

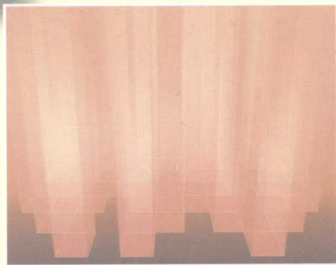


普通高等教育“十一五”国家级规划教材
高 职 高 专 计 算 机 系 列

数据结构

(Java语言版)

王学军 主编
宋汉珍 主审





清华大学出版社 “十二五” 国家重点图书出版规划项目
清华大学出版社 “十二五” 国家重点图书出版规划项目

数据结构

(Java版)

清华大学出版社
清华大学出版社



清华大学出版社

普通高等教育“十一五”国家级规划教材

高职高专计算机系列

数据结构（Java 语言版）

王学军 主编

宋汉珍 主审

人民邮电出版社

北 京

图书在版编目(CIP)数据

数据结构: Java 语言版 / 王学军主编. —北京: 人民邮电出版社, 2008.8

普通高等教育“十一五”国家级规划教材. 高职高专计算机系列

ISBN 978-7-115-18577-8

I. 数… II. 王… III. ①数据结构—高等学校: 技术学校—教材②JAVA 语言—程序设计—高等学校: 技术学校—教材 IV. TP311.12 TP312

中国版本图书馆 CIP 数据核字(2008)第 114291 号

内 容 提 要

本书共分 10 章, 重点介绍 3 种基本数据结构及其应用, 主要内容包括绪论、Java 语言基础知识、线性表、栈和队列、数组和广义表、串、树与二叉树、图、查找和排序等。本书采用 Java 语言描述数据结构中的算法, 每章配有一定数量的具有完整程序的实例, 并在最后提供难易适中、与所讲理论知识相配套的习题, 帮助读者学习和理解理论知识。

本书面向高等职业院校学生, 语言通俗易懂, 每章都由实例引入, 理论和实践紧密结合。全书重点突出基本理论和基本算法的实现过程, 强调实践性和实用性。另外本书配有电子教案和习题解答, 可从人民邮电出版社的网站(www.ptpress.com.cn)下载。

本书可作为高职高专院校计算机及相关专业“数据结构”课程的教材, 也可作为各类计算机培训班的教材。

普通高等教育“十一五”国家级规划教材
高职高专计算机系列

数据结构 (Java 语言版)

-
- ◆ 主 编 王学军
 - 主 审 宋汉珍
 - 责任编辑 张孟玮
 - 执行编辑 王亚娜
 - ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号
 - 邮编 100061 电子函件 315@ptpress.com.cn
 - 网址 <http://www.ptpress.com.cn>
 - 北京隆昌伟业印刷有限公司印刷
 - ◆ 开本: 787×1092 1/16
 - 印张: 15.75
 - 字数: 384 千字 2008 年 8 月第 1 版
 - 印数: 1—3 000 册 2008 年 8 月北京第 1 次印刷

ISBN 978-7-115-18577-8/TP

定价: 25.00 元

读者服务热线: (010)67170985 印装质量热线: (010)67129223

反盗版热线: (010) 67171154

前 言

“数据结构”是计算机及相关专业的重要专业基础课程。随着计算机技术的不断发展,数据结构的内容也需要做适当调整,重点要提高学生的实践能力。本书就是以计算机及相关专业学生必须掌握的知识为核心,以高等职业教育所培养的学生应该具备的能力为依据,以突出实践性和实用性为目的,进行设计编写的。

本书编写人员都是长期从事高等职业教育计算机教学的一线教师,具有丰富的教学 and 实践经验,同时编写组中还有在企业从事过 Java 软件开发的工程人员。

本书的主要特色如下。

(1) 内容选取合理,组织得当。本书根据高等职业教育计算机人才培养目标组织内容,理论部分以够用为度,重点突出实践性和实用性。每章都由实例引入,并且配备一定数量的扩展实例(其中包括一部分工程实例),有助于对理论知识的消化、理解。

