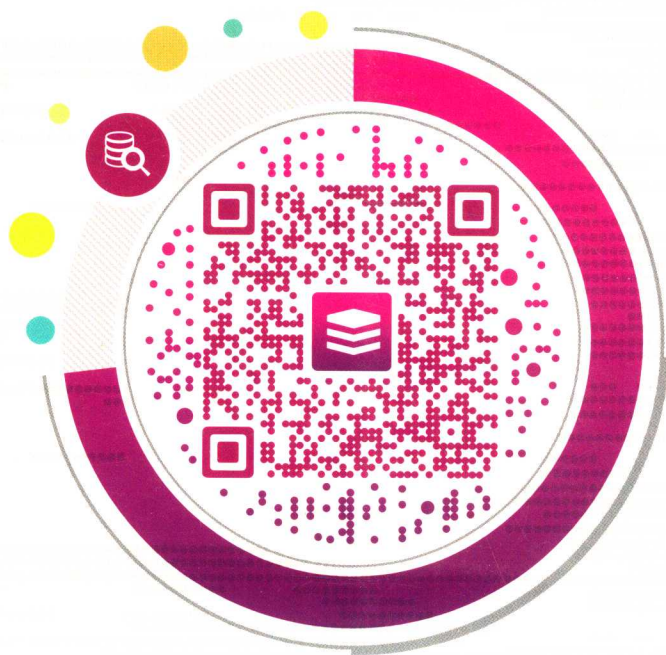


21世纪高等学校计算机类课程创新规划教材 · 微课版



新编数据结构

案例教程(C/C++语言)

微课版

◎ 薛晓亚 主编 周丽平 马金霞 陈延波 副主编

21个综合案例

配套资源丰富

600分钟
63个微课视频

清华大学出版社



内 容 简 介

本书在主要介绍数据的逻辑结构、数据的存储结构、数据的运算等基本知识的基础上,从抽象数据类型的角度,讨论各种基本类型的数据结构及相关应用。

全书共分5篇:第1篇(第1章)为绪论篇,着重介绍数据结构的相关概念和算法的基础知识;第2篇(第2~5章)为线性结构篇,着重讨论线性结构的概念和基本运算的算法实现,介绍了一般的线性结构和特殊的线性结构在不同存储结构之下的基本操作和应用;第3篇(第6章)为树形结构篇,着重介绍基本的树形结构——二叉树在不同存储结构之下如何实现基本操作和应用;第4篇(第7章)为图形结构篇,介绍图形结构在不同存储结构之下的基本操作和应用;第5篇(第8~10章)为数据运算篇,首先介绍数据的查找和排序基本运算的算法实现,接着介绍常见的查找和排序方法,分析并对比它们的算法效率,最后介绍数据结构的基础知识在程序设计竞赛中的应用。全书提供了大量应用实例,每种算法都采用C/C++语言进行描述,帮助读者理解基础理论。

本书叙述清楚,便于教学和读者自学,适合作为高等院校计算机专业及信息相关专业的教材,也可作为计算机应用技术人员的参考书。

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

新编数据结构案例教程: C/C++语言: 微课版/薛晓亚主编. —北京: 清华大学出版社, 2019
(21世纪高等学校计算机类课程创新规划教材·微课版)
ISBN 978-7-302-51089-5

I. ①新… II. ①薛… III. ①C语言—数据结构—高等学校—教材 IV. ①TP311.12 ②TP312.8

中国版本图书馆CIP数据核字(2018)第195636号

责任编辑: 刘 星 常建丽

封面设计: 刘 健

责任校对: 李建庄

责任印制: 李红英

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: 三河市君旺印务有限公司

经 销: 全国新华书店

开 本: 185mm×260mm 印 张: 23.25

字 数: 564千字

版 次: 2019年7月第1版

印 次: 2019年7月第1次印刷

印 数: 1~1500

定 价: 59.00元

产品编号: 077139-01

前 言

一、为什么要写本书

随着信息处理技术和计算机技术的飞速发展,计算机在各个学科和领域得到了广泛的应用,而随着计算机处理的数据量迅速增大,数据类型随之增多,结构复杂的数据和数据关系的处理是我们必须面对的问题,由此设计一个结构好、效率高的软件,就必须分析并设计出好的数据结构,以便优质地处理数据的存储、数据传输和输出处理等操作。

