定一个姓名时,就只能从通信录的第 1 个姓名开始,逐个与给定的姓名进行比较,直至找到指定的姓名,接着打印出他的住址,或者查完整个通信录还没找到此人,这时给出相应的“查无此人”的标志。这种方法很简单,当职工人数较少时是可行的。但对于一个大单位来说,这就相当费时间,效率太低。

另一种方法是,对职工通信录进行适当的组织,如图 1.1(b)所示,即按字母的顺序排列姓名和相应的住址,而且还可以再构造一个索引表,用这个表来登记姓名中每个具有相同开头字母的第 1 个姓名在通信录中的起始位置,见图 1.1(a)。这样,情况可大为改善,即:若查找某个职工,例如,查“Min Rui”的住址,则可先从索引表中找到以“M”开头的姓名的起始位置,然后,从此位置开始往下顺序查找,而不必去查看其他字母开头的职工姓名。由于采用了新的结构,因而可以设计出一个完全不同的算法。

姓名		住址	
1	Ai Xiaoming 艾小明	中山路	1036 号
11	An Jing 安 静	中北路	128 号
	⋮	⋮	
181	Ban Lili 班莉莉	南三街	72 号
191	Bao Changqing 包长青	东一路	89 号
201	Bi Zhenghua 毕正华	北大街	101 号
	⋮	⋮	
9071	Ma Xiaoshan 马小山	西小巷	51 号
9081	Mi Gu 米 谷	东方大道	甲 1 号
9091	Min Jie 闵 洁	大红门	3 号
9101	Min Rui 闵 锐	南市口	227 号
	⋮	⋮	

索引	指针
A	1
B	181
⋮	
M	9071
⋮	

(a) 索引表

(b) 职工通信录

图 1.1 职工通信录在计算机中的存储示意

上述职工通信录的组织方式问题就是一个数据结构问题。两种不同的数据结构,可得出两个完全不同的算法。由此可见,计算机的算法与数据结构密切相关,即每一个算法无不依赖于具体的数据结构,而数据结构直接影响着算法的选择和算法的效率。

下面对例 1.1 所述职工通信录作进一步讨论。当有新职工调入时,通信录中需要增添新职工的姓名和住址;当有职工调离时,应从通信录中删除调离职工的姓名和住址。这就要求在原有的通信录上进行插入(Insert)和删除>Delete)运算。对于一种已生成的结构,如何实现插入和删除运算?即把要增添职工的姓名和住址插入到表头还是表尾,或者是中间某个合适的位置上,这要由具体的问题和所用的数据结构决定。此外,插入后,对原有的数据是否有影响?有什么样的影响?删除某职工的姓名和住址以后,其他职工的姓名和住址在通信录中的位置是否要跟着变动?若需要变动,又应如何进行?这一系列的问题,说明为适应数据的增减,还必须对每种数据结构定义一套运算。并且,这里只涉及了插入和删除运算,而在实际问题中,还会提出一些其他的运算。例如,某职工住址改变了,为适应这种需要还要定义修改(Modify)运算等。也就是说,数据结构还必须给出适用于每种结构类型所定义的各种运算。

通过以上讨论,可以广义地认为:数据结构是一门研究程序设计中计算机操作的对象(数据)以及它们之间的关系和运算的技术。这也是本书所要介绍的主要内容。

1.3 基本术语

数据结构中常用的名词和术语主要有:

1) 数据(Data)

计算机中的数据是广义的,包括数、字符、字符串、表、文件、声音、图形、图像等,即所有能输入到计算机中被计算机加工处理的信息都属于数据的范畴。

2) 数据元素(Data element)

数据元素是数据的基本单位,即数据集合中的个体,在计算机程序中通常作为一个整体来考虑和处理。每个数据元素可以只有一个数据项,通常称为域(Field),也可以由若干个数据项组成。例如,学生信息管理系统中,每个学生的情况为一个数据元素,而其中的学号、姓名、性别、年龄等信息分别称为数据项。数据项是数据的最小单位。数据元素的同义语有结点(Node)、顶点(Vertex)和记录(Record)等。

3) 数据对象(Data object)

数据对象是性质相同的数据元素的集合,它是数据的一个子集。例如,整型数据对象是集合 $\{0, \pm 1, \pm 2, \dots\}$ 。数据对象可以是有限的,也可以是无限的。

4) 数据结构(Data structure)

数据结构是相互之间存在一种或多种特定关系(即结构)的数据元素的集合,这是人们对数据结构的一种狭义解释。

由于同一数据对象中的数据元素都不是孤立的,而是彼此相关的,因此,数据结构所研究和解决的问题是:数据元素之间各种抽象化的相互关系(即结构)和这种关系在计算机中的存储表示,并对每种结构定义出各自的运算和设计出相应的算法。通常所说的数据结构与数据对象不同,在描述一种数据结构时,不但要描述数据对象,还要描述数据元素之间的相互关系。

严格地说,数据的结构包括两个方面:数据的逻辑结构(Logical structure)和数据的物理结构(Physical structure)。

5) 数据的逻辑结构

数据的逻辑结构是指各数据元素之间的逻辑关系,是用户按使用需要建立起来,并是一种呈现在用户面前的数据元素的结构形式。图 1.2 所示为某大学教师人事档案信息管理系统中

