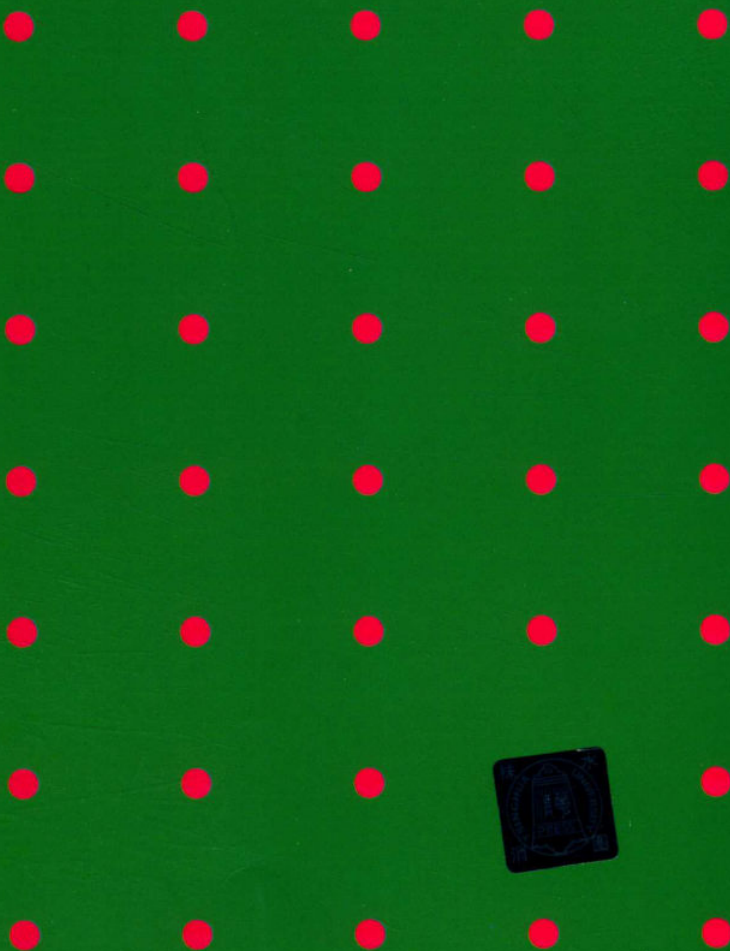


普通高校本科计算机专业特色教材精选·算法与程序设计

数据结构 (C语言描述)

董洁 卞鹏 孙雪洋 编著



清华大学出版社

普通高校本科计算机专业特色教材精选·算法与程序设计

数据结构（C语言描述）

董洁 卞鹏 孙雪洋 编著

贵州师范学院内部使用

清华大学出版社
北京

内 容 简 介

本书系统地介绍了各种常用的数据结构与算法的基本理论和基本方法,共8章。其中第1章为绪论,引入数据、数据结构、抽象数据类型和算法等基本概念;第2~6章从抽象数据类型的角度讨论各种常用的数据结构及其应用,包括线性表、栈、队列、串、数组、广义表、树和二叉树以及图等;第7章和第8章分别讨论查找和排序的各种实现方法及其综合分析比较。通过介绍并探讨数据的组织、算法设计及其时间和空间效率的分析方法,培养读者针对具体问题的应用背景,选择合适的数据结构,设计并编写复杂程序的能力。

本书采用类C语言作为数据结构和算法的描述工具,尽量考虑C语言的特色,兼顾数据结构和算法的可读性,结构清晰,内容充实,语言精练,主要章节配有微课讲解视频的二维码,易于理解,强调系统性和实用性的结合。

本书可以供高等院校学生使用,也可以作为硕士研究生入学考试的参考书,还可以供各类学习数据结构的人员参考使用。

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: C语言描述/董洁,卞鹏,孙雪洋编著. —北京:清华大学出版社,2019
普通高校本科计算机专业特色教材精选·算法与程序设计
ISBN 978-7-302-53205-7

I. ①数… II. ①董… ②卞… ③孙… III. ①数据结构 ②C语言—程序设计 IV. ①TP311.12
②TP312.8

中国版本图书馆CIP数据核字(2019)第129397号

责任编辑:袁勤勇 杨枫

封面设计:傅瑞学

责任校对:焦丽丽

责任印制:杨艳

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>, 010-83470236

印 刷 者:北京富博印刷有限公司

装 订 者:北京市密云县京文制本装订厂

经 销:全国新华书店

开 本:185mm×260mm

印 张:18.75

字 数:428千字

版 次:2019年11月第1版

印 次:2019年11月第1次印刷

定 价:49.00元

产品编号:080921-01

前言

PREFACE

“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，同时是操作系统、数据库原理、编译原理、软件工程、人工智能等多门后续课程的基础。随着计算机应用范围的不断扩大，学习和使用计算机的人群已经不限于计算机专业，许多涉及信息处理的理工类、管理类专业也把“数据结构”作为一门必修的基础课，以便有效地使用计算机，充分发挥计算机的功能。因此，学好“数据结构”，对于计算机及其相关专业的学生，具有十分重要的意义。

数据结构主要分析研究的是计算机处理的数据对象的特性以及数据元素之间的关系，以便为应用涉及的数据选择适当的逻辑结构、存储结构和相应的算法，并初步掌握算法的时间和空间分析的技术，培养学生设计复杂程序的能力。作者长期以来一直选用严蔚敏教授编写的数据结构教材进行教学，该教材具有概念表述严谨、逻辑推理严密等许多优点，但课程内容涉及数据的组织原理和算法比较抽象，对地方院校的学生来说难度过大，编者将多年的教学经验做了系统的总结，根据学生的实际情况，应用型人才培养的需要以及“数据结构”课程的特点，将理论与实践相结合，采用算法配以图形展示和操作步骤描述等方法，把抽象的原理具体化，精心组织编写了本教材。从便于初学者学习的角度出发，对课程内容做层次化处理，以利于读者更好地学习和掌握课程内容，为后续课程的学习打下良好的基础。