(2) 算法实现方式先进,适合学生学习。本书采用 Java 语言作为数据结构算法的实现语言,体现面向对象语言的特色,使数据结构的相关算法实现更加方便,教学效果更加突出。

(3) 定位准确、恰当,适合高职高专院校学生使用。本书主要面向高职高专院校学生,目的是把学生培养成高等技术应用型生产一线人才。

(4) 突出适合高等职业教育的任务驱动方法。本书所有内容都以学习任务的方式给出,帮助教师把握在教学中应该注意的重点、难点。

本书建议学时为 70~90 学时,其中理论学时为 50~60 学时,实践环节学时为 20~30 学时。授课时应以任务驱动为主,理论结合实际,突出实例在本课程教学中的作用。

本书由王学军担任主编,董国增、郑阳平、张暑军担任副主编。编写任务分工如下:第 1、3、4、9 章由王学军编写,第 7 章由董国增编写,第 2、5、8 章由郑阳平编写,第 6、10 章由张暑军编写。宋汉珍教授担任主审。全书由王学军、郑阳平统稿,张暑军对书中的程序进行了审核。主审宋汉珍教授对本书的编写计

划、大纲及全书内容进行了详细审阅，并提出了许多建议。另外，在本书的编写和前期研讨过程中，还得到了马晓晨、谢懿、郝春雷、李海明、牟学鹏、张清涛、李杰等教师的大力帮助；同时，在程序调试和校验审核方面也得到了何林芳、王小红、薛显、郭开文、马永明、程瑞、胡国庆等同学的大力支持。在此一并表示感谢！

由于作者水平有限，书中难免存在不妥之处，敬请读者批评指正。

编者

2008年5月

目 录

第 1 章 绪论	1	2.4.8 类的多态性	27
1.1 数据结构的 3 种基本结构	1	2.4.9 抽象类和内部类	28
1.1.1 线性结构	1	2.4.10 接口	29
1.1.2 层次结构	3	2.4.11 包	29
1.1.3 网状结构	4	2.4.12 异常处理	31
1.2 数据结构研究的主要问题	6	2.4.13 Java 标准数据流	33
1.3 算法及描述	7	2.5 Java 语言中的“指针”实现	35
1.3.1 算法与算法特性	7	2.6 JDK1.5 新增特性	38
1.3.2 算法表示	8	2.6.1 泛型	39
1.4 算法效率分析	10	2.6.2 增强的集合遍历结构	39
习题	11	2.6.3 自动装箱/拆箱	39
		2.6.4 枚举类型	40
第 2 章 Java 语言基础知识	13	2.6.5 静态 import	40
2.1 实例引入	13	2.6.6 从终端读取数据	41
2.2 Java 语言概述	14	2.6.7 格式化输出	41
2.3 面向对象程序设计简述	14	2.6.8 可变参数	41
2.3.1 面向对象程序设计的 基本概念	15	习题	42
2.3.2 面向对象程序设计的 基本特征	15	第 3 章 线性表	44
2.4 Java 语言基础知识	16	3.1 实例引入	44
2.4.1 数据类型	17	3.2 线性表的概述	45
2.4.2 运算符	18	3.2.1 线性表的概念	45
2.4.3 流程控制	18	3.2.2 线性表的存储结构及 操作	46
2.4.4 数组	21	3.3 顺序表的基本操作及实现	46
2.4.5 类与对象	23	3.3.1 顺序表的概述	46
2.4.6 类的封装性	26	3.3.2 顺序表的基本操作及 实现	47
2.4.7 类的继承性	27		

