



“十二五”普通高等教育本科国家级规划教材 计算机系列教材

数据结构与算法 (第2版)

汪 沁 奚李峰 主 编
邓 芳 金 冉 刘晓利 陈 慧 副主编

清华大学出版社



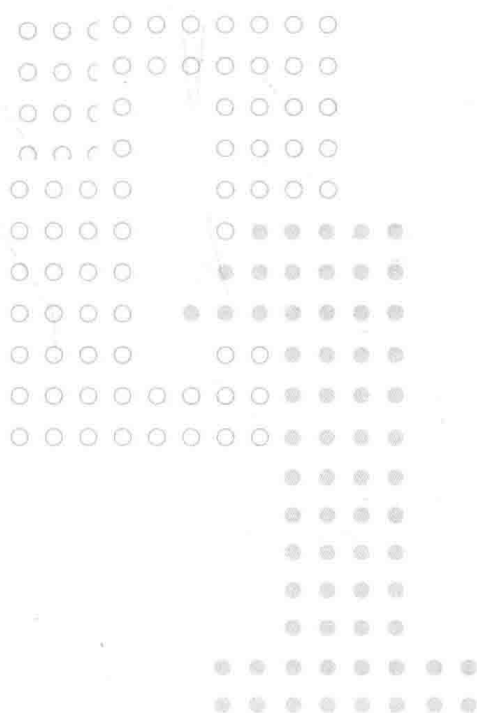
“十二五”普通高等教育本科国家级规划教材 计算机系列教材

汪 沁 奚李峰 主 编

邓 芳 金 冉 刘晓利 陈 慧 副主编

数据结构与算法

(第2版)



清华大学出版社

北京

内 容 简 介

本书系统地介绍了各种数据结构的特点、存储结构及相关算法。书中采用 C 语言描述算法。主要内容包括数据结构的基本概念、算法描述和算法分析初步;线性表、堆栈、队列、串、数组、树、图等结构;查找、排序等。每章后面配有小结、习题、讨论题。本书有配套的完整的习题与实验指导书,每一章节都给出了完整的 C 语言和 C++ 源程序示例。

本书叙述清晰,深入浅出,注意实践,便于教学与实践。

本书既可作为高等院校计算机专业的教材,也可供从事计算机应用与工程工作的科技工作者自学参考。

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

数据结构与算法/汪沁,奚李峰主编. —2 版. —北京:清华大学出版社,2018

(计算机系列教材)

ISBN 978-7-302-49953-4

I. ①数… II. ①汪… ②奚… III. ①数据结构—高等学校—教材 ②算法分析—高等学校—教材 IV. ①TP311.12

中国版本图书馆 CIP 数据核字(2018)第 097061 号

责任编辑:张 民 李 晔

封面设计:常雪影

责任校对:李建庄

责任印制:宋 林

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:北京密云胶印厂

经 销:全国新华书店

开 本:185mm×260mm

印 张:17.25

字 数:400 千字

版 次:2013 年 2 月第 1 版 2018 年 6 月第 2 版

印 次:2018 年 6 月第 1 次印刷

定 价:45.00 元

产品编号:076393-01

前 言

数据结构与算法是计算机专业重要的专业基础课程与核心课程之一。从理论上讲,通过学习数据结构可以使学生掌握对不同数据结构的组织方法和对具体数据结构所实施的若干算法,并能分析算法的优劣。学习数据结构与算法的最终目的是提高学生的程序设计水平和能力。

对于应用型人才培养应该注重能力的培养,而不是只满足于理论的掌握。因此,在本书的编写过程中遵循谭浩强教授提出的新三部曲“提出问题—解决问题—归纳分析”的写法,强调从实践中获取知识。本书给出了能够解决实际问题的大量算法,希望学生在阅读和总结这些算法的基础上提高程序设计的水平。因此,本书的大部分算法只要经过简单的修改就能上机运行,具有很好的实用价值,也给学习者带来了方便。

