

21世纪高等教育计算机规划教材

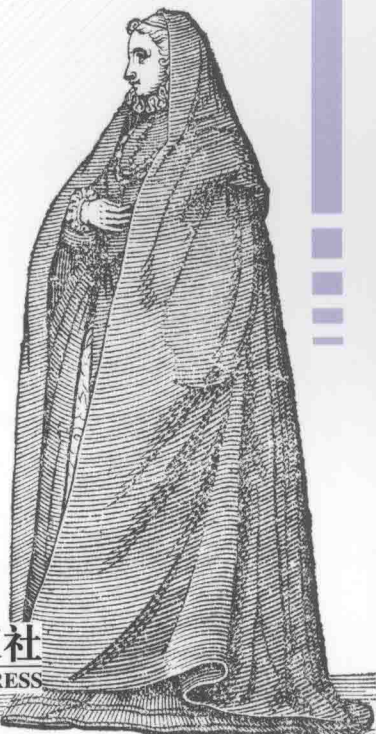
COMPUTER

数据结构 (Java 语言描述)

Data Structures in Java

■ 罗福强 杨剑 刘英 编著

- 以 Java 为基础，将面向对象的思想融入到数据结构设计和算法设计之中
- 强化串、矩阵、树和图等数据结构的设计和应用，为新兴技术的应用提供更多支持
- 强调动手能力培养，提供丰富的习题和实验实训内容



中国工信出版集团



人民邮电出版社
POSTS & TELECOM PRESS

21世纪高等教育计算机规划教材

COMPUTER

数据结构 (Java 语言描述)

Data Structures in Java

罗福强 杨剑 刘英 编著



人民邮电出版社

北京

图书在版编目 (CIP) 数据

数据结构：Java语言描述 / 罗福强, 杨剑, 刘英编著. — 北京：人民邮电出版社, 2016. 1
21世纪高等教育计算机规划教材
ISBN 978-7-115-41368-0

I. ①数… II. ①罗… ②杨… ③刘… III. ①数据结构—高等学校—教材②C语言—程序设计—高等学校—教材 IV. ①TP311.12②TP312

中国版本图书馆CIP数据核字(2015)第314576号

内 容 提 要

“数据结构”课程是国内计算机相关专业的基础课程。为了有利于促进新兴信息技术（例如：云计算、大数据、物联网、智能手机等）人才培养，本书强化了诸如串、矩阵、树和图等数据结构的设计和应用。本书共9章，可划分为四个部分。第一部分（第1章）是数据结构的基本概念部分；第二部分（第2~6章）是基本的数据结构部分，包括线性结构（线性表、栈、队列、串、矩阵、广义表）与非线性结构（树、图）；第三部分（第7~8章）是基本技术部分，包括查找与排序方法；第四部分（第9章）是综合项目实训部分，通过“校园导游系统设计”案例展示数据结构的应用与实现过程。本书面向应用型本科院校学生，立足于把数据结构的基本概念和基本算法讲清楚，讲透彻，知识结构完整，提供大量的应用案例，不仅配备了丰富、符合教学目标的课后习题，还配备了富有针对性的实训任务和综合项目实训，因此既方便教学，又方便自学。

本书可作为应用型高等院校计算机相关专业的数据结构课程的教材或参考资料，也可供软件开发人员参考使用。

-
- ◆ 编 著 罗福强 杨剑 刘英
责任编辑 刘 博
责任印制 沈 蓉 彭志环
 - ◆ 人民邮电出版社出版发行 北京市丰台区成寿寺路11号
邮编 100164 电子邮件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
固安县铭成印刷有限公司印刷
 - ◆ 开本：787×1092 1/16
印张：15 2016年1月第1版
字数：391千字 2016年1月河北第1次印刷
-

定价：39.80元

读者服务热线：(010)81055256 印装质量热线：(010)81055316

反盗版热线：(010)81055315

在物质世界中,一切事物投影到计算机世界之中全部都表现为数据。这些数据来源于现实,表示某种具体的意义,具有统一的表示方法,因而构成计算机程序处理的对象。研究数据在计算机中的表示方法、关联方法、存储方法以及在其之上的处理方法,就构成了数据结构课程的主要内容。

随着以云计算、物联网、大数据、移动互联为代表的第3次信息技术浪潮的发展,无论软件设计思想、程序设计语言、软件开发环境,还是最终的应用场景如今都发生了深刻的变革。首先,从程序设计方法来讲,传统的以C语言为代表的结构化程序设计方法早已转变为面向对象的程序设计方法了,而且在云计算技术的引领之下程序设计方法正在逐渐向面向服务的设计方法(SOA+RESTful)发展和转移。其次,从人才培养方面来说,不但出现了新专业,如软件工程、网络工程、物联网工程等;而且专业方向也越分越细了,如云计算技术方向、大数据技术方向、智能手机方向等。

因此,按“C语言程序设计→C++程序设计→数据结构→……”的线路开展教学的课程设置模式越来越无法适应第3次IT浪潮发展的要求了。也正因为如此,越来越多的院校在试着打破传统的课程设置模式。例如,根据专业方向的需要直接将Java语言作为入门语言组织教学,为此就形成了诸如“Java程序设计→数据结构→大数据技术(Hadoop+HBase)→……”之类课程设置方案。

