



“十二五”普通高等教育本科国家级规划教材

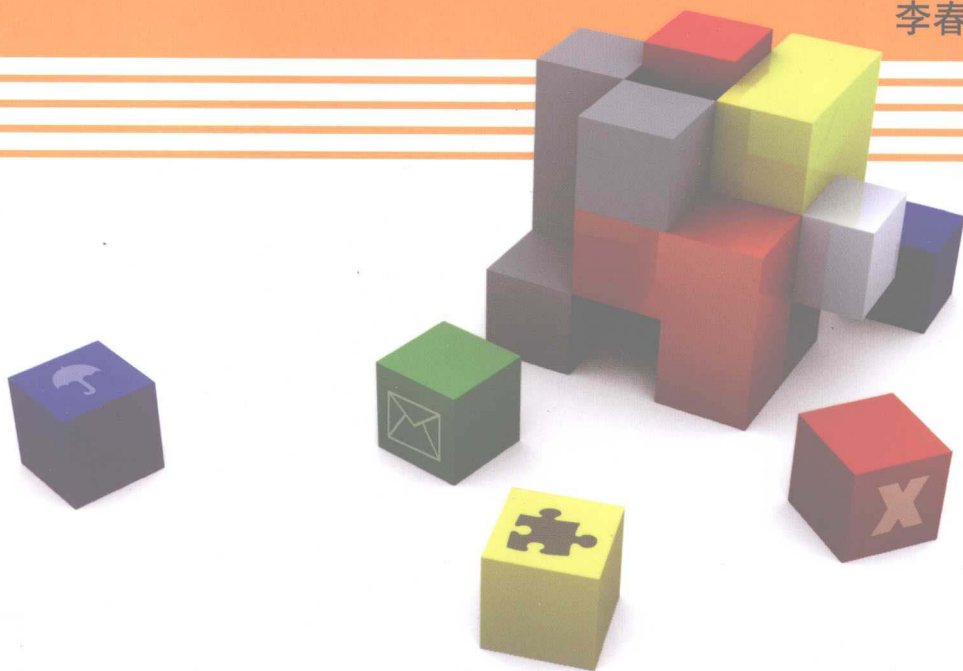


高等学校数据结构课程系列教材

(第4版)

数据结构教程

李春葆 主编



清华大学出版社

高等学校数据结构课程系列教材

数据结构教程(第4版)

李春葆 主编

尹为民 蒋晶珏 喻丹丹 安 杨 编著

清华大学出版社
北京

内 容 简 介

本书在前3版的基础上,根据教育部新的考研大纲和大量读者来信提出的要求进行了修订。本书内容包括绪论、线性表、栈和队列、串、递归、数组和广义表、树和二叉树、图、查找、内排序、外排序和文件,还给出了6个综合实验题、实验报告格式、引用型参数的说明、顺序表和顺序栈以及顺序队列使用指针引用型参数的说明、书中部分算法清单、全国计算机专业数据结构2011年联考大纲。

本书适合高等院校计算机及相关专业本科生和研究生使用。

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

数据结构教程/李春葆主编.—4版.—北京:清华大学出版社,2013.1

(高等学校数据结构课程系列教材)

ISBN 978-7-302-25087-6

I. ①数… II. ①李… III. ①数据结构—教材 IV. ①TP311.12

中国版本图书馆CIP数据核字(2011)第046563号

责任编辑:魏江江 薛 阳

封面设计:杨 兮

责任校对:梁 毅

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 装 者:北京鑫海金澳胶印有限公司

经 销:全国新华书店

开 本:185mm×260mm 印 张:24 字 数:607千字

版 次:2013年1月第4版 印 次:2013年2月第2次印刷

印 数:82501~85500

定 价:39.00元

前言

数据结构是计算机学科的必修课程,涵盖了计算机学科的设计、数值分析、操作系统和编译原理等课程所涉及的大部分相关算法的实现。学好该课程,不仅对这些后续课程的学习有很大帮助,而且能在实际中发挥其广泛的用途。