3.4 链表的基本操作及实现	50	5.3 特殊矩阵	88
3.4.1 链表	50	5.3.1 对称矩阵	89
3.4.2 链表的分类	52	5.3.2 三角矩阵	90
3.4.3 单链表的基本运算及 实现	53	5.3.3 对角矩阵	91
3.4.4 其他形式的链表的 相关运算	54	5.4 稀疏矩阵	91
3.4.5 算法实例	55	*5.5 广义表	98
3.5 线性表的应用	56	5.5.1 广义表的概念	98
3.5.1 顺序表的连接	56	5.5.2 广义表的存储结构	100
3.5.2 字符串的逆转算法	57	习题	100
习题	60		
第 4 章 栈和队列	61	第 6 章 串	102
4.1 实例引入	61	6.1 实例引入	102
4.2 栈的相关概述	62	6.2 串的概述	103
4.2.1 栈的定义	62	6.3 串的顺序存储结构	104
4.2.2 栈的相关概念	62	6.3.1 通过 String 类处理串	104
4.2.3 栈的操作过程	62	6.3.2 通过 StringBuffer 类 处理串	106
4.2.4 栈的存储结构	64	6.4 串的链式存储结构	108
4.3 用数组实现顺序栈及操作	65	6.4.1 链串的实现	108
4.4 用类实现链式栈及相应操作	67	6.4.2 链串基本算法	109
4.5 队列的相关概述	70	习题	111
4.5.1 队列的定义	70		
4.5.2 队列的相关概念	70	第 7 章 树与二叉树	113
4.5.3 队列的存储结构	70	7.1 实例引入	113
4.6 用数组实现顺序队列及 相应操作	71	7.2 树	114
4.7 用类实现链队列及相应操作	74	7.2.1 树的定义	114
4.8 栈和队列的实例应用	76	7.2.2 树的表示方法	115
习题	81	7.2.3 树的抽象数据类型	116
		7.2.4 树的存储结构	117
		7.3 二叉树	119
		7.3.1 二叉树的定义	119
		7.3.2 二叉树的性质	120
		7.3.3 二叉树的抽象数据 类型	122
		7.3.4 二叉树的存储结构	122
		7.4 二叉树的节点类及二叉树类	124
		7.4.1 二叉树节点类	124
		7.4.2 二叉树类	125
第 5 章 数组和广义表	82		
5.1 实例引入	82		
5.2 数组	83		
5.2.1 数组的基本概念	83		
5.2.2 一维数组	84		
5.2.3 二维数组	84		

7.5 二叉树的遍历	126	8.5.3 Prim 算法	175
7.5.1 二叉树遍历算法	126	8.6 最短路径问题	177
7.5.2 二叉树遍历算法的实现	127	8.7 拓扑排序	183
7.5.3 非递归的二叉树遍历算法	129	8.7.1 有向无环图	183
7.5.4 二叉树遍历的应用	130	8.7.2 拓扑排序	183
7.6 线索二叉树	131	8.8 AOE 网与关键路径	187
7.6.1 线索二叉树的定义	132	8.8.1 AOE 网	187
7.6.2 线索二叉树的存储结构	132	8.8.2 关键路径	187
7.6.3 遍历线索二叉树	133	8.9 综合示例	191
7.6.4 构造中序线索二叉树	134	习题	200
7.7 树和森林	135	第 9 章 查找	203
7.7.1 树、森林与二叉树的转换	135	9.1 实例引入	203
7.7.2 树和森林的遍历	137	9.2 基本概念与术语	205
7.8 树的应用	138	9.2.1 查找的概念	205
7.8.1 二叉排序树	138	9.2.2 查找方法	205
7.8.2 哈夫曼树和哈夫曼编码	144	9.3 顺序查找法	205
*7.8.3 判定树	152	9.4 折半查找法	207
习题	154	9.5 二叉排序树法	209
第 8 章 图	156	9.6 哈希查找法	213
8.1 实例引入	156	9.6.1 哈希查找概念	214
8.2 图的基本概念	157	9.6.2 哈希函数	215
8.2.1 图的定义	157	9.6.3 冲突解决方法	217
8.2.2 图的相关概念	159	9.7 应用实例	219
8.3 图的存储结构	162	习题	222
8.3.1 邻接矩阵	162	第 10 章 排序	224
8.3.2 邻接表	164	10.1 实例引入	224
8.4 图的遍历	165	10.2 排序的概念	225
8.4.1 深度优先搜索遍历	165	10.3 排序的分类	225
8.4.2 广度优先搜索遍历	169	10.3.1 按照存储交换分类	225
8.5 生成树和最小生成树	173	10.3.2 按照内部排序的过程分类	226
8.5.1 生成树	173	10.3.3 按照排序的稳定性分类	226
8.5.2 Kruskal 算法	174	10.4 插入排序	226
		10.4.1 直接插入排序	226
		10.4.2 希尔排序	228