本教材正是基于这种时代大变革的要求而编写的教材。全书以Java为基础,将面向对象的思想融入到数据结构设计和算法设计之中,通过精选基础理论内容、降低理论难度和抽象性、加强实践环节等措施来增强学生的面向对象程序设计能力和应用能力培养和提高,并力求取得较好的教学效果。同时,基于Java语言已经提供了诸如栈、队列、链表、字符串、数组、集合等内置数据结构的状况,兼顾大数据技术、物联网技术等新专业方向的对地理位置、图像、视频等数据处理的需要,本教材强化了诸如串、矩阵、树和图等数据结构的设计和应用,从而为新兴的技术应用提供更多的支持。

本书在编写时力求体现以下5大特色。第一,教学定位清楚,本书面向应用型本科院校学生,立足于把数据结构的基本概念和基本算法讲清楚,讲透彻,第二,教学内容先进,全书以Java语言为工具,用面向对象的思想来描述各种的数据结构的定义和相关操作算法的实现。第三,教学目标明确,知识结构完整,在教学内容安排方面强调既要方便教学,又要方便自学,因此针对数据结构的基本算法提供完整的Java源代码实现,而针对其他算法只提供分析和设计思路,源代码留给读者拓展实现。第四,教学理念先进,坚持以应用为纲,避免了传统数据结构教材重理论轻实用的弊端,因此本书针对每一种数据结构都特别突出相关数据结构的应用与实践。第五,教学模式完善,坚持以能力培养为中心,全书不仅配备了丰富、符合教学目标的课后习题,还为每章配备了从知识点出发进行设计的、能真正培养学生应用能力的实训。本书最后一章提供了综合项目实训,通过6个实训任务来提高

学生的综合应用与实践能力。

本书由四川大学锦城学院的罗福强老师、刘英老师和电子科技大学成都学院的杨剑老师共同编写。其中，刘英老师编写第 1 章的部分和第 2、3 章，罗福强老师编写了第 1 章的部分和第 4、5、9 章，杨剑老师编写了第 6、7、8 章。本书最后由罗福强统稿和审核。

特别感谢四川大学锦城学院的陶德元教授对本书的编写提出了很多中肯的意见。同时，也向所有其他帮助和支持本书编写的人表示感谢！

编 者

2015 年 12 月

目 录

第 1 章 概述	1
1.1 数据结构的作用和意义	1
1.1.1 数据结构的作用	2
1.1.2 数据结构的意义	2
1.2 基本概念和术语	4
1.2.1 基本概念和术语	4
1.2.2 数据的逻辑结构	7
1.2.3 数据的物理结构	8
1.3 面向对象的数据结构表示	9
1.3.1 Java 面向对象基础	9
1.3.2 面向对象的抽象数据类型	12
1.3.3 使用 Java 语言描述数据结构的 优势	13
1.4 算法和算法分析	15
1.4.1 算法的基本概念	15
1.4.2 算法效率的度量	16
1.4.3 算法效率分析	17
1.5 习题	19
第 2 章 线性表	21
2.1 线性表的逻辑结构	22
2.1.1 线性表的概念	22
2.1.2 线性表的基本操作	23
2.2 线性表的顺序表示和实现	23
2.2.1 线性表的顺序表示	23
2.2.2 顺序表的实现	24
2.2.3 顺序表的应用	30
2.3 线性表的链式表示和实现	31
2.3.1 线性表的链式表示	31
2.3.2 单链表的实现	31
2.3.3 循环链表	37
2.3.4 双向链表	38
2.3.5 链表的应用——链表合并与一元 多项式相加	39
2.4 习题	43
2.5 实训	45
第 3 章 栈和队列	47
3.1 栈	47
3.1.1 栈的定义及基本运算	47
3.1.2 顺序栈	49
3.1.3 链栈	51
3.2 队列	53
3.2.1 队列的定义及基本运算	53
3.2.2 顺序队列	53
3.2.3 链队列	58
3.3 栈和队列的应用举例	60
3.3.1 栈应用之一：数制转换	60
3.3.2 栈的应用之二：括号匹配	61
3.3.3 栈的应用之三：表达式求值	63
3.3.4 队列应用之一：模拟服务台前的排 队现象问题	67
3.3.5 队列应用之二：批量任务的 处理	69
3.4 习题	70
3.5 实训	71
第 4 章 串、矩阵和广义表	73
4.1 串及其运算	73
4.1.1 串的基本概念	73
4.1.2 串的基本操作	74
4.2 串的顺序存储与实现	74
4.2.1 顺序存储结构	75
4.2.2 串的实现	75
4.2.3 模式匹配	77
4.3 矩阵	79
4.3.1 特殊矩阵	79
4.3.2 稀疏矩阵	81
4.4 广义表	82