(1) 深入浅出,通俗易懂。对数据结构的基本概念、基本理论的阐述注重科学严谨。同时从应用出发,对新概念的引入从实例着手。对各种基本算法描述尽量详细,叙述清楚。本书在讲解数据的存储结构时,使用了大量的图示和表格,有助于学生对数据结构的理解。

(2) 理论联系实际。为了巩固所学的理论知识,每章都附有练习题和讨论题,供学生进行书面练习、上机作业时选用和讨论。针对学生中普遍存在的“只懂概念不懂编程”的问题,配套有完整的习题与实验指导书,每一章节都给出了完整的 C 语言和 C++ 源程序示例,供学生参考模拟,从而提高学生的程序设计能力。数据结构课程的一个重要任务是培养学生进行复杂程序设计的能力,目的在于提高学生的程序设计能力和进行规范化程序设计的素养。

(3) 循序渐进,逐步加深。由于采用了 C 语言和 C++ 语言面向对象的方法描述数据结构,对于低年级学生来说存在一定难度。为了使读者更好地学习数据结构自身的知识内容,克服描述工具所带来的困难,本书对此做了独特处理。

本书可以作为普通高等院校计算机专业本科和专升本的教材。由于资源翔实、通俗易懂,对书中内容适当取舍之后,也可作为高等职业技术和专科教育的计算机专业教材。同时,本书还可作为研究生考试和各类认证考试的复习参考书,以及计算机应用工作者和工程技术人员参考书。

本书由汪沁、奚李峰主编。其中,第 1~3 章、第 9 章和实验指导由汪沁、奚李峰编写;第 6 章、第 10 章由邓芳编写;第 4 章、第 7 章由刘晓利编写;第 5 章、第 8 章由金冉、陈慧编写。全书由汪沁、奚李峰统编。

考虑到在数据结构与算法的学习中,教师需要在课堂上对大量的算法进行讲解,而学生应该在此基础上大量阅读并理解数据结构经典算法,因此本书对算法都进行了较为详细的注释。对一些难度比较大的算法,在用语言描述之前,还对算法进行了分析。

由于编者水平有限,疏漏在所难免,欢迎广大读者批评指正并提出宝贵意见。

作者的电子邮箱地址: qinwang@126.com。

编 者
2018年5月

目 录

第1章 绪论 /1

- 1.1 数据结构的概念 /2
 - 1.1.1 引言 /2
 - 1.1.2 数据结构的有关概念与术语 /5
- 1.2 抽象数据类型 /7
- 1.3 算法描述与分析 /11
 - 1.3.1 什么是算法 /11
 - 1.3.2 算法分析技术初步 /13
- 1.4 基本的算法策略 /17
 - 1.4.1 穷举法 /17
 - 1.4.2 递推法与迭代法 /18
 - 1.4.3 分治法 /20
 - 1.4.4 贪心算法 /22
 - 1.4.5 动态规划 /22
- 1.5 案例分析 /25
- 1.6 小结 /26
- 讨论小课堂1 /27
- 习题1 /28

第2章 线性表 /30

- 2.1 线性表的定义及其运算 /30
 - 2.1.1 线性表的定义 /30
 - 2.1.2 线性表的抽象数据类型 /31
- 2.2 线性表的顺序存储结构及实现 /32
 - 2.2.1 顺序存储结构 /32
 - 2.2.2 线性表在向量中基本运算的实现 /34
- 2.3 线性表的链表存储结构 /39
 - 2.3.1 单链表 /39
 - 2.3.2 线性链表基本运算的实现 /42
- 2.4 循环链表和双向链表 /49
 - 2.4.1 循环链表 /49
 - 2.4.2 双向链表 /50
 - 2.4.3 顺序存储结构与链表存储结构的综合分析与比较 /51

- 2.5 单链表的应用 /52
 - 2.5.1 多项式相加的链表存储结点 /52
 - 2.5.2 多项式相加的算法实现 /53
- 2.6 小结 /54
- 讨论小课堂2 /55
- 习题2 /55