10.5 交换排序.....	230	10.7.1 归并排序	237
10.5.1 冒泡排序	230	10.7.2 基数排序	238
10.5.2 快速排序	231	10.8 排序的工程应用举例	240
10.6 选择排序.....	233	习题	242
10.6.1 直接选择排序	233	参考文献	244
10.6.2 堆排序	234		
10.7 其他排序.....	236		

第 1 章 绪论

【内容简介】

本章通过实例引入数据结构的概念，主要介绍线性结构、层次结构（树形结构）以及网状结构（图形结构）等常用数据结构的基本概念，算法的概念以及描述算法的一般规则，算法的时间复杂度和空间复杂度的简单分析与评价等。

【知识要点】

- ◇ 数据结构中的常用术语；
- ◇ 线性结构、层次结构和网状结构的结构特点；
- ◇ 算法的定义、特性以及描述规则；
- ◇ 时间复杂度、空间复杂度的定义以及评价规则。

【教学提示】

本章作为该课程的入门，共设 3 学时，对于有 Java 语言基础的学生，可安排 2 学时，重点讲解数据结构的基本概念及相关知识。在本章的学习中，要注意对数据结构概念的引入，重点是对实例的理解以及如何利用 Java 语言知识来理解数据结构的相关概念。算法的空间复杂度等相关内容可作为选学内容。

1.1 数据结构的 3 种基本结构

【学习任务】 了解该实例的含义，进而初步了解数据结构的相关概念。

【例 1.1】 某大学拟建立校园网络，设计了如图 1.1 所示的网络拓扑结构图。

现对该网络拓扑结构图进行分析。首先通过观察发现，该图中有若个交换机，需要了解其性能参数、接口配置、相互之间的联系等信息。下面通过从该校园网中交换机的基本信息、交换机之间的层次关系、交换机之间的传输距离等问题着手，引入数据结构中 3 种基本结构的概念。

1.1.1 线性结构

1. 通过对交换机信息分析，引入线性结构

该校园网的交换机信息如表 1.1 所示。通过该表可以看出，每个交换机的信息构成了一

个整体，而这些交换机信息又构成了一个整体，而单纯从这些信息角度看，它构成一种顺序关系，称其为线性结构。

表 1.1 某学校校园网的交换机信息

名 称	性 质	数 量	传 输 速 率
STAR-S6808	核心路由交换机	1	1000Mbit/s
S3550-12G	楼宇交换机	6	1000Mbit/s
STAR-1926G+	楼层交换机	若干	100Mbit/s