4.4.1 广义表的逻辑结构	82	5.7 习题	125
4.4.2 广义表的存储结构及实现	83	5.8 实训	128
4.5 串的应用: 文本编辑	84	第 6 章 图	129
4.6 矩阵的应用: 矩阵运算与实现	87	6.1 图的定义及其常用术语	130
4.6.1 矩阵运算的意义	87	6.1.1 图的定义	130
4.6.2 矩阵的加法	88	6.1.2 图的常用术语及含义	130
4.6.3 矩阵的乘法	89	6.2 图的存储结构	132
4.6.4 矩阵的转置	90	6.2.1 邻接矩阵	132
4.7 习题	91	6.2.2 邻接表	136
4.8 实训	91	6.3 图的遍历	141
第 5 章 树和二叉树	93	6.3.1 深度优先搜索	141
5.1 树的定义和基本术语	94	6.3.2 广度优先搜索	144
5.1.1 树的定义	94	6.4 生成树和最小生成树	146
5.1.2 树的表示方法	94	6.4.1 生成树	146
5.1.3 树的术语	95	6.4.2 最小生成树	147
5.2 二叉树	96	6.5 图的应用	152
5.2.1 二叉树基本概念	96	6.5.1 最短路径	152
5.2.2 二叉树的性质	97	6.5.2 拓扑排序	157
5.2.3 二叉树的存储结构	97	6.5.3 关键路径	158
5.2.4 二叉树的基本操作及实现	100	6.6 习题与解析	161
5.3 二叉树遍历	102	6.7 实训	165
5.3.1 二叉树遍历	102	第 7 章 查找	166
5.3.2 线索二叉树	107	7.1 基本概念	166
5.3.3 线索二叉树的基本操作实现	108	7.2 静态查找表	168
5.4 树和森林	111	7.2.1 顺序查找	168
5.4.1 树的存储结构	111	7.2.2 折半查找	170
5.4.2 树和森林与二叉树之间的转换	113	7.3 动态查找表	173
5.4.3 树的遍历	114	7.3.1 二叉排序树	173
5.4.4 森林的遍历	115	7.3.2 平衡二叉树	174
5.5 二叉树的应用: Huffman 树与 Huffman 编码	115	7.3.3 B-树	176
5.5.1 哈夫曼树	115	7.4 哈希表	180
5.5.2 哈夫曼树的构造算法	117	7.4.1 哈希表的概念	180
5.5.3 哈夫曼编码	119	7.4.2 哈希函数的构建	181
5.6 二叉树的应用: 二叉排序树与查找	121	7.4.3 处理冲突	182
5.6.1 二叉排序树	121	7.4.4 哈希表的查找及其分析	185
5.6.2 二叉排序树的查找	122	7.5 习题与解析	186
5.6.3 二叉排序树的插入	123	7.6 实训	188
5.6.4 二叉排序树的删除	124		

第 8 章 排序	189	8.8 实训.....	207
8.1 基本概念.....	189	第 9 章 综合项目实训	209
8.2 插入排序.....	191	9.1 实习实训题目及设计要求.....	209
8.2.1 直接插入排序.....	191	9.1.1 评分参考评准.....	209
8.2.2 希尔排序.....	193	9.1.2 智能仓库控制系统.....	210
8.3 交换排序.....	194	9.1.3 运动会分数统计.....	210
8.3.1 冒泡排序.....	194	9.1.4 学生成绩管理系统.....	211
8.3.2 快速排序.....	196	9.1.5 飞机售票系统.....	211
8.4 选择排序.....	198	9.1.6 仓库货物管理系统.....	212
8.4.1 简单选择排序.....	198	9.1.7 校园导游系统设计.....	212
8.4.2 堆排序.....	199	9.2 实习实训与课程设计报告模板.....	212
8.5 归并排序(二路归并排序).....	202	参考文献	231
8.6 各种排序方法的比较.....	204		
8.7 习题与解析.....	205		

第 1 章

概述

建议学时：2 学时

总体要求

- 了解数据结构的意义、数据结构在计算机领域的地位和作用
- 掌握数据结构各名词、术语的含义和有关的基本概念，以及数据的逻辑结构和存储结构之间的关系
- 了解使用 Java 语言对数据结构进行抽象数据类型的表示和实现的方法
- 了解算法的五要素
- 掌握计算语句频度估算算法时间复杂度的方法

相关知识点

- 相关术语：数据、数据元素、数据对象、数据结构
- 数据逻辑结构：集合、线性结构、树和图
- 数据的物理结构：顺序和非顺序结构
- 算法的五要素和时间复杂度及空间复杂度

学习重点

- 数据的逻辑结构和存储结构及其之间的关系
- 算法时间复杂度的计算

学习难点

- 算法时间复杂度的计算

计算机科学是一门研究信息表示、组织和处理的科学，而信息的表示和组织直接关系到处理信息的效率。随着计算机产业的迅速发展和计算机应用领域的不断扩大，计算机应用已不仅仅限于早期的科学计算，而是更多地用于控制、管理和数据处理等方面，随之而来的是处理的数据量越来越大，数据类型越来越多，数据结构越来越复杂。因此，如要编制一个高效的处理程序，就需要解决如何合理地组织数据，建立合适的数据结构，设计好的算法，来提高程序执行的效率等问题。“数据结构”这门学科就是在这样的背景下逐步形成和发展起来的。