第3章 栈和队列 /57

- 3.1 栈 /57
 - 3.1.1 栈的定义 /57
 - 3.1.2 栈的抽象数据类型 /58
- 3.2 栈的顺序存储结构及实现 /59
 - 3.2.1 栈的顺序存储结构 /59
 - 3.2.2 顺序栈的定义 /60
- 3.3 栈的链表存储结构及实现 /62
- 3.4 栈的应用 /65
 - 3.4.1 表达式的计算 /65
 - 3.4.2 子程序的嵌套调用 /67
 - 3.4.3 递归调用 /68
- 3.5 队列 /69
 - 3.5.1 队列的定义及运算 /69
 - 3.5.2 队列的抽象数据类型 /70
- 3.6 队列的顺序存储结构及实现 /70
- 3.7 队列的链表存储结构及实现 /74
- 3.8 队列的应用 /77
- 3.9 算法实例——Hanoi 塔问题 /78
- 3.10 小结 /79
- 讨论小课堂3 /80
- 习题3 /81

第4章 串 /83

- 4.1 串的基本概念 /83
 - 4.1.1 串的定义 /83
 - 4.1.2 串的抽象数据类型 /84
- 4.2 串的存储与基本操作的实现 /85
 - 4.2.1 定长顺序串 /86
 - 4.2.2 堆串 /86
 - 4.2.3 块链串 /87

- 4.2.4 串操作的实现 /88
- 4.3 串的模式匹配 /91
 - 4.3.1 朴素模式匹配算法 /92
 - 4.3.2 模式匹配的 KMP 算法 /92
- 4.4 串的应用举例: 文本编辑 /97
- 4.5 小结 /99
- 讨论小课堂 4 /99
- 习题 4 /100

第 5 章 数组和广义表 /101

- 5.1 数组 /102
 - 5.1.1 数组的基本概念 /102
 - 5.1.2 二维数组 /102
 - 5.1.3 数组的顺序存储方式 /103
- 5.2 矩阵的压缩存储 /104
 - 5.2.1 特殊矩阵 /104
 - 5.2.2 稀疏矩阵 /107
- 5.3 广义表 /112
 - 5.3.1 广义表的定义 /112
 - 5.3.2 广义表的存储结构 /113
- 5.4 案例分析 /116
 - 5.4.1 概述和方法 /116
 - 5.4.2 算法和程序 /118
- 5.5 小结 /120
- 讨论小课堂 5 /120
- 习题 5 /120

第 6 章 树与二叉树 /122

- 6.1 树的概念及术语 /123
 - 6.1.1 树的定义 /123
 - 6.1.2 树的抽象数据类型 /124
 - 6.1.3 树的表示方式 /125
- 6.2 二叉树 /125
 - 6.2.1 二叉树的定义 /125
 - 6.2.2 二叉树的抽象数据类型 /126
 - 6.2.3 二叉树的重要性质 /127
 - 6.2.4 二叉树的存储结构 /128
- 6.3 二叉树的遍历 /130

- 6.3.1 先序遍历 /131
- 6.3.2 中序遍历 /131
- 6.3.3 后根遍历 /132
- 6.3.4 按层遍历 /133
- 6.3.5 非递归遍历算法 /133
- 6.3.6 二叉树的建立 /136
- 6.3.7 二叉树遍历的应用举例 /137
- 6.4 二叉树与树、森林的转换 /139
 - 6.4.1 树与二叉树的转换 /139
 - 6.4.2 森林与二叉树的转换 /140
- 6.5 树的存储结构 /141
 - 6.5.1 树的双亲表示法 /142
 - 6.5.2 孩子表示法 /142
 - 6.5.3 孩子兄弟表示法 /143
- 6.6 树的遍历 /144
 - 6.6.1 一般树的遍历 /144
 - 6.6.2 森林的遍历 /145
- 6.7 二叉树的应用 /146
 - 6.7.1 哈夫曼树 /146
 - 6.7.2 哈夫曼树的构造 /146
 - 6.7.3 哈夫曼树的实现算法 /148
 - 6.7.4 哈夫曼编码 /149
- 6.8 小结 /150
- 讨论小课堂 6 /150
- 习题 6 /150