“数据结构”作为计算机专业的一门核心基础课程,是计算机程序设计的重要理论和技术基础。通过本课程的学习,学生不仅可掌握各种组织方式下数据的存储、运算,而且还能使学生熟悉程序设计的基础方法,提高利用数据结构和算法解决实际问题的能力。

二、内容特色

本书有如下特色。

(1) 结构清晰、内容全面、文字描述简单明了、可读性强。

(2) 图文并茂,全书使用 150 余幅图描述数据结构概念、算法的基本思想、算法的执行过程。

(3) 强调数据结构中的 3 种逻辑结构和 2 种存储表示。

全书强调 3 种逻辑结构,即线性结构、树形结构、图形结构,对每一种结构都采用 2 种存储方式(即顺序存储和链式存储)表示,但必须注意每一种逻辑结构要结合其特点选择合适的存储方式表示。

(4) 由浅入深归纳总结每种数据结构的算法设计方法。

例如,利用创建单链表的算法,可以帮助实现链表的逆置、拆分、合并、排序等操作,利用二叉树的遍历算法,可以帮助实现查找结点、计算结点数量、计算二叉树高度、构造二叉树等。同样,图的很多算法都是基于遍历算法的。如果读者掌握了基本算法的设计方法,设计相关问题就容易多了。

三、结构安排

本书共分 5 篇 10 章,具体内容如下。

第 1 篇即第 1 章,为绪论篇。

第 1 章为绪论,主要介绍数据结构的基本概念、数据的存储表示和算法的时间复杂度等内容。

第 2 篇即第 2~5 章,为线性结构篇。

第 2 章为线性表(线性表是最基本的数据结构),主要介绍了线性表的概念、线性表的抽象数据类型、线性表的两种存储结构(顺序表和链表)和常见的基本算法设计,并通过示例深入理解线性表的应用。

第3章为栈与队列,是操作受限制的线性表,主要介绍栈的概念、栈的抽象数据类型、栈的两种存储结构(顺序栈和链栈)和基本运算算法设计、栈的应用算法设计;队列的概念、队列的抽象数据类型、队列的两种存储结构(顺序队列和链队列)和各种基本运算算法设计、队列的应用算法设计。

第4章为串,是特殊的线性表,主要介绍了串的概念、串的存储结构、串的几种基本运算算法设计和串的模式匹配算法设计。

第5章为数组和广义表,主要介绍了数组的概念、数组按行和按列两种存储方式实现、几种特殊矩阵的压缩存储方式、稀疏矩阵压缩存储及转置算法设计;广义表的概念、广义表的存储结构及相关算法设计。

第3篇即第6章,为树形结构篇。

第6章为树和二叉树,主要介绍树的概念及逻辑表示、树的性质、树的存储结构;介绍二叉树的概念、二叉树的性质、二叉树的基本运算算法设计、二叉树的遍历运算算法设计(非递归方式和递归方式)、线索二叉树的概念和构造、哈夫曼树的概念和构造、哈夫曼编码构造等。

第4篇即第7章,为图形结构篇。

第7章为图,主要介绍图的基本概念和逻辑表示、图的存储结构、图的基本运算算法设计、图的遍历算法(DFS和BFS)及相关应用,尤其是工程上常用的最小生成树、最短路径算法、AOV网及AOE网的基本算法等。

第5篇即第8~10章,为数据运算篇。

第8章为查找,主要介绍查找的概念、查找效率的度量标准。本章分为静态表的查找、动态查找表和哈希表查找,分析并对比各种查找方法的查找效率。

第9章为排序,主要介绍排序的概念、排序效率的度量标准,插入排序、交换排序、选择排序、归并排序、基数排序的算法设计,并对各排序算法的时间复杂度和空间复杂度进行分析和比较。