1.1 数据结构的作用和意义

数据是外部世界信息的计算机化，是计算机加工处理的对象。运用计算机处理数据时，必须解决四个方面的问题：一是如何在计算机中方便、高效地表示和组织数据；二是如何在计算机存

储器 (内存和外存) 中存储数据; 三是如何对存储在计算机中的数据进行操作, 可以有哪些操作, 如何实现这些操作以及如何对同一问题的不同操作方法进行评价; 四是必须理解每种数据结构的性能特征, 以便选择一个适合于某个特定问题的数据结构。这些问题就是数据结构这门课程所要研究的主要问题。

1.1.1 数据结构的作用

我们知道, 虽然每个人都懂得英语的语法与基本类型, 但是对于同样的题目, 每个人写出的作文, 水平却高低不一。程序设计也和写英语作文一样, 虽然程序员都懂得语言的语法与语义, 但是对于同样的问题, 不同的程序员会写出来不一样的程序。有的人写出来的程序效率很高, 有的人却用复杂的方法来解决一个简单的问题。

当然, 程序设计水平的提高仅仅靠看几本程序设计书是不行的。只有多思索, 多练习, 才能提高自己的程序设计水平, 否则, 书看得再多, 提高也不大。程序设计水平要想提高, 要多看别人写的程序, 多去思考问题。从别人写的程序中, 我们可以发现效率更高的解决方法; 从思考问题的过程中, 我们可以了解解决问题的方法常常不只一个。运用先前解决问题的经验, 来解决更复杂更深入的问题, 是提高程序设计水平的最有效途径。

数据结构正是前人在思索问题的过程中所想出的解决方法。一般而言, 在学习程序设计一段时间后, 学习“数据结构”便能让你的程序设计水平上一个台阶。如果只学会了程序设计的语法和语义, 那么你能解决程序设计三分之一的问题, 而且运用的方法并不是最有效的。但如果学会了数据结构的概念, 就能在程序设计上, 运用最有效的方法来解决绝大多数的问题。

《数据结构》这门课程的目的有三个。第一个是讲授常用的数据结构, 这些数据结构形成了程序员处理数据问题的基本工具。对于许多常见的问题, 这些数据结构是理想的选择。程序员可以直接拿来或经过少许的修改就可以使用, 非常方便。第二个是讲授常用的算法, 这和数据结构一样, 是人们在长期实践过程中的总结, 程序员也可以直接拿来或经过少许的修改就可以使用, 并且可以通过算法训练来提高程序设计水平。第三个目的是通过程序设计的技能训练促进程序员综合能力的提高。

1.1.2 数据结构的意义

当我们用计算机解决一个问题时, 必须告诉计算机如何去做。这需要先分析问题, 确定一个适合的数据模型; 然后, 需要设计一个求解这个数据模型的算法, 最后编写程序, 经过反复调试直至得到正确结果。这就像我们求解一个数学的应用题时, 需要通过问题的描述列出一个方程或方程组, 然后求解该方程 (组) 一样。但是, 需要计算机求解的大多数问题比数学方程要复杂得多。下面给出几个简单的例子加以说明。

【例 1-1】某公司有 50 名员工, 现在需要设计一个管理系统, 完成对员工信息的查找、修改、插入或删除。

首先需考虑如何表示这 50 名员工的信息。员工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, 这是一种线性结构。可以建立一个线性表, 线性表中的每一个元素表示一个员工信息, 如表 1-1 所示, 对员工信息的查找、修改、插入或删除都应该基于该线性表进行操作。

员工信息是按工号一个接一个存放的。要查找某个员工信息, 可以从第一个员工开始, 依次向后一一比较, 找到后, 就可以修改了。如果要插入一个员工信息, 可以先找到插入位置, 把插入位置到最后的所有员工信息依次向后移动, 空出位置后插入。如果要删除一个员工信息, 可以

先找到删除位置，然后将后面的员工信息依次前移即可。

表 1-1 员工基本信息表

工号	姓名	性别	生日	...
K0705	王鹏	男	1975.06	
K0722	李煜	男	1978.04	
K0809	赵斌	女	1978.11	
K0916	侯超	男	1979.03	
K1005	张小兵	男	1980.07	
K1204	周宇	女	1981.12	
...

【例 1-2】计算机和人对弈问题。由于对弈的过程是在一定的规则下进行的，为使计算机能灵活对弈就必须将对弈过程中所有可能发生的情况以及相应的对策都考虑周全，同时，作为一名“好”的棋手，还应能预测棋局的发展趋势，所以，为使计算机能够和人进行对弈，必须事先将对弈的策略存入计算机，图 1.1 (a) 所示为一个九宫格的棋盘格局。