第7章 图 /153

- 7.1 图的基本概念 /153
 - 7.1.1 图的定义 /153
 - 7.1.2 图的术语 /155
 - 7.1.3 图的抽象数据类型 /156
- 7.2 图的存储结构 /157
 - 7.2.1 图的邻接矩阵 /157
 - 7.2.2 邻接矩阵表示法的描述 /159
 - 7.2.3 邻接矩阵表示下的基本操作的实现 /160
 - 7.2.4 图的邻接链表 /161
 - 7.2.5 图的邻接表表示法的描述 /162
 - 7.2.6 邻接表表示下基本操作的实现 /163

- 7.3 图的遍历与图的连通性 /165
 - 7.3.1 图的深度优先遍历 /166
 - 7.3.2 图的广度优先遍历 /168
 - 7.3.3 非连通图和连通分量 /170
- 7.4 图的最小生成树 /170
 - 7.4.1 最小生成树的基本概念 /170
 - 7.4.2 普里姆(Prim)算法 /171
 - 7.4.3 克鲁斯卡尔(Kruskal)算法 /174
- 7.5 最短路径 /175
 - 7.5.1 从某顶点到其余各顶点的最短路径 /175
 - 7.5.2 每对顶点之间的最短路径 /178
- 7.6 拓扑排序与关键路径 /180
 - 7.6.1 拓扑排序 /180
 - 7.6.2 关键路径 /183
- 7.7 图的应用 /189
 - 7.7.1 图在路由器寻径中的应用 /189
 - 7.7.2 图在物流信息系统中应用 /190
- 7.8 小结 /190
- 讨论题 7 /191
- 习题 7 /191

第 8 章 查找 /193

- 8.1 查找的基本概念 /194
- 8.2 静态查找表 /195
 - 8.2.1 顺序表的查找 /195
 - 8.2.2 有序表的折半查找 /196
 - 8.2.3 索引顺序表查找 /200
- 8.3 动态查找表 /201
 - 8.3.1 二叉排序树 /201
 - 8.3.2 平衡二叉树 /210
- 8.4 案例分析 /214
 - 8.4.1 直方图问题 /214
 - 8.4.2 箱子装载问题 /216
- 8.5 小结 /218
- 讨论小课堂 8 /218
- 习题 8 /219

第 9 章 排序 /220

- 9.1 排序的基本概念 /220

9.2	插入排序	/221
9.2.1	直接插入排序	/221
9.2.2	折半插入排序	/222
9.2.3	希尔排序	/222
9.3	交换排序	/224
9.3.1	冒泡排序	/224
9.3.2	快速排序	/225
9.4	选择排序	/229
9.4.1	简单选择排序	/229
9.4.2	堆排序	/230
9.5	归并排序	/233
9.6	基数排序	/235
9.7	小结	/239
	讨论小课堂 9	/240
	习题 9	/240

第10章 索引结构与哈希 /242

10.1	静态索引结构	/242
10.1.1	索引表	/242
10.1.2	索引表查找	/242
10.2	动态索引结构(B-树和 B+树)	/245
10.2.1	B-树的定义	/245
10.2.2	B-树的运算	/246
10.2.3	B+树	/249
10.3	键树及 Trie 树	/250
10.3.1	键树的定义	/250
10.3.2	双链树	/251
10.3.3	Trie 树	/252
10.4	哈希表及其查找	/253
10.4.1	哈希表与哈希函数	/253
10.4.2	构造哈希函数的常用方法	/254
10.4.3	解决冲突的主要方法	/256
10.4.4	哈希查找的性能分析	/260
10.5	小结	/261
	讨论小课堂 10	/262
	习题 10	/262

参考文献 /265

第1章 绪 论

计算机科学是一门研究数据表示和数据处理的科学。现在,计算机处理的对象已由单纯的数值发展到字符、图像、声音等,表示这些对象的数据成分往往不是单一的,而是多成分且形成了一定的结构。因此,在程序设计中,除了应精心设计算法外,还应精心组织数据(包括原始数据、中间结果、最终结果),使之形成一定的组织形式(数据结构),以便让计算机尽可能高效率地处理。因此,要设计出一个结构好、效率高的程序,必须研究数据的特性及数据间的相互关系及其对应的存储表示,并利用这些特性和关系设计出相应的算法和程序。“数据结构”就是在此背景下逐步形成、发展起来的。通过这门课的学习,可以掌握非数值计算程序设计中用到的基本方法和技巧。

【案例引入】

七桥问题(seven bridges problem)。在哥尼斯堡的一个公园里,有七座桥将普雷格尔河中两个岛及岛与河岸连接起来,如图 1.1 所示。问是否可能从这四块陆地中任一块出发,恰好通过每座桥一次,再回到起点?