计算机是进行数据处理的工具,而数据结构主要研究数据的各种组织形式以及建立在这些组织形式之上的各种运算算法的实现,它不仅为用计算机语言进行程序设计提供了方法性的理论指导,还在一个更高的层次上总结了程序设计的常用方法和常用技巧。

本教程是作者针对数据结构课程概念多、算法灵活和抽象性强等特点,在总结长期教学经验的基础上编写的。全书分为 13 章和 6 个附录,第 1 章为绪论,介绍数据结构的基本概念,特别强调算法分析的方法;第 2 章为线性表,介绍线性表的两种存储结构——顺序表和链表与其基本运算算法的实现;第 3 章为栈和队列,介绍这两种特殊的线性结构的概念与应用;第 4 章为串,介绍串的概念与模式匹配算法;第 5 章为递归,较深入地讨论计算机学科中递归算法的设计方法;第 6 章为数组和广义表,介绍数组、稀疏矩阵和广义表的概念与相关运算算法的实现;第 7 章为树和二叉树,介绍树和二叉树的概念与各种运算算法的实现,其中特别突出二叉树的各种递归算法;第 8 章为图,介绍图的概念和图的各种运算算法的实现;第 9 章为查找,介绍各种查找算法的实现;第 10 章为内排序,介绍各种内排序算法的实现;第 11 章为外排序,介绍各种外排序算法的实现;第 12 章为文件,介绍各类文件的组织结构;第 13 章为采用面向对象的方法描述算法,介绍了面向对象的概念和采用 C++ 语言描述数据结构算法的方法;附录 A 给出 6 个综合实验题;附录 B 给出实验报告格式;附录 C 是引用型参数的说明;附录 D 是顺序表、顺序栈和顺序队列使用指针引用型参数的说明;附录 E 给出书中部分算法清单;附录 F 为教育部颁发的全国计算机专业数据结构 2012 年联考大纲。

数据结构是一门应用性非常强的课程,学生在掌握各种数据结构特别是存储结构的基础上,一定要尽可能多地上机练习,通过实验把难以理解的抽象概念转化为实实在在的计算机能够正确运行的程序,这样才能将所学知识

和实际应用结合起来,吸取算法的设计思想的精髓,提高运用这些知识解决实际问题的能力。因此,本教程突出上机练习内容,除最后一章外其余各章都给出了大量的上机实验题(属验证设计型实验),供教师和学生选用,附录 A 还给出 6 个综合性较强的实验题(属综合设计型实验),目的是全面考查学生综合运用数据结构知识的能力,教师一般可以在本课程学习末期或者在专门的数据结构集中实习课(通常为 36 课时)中向学生布置。

为了便于学习和上机实验,我们还编写了与本教程配套的《数据结构教程学习指导》和《数据结构教程上机实验指导》两本书,与本教程构成一个完整的教学系列。本系列中所有程序均在 Visual C++ 6.0 环境下调试通过。

本教程和配套的上机实验指导、学习指导的编写得到武汉大学教务部“数据结构综合教学改革”教学项目的支持,是本群组许多教师多年来在数据结构课程教学研究和教学改革中的经验与成果的结晶。本教程在编写过程中得到王丽娜、黄传河和黄竞伟等多位教授、博导的大力支持,也得到很多使用本书的老师和同学的热心帮助,作者在此表示衷心感谢。

本书在前 3 版的基础上,根据教育部新的考研大纲和大量读者来信提出的要求进行了修订,例如,广义表在新考纲中没有出现,所以将其合并到数组部分,同时对全书中多个算法进行了优化。另外,编者在 VC++ 6.0 环境中实现了书中各数据结构的基本运算算法和算法设计例题,这些源程序和本书的课件可以从 <http://www.tup.com.cn> 网站免费下载。