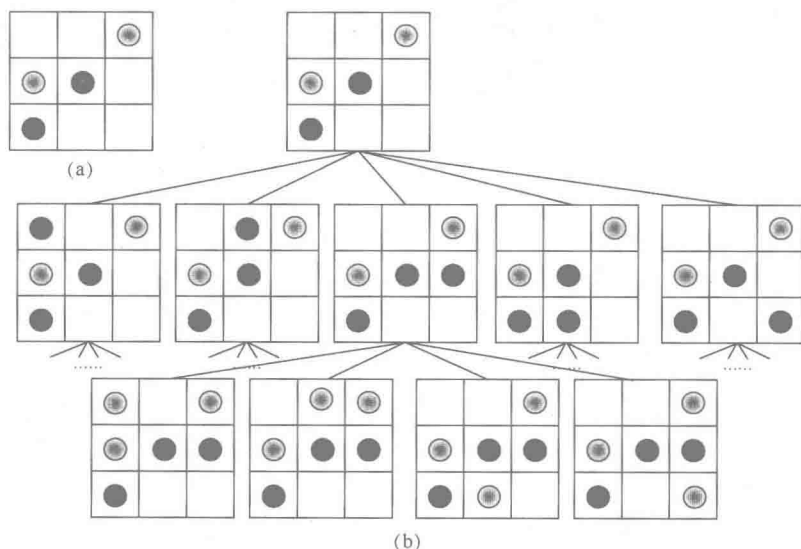


图 1.1 九宫格方盘对弈“树”

黑白双方交替落子，如将从对弈开始到结束过程中可能出现的棋盘格局画出来，则可得到一个倒长的“树”，图 1.1 (b) 所示的是其中的一部分。“树根”是对弈开始前的棋盘格局，而“叶子”是可能出现的结局。可以看到与线性结构不同，一个格局可以派生出几个格局，也就是说，一个格局只能有一个前驱，而可以有多个后继。如果计算机对弈开始前就计算出这样一颗树，就可以知道在对弈过程哪一种走法获胜的概率大一些，就像一位高手能预测棋局发展的趋势一样，从而选择一种较好的走法。

【例 1-3】田径比赛赛程安排问题。在一名选手参加多个项目的情况下，这些项目不能同时开始，否则会产生冲突。假设有一个比赛的参赛项目表如表 1-2 如示，那么 A、B、E 就不能同时开始。那么应如何安排赛程呢？

表 1-2

选手参赛项目表

姓名	参赛项		
张	A	B	E
王	C	D	
李	C	E	F
赵	D	F	A
刘	B	F	

在此例中，可以把一个参赛项表示为图中的一个顶点，而当两个项目不能同时举行时，以两个顶点之间的连线表示互相矛盾的关系。如图 1.2 (a) 所示，每个圆圈表示一个比赛项目，两个圆圈之间的连线表示这两个圆圈不能在同一时间安排。所以当安排项目 A 时，只能同时安排没有和 A 连线的项目，在此例中，应为 C，可以按此方法将没有冲突（互相没有连线）的项目用同一种颜色涂色，图 1.2 (b) 所示为一种涂色结果。该结果表示的安排方法为 (A、C)，(B、D)，(E)，(F)。

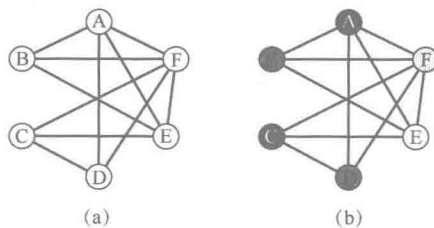


图 1.2 赛程安排示意图

通常，这种模型中，每个项目表示为一个顶点。每个顶点可以和其他任意个顶点联系。这类问题的数学模型是一种称为“图”的数据结构。

综合前面 3 个例子，这类非数值计算问题的数学模型是诸如表、树、图之类的数据结构，因此，数据结构是一门研究非数值计算的程序设计问题中计算机的操作对象以及它们之间的关系及操作的学科。

数据结构是一门介于数学、计算机硬件和计算机软件三者之间的核心课程。数据结构不仅是一般程序设计的基础，而且是设计和实现编译程序、操作系统、数据库系统及其他系统程序和大型应用程序的重要基础。

学习“数据结构”既为进一步学习其他计算机专业课程提供必要的准备知识，也有助于提高软件设计和程序编制水平。1968 年，美国 D.E.Kunth 教授开创了数据结构的最初体系，他的名著《计算机程序设计技巧》较为系统地阐述了数据的逻辑结构和存储结构及其操作。随着计算机科学的飞速发展和应用领域的不断扩大，到 20 世纪 80 年代初期，数据结构的基础研究日臻成熟，已成为一门完整的学科。

1.2 基本概念和术语

1.2.1 基本概念和术语

本节将对一些常用的概念和术语进行介绍，这些概念和术语在以后的章节中会多次出现。

1. 数据

数据即信息的载体,是对客观事物的符号表示,凡能输入到计算机中并被计算机程序处理的符号都可称之为**数据**,如整数、实数、字符、文字、声音、图像等都是数据,如图 1.3 所示。



图 1.3 数据范畴

2. 数据元素

数据元素是数据的基本单位,它在计算机处理和程序设计中通常作为独立个体。数据元素一般由一个或多个数据项组成,一个数据元素包含多个数据项时,常称为记录、结点等。数据项也称为域、字段、属性、表目、顶点。