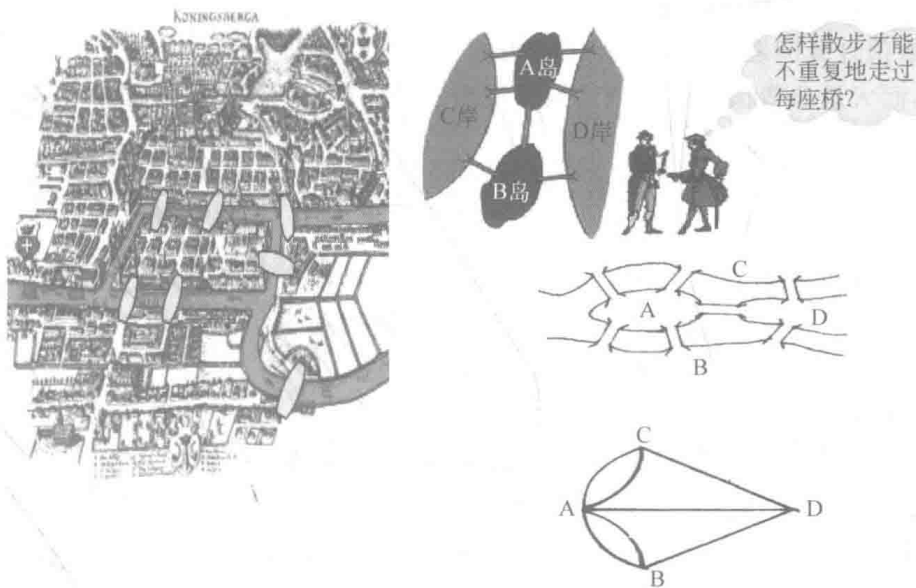


图 1.1 七桥问题图

遇到这样的案例,我们如何运用数据结构的知识去解决问题呢?实际上该题目是一个比较重要的问题,是有关数据结构——图知识方面的问题,是对欧拉图的路径实现问题。用图形方式验证七桥问题,可以证明该问题是不可实现的。

问题:你过去是否听说过“数据结构”?你知道数据结构是讨论什么内容的学科吗?

1.1 数据结构的概念

数据结构是计算机科学与技术专业的专业基础课,是十分重要的核心课程。所有的计算机系统软件和应用软件都要用到各种类型的数据结构。因此,要想更好地运用计算机来解决实际问题,仅掌握几种计算机程序设计语言是难以应付众多复杂的课题的。要想有效地使用计算机、充分发挥计算机的性能,还必须学习和掌握好数据结构的有关知识。

数据结构涉及各方面的知识,如计算机硬件范围的存储装置和存取方法;在计算机软件范围中的文件系统,数据的动态存储与管理,信息检索;数学范围的许多算法知识,还有一些综合性的知识,如编码理论、算子关系、数据类型、数据表示、数据运算、数据存取等。因此,数据结构是数学、计算机硬件、软件三者之间的一门核心课程。打好“数据结构”这门课程的扎实基础,对于学习计算机专业的其他课程,如操作系统、编译原理、数据库管理系统、软件工程、人工智能等都是十分有益的。

1.1.1 引言

程序设计的实质即为计算机处理问题编制一组“指令”,首先需要解决两个问题,即算法和数据结构。算法即处理问题的策略,而数据结构即为问题的数学模型。

问题:什么是非数值计算问题?举例说明。

很多数值计算问题的数学模型通常可用一组线性或非线性的代数方程组或微分方程组来描述,而大量非数值计算问题的数学模型正是本门课程要讨论的数据结构。

例如:

(1) 在计算机内部,1和2是用二进制00000001和00000010来表示的,迷宫、棋盘在计算机内部如何表示呢?