由于水平所限,尽管编者不遗余力,仍可能存在错误和不足之处,敬请读者批评指正,特别希望使用本书的教师与作者探讨,共同提高我国计算机专业数据结构课程的教学水平。

编 者

目录

第 1 章 绪论	1
1.1 什么是数据结构	1
1.1.1 数据结构的定义	1
1.1.2 逻辑结构类型	4
1.1.3 存储结构类型	6
1.1.4 数据类型和数据结构	7
1.2 算法及其描述	10
1.2.1 什么是算法	10
1.2.2 算法描述	11
1.3 算法分析	13
1.3.1 算法设计的目标	13
1.3.2 算法效率分析	13
1.3.3 算法存储空间分析	16
1.4 数据结构+算法=程序	17
1.4.1 程序和数据结构	17
1.4.2 算法和程序	17
1.4.3 算法和数据结构	17
1.4.4 数据结构的发展	23
本章小结	24
练习题 1	24
上机实验题 1	26
第 2 章 线性表	27
2.1 线性表及其逻辑结构	27
2.1.1 线性表的定义	27
2.1.2 线性表的抽象数据类型描述	28
2.2 线性表的顺序存储结构	29

2.2.1	线性表的顺序存储结构——顺序表	29
2.2.2	顺序表基本运算的实现	31
2.3	线性表的链式存储结构	38
2.3.1	线性表的链式存储结构——链表	38
2.3.2	单链表	40
2.3.3	双链表	47
2.3.4	循环链表	51
2.4	线性表的应用	52
2.5	有序表	56
2.5.1	有序表的抽象数据类型描述	56
2.5.2	有序表的存储结构及其基本运算算法	57
2.5.3	有序表的归并算法	57
2.5.4	有序表的应用	59
	本章小结	61
	练习题 2	61
	上机实验题 2	61
第 3 章	栈和队列	64
3.1	栈	64
3.1.1	栈的定义	64
3.1.2	栈的顺序存储结构及其基本运算的实现	66
3.1.3	栈的链式存储结构及其基本运算的实现	68
3.1.4	栈的应用	70
3.2	队列	80
3.2.1	队列的定义	80
3.2.2	队列的顺序存储结构及其基本运算的实现	81
3.2.3	队列的链式存储结构及其基本运算的实现	85
3.2.4	队列的应用	89
3.2.5	双端队列	93
	本章小结	95
	练习题 3	95
	上机实验题 3	96
第 4 章	串	99
4.1	串的基本概念	99
4.2	串的存储结构	100
4.2.1	串的顺序存储结构——顺序串	100
4.2.2	串的链式存储结构——链串	105

4.3 串的模式匹配	110
4.3.1 Brute-Force 算法	110
4.3.2 KMP 算法	112
本章小结	119
练习题 4	119
上机实验题 4	119
第 5 章 递归	121
5.1 什么是递归	121
5.1.1 递归的定义	121
5.1.2 何时使用递归	122
5.1.3 递归模型	123
5.1.4 递归与数学归纳法	124
5.2 递归调用的实现原理	125
5.3 递归算法的设计	127
5.3.1 递归算法设计的步骤	127
5.3.2 递归数据结构的递归算法设计	128
5.3.3 递归求解方法的递归算法设计	130
本章小结	131
练习题 5	132
上机实验题 5	132
第 6 章 数组和广义表	133
6.1 数组	133
6.1.1 数组的基本概念	133
6.1.2 数组的存储结构	134
6.1.3 特殊矩阵的压缩存储	136
6.2 稀疏矩阵	138
6.2.1 稀疏矩阵的三元组表示	138
6.2.2 稀疏矩阵的十字链表表示	141
6.3 广义表	145
6.3.1 广义表的定义	145
6.3.2 广义表的存储结构	147
6.3.3 广义表的运算	148
本章小结	151
练习题 6	151
上机实验题 6	152