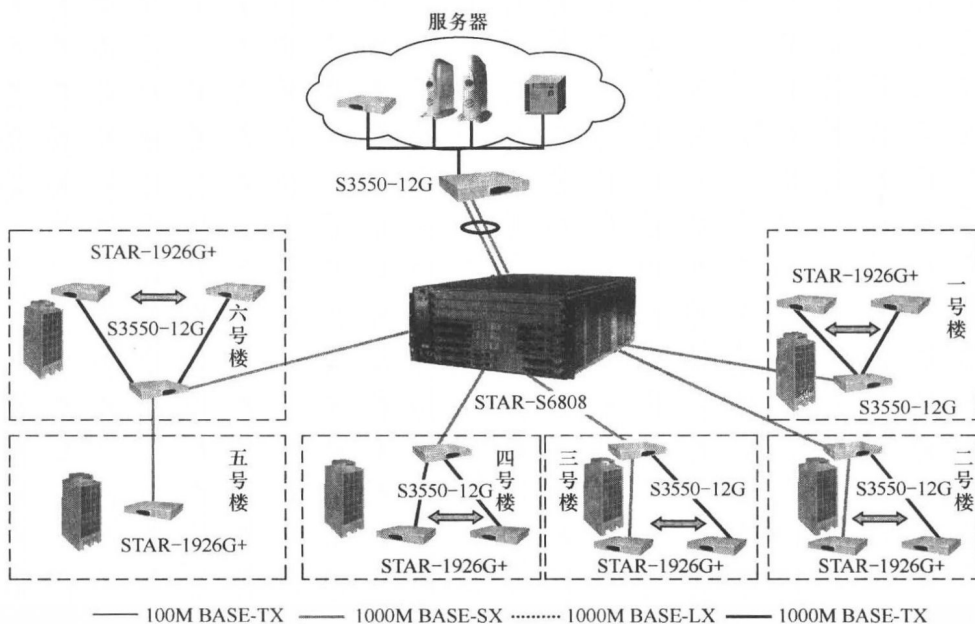


图 1.1 某学校校园网络拓扑结构图

在日常生活经常会遇到和上面结构类型相似的表结构，例如工资管理系统、人事管理系统、仓库库存管理系统、图书管理系统等。

【例 1.2】 图书管理系统。某学校图书信息包括图书编号、书名、数量和价格等方面的信息，如表 1.2 所示，一行表示一条数据记录（简称记录），即表示某种图书的信息；一列代表一个属性，称其为字段，表示该记录中某一方面的属性。每种图书信息的位置有先后次序，它们之间形成一种线性关系。

表 1.2 某学校的图书信息系统

图 书 编 号	书 名	数 量	价 格
200101	计算机组成原理	35	35
200405	大学英语	35	28
200436	高等数学	140	24
200617	计算机基础	70	18

续表

图书编号	书名	数量	价格
...	...	...	
200705	机械设计基础	35	21

对上述具有线性结构的信息可以进行的主要操作有查找系统中的某个信息、修改系统中的某条记录信息、在固定的位置插入和删除相应的数据信息等，即查询、插入、删除、修改等相关操作。

2. 数据的相关概念

数据是数据结构最基本的概念，数据的构成及数据的性质是掌握数据结构概念的基础。数据分为数值型数据和非数值型数据。数据通过编码成为能被计算机识别、存储和处理的符号。根据数据的不同划分和分类，可以得出数据的一组相关概念。

数据(Data)是描述客观事物的数据集合。例如在【例 1.2】中，每个描述图书的记录就是一个数据。这些数据有一个共同的特点，即它们都是可以被输入到计算机中并能被计算机识别、存储和处理的符号。

数据元素(Data Element)是构成数据的基本单位。有些数据是由单个元素构成的，例如{1, 2, 3, 4, 5, ..., 100}中的每个数字就是一个数据，而有些数据是由一些元素构成的。对于【例 1.1】中交换机的信息和【例 1.2】中描述图书的信息都是由一组数据构成的。

数据项(Data Item)是数据结构中的最小单位。当数据元素由多个项构成时，其每个分项称为数据项，例如，图书信息系统中的图书编号、书名、数量、价格等都是数据项。

数据对象(Data Object)是指相同性质的数据元素构成的集合。在【例 1.1】中的交换机信息和【例 1.2】中的图书信息，都具有相同的性质和相同的数据类型，这样的数据构成的集合就是一个数据对象。