(2) 交叉路口的红绿灯管理。如今十字路口横竖两个方向都有三个红绿灯,分别控制左拐、直行和右拐,那么如何控制这些红绿灯既使交通不堵塞,又使流量最大呢?若要编制程序解决问题,首先要解决一个如何表示的问题。

(3) 七桥问题。

在各种高级语言程序设计的基本训练中,解决某一实际问题的步骤一般是:分析实际问题;确定数学模型;编写程序;反复调试程序直至得到正确结果。所谓数学模型,一般指具体的代数方程等。然而,有些实际问题无法用数学方程表示。现在具体分析一个典型实例,它们的主要特点是处理数据信息的存储与检索等,而不是单纯的数值计算。

实例:管理办公室的钥匙。

方法1:按照房间号码顺序存放钥匙,如图1.2所示。

方法2:按照楼号和楼层分抽屉号和抽屉格子来存放钥匙,如图1.3所示。

问题:取钥匙的方法一样吗?

例如,寻找3105房间(3号楼1层5号房间)的钥匙。

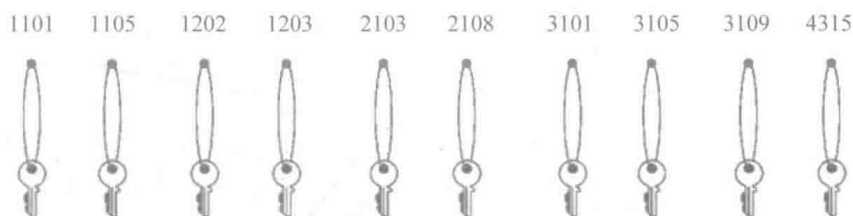


图 1.2 按照房间存放钥匙图

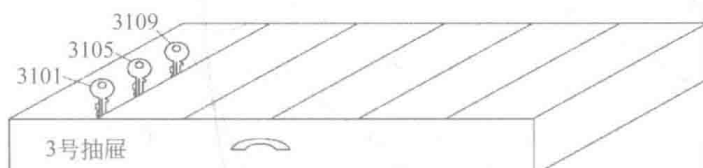


图 1.3 按照楼号和楼层存放钥匙图

方法 1: 直接按照房间号码顺序查找。

方法 2:

(1) 找到放钥匙的办公桌;

(2) 打开 3 号抽屉;

(3) 在 3 号抽屉的第 1 个矩形格子中逐个查看, 找到标号 3105 的钥匙, 取到钥匙。

方法 2 在数据的处理上要比方法 1 复杂一些, 但程序的效率有所提高, 尤其是加快了查询的速度。

原因:

(1) 程序存储钥匙信息的方式不同。

(2) 存储方式导致了程序必须执行不同的操作才能完成存储、查询等任务。

我们需要研究三点, 包括钥匙间的逻辑关系(钥匙间的联系)、钥匙信息在计算机中的存储方式和对钥匙信息的操作(存储、查询等), 而这正是数据结构要研究的三个问题。数据结构研究的是非数值问题中数据之间的逻辑关系、具有逻辑关系的数据在计算机内的表示方式(存储方式)以及对数据的操作。

挂在墙上的钥匙之间的逻辑关系是线性关系, 其特征是: 一把钥匙只与其左边的钥匙和右边的钥匙有逻辑关系(线性关系), 如图 1.4 所示。

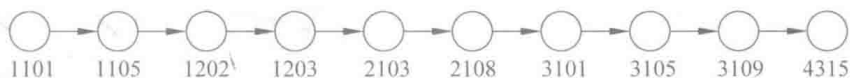


图 1.4 挂在墙上的钥匙之间的逻辑关系图

放在抽屉里的钥匙之间的逻辑关系是非线性关系, 如图 1.5 所示。

具有逻辑关系的数据在计算机内存储时, 所采取的存储方式应该能够描述数据之间的逻辑关系。同样是线性关系, 数据的存储结构可采用顺序存储或链式存储的方法, 如图 1.6 所示。

问题: 顺序存储和链式存储的区别是什么?