第10章为ACM经典案例,主要介绍以数据结构理论知识为基础的深化应用,探讨如何综合应用数据结构基础知识参与程序设计竞赛,增强学生的竞技精神。

本书的第1、2章由周丽平编写,第3、5、6、7、8、10由薛晓亚编写,第4章由马金霞编写,第9章由陈延波编写。全书由薛晓亚统稿。

四、读者对象

- 对数据结构课程感兴趣的读者。
- 计算机相关专业的本科生、专科生。
- 职业技术类院校计算机相关专业本科生、专科生。

五、致谢

感谢张秀国、高伟林、赵晓庆、田路阳、李冉冉等在本书的资料整理及校对过程中所付出的辛勤劳动。

由于编者的水平和经验有限,加之时间比较仓促,疏漏之处在所难免,敬请读者批评指正。



源代码下载

编者

2019年2月

目 录

第 1 篇 绪 论 篇

第 1 章 绪论	3
1.1 什么是数据结构	3
1.1.1 数据结构的产生与发展	3
1.1.2 数据结构的基本概念	3
1.1.3 逻辑结构的种类	5
1.1.4 数据的存储结构	7
1.2 抽象数据	10
1.2.1 数据类型	10
1.2.2 抽象数据类型的表示与实现	12
1.3 算法及其性能分析	14
1.3.1 算法	14
1.3.2 算法设计的目标	14
1.3.3 算法的时间复杂度度量	16
1.3.4 算法的空间复杂度度量	21
1.4 STL 概述	22
1.4.1 STL 的发展和特点	22
1.4.2 C++ 标准库和 STL	22
1.4.3 数据结构和 STL 的关系	23
1.5 综合案例	24
1.5.1 哥德巴赫猜想问题	24
1.5.2 连续整数问题	26
本章小结	26

第 2 篇 线性结构篇

第 2 章 线性表	29
2.1 线性表的抽象数据类型	29
2.1.1 线性表的定义	29

2.1.2	线性表的抽象数据类型描述	30
2.2	线性表的顺序存储结构	33
2.2.1	线性表的顺序存储结构——顺序表	33
2.2.2	顺序表基本运算的实现	35
2.3	线性表的链式存储结构	43
2.3.1	线性表的链式存储结构——链表	43
2.3.2	单链表基本运算的实现	46
2.3.3	双链表	54
2.3.4	循环链表	58
2.3.5	STL 与链表	60
2.4	综合案例	66
2.4.1	一元多项式的表示及相加运算	66
2.4.2	魔法师发牌问题	67
2.4.3	约瑟夫问题	68
	本章小结	69
第3章	栈与队列	70
3.1	栈	70
3.1.1	栈的概述	70
3.1.2	栈的顺序存储结构	71
3.1.3	栈的链式存储结构	75
3.2	栈综合案例	77
3.2.1	进制转换	77
3.2.2	表达式求值	79
3.2.3	检验表达式中的括号匹配情况	81
3.2.4	栈与递归问题	82
3.3	队列	85
3.3.1	队列的定义和抽象数据类型	85
3.3.2	队列的顺序存储	86
3.3.3	队列的链式存储	91
3.3.4	优先级队列	94
3.4	STL 中的栈与队列	96
3.4.1	STL 中的栈	96
3.4.2	STL 中的队列	97
3.4.3	STL 中的优先队列的使用方法	98
3.5	队列综合案例	99
3.5.1	打印杨辉三角形	99
3.5.2	报数问题	102
3.5.3	舞伴问题	103

本章小结	104
第 4 章 串	106
4.1 串的基本概念和抽象数据类型	106
4.1.1 串的基本概念	106
4.1.2 串的抽象数据类型	107
4.2 串的存储结构	108
4.2.1 串的顺序存储结构——顺序串	108
4.2.2 串的链式存储结构——链串	113
4.3 串的模式匹配	116
4.3.1 串的古典匹配算法	116
4.3.2 串的 KMP 算法	119
4.4 综合案例	124
4.4.1 文本编辑	124
4.4.2 建立词索引表	125
本章小结	128
第 5 章 数组和广义表	129
5.1 数组的定义及抽象数据类型	129
5.1.1 数组的定义	129
5.1.2 数组的抽象数据类型	129
5.2 数组的顺序存储与寻址	130
5.2.1 以行序为主序	131
5.2.2 以列序为主序	132
5.3 特殊矩阵及其压缩存储	133
5.3.1 对称矩阵	133
5.3.2 下(上)三角矩阵	134
5.3.3 对角矩阵	135
5.4 稀疏矩阵	136
5.4.1 稀疏矩阵的三元组表示	136
5.4.2 稀疏矩阵的十字链表表示	141
5.5 广义表	145
5.5.1 广义表的定义	145
5.5.2 广义表的存储结构	146
5.5.3 广义表的运算	148
5.6 综合案例	151
5.6.1 大整数相乘	151
5.6.2 荷兰国旗问题	153
本章小结	154