1.1.2 层次结构

1. 通过校园网交换机之间的层次关系，引入层次结构

按照交换机之间的管理和被管理的关系，形成了一种层次结构（也称为树形结构），如图 1.2 所示。每个交换机都称做该结构中的节点，节点之间形成了一对多的树形关系。

和如图 1.2 所示的结构类似的还有计算机目录之间的关系、公司部门的结构关系等。

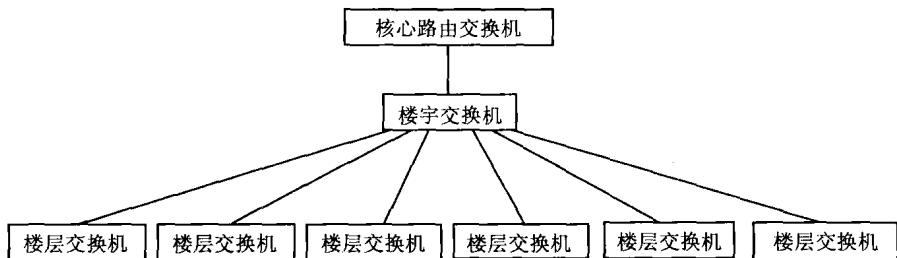


图 1.2 某校园网交换机之间的层次关系示意图

【例 1.3】计算机某磁盘（以 C 盘为例）的目录结构如图 1.3 所示，该磁盘的根目录下有 4 个子目录（USER、WINDOWS、DOWNLOADS、WMPUB），每个子目录下面又设有两个子目录，它们之间形成了一种层次关系，这就形成了一种树形结构，每个目录都称做该结构中的节点，节点之间形成了一对多的关系。

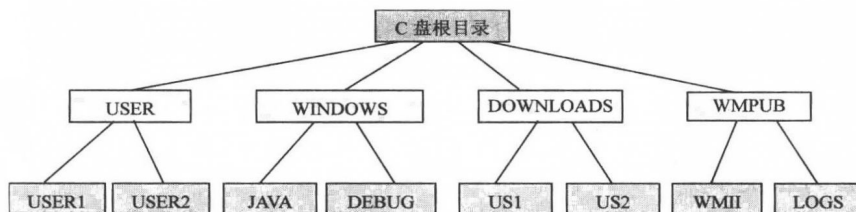


图 1.3 计算机磁盘目录结构示意图

对树形结构可以进行的操作主要有：节点的查找、节点信息的修改、节点的插入和删除等。

2. 数据结构的相关概念

通过上面的两个例子可以看出，数据是构成数据结构的基础，而数据之间的关系是进行数据操作的基础。下面介绍数据结构的相关概念。

数据结构 (Data Structure) 是指具有某种联系的数据元素以及元素之间所构成的各种关系组成的集合。数据元素不是孤立存在的，正因为它们在它们之间总存在某种相互关系，才构成了数据元素之间的各种关系，这些关系称为结构。数据的结构可分为数据的逻辑结构和数据的物理结构。

逻辑结构 (Logical Structure) 是指构成数据结构的数据元素相互之间本身具有的逻辑关系，例如【例 1.1】中的交换机信息和【例 1.2】中的图书信息就都是线性关系，图 1.2 中交换机之间的层次关系和图 1.3 中目录之间的层次关系都属于逻辑关系。物理结构（或存储结构）是指构成数据结构的数据元素及其关系在计算机中的描述和表示。一种数据结构可对应一种或多种物理结构。

3. 数据结构的描述

数据结构是由两个集合构成的一个二元组 $\langle D, R \rangle$ 。其定义如下：

$$\langle D, R \rangle = \{$$

$$\{d_i | 1 \leq i \leq n, n \geq 1\} \quad // \text{表示构成数据结构的数据元素的集合，其中 } d_i \text{ 表示第 } i \text{ 个数据元素，}$$

$$\quad \quad \quad n \text{ 为 } D \text{ 中数据元素的个数}$$

$$\{r_j | 1 \leq j \leq m, m \geq 1\} \quad // \text{表示数据元素之间的各种关系，} r_j \text{ 表示数据元素之间的第 } j \text{ 个关系，}$$

$$\quad \quad \quad m \text{ 为 } D \text{ 上的关系个数}$$

$$\}$$

1.1.3 网状结构

1. 通过对交换机之间的位置和距离进行分析，引入图形结构

图 1.4 所示为另一学校的网络拓扑结构，在其网络布线和施工的过程中，需要了解交换

机之间的位置，以及它们之间的距离等关键性的问题。

由图 1.4 可看出，各节点之间形成了一种网状结构（也称为图形结构），该结构的特点是每个节点之间都可以建立联系，形成了一种多对多的网状关系。与该结构类似的还有交通图、地图、通信网络图等。