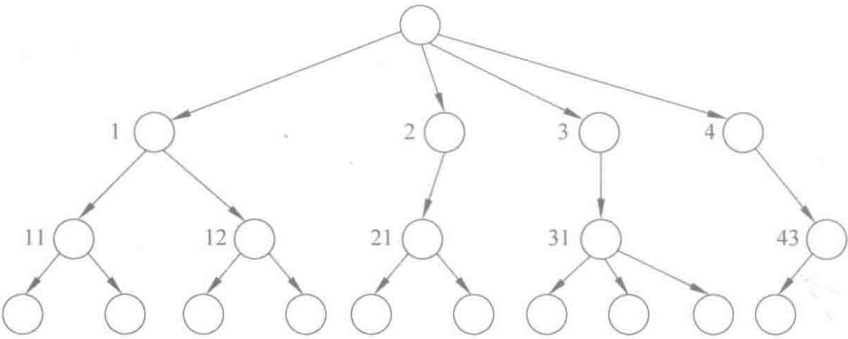
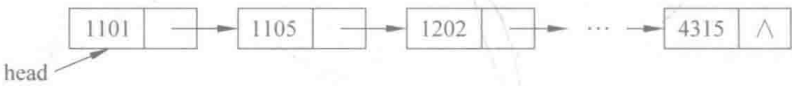


图 1.5 放在抽屉里的钥匙之间的逻辑关系图

顺序存储方法是把逻辑上相邻的元素存储在物理位置相邻的存储单元中，由此得到的存储表示称为顺序存储结构。顺序存储结构是一种最基本的存储表示方法，通常借助于程序设计语言中的数组来实现，如图 1.6(a)所示。

数组下标	数组a内容
0	1101
1	1105
2	1202
3	1203
4	2103
5	2108
6	3101
7	3105
8	3109
9	4315
10	
⋮	

(a) 顺序存储图



(b) 链式存储图

图 1.6 顺序存储和链式存储图

链式存储方法对逻辑上相邻的元素不要求其物理位置相邻，元素间的逻辑关系通过附设的指针字段来表示，由此得到的存储表示称为链式存储结构，链式存储结构通常借助于程序设计语言中的指针类型来实现，如图 1.6(b)所示。

以上所举例子中的数学模型正是数据结构要讨论的问题。因此，简单地说，数据结构是一门讨论“描述现实世界实体的数学模型(非数值计算)及其上的操作在计算机中如何表示和实现”的学科。

问题：你以前是否了解“数据结构”这个概念？

1.1.2 数据结构的有关概念与术语

1. 数据

在计算机科学中,数据(data)是所有能被输入到计算机中,且能被计算机处理的符号(数字、字符等)的集合,它是计算机操作对象的总称。是计算机处理的信息的某种特定的符号表示形式。数据不仅包括整型、实型等数值数据,也包括文本数据、声音、视频、图像等非数值数据。

2. 数据元素

数据元素(data element)是数据(集合)中的一个“个体”,在计算机中通常作为一个整体进行考虑和处理,是数据结构中讨论的“基本的独立单位”,它也被称为元素、结点、记录等。例如,学生信息系统中的班级信息表中的一个记录(例如“001 张三 男 96”)、八皇后问题中状态树的一个状态、教学计划编排问题中的一个顶点等,都被称为一个数据元素。在复杂的数据结构中,数据元素往往由若干个数据项组成,数据项是具有独立含义的最小标识单位,也称为字段或域。例如“001 张三 男 96”这个数据元素由 001(学生编号)、张三(姓名)、男(性别)、96(成绩) 4 个数据项组成。

因此有两类数据元素:一类是不可分割的“原子”型数据元素,如:整数“8”、字符“S”等;另一类是由多个款项构成的数据元素,其中每个款项被称为一个“数据项”。

例如,描述一个职员的信息的数据元素可由下列 6 个数据项组成:姓名、性别、出生日期、电话、家庭地址、职位。其中的出生日期又可以由 3 个数据项“年”“月”和“日”组成,则称“出生日期”为“组合项”,而其他不可分割的数据项为“原子项”。数据项(data item)是数据结构中讨论的“最小单位”。