第3篇 树形结构篇

第6章 树和二叉树	157
6.1 树	157
6.1.1 树的定义	157
6.1.2 树的术语	158
6.1.3 树的基本性质	159
6.1.4 树的抽象数据类型	160
6.2 二叉树	160
6.2.1 二叉树的定义	160
6.2.2 二叉树的性质	162
6.2.3 二叉树的抽象数据类型	163
6.2.4 二叉树的存储结构	164
6.3 二叉树的基本操作	166
6.3.1 中序遍历	166
6.3.2 先序遍历	166
6.3.3 后序遍历	167
6.3.4 层次遍历	167
6.3.5 二叉树遍历的应用	170
6.3.6 二叉树遍历的非递归实现	173
6.4 线索二叉树	181
6.4.1 线索二叉树的概念	181
6.4.2 线索化二叉树	183
6.4.3 遍历线索二叉树	184
6.5 树与森林	185
6.5.1 树的存储结构	185
6.5.2 森林与二叉树的转换	188
6.5.3 树的遍历与森林的遍历	191
6.6 哈夫曼树及其应用	192
6.6.1 哈夫曼树的基本概念	192
6.6.2 哈夫曼树构造算法	192
6.6.3 哈夫曼编码	194
6.7 STL 中实现树结构	197
6.7.1 STL 中的 vector	197
6.7.2 STL 中的 map	200
6.8 综合案例——学校建模问题	203
本章小结	207

第4篇 图形结构篇

第7章 图	211
7.1 图的概念	211
7.1.1 图的定义和术语	211
7.1.2 图的抽象数据类型	216
7.2 图的存储表示	216
7.2.1 邻接矩阵	217
7.2.2 邻接表	218
7.2.3 十字链表	219
7.3 图的遍历与连通性	222
7.3.1 深度优先遍历	222
7.3.2 广度优先遍历	223
7.3.3 连通分量	226
7.4 最小生成树	230
7.4.1 普里姆算法	230
7.4.2 克鲁斯卡尔算法	233
7.5 最短路径	238
7.5.1 单源最短路径	239
7.5.2 全源最短路径	244
7.6 活动网络	248
7.6.1 用顶点表示活动的 AOV 网络	248
7.6.2 AOE 图与关键路径	250
7.7 综合案例	253
7.7.1 道路修建问题	253
7.7.2 回家路线问题	255
7.7.3 棍子还原问题	257
本章小结	259

第5篇 数据运算篇

第8章 查找	263
8.1 查找的基本概念	263
8.2 静态表的查找	264
8.2.1 顺序查找	264
8.2.2 折半查找	265
8.2.3 斐波那契查找	268
8.2.4 分块查找	269

8.3	动态查找表	270
8.3.1	二叉排序树	270
8.3.2	平衡二叉树	278
8.3.3	B-树	284
8.3.4	B+树	290
8.4	哈希表查找	291
8.4.1	哈希表的基本概念	291
8.4.2	哈希函数构造方法	292
8.4.3	哈希冲突解决方法	293
8.4.4	哈希表上的查找分析	296
8.5	STL 中的查找	301
8.6	综合案例——拼写检查问题	302
	本章小结	304
第 9 章	排序	305
9.1	排序的基本概念	305
9.2	插入排序	306
9.2.1	直接插入排序	306
9.2.2	希尔排序	308
9.3	交换排序	310
9.3.1	冒泡排序	311
9.3.2	快速排序	312
9.4	选择排序	315
9.4.1	简单选择排序	315
9.4.2	锦标赛排序	317
9.4.3	堆排序	318
9.5	二路归并排序	322
9.6	基数排序	325
9.7	内部排序方法的比较	328
9.8	STL 中的排序	329
9.9	综合案例——比赛排名问题	330
	本章小结	332
第 10 章	ACM 经典案例	333
10.1	递归算法	333
10.1.1	三柱汉诺塔问题	333
10.1.2	传染病问题	335
10.1.3	N 皇后问题	337
10.2	DFS 与 BFS 问题	341