数据的逻辑结构示意图。

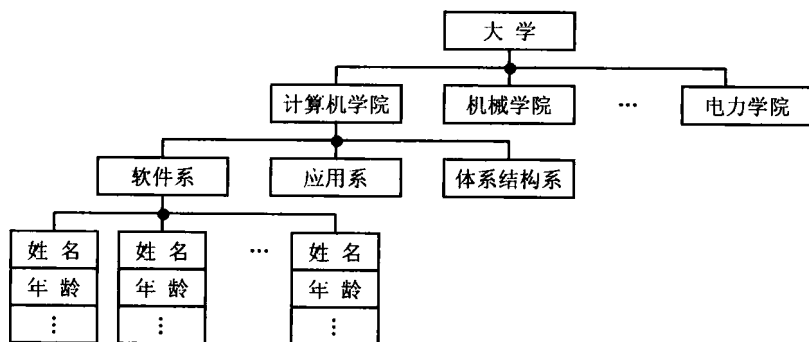


图 1.2 数据的逻辑结构

数据的逻辑结构通常分为 4 类：

- (1)集合：结构中的数据元素除了同属于一种类型外，别无其他关系。
- (2)线性结构：结构中的数据元素之间存在一对一的关系。
- (3)树型结构：结构中的数据元素之间存在一对多的关系。
- (4)图状结构或图状结构：结构中的数据元素之间存在多对多的关系。

对上述这些类型的数据结构，后续章节均有较详细的介绍。

6) 数据的物理结构

数据的物理结构又称为数据的存储结构，它是指数据在计算机内实际的存储形式。例如，要把图 1.2 所示各数据存储到计算机中，不仅要考虑数据存放的先后次序，而且还要考虑那些反映教师各种情况的数据项，如姓名、年龄、工资等，如何在计算机中存放，亦即要考虑数据的物理结构。

数据的逻辑结构和物理结构是密切相关的。在后面章节的学习中将会看到，有的结构类型中这二者是一致的，有的则是不一致的。为了叙述上的方便和避免产生混淆，常把数据的逻辑结构简称为数据结构，而将数据的物理结构称为存储结构。

每种数据结构都可通过映像的方式得到相应的存储结构。常用的映像方式有两种：顺序映像和非顺序映像。由此，可得出两种不同的存储结构：顺序存储结构和链式存储结构。

顺序存储结构是借助元素在存储器中的相对位置来表示数据元素之间的逻辑关系的，而非顺序存储结构则是借助指示元素地址的指针来表示数据元素之间的逻辑关系的。

人们也常常用 Hash 和索引方式映像，从而可得到 Hash 存储结构和索引存储结构。

7) 数据类型(Datatype)

数据类型是程序设计语言中各变量可取的种类。各种程序设计语言不仅规定了每种类型变量的取值范围，而且还规定了该类型变量能执行的运算，也有些语言允许由内部类型构造出新的数据类型。因此，可以把数据类型看作是程序设计语言中已经实现的数据结构。

数据结构是在程序设计语言的基础上由用户建立起来的，它依靠语言提供的数据类型来描述数据的逻辑结构，也依靠语言提供的各种设施来定义、描述运算及其算法。这些运算按实际问题的需要由用户自己定义，而不是由语言系统事先规定。

因此，数据类型和数据结构的主要区别是：前者面向系统，后者面向对象，后者是高一层的数据抽象。如果程序设计语言提供有抽象数据类型设施，则二者可以统一起来。

8) 抽象数据类型 (Abstract data type)

抽象数据类型是指一个数学模型以及定义在该模型上的一组操作。抽象数据类型的定义仅仅取决于它的一组逻辑特性,而与它在计算机内的表示和实现无关。即不论其内部结构如何变化,只要它的数学特性不变,就不会影响其外部的使用。

在面向对象的程序设计(Object Oriented Programming,简称 OOP)技术中,数据和对数据的操作是融合在一起的。一般借助“对象”来描述抽象数据类型,并且一旦定义了一个对象,就可以使用对象的名字来说明变量,调用其中的操作,从而实现信息的隐蔽和封装。

抽象数据类型可以通过高级语言中已有的数据类型来表示和实现。

9) 数据结构的运算

数据结构的主要运算包括以下几种:

- (1) 建立(Create)一个数据结构的运算;
- (2) 消除(Destroy)一个数据结构的运算;
- (3) 从数据结构中删除>Delete)一个数据元素的运算;
- (4) 把一个数据元素插入(Insert)到另一个数据结构中的运算;
- (5) 对一个数据结构进行访问(Access)的运算;
- (6) 对一个数据结构进行修改(Modify)的运算;
- (7) 对一个数据结构进行排序(Sort)的运算;
- (8) 对一个数据结构进行查找(Search)的运算。

以上是数据结构中的一些主要运算,除此以外,还有一些其他的运算。

1.4 算法和算法的分析

1.4.1 算法

算法(Algorithm)是计算机科学和技术中一个十分重要的概念。本书在讨论各种数据结构基本运算的同时,都将会给出相应的算法描述。

算法是对特定问题求解步骤的一种描述,也是执行特定计算的有穷过程。该过程应具有下述几个特性:

- (1) 动态有穷性:当执行一个算法时,不论是何种情况,在经过了有限步骤后,该算法一定会终止。
- (2) 确定性:算法中的每一条指令都必须是清楚的,执行算法时不会产生歧义性。并且,任何条件下算法只有惟一的执行路径,即对于相同的输入定有相同的输出。
- (3) 输入:一个算法有零个或多个由外界提供的量。
- (4) 输出:一个算法产生一个或多个输出。
- (5) 可行性:算法是可行的,即算法中描述的操作都可以通过已实现的基本运算的执行来完成。

1.4.2 算法的描述

描述算法的方法很多,例如,可用自然语言(中文、英文等)、计算机语言、流程图等来描述算法。本书中使用类 C 语言来描述算法(类 C 语言是精选的 C 语言的一个子集,同时也作了