3. 数据对象

数据对象(data object)是具有相同特性的数据元素的集合,如整数、实数等。所谓性质相同,是指数据元素具有相同数量和类型的数据项。它是数据的一个子集。例如,学生信息系统中的班级学生信息表就是一个数据对象,这个对象中的数据元素都是由“学生编号”“姓名”“性别”“成绩”4 个数据项组成的;班级中的全部男生的信息也是一个数据对象,它是班级学生信息的一个子集。通常简称数据对象为数据。

4. 数据结构

数据结构(data structure)以某种内在联系将由数据项组成的数据元素组织成为一个数据对象,学习数据结构的组织形式以及相关运算非常重要。

数据结构是指互相之间存在着一种或多种关系的数据元素的集合。在任何问题中,数据元素之间都不会是孤立的,它们之间都存在着这样或那样的关系,这种数据元素之间的关系称为结构。

从上面所介绍的数据结构的概念中可以知道,一个数据结构有两个要素:一个是数据元素的集合,另一个是关系的集合。在形式上,数据结构通常可以采用一个二元组来表示。

数据结构的定义: 数据结构是一个二元组

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

其中,D 是数据元素的有限集,R 是 D 上关系的有限集。

根据数据元素间关系的不同特性,通常有下列 4 类基本的结构:

- 集合结构。在集合结构中,数据元素间的关系是“属于同一个集合”。集合是元素关系极为松散的一种结构。
- 线性结构。该结构的数据元素之间存在着一对一的关系。
- 树形结构。该结构的数据元素之间存在着一对多的关系。
- 图形结构。该结构的数据元素之间存在着多对多的关系,图形结构也称作网状结构。图 1.7 为表示上述 4 类基本结构的示意图。

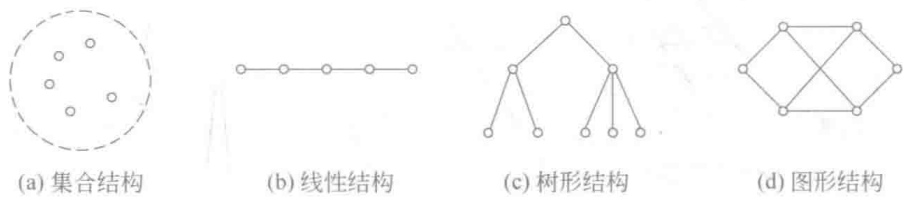


图 1.7 4 类基本结构的示意图

由于集合是数据元素之间关系极为松散的一种结构,因此也可用其他结构来表示。

数据结构包括数据的逻辑结构和数据的物理结构。数据逻辑结构是对数据元素之间存在的逻辑关系的描述,它可以用一个数据元素的集合和定义在此集合上的若干关系表示。

数据物理结构是数据逻辑结构在计算机中的表示和实现,故又称数据“存储结构”。

逻辑结构反映的是数据元素之间的关系,它们与数据元素在计算机中的存储位置无关,是数据结构在用户面前所呈现的形式。

上述对数据结构的定义还只是数学上的抽象概念,并没有涉及计算机,完整的数据结构定义还应该包括它在计算机中的表示,即数据的存储结构。

数据的存储结构关系有两种表示方法:

(1) 顺序存储,如图 1.8 所示。以“B 相对于 A 的存储位置”表示“B 是 A 的后继”,例如,令 B 的存储位置和 A 的存储位置之间相差一个预设常量 X,X 本身是个隐含值,由此得到的数据存储结构为“顺序存储结构”。

(2) 链式存储,如图 1.9 所示。以和 A 绑定在一起的附加信息(指针)表示后继关系,这个指针即为 B 的存储地址,由此得到的数据存储结构为“链式存储结构”。



图 1.8 顺序存储结构

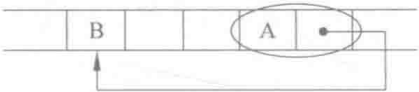


图 1.9 链式存储结构