第7章 树和二叉树	154
7.1 树的基本概念	154
7.1.1 树的定义.....	154
7.1.2 树的逻辑表示方法.....	155
7.1.3 树的基本术语.....	156
7.1.4 树的性质.....	157
7.1.5 树的基本运算.....	158
7.1.6 树的存储结构.....	159
7.2 二叉树的基本概念	162
7.2.1 二叉树的定义.....	162
7.2.2 二叉树的性质.....	163
7.2.3 二叉树与树、森林之间的转换	164
7.3 二叉树的存储结构	167
7.3.1 二叉树的顺序存储结构.....	167
7.3.2 二叉树的链式存储结构.....	168
7.4 二叉树的基本运算及其实现	169
7.4.1 二叉树的基本运算概述.....	169
7.4.2 二叉树的基本运算算法实现.....	169
7.5 二叉树的遍历	172
7.5.1 二叉树遍历的概念.....	172
7.5.2 二叉树遍历递归算法.....	173
7.5.3 二叉树遍历非递归算法.....	177
7.5.4 层次遍历算法.....	182
7.6 二叉树的构造	184
7.7 线索二叉树	189
7.7.1 线索二叉树的概念.....	189
7.7.2 线索化二叉树.....	191
7.7.3 遍历线索化二叉树.....	192
7.8 哈夫曼树	193
7.8.1 哈夫曼树概述.....	193
7.8.2 哈夫曼树的构造算法.....	194
7.8.3 哈夫曼编码.....	195
7.9 用并查集求解等价问题	197
7.9.1 什么叫并查集.....	197
7.9.2 并查集的算法实现.....	199
本章小结.....	201
练习题7	201
上机实验题7	202

第 8 章 图	204
8.1 图的基本概念	204
8.1.1 图的定义	204
8.1.2 图的基本术语	205
8.2 图的存储结构	207
8.2.1 邻接矩阵存储方法	207
8.2.2 邻接表存储方法	208
8.3 图的遍历	211
8.3.1 图的遍历的概念	211
8.3.2 深度优先遍历	211
8.3.3 广度优先遍历	213
8.3.4 非连通图的遍历	214
8.3.5 图遍历算法的应用	215
8.4 生成树和最小生成树	223
8.4.1 生成树的概念	223
8.4.2 无向图的连通分量和生成树	224
8.4.3 普里姆算法	225
8.4.4 克鲁斯卡尔算法	227
8.5 最短路径	231
8.5.1 路径的概念	231
8.5.2 从一个顶点到其余各顶点的最短路径	231
8.5.3 每对顶点之间的最短路径	236
8.6 拓扑排序	240
8.7 AOE 网与关键路径	242
本章小结	246
练习题 8	246
上机实验题 8	247
第 9 章 查找	249
9.1 查找的基本概念	249
9.2 线性表的查找	250
9.2.1 顺序查找	250
9.2.2 折半查找	251
9.2.3 索引存储结构和分块查找	253
9.3 树表的查找	256
9.3.1 二叉排序树	256
9.3.2 平衡二叉树	264
9.3.3 B-树	270

9.3.4 B+树	275
9.4 哈希表查找	276
9.4.1 哈希表的基本概念	276
9.4.2 哈希函数构造方法	277
9.4.3 哈希冲突解决方法	278
9.4.4 哈希表上的运算	281
本章小结	285
练习题 9	285
上机实验题 9	286
第 10 章 内排序	287
10.1 排序的基本概念	287
10.2 插入排序	288
10.2.1 直接插入排序	288
10.2.2 折半插入排序	290
10.2.3 希尔排序	291
10.3 交换排序	293
10.3.1 冒泡排序	293
10.3.2 快速排序	295
10.4 选择排序	299
10.4.1 直接选择排序	299
10.4.2 堆排序	301
10.5 归并排序	305
10.6 基数排序	307
10.7 各种内排序方法的比较和选择	310
本章小结	311
练习题 10	311
上机实验题 10	312
第 11 章 外排序	313
11.1 外排序概述	313
11.2 磁盘排序	314
11.2.1 生成初始归并段	315
11.2.2 多路平衡归并	317
11.2.3 最佳归并树	320
11.3 磁带排序	322
11.3.1 多路平衡归并排序	322
11.3.2 多阶段归并排序	323
本章小结	325