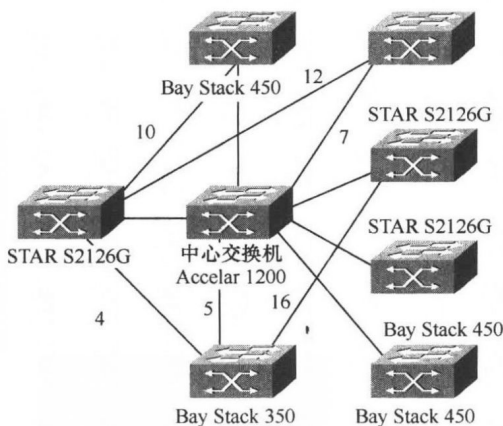


图 1.4 某学校校园网交换机之间的位置和距离示意图

【例 1.4】 哥尼斯堡七桥问题。在 18 世纪的东普鲁士的哥尼斯堡城市，有条横贯全城的普雷格尔河和两个岛屿，在河的两岸与岛屿之间架设了 7 座桥，把它们连接起来（见图 1.5 所示）。将如图 1.5 所示的问题抽象成一个数学问题，就得到如图 1.6 所示的无向图。

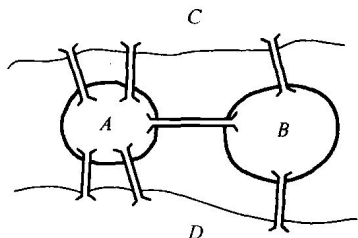


图 1.5 哥尼斯堡七桥问题示意图

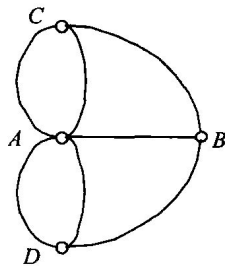


图 1.6 七桥问题抽象化示意图

由图 1.5 可以看出，A、B、C、D 4 个节点之间都可以产生联系，即多对多的关系，这就是数据结构中的网状结构（也称为图形结构）。

在网状结构中可以进行的操作有：检索顶点，查找某顶点到其他顶点之间的路径，求最短距离，求关键路径等。数据结构既可以描述数值型数据的特点，也可以描述非数值型数据的计算问题。

2. 数据结构的分类

根据数据结构中相关数据元素之间的不同关系，可将数据结构分为线性结构、树形结构和图形结构 3 类。

- ① 线性结构：该结构的数据元素之间形成一对一的线性关系，【例 1.2】就是线性结构。
- ② 树形结构（层次结构）：该结构的数据元素之间存在着一对多的关系，【例 1.3】就是

树形结构。

③ 图形结构 (网状结构): 该结构的数据元素之间存在着多对多的关系, 【例 1.4】就是网状结构。

数据类型 (Data Type) 和数据结构之间有着密切的联系, 在许多由高级语言编写的程序中, 对于常量、变量或表达式都有一个所属的确定数据类型 (即本身的数据特点)。根据数据特点, 决定了它在程序执行期间的取值范围, 以及在这些值上允许进行的操作。在数据结构中, 专门提出抽象数据类型这个概念, 用来描述数学模型及相关操作。

抽象数据类型 (Abstract Data Type, ADT) 是对特定数学模型的数学描述, 包括操作的数据, 数据之间的关系, 以及在该关系上可以实现的操作, 即数据类型一般由元素、关系及操作三要素组成。抽象数据类型独立于各种操作的具体实现。它的优点是将数据和操作封装在一起, 使得用户程序只能通过 ADT 里定义的某些操作来访问其中的数据, 从而实现了信息隐藏。在 Java 语言中, 可以用类的说明来表示 ADT, 用类的实现来实现 ADT。

抽象数据类型可描述如下:

ADT 抽象数据类型名

```
{
    数据对象: (数据对象的定义);
    数据关系: (数据关系的定义);
    基本操作: (基本操作的定义);
};
```