10.2.1 DFS 之迷宫难题	345
10.2.2 BFS 之管道和指针游戏	349
本章小结	353
附录 A 全国计算机专业数据结构考研大纲	354
参考文献	357

第1篇 绪论篇

数据结构针对非数值计算的程序设计问题,研究计算机的操作对象以及它们之间的关系和操作,是介于数学、计算机硬件和计算机软件三者之间的一门核心课程。学习此课程的目的是了解计算机处理对象的特性,将实际问题中涉及的处理对象在计算机中表示出来并对它们进行处理。例如公交车线路问题,如何换乘车用最短的时间到达目的地,或者如何用最短的距离到达目的地等。本章是数据结构的基础,主要介绍数据结构的基本概念、数据的逻辑结构和存储结构、抽象数据类型和算法性能分析等基础知识。

1.1 什么是数据结构

1.1.1 数据结构的产生与发展

“数据结构”是计算机及相关专业的专业基础课之一,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)的学习打下坚实的基础。

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。因此,在程序设计中,往往遵循“程序=数据结构+算法”的法则。

在了解数据结构的重要性之后,开始讨论数据结构的相关定义。本节先从一个简单的学生表示例入手,然后给出数据结构严格的定义,接着分析数据结构的常见类型,最后给出数据结构和数据类型之间的区别和联系。

1.1.2 数据结构的基本概念

1. 数据

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,数据是指所有能被输入到计算机中,且能被计算机处理的一切符号的集合,它是计算机能操作的对象的总称,也是计算机处理信息的某种特定的



视频讲解

符号表示形式。例如,201702 班学生数据就是该班全体学生记录的集合。

注意:数据的范围是随着计算机的发展而不断扩大的。

人们通常以**数据元素**作为数据的基本单位研究数据。例如,201702 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、结点、顶点或记录。一个数据元素也可以由若干数据项组成。

2. 数据项

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,201702 班中每个数据元素(即学生记录)由学号、姓名、性别、籍贯、电话等数据项组成。数据对象是性质相同的数据元素的集合,是数据的子集。

3. 数据结构

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具有某种或某几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。数据结构包括如下三个方面。

- 描述数据元素之间的逻辑关系,即数据的**逻辑结构**。一般情况下,数据结构和逻辑结构是等价的。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,与计算机硬件无关。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。
- 数据元素及其关系在计算机内存中的存储方式,即数据的**存储结构**,也称为数据的物理结构。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,也称为映像,也就是数据的逻辑结构在计算机中的存储方式是物理结构,它与计算机有关。
- 施加在数据上的操作,即数据的运算。

表 1.1 中,整个学生表为学生数据对象,该数据表中包含 6 个学生记录信息,每一行学生记录(例如,张三的记录信息)代表一个数据元素。张三的记录信息中包含学号、姓名、性别、籍贯、电话 5 个数据项。

表 1.1 学生数据表

学 号	姓 名	性 别	籍 贯	电 话
01	张三	男	北京	666301
02	李四	男	南京	666302
03	王五	男	北京	666301
04	赵六	女	郑州	666303
05	钱七	女	青岛	666303
06	孙八	女	郑州	666306

一个数据元素

} 一个数据项

为了更确切地描述数据结构,通常采用二元组表示。

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

其中,Data_Struct 是一种数据结构,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(或结点), n 为 D 的基数, 即数据元素的个数。特别地, 当 $n=0$ 时, 表示 D 是一个空集, 因而 $Data_Struct$ 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, 当 $m=0$ 时, 表示 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。

1.1.3 逻辑结构的种类



视频讲解

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。根据元素间的关系, 将数据的逻辑结构分为以下 4 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。例如, 整数集合、字母表集合。

如图 1.1 所示, 元素 A 、 B 、 C 、 D 、 E 除了同属于一个集合外, 相互孤立, 无其他关系。

2. 线性结构

线性结构是指该结构中的结点之间存在一对一的关系。其特点是: 开始结点和终端结点都是唯一的, 除了开始结点和终端结点以外, 其余结点都有且仅有一个前驱, 有且仅有一个后继。线性表就是一种典型的线性结构。