练习题 11	325
上机实验题 11	325
第 12 章 文件	327
12.1 文件的基本概念	327
12.1.1 什么是文件	327
12.1.2 文件的逻辑结构及操作	328
12.1.3 文件的存储结构	328
12.2 顺序文件	328
12.3 索引文件	329
12.3.1 ISAM 文件	330
12.3.2 VSAM 文件	333
12.4 哈希文件	335
12.5 多关键字文件	335
12.5.1 多重表文件	335
12.5.2 倒排文件	336
本章小结	337
练习题 12	337
上机实验题 12	337
第 13 章 采用面向对象的方法描述算法	339
13.1 面向对象的概念	339
13.2 用 C++ 描述面向对象的程序	341
13.2.1 类	341
13.2.2 类对象	343
13.2.3 构造函数和析构函数	344
13.2.4 派生类	347
13.3 用 C++ 描述数据结构算法	349
13.3.1 顺序表类	349
13.3.2 链栈类	352
13.3.3 二叉树类	354
附录A 综合实验题	358
综合实验题 1 链表综合算法设计	358
综合实验题 2 求复杂表达式的值	358
综合实验题 3 用二叉树实现家谱的相关运算	358
综合实验题 4 求无向图中满足约束条件的路径	359
综合实验题 5 分析二分查找成功时的平均查找长度	359
综合实验题 6 求各种排序算法的执行时间	359

附录 B 实验报告格式	360
附录 C 引用型参数的说明	361
附录 D 顺序表、顺序栈和顺序队列使用指针引用型参数的说明	363
附录 E 书中部分算法清单	366
附录 F 全国计算机专业数据结构 2012 年联考大纲	369
参考文献	372

绪 论

第 1 章

“数据结构”是计算机及相关专业的专业基础课之一,是一门十分重要的核心课程,主要学习用计算机实现数据组织和数据处理的方法。它也为计算机专业的后续课程(如操作系统、编译原理、数据库原理和软件工程等)学习打下坚实的基础。

另外,随着计算机应用领域的不断扩大,非数值计算问题占据了当今计算机应用的绝大部分,简单的数据类型已经远远不能满足需要,各数据元素之间的复杂关系已经不是普通数学方程式所能表达的了,且无论是设计系统软件还是应用软件都会用到各种复杂的数据结构。因此,掌握好数据结构课程的知识,对于提高解决实际问题的能力将会有很大的帮助。实际上,一个“好”的程序无非是选择了一个合理的数据结构和一个好的算法,而好的算法的选择很大程度上取决于描述实际问题所采用的数据结构。所以,要想编写出“好”的程序,仅仅学习计算机语言是不够的,必须扎实地掌握数据结构的基本知识和基本技能。

1.1 什么是数据结构

在了解数据结构的重要性之后,我们开始讨论数据结构的定义。本节先从一个简单的学生表例子入手,继而给出数据结构的严格定义,接着分析数据结构的几种类型,最后给出数据结构和数据类型之间的区别与联系。

1.1.1 数据结构的定义

数据是描述客观事物的数和字符的集合。例如,日常生活中使用的各种文字、数字和特定符号都是数据。从计算机的角度看,数据是所有能被输入到计算机中,且能被计算机处理的符号的集合,它是计算机能操作的对象总称,也是计算机处理信息的某种特定的符号表示形式(例如,200902 班学生数据就是该班全体学生记录的集合)。

人们通常以数据元素作为数据的基本单位(例如,200902 班中的每个学

生记录都是一个数据元素)。在有些情况下,数据元素也称为元素、节点、顶点、记录等。有时候,一个数据元素可以由若干个数据项组成。数据项是具有独立含义的最小数据单位,也称为字段或域(例如,200902班中每个数据元素即学生记录是由学号、姓名、性别和班号等数据项组成)。数据对象是性质相同的数据元素的集合,它是数据的子集。