如表 1-3 所示,每个同学的信息即表中的一行,是一个数据元素。每个数据元素又包含若干个数据项,如专业、学号等。

表 1-3 学生信息统计表

专业	专业方向	学号	姓名	联系电话
信息科学	嵌入式方向	110810110	黄小虎	18456384714
信息科学	移动方向	110810117	龚力	15881452292
通信工程	网络通信方向	110840128	李俊	18212385458
通信工程	嵌入式方向	110840139	张静静	18202343102
信息工程	物联网方向	110830104	张婷	15682054175

3. 数据对象

数据对象是具有相同特征的数据元素的集合,是数据的一个子集,如一个整型数组、一个字符串数组都是一个数组对象。

例如要将表 1-3 中的学生信息按照学号大小进行排序,排序时比较的是各个数据元素中学号这一数据项的大小。此时整个表中的数据元素就是待处理的数据对象。

4. 数据结构

数据结构简称 DS(Data Structure),是数据及数据元素的组织形式。任何数据都不是彼此孤立的,通常把相关联的数据按照一定的逻辑关系组织起来,这样就形成了一个数据结构。

数据结构包含两个方面的内容,一是数据对象,二是数据对象中数据元素之间内在的关系。数据结构通常有四类基本形式:集合结构,线性结构,树型结构,图形结构或网状结构。

(1)集合结构:集合结构中的数据元素除了同属于一个集合外,它们之间没有其他关系。各个数据元素是“平等”的,它们的共同属性是“同属于一个集合”。数据结构中的集合关系就类似于数学中的集合。

【例 1-4】电视机、冰箱等家用电器构成的就是一个集合结构,如图 1.4 所示。



图 1.4 家用电器构成的集合结构

(2)线性结构:线性结构中的数据元素之间是一一对应的关系。

【例 1-5】节气是中国古代订立的一种用来指导农事的补充历法,是汉族劳动人民长期经验的积累和智慧的结晶。春季的节气有立春、雨水、惊蛰、春分、清明、谷雨,这些节气一个接一个就是一种线性关系,如图 1.5 所示。

立春 — 雨水 — 惊蛰 — 春分 — 清明 — 谷雨

图 1.5 春季节气构成的线性结构

(3) 树型结构：树形结构中的数据元素之间存在一种一对多的层次关系。

【例 1-6】院系机构的组织就是一种树型结构，如图 1.6 所示。计科系和电子系的地位是“平等”的，电子系开设有信息工程、通信、微电子等专业，它们受电子系的“领导”。反过来看，无论是通信工程，还是微电子，都属于电子系，无论是计科系，还是电子系，都属于某学院，即树型结构中正关系是一对多，逆关系是一对一。



图 1.6 某院系组织构成的树型结构

(4) 图状结构：图状结构中的数据元素是多对多的关系。

【例 1-7】在一个学生选课系统中，数据库的设计采用的就是网状结构，如图 1.7 所示。把 S1 学生和他的选课记录（选修的 C1、C2 两门课程的选课记录）链接起来，同样把 S2、S3、S4 学生和他们的选课记录链接起来，把 C1 课程和选修了 C1 课程的选课记录（有 S1、S2、S3、S4 学生选修了 C1）链接起来，同样把 C2、C3 课程和选修了这些课程的选课记录链接起来，通过分析可以发现一个学生可以选修若干门课程，某一课程可以被若干同学选修，学生与课程之间是多对多的关系。

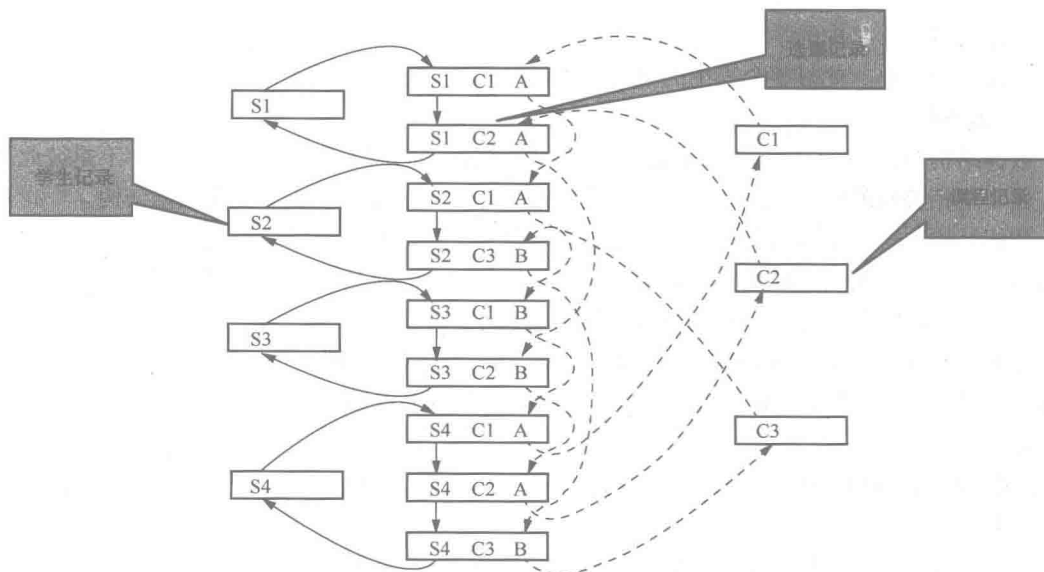


图 1.7 学生选课系统中数据库设计采用的图状结构

5. 数据类型

数据类型是一组具有相同性质的操作对象以及该组操作对象上的运算方法的集合，如整数类型、字符类型等。每一种数据类型都有自身特点的一组操作方法，即运算规则。JAVA 中提供的基本的数据类型有 int, double, long, float, short, byte, character, boolean。