如图 1.2 所示, 线性结构中每个数据结点有且仅有一个前驱结点(除第一个结点外), A 结点为第一个结点(首元结点), B 的前驱结点为 A , C 的前驱结点为 B , 其他以此类推。这种数据结构的特点就是结点之间存在一对一的关系, 即线性关系, 因此是一种线性结构。例如, 表 1.1 中每一行记录之间的关系, 或者生活中在食堂排队打饭, 在银行排队办理业务等均均为典型的线性结构。

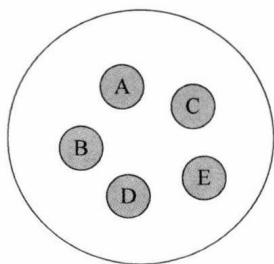


图 1.1 集合结构图示



图 1.2 线性结构图示

3. 树形结构

树形结构是指该结构中的结点之间存在一对多的关系。其特点是每个结点最多只有一个前驱, 但可以有多个后继, 且终端结点可以有多个。从上到下结点间是一对多的关系, 从下到上结点间是一一对一的关系, 二叉树就是一种典型的树形结构。

如图 1.3 所示, 树形结构的根结点有且仅有一个(R_1 为根结点), 每个结点有且只有一个前驱结点(除树根结点外), 但可以有多个后继结点(叶结点可看作具有 0 个后继结点)。

图中,R1的后继结点为R2和R3,R2的后继结点为R4和R5,R4和R5的前驱结点为R2,R2和R3的前驱结点为R1。例如,家族的族谱、公司的人事组织结构等均为典型的树形结构。

4. 图形结构

图形结构是指该结构中的结点之间存在多对多的关系。其特点是:每个结点的前驱和后继的个数都可以是任意的。图形结构可能没有开始结点和终端结点,也可能有多个开始结点、终端结点。

如图1.4所示,每个结点有多个前驱结点和多个后继结点。结点北京的后继结点为成都,结点北京的前驱结点为郑州和沈阳,结点郑州的后继结点为北京和沈阳,结点郑州的前驱结点为成都和武汉,其他结点以此类推。例如,青岛市各建筑物的分布图、公交换乘线路等均为典型的图形结构。

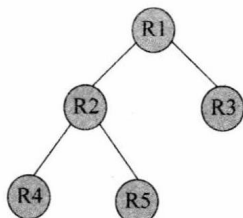


图 1.3 树形结构图示

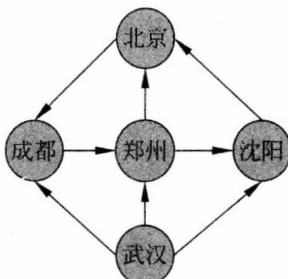


图 1.4 图形结构图示

注意：树形结构和图形结构统称为非线性结构,该类结构中的结点之间存在一对多或多对多的关系。由图形结构、树形结构和线性结构的定义可知,线性结构是树形结构的特殊情况,而树形结构又是图形结构的特殊情况。

【例 1.1】 有一种数据结构 $B1 = (D, R)$, 其中:

$$D = \{a, b, c, d, e\}$$

$$R = \{\langle a, b \rangle, \langle b, c \rangle, \langle c, d \rangle, \langle d, e \rangle, \langle e, a \rangle\}$$

画出其逻辑结构表示。

解：B1 对应的逻辑结构图如图 1.5 所示。

从该例中可以看出,每个结点都有一个前驱结点和一个后继结点。这种数据结构的特点是:每个结点的前驱和后继的个数都是任意的,数据元素之间是多对多的关系,即图形结构。

【例 1.2】 有一种数据结构 $B2 = (D, R)$, 其中:

$$D = \{a, b, c, d, e\}$$

$$R = \{\langle a, b \rangle, \langle b, c \rangle, \langle c, d \rangle\}$$

画出其逻辑结构表示。

解：B2 对应的逻辑结构图如图 1.6 所示。

从该例中可以看出,结点 a、b、c、d 之间满足线性结构关系,而结点 e 和其他结点不存在逻辑关系,可以认为,结点 e 和其他结点的关系是任意的,于是整个数据结构的特点满足数据元素之间是多对多的关系,即图形结构。