数据结构是指所有数据元素以及数据元素之间的关系,可以看作是相互之间存在着某种特定关系的数据元素的集合,即可把数据结构看成是带结构的数据元素的集合。数据结构包括如下几个方面:

(1) 数据元素之间的逻辑关系,即数据的逻辑结构,它是数据结构在用户面前呈现的形式。

(2) 数据元素及其关系在计算机存储器中的存储方式,即数据的存储结构,也称为数据的物理结构。

(3) 施加在数据上的操作,即数据的运算。

数据的逻辑结构是从逻辑关系(主要是指数据元素的相邻关系)上描述数据的,它与数据的存储无关,是独立于计算机的。因此,数据的逻辑结构可以看作是从具体问题抽象出来的数学模型。

数据的存储结构是逻辑结构用计算机语言的实现或在计算机中的表示(亦称为映像),也就是逻辑结构在计算机中的存储方式,它是依赖于计算机语言的。一般只在高级语言(例如C/C++语言)的层次上讨论存储结构。

数据的运算是定义在数据的逻辑结构之上的,每种逻辑结构都有一组相应的运算。例如,最常用的运算有检索、插入、删除、更新、排序等。数据的运算最终需在对应的存储结构上用算法实现。

因此,数据结构是一门讨论“描述现实世界实体的数学模型(通常为非数值计算)及其之上的运算在计算机中如何表示和实现”的学科。

【例 1.1】 有一个学生表(数据),如表 1.1 所示。这个表中的数据元素是学生记录,每个数据元素由 4 个数据项(即学号、姓名、性别和班号)组成。讨论其存储结构。

表 1.1 学生表

学号	姓名	性别	班号
1	张斌	男	9901
8	刘丽	女	9902
34	李英	女	9901
20	陈华	男	9902
12	王奇	男	9901
26	董强	男	9902
5	王萍	女	9901

解: 该表中的记录顺序反映了数据元素之间的逻辑关系,可以用学号标识每个学生记录,这种逻辑关系可以表示为: $\langle 1, 8 \rangle, \langle 8, 34 \rangle, \langle 34, 20 \rangle, \langle 20, 12 \rangle, \langle 12, 26 \rangle, \langle 26, 5 \rangle$ 。其中 $\langle a_i, a_{i+1} \rangle$ 表示元素 a_i 和 a_{i+1} 之间是相邻的,且 a_i 在 a_{i+1} 之前, a_{i+1} 在 a_i 之后。

这些数据在计算机存储器中的存储方式就是其存储结构。通常可以采用 C/C++ 语言

中的结构体数组和链表两种方式实现其存储结构。

存放上述学生表的结构体数组 Stud 定义如下：

```
struct
{
    int no;           //存储学号
    char name[8];    //存储姓名
    char sex[2];     //存储性别
    char class[4];   //存储班号
} Stud[7] = {{1,"张斌","男","9901"}, ..., {5,"王萍","女","9901"}};
```

数组 Stud[] 中各元素在内存中顺序存放, 即 Stud[i] 存放在 Stud[i+1] 之前, 而 Stud[i+1] 存放在 Stud[i] 之后。

存放学生表的链表的节点类型 StudType 定义为：

```
typedef struct studnode
{
    int no;           //存储学号
    char name[8];    //存储姓名
    char sex[2];     //存储性别
    char class[4];   //存储班号
    struct studnode * next; //存储指向下一个学生的指针
} StudType;
```

学生表中每个学生记录采用一个 StudType 类型的节点存储, 一个学生节点的 next 域指向逻辑结构中它的后继学生记录对应的节点, 从而构成一个链表。其存储结构如图 1.1 所示, 首节点的指针为 head, 用它来标识该学生链表。由 head 所指节点的 next 域得到下一个节点的地址, 然后再由它得到下一个节点的地址, …… , 这样就可以找到任何一个节点的地址。