【例 1.5】 【例 1.2】 中线性结构对应的抽象数据类型定义如下:

ADT book_list {

数据元素: 每条图书记录 x_i 为一数据元素, $i=1, 2, \dots, n$ ($n \geq 1$)。

逻辑结构: 所有数据元素 x_i 存在线性关系 (x_i, x_{i+1}) , x_1 无前趋, x_n 无后继。

基本操作:

```
Initbook_List(L);           //建立空线性表
Lengthbook_List(L);         //求线性表长度
GetElement(L,i);            //取线性表中的第 i 个元素
Searchbook_List(L,x);       //确定元素 x 在线性表 L 中的位置
Insertbook_List(L, i,x);     //在线性表中第 i 个位置插入数据元素 x
Deletebook_List(L,i);        //删除线性表中第 i 个位置的数据元素
...
};
```

1.2 数据结构研究的主要问题

【学习任务】 了解数据结构所研究的主要问题, 为后面的学习奠定基础。

数据结构是一门研究数据的各种逻辑结构和存储结构, 以及对数据各种操作的课程, 即研究数据的逻辑结构、数据的物理结构以及对数据的操作实现。所有的计算机系统软件和应用软件的编写都要用到各种类型的数据结构。数据结构将解决计算机中处理的大量数据元素, 大量的数据类型, 以及越来越复杂的数据之间的关系。

例如，在【例 1.2】的图书管理系统中，每一种类的书和其他书之间整体上构成了前后对应的线性关系，即逻辑结构，同时还要了解这样的数据如何在计算机中进行存储，即存储结构（假设使用数组进行存储），这样才可以讨论其相应操作，如插入、删除、查询、修改等。

在高级程序设计语言中，数据结构描述的是所能表示并可以存储的数据种类和数据实体（即物体），数据实体是指在一种数据类型中的所有数据构成的集合；数据结构就是数据实体（即物体）中各元素之间的关系。

例如，数组 $A=\{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$ 表示的是小于等于 10 的自然数集合，其中每个数是数据实体，而各元素之间的关系就是前后元素之间的一对一关系，即 1 的后面是 2，2 的后面是 3，……，这样就在集合 A 中实现了数据之间的线性关系。

数据结构是计算机类专业的一门专业基础的课程，是学习操作系统、数据库原理等专业课的基础。数据结构涉及了数学范围的诸多知识，计算机硬件范围的编码理论、存取装置和存取方法等知识，软件范围的文件系统、数据的动态存储管理和信息管理 etc. 知识。所以说数据结构是介于数学、计算机硬件及软件三者之间的一门核心课程。许多学校将数据结构课程设为计算机类专业的主干课程。

1.3 算法及描述

【学习任务】 算法的理解及其特征，重点掌握由数学问题演变成算法的过程，以及算法的特征及评判标准。

当利用计算机来解决一个具体问题时，一般需要经过如下几个步骤：首先要从具体问题抽象出一个适当的数学模型，然后设计或选择算法来解决此问题，最后进行编程，直至得到正确的解答，这就是所谓的算法。随着计算机应用领域的扩大和软、硬件的发展，非数值计算问题显得越来越重要。据统计，当今处理非数值计算性问题占用了 90% 以上的机器时间。这类问题涉及的数据元素之间的相互关系更为复杂，解决这类问题的关键是要设计出适合的数据结构，这样才能有效地解决问题。算法的设计取决于数据的逻辑结构，算法的实现取决于数据的物理存储结构。同时还要考虑执行算法时的时间和空间效率。

1.3.1 算法与算法特性

根据数据结构所研究问题的性质，可以得出如下的结论：

数据结构=逻辑结构+存储结构+相应算法（或操作）

因此，算法在数据结构中起着非常重要的作用。

1. 算法的概念

算法（Algorithm）是指为完成某项任务或者特定问题所设计的一种求解步骤的描述。算法可以理解成指令的有限序列，其中每一条指令可由一个或多个操作组成。

说明：算法的含义与程序十分相似，但又有区别。算法是解决问题的步骤分析，而程序