由于集合中的元素的关系极为松散,可用其他数据结构来表示,所以本书不做专门介绍。

从数据类型和数据结构的概念可知,二者的关系非常密切。数据类型可以看作是简单的数据结构。数据的取值范围可以看作是数据元素的有限集合,而对数据进行操作的集合可以看作是数据元素之间关系的集合。

1.2.2 数据的逻辑结构

数据的逻辑结构 (Logic Structure) 是从具体问题抽象出来的数学模型,与数据在计算机中的具体存储没有关系。数据的逻辑结构独立于计算机,是数据本身所固有的特性。从逻辑上可以把数据结构分为线性结构和非线性结构,上述数据结构的定义就是数据的逻辑结构 (Logic Structure),主要包括:集合、线性、树和图形结构,其中集合、树和图形结构都属于非线性结构。

数据的逻辑结构有两个要素:一是数据元素的集合,通常记为 D ;二是 D 上的关系集,它反映了 D 中各数据元素之间的前驱后继关系,通常记为 R ,即一个数据结构可以表示成二元组 $B=(D,R)$ 。

【例 1-8】一年四季可表示成 $B_1=(D_1,R_1)$,其中

$D_1=\{\text{春,夏,秋,冬}\}$, $R_1=\{\langle\text{春,夏}\rangle,\langle\text{夏,秋}\rangle,\langle\text{秋,冬}\rangle\}$;

序偶关系 $\langle\text{春,夏}\rangle$ 表示:春季之后的下一个季节是夏季,反过来夏季的前一个季节是春季,其他依此类推,通过关系集 R_1 就可以描述出一年四季的更替顺序春、夏、秋、冬,同时我们可以发现在关系集 R_1 中,夏季和秋季都有一个直接前驱和一个直接后继,春季作为一年之首是没有前驱的,但是有一个直接后继夏,冬季作为一年之尾没有后继,但是有一个前驱,所以二元组 B_1 描述的是一个一对一关系的线性结构。

注意, $\langle x,y \rangle$ 意为 x 和 y 之间存在“ x 领先于 y ”的次序关系。而 (x,y) 表示 x 和 y 之间没有次序上的关系。

【例 1-9】某单位的管理关系可表示成 $B_2=(D_2,R_2)$,其中

$D_2=\{\text{总经理, 部门经理 A, 部门经理 B, 组长 A, 组长 B, 组长 C, 组长 D, 职工 A, 职工 B, 职工 C, 职工 D, 职工 E, 职工 F, 职工 G}\}$

$R_2=\{\langle\text{总经理, 部门经理 A}\rangle,\langle\text{总经理, 部门经理 B}\rangle,\langle\text{部门经理 A, 组长 A}\rangle,\langle\text{部门经理 A, 组长 B}\rangle,\langle\text{部门经理 B, 组长 C}\rangle,\langle\text{部门经理 B, 组长 D}\rangle,\langle\text{组长 A, 职工 A}\rangle,\langle\text{组长 A, 职工 B}\rangle,\langle\text{组长 A, 职工 C}\rangle,\langle\text{组长 D, 职工 D}\rangle,\langle\text{组长 D, 职工 E}\rangle,\langle\text{组长 D, 职工 F}\rangle,\langle\text{组长 D, 职工 G}\rangle\}$;

通过分析关系集 R_2 可知该单位人员的关系为:一个总经理管理若干部门经理,每个部门经理管理若干小组长,每个小组长管理若干职工。但是每个职工的直接领导,即组长,只有一个,每个组长的直接领导,即部门经理,只有一个,每个部分经理的直接领导总经理只有一个,即二元组描述的是一个一对多关系的树型结构,如图 1.8 所示。

【例 1-10】A 某的人际关系可以表示成 $B_3=(D_3,R_3)$,其中

$D_3=\{\text{A 某, B 某, C 某, D 某, E 某, F 某, G 某, H 某}\}$,

$R_3=\{\langle\text{A 某,B 某}\rangle,\langle\text{A 某,C 某}\rangle,\langle\text{B 某, D 某}\rangle,\langle\text{B 某,, E 某}\rangle,\langle\text{B 某, F 某}\rangle,\langle\text{C 某, F 某}\rangle,\langle\text{C 某, G 某}\rangle,\langle\text{E 某, H 某}\rangle,\langle\text{F 某, H 某}\rangle\}$;

通过分析关系集 R_3 可知 A 某的人际关系为:每个人可以有多个朋友,同一个人也可以是多个人共同的朋友。例如, F 某既是 B 某的朋友,也是 C 某的朋友,即二元组描述的是一个多对多关系的图形结构,如图 1.9 所示。