对于学生表这种数据结构, 可以进行一系列的运算, 例如, 增加一个学生记录、删除一个学生记录、查找性别为“女”的学生记录、查找班号为“9902”的学生记录等。从前面介绍的两种存储结构可以看出, 同样的运算, 在不同的存储结构中其实现过程是不同的。

例如查找学号为 20 的学生的姓名, 对于 Stud 数组, 可以从 Stud[0] 开始比较, Stud[0]. no 不等于 20, 再与 Stud[1]. no 比较, …… , 直到找到 Stud[3]. no 等于 20, 返回 Stud[3]. name。而对于 head 为首节点指针的链表, 从 head 所指节点开始比较, head->no 不等于 20, 从它的 next 域得到下一个节点的地址, 再与下一个节点的 no 域比较, …… , 直到某节点的 no 域等于 20, 返回其 name 域。

从上例可以得出结论: 对于一种数据结构, 其逻辑结构总是唯一的, 但它可能对应多种存储结构, 并且在不同的存储结构中, 同一运算的实现过程可能不同。



图 1.1 学生表的链表存储结构

为了更确切地描述一种数据结构,通常采用二元组表示:

$$B=(D,R)$$

其中, B 是一种数据结构,它由数据元素的集合 D 和 D 上二元关系的集合 R 组成,即:

$$D=\{d_i \mid 1 \leq i \leq n, n \geq 0\}$$

$$R=\{r_j \mid 1 \leq j \leq m, m \geq 0\}$$

其中 d_i 表示集合 D 中的第 i 个数据元素(或节点), n 为 D 中数据元素的个数,特别地,若 $n=0$,则 D 是一个空集,因而 B 也就无结构可言,有时把这种情况认为是具有任意结构。 r_j 表示集合 R 中的第 j 个关系, m 为 R 中关系的个数,特别地,若 $m=0$,则 R 是一个空集,表明集合 D 中的数据元素间不存在任何关系,彼此是独立的,这和数学中集合的概念是一致的。

R 中的一个关系 r 是序偶的集合,对于 r 中的任一序偶 $\langle x, y \rangle (x, y \in D)$,把 x 叫做序偶的第一节点,把 y 叫做序偶的第二节点,称序偶的第一节点为第二节点的前驱节点,称第二节点为第一节点的后继节点。如在序偶 $\langle x, y \rangle$ 中, x 为 y 的前驱节点,而 y 为 x 的后继节点。

若某个节点没有前驱节点,则称该节点为开始节点;若某个节点没有后继节点,则称该节点为终端节点。

对于对称序偶,即 $\langle x, y \rangle \in R$,且 $\langle y, x \rangle \in R (x, y \in D)$,可用圆括号代替尖括号,即 $(x, y) \in R$ 。

【例 1.2】 有一个如表 1.2 所示的城市表,假设区号是唯一的,给出其逻辑结构的二元组表示。

表 1.2 城市表

区 号	城 市 名	说 明
010	Beijing	首都
021	Shanghai	直辖市
027	Wuhan	湖北省省会
029	Xian	陕西省省会
025	Nanjing	江苏省省会

解:城市表中共有 5 个记录,其逻辑结构的二元组表示如下:

$$\text{City}=(D,R)$$

$$D=\{010,021,027,029,025\}$$

$$R=\{r\}$$

$$r=\{\langle 010,021 \rangle, \langle 021,027 \rangle, \langle 027,029 \rangle, \langle 029,025 \rangle\}$$

数据结构还可以利用图形形象地表示出来,图形中的每个节点对应一个数据元素,两节点之间的连线对应关系中的一个序偶。

1.1.2 逻辑结构类型

在不会产生混淆的前提下,常常将数据的逻辑结构简称为数据结构。数据的逻辑结构主要有以下几类。

(1) 集合。

集合是指数据元素之间除了“同属于一个集合”的关系外,别无其他关系。