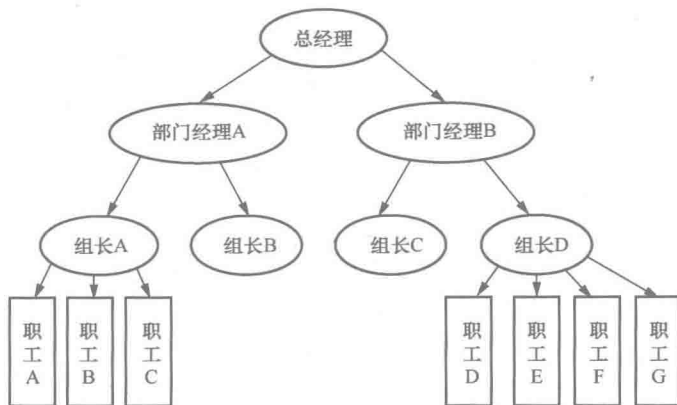


图 1.8 单位的管理关系

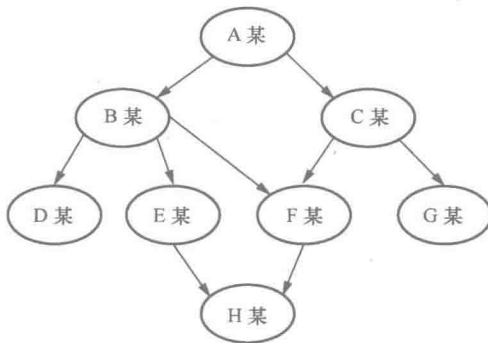


图 1.9 A 某人际关系

上述数据结构的定义仅是对操作对象的一种数学描述，是从操作对象抽象出来的数学模型，结构定义的“关系”抽述的是数据元素之间的逻辑关系，所以称为逻辑结构。

1.2.3 数据的物理结构

我们讨论数据结构的目的是为了在计算机中实现对它的操作，因此还需要研究在计算机中如何表示和存储数据结构，即数据的物理结构 (Physical Structure)。数据的物理结构又称存储结构，它的实现依赖于具体的计算机语言。数据存储结构有顺序和链式两种不同的方式，顺序存储的特点是数据元素在存储器的相对位置来体现数据元素相互间的逻辑关系，顺序存储结构通常用高级编程语言中的一维数组来描述或实现。

例如：一个有序的数字序列 (25, 34, 48, 57, 63)，25 有一个直接后继 34，如果该有序表用数组存储，通过图 1.10 可以发现相邻的 25 和 34 在存储器中的地址也是相邻的，即数据元素在存储器中的相对位置可以体现数据元素在有序表中的前驱和后继关系，整个数组在存储器中占用的是一片连续的存储空间。

链式存储结构是通过一组任意的存储单元来存储数据元素的，而这些存储单元可以是连续的，也可以是不连续的。为建立起数据元素之间的逻辑关系，对任一数据元素 a，除了存放元素自身信息 a 之外，还需要存放与 a 有关系的其他元素的地址 150，通过这个地址就可以找到下一个数据元素 b，依此类推，可以发现字符序列 ('a', 'b', 'c', 'd', 'e', 'f') 在存储器中的地址是不连续的。

...	
0300	25
0304	34
0308	48
030C	57
0310	63
...	

图 1.10 顺序存储结构示意图

110	e	200
...		...
150	b	190
160	a	150
...		...
190	c	210
200	f	NULL
210	d	110

图 1.11 链式存储结构示意图

在顺序存储结构的基础上，又可延伸出另外两种存储结构，即索引存储和散列存储。

索引存储就是在数据文件的基础上增加了一个索引表文件。通过索引表建立索引，可以把一个顺序表分成几个顺序子表，其目的是在查询时提高查找效率，避免盲目查找。

散列存储就是通过数据元素与存储地址之间建立起某种映射关系，使每个数据元素与每一个存储地址之间尽量达到一一对应的目的。这样，查找时同样可以大大提高效率。

数据的逻辑结构和物理结构是密切相关的两个方面，任何一个算法的设计取决于选定的逻辑结构，而算法的实现依赖于采用的存储结构。综上所述，数据结构主要描述的是数据元素之间的逻辑关系、数据在计算机系统存储方式和数据的运算三个方面的内容，即数据的逻辑结构、存储结构和数据的操作集合。

1.3 面向对象的数据结构表示

1.3.1 Java 面向对象基础

1. 类的声明与实例化

Java 语言是面向对象的程序设计语言，类和对象是面向对象的核心。Java 语言提供了创建类和创建对象的语法支持。定义类的简单语法如下。

```
[修饰符] class 类名
{
    零个到多个构造器定义
    零个到多个成员变量
    零个到多个成员方法
}
```

在上面的语法格式中，修饰符可以是 `public`、`final`、`abstract`，或者完全省略这三个修饰符，类名必须是一个合法的标识符。

例如，圆是大家都熟悉的一种几何图形，半径是圆的一个属性，它的值决定了圆的大小，下面我们就来定义一个表示圆的类。

```
public class Circle
{
```