全书中每一章开始都设有学习目标，并给出知识结构图，以供教学过程参考。采用类C语言作为数据和算法的描述语言。对于每一种基本数据结构，给出相关定义后，用规范化的ADT(抽象数据类型)进行描述，使读者能从面向对象的角度理解和把握概念的本质；在对数据的存储结构和算法进行描述时，尽量考虑C语言的特色，同时兼顾数据结构和算法的可读性。对各种数据结构的定义和实现简洁、清晰，算法讲解更加细致，按“基本思想、算法步骤、C语言描述、算法分析”四级模式精心组织教学内容，将用文字描述的算法步骤与用类C语言表述的算法描述一

一对应。部分算法给出了对应的程序设计代码,便于学生深入理解和上机实践,锻炼学生的实际应用能力。

全书共 8 章,第 1 章为绪论,引入数据、数据结构、抽象数据类型、算法、算法复杂度等基本概念,是全书的基础;第 2~6 章从抽象数据类型的角度,分别讨论不同的数据结构;第 7 章和第 8 章分别讨论两种重要的常用操作。其中第 2 章讨论线性表,介绍了线性表的基本概念、两种存储结构,不同存储结构下的操作实现以及一些简单应用,并给出了部分算法的程序实现;第 3 章讨论栈与队列,介绍了栈与队列的基本概念、特点,不同存储结构下的操作实现,以及递归等实际应用的算法实现;第 4 章讨论串、数组和广义表,介绍了串的概念、串的存储及模式匹配算法;数组及其元素的存取、压缩存储和基于压缩存储的算法;广义表的基本概念及其存储方式;第 5 章讨论树,介绍了树和二叉树的概念、各种存储结构,以及遍历、线索化二叉树、树、森林与二叉树的转换、Huffman 树的概念与实现;第 6 章讨论图,介绍了图的相关概念、图的存储方式,以及图的不同遍历方法、最小生成树、拓扑排序、关键路径和最短路径的概念与实现;第 7 章是查找,介绍了查找的概念与分类、各种查找方法的实现及复杂度分析;第 8 章是排序,介绍了排序的概念、排序的分类,重点介绍了各种内部排序方法的实现。

本书受到辽宁省教育厅精品资源共享课的资助,获得沈阳建筑大学、沈阳科技学院、辽宁科技学院、沈阳理工大学、沈阳城市学院等多个院校老师的大力支持,其中第 1~3 章由董洁、孙雪洋编写,第 4~6 章由卞鹏、曹科研、刘也凡编写,第 7 章由董洁、赵明编写,第 8 章由董洁、李筠、朱元华编写,由董洁、刘前对全文进行通审和定稿。

本书各章节主要内容分别配有微课讲解视频,其中第 1~3 章由董洁录制,第 4、5 章由赵明、董洁录制,第 6 章由董洁、孙焕良录制,第 7 章由董洁、任义录制,全部视频由卞鹏剪辑完成。

本书可作为高等院校计算机科学与技术、软件工程、信息工程、信息与计算科学、信息管理与信息系统等专业的教材、参考书或考研辅导用书,也可供其他相关理工类专业或工程技术人员参考。对于计算机、信息类专业,可讲授 64 学时,对于非信息类专业,可适当删减,讲授 48 学时。

由于作者水平有限,书中难免有和疏漏之处,恳请广大读者指正。

编者

2019 年 3 月

目 录

CONTENTS

| | |
|------------------------------|----|
| 第 1 章 绪论 | 1 |
| 1.1 概述 | 2 |
| 1.1.1 数据结构的研究内容 | 2 |
| 1.1.2 数据结构的发展过程 | 4 |
| 1.2 基本概念和术语 | 4 |
| 1.2.1 数据、数据元素、数据项和数据对象 | 4 |
| 1.2.2 逻辑结构和存储结构 | 4 |
| 1.2.3 数据类型和抽象数据类型 | 8 |
| 1.3 算法和算法分析 | 12 |
| 1.3.1 算法的定义及特性 | 12 |
| 1.3.2 算法与数据结构、程序的关系 | 13 |
| 1.3.3 评价算法的基本标准 | 13 |
| 1.3.4 算法时间的度量 | 13 |
| 1.3.5 算法的空间复杂度 | 17 |
| 小结 | 19 |
| 习题 | 19 |
| 第 2 章 线性表 | 23 |
| 2.1 线性表的概念 | 24 |
| 2.1.1 线性表的定义和特点 | 24 |
| 2.1.2 线性表的类型定义 | 24 |
| 2.2 线性表的顺序表示和实现 | 28 |
| 2.2.1 线性表的顺序存储表示 | 28 |
| 2.2.2 顺序表的结构定义 | 29 |
| 2.2.3 顺序表基本操作的实现 | 30 |

| | | |
|------------|-----------------|-----------|
| 2.3 | 线性表的链式表示和实现 | 36 |
| 2.3.1 | 单链表的定义和表示 | 36 |
| 2.3.2 | 单链表基本操作的实现 | 39 |
| 2.3.3 | 循环链表 | 48 |
| 2.3.4 | 双向链表 | 49 |
| 2.3.5 | 静态链表 | 52 |
| 2.4 | 线性表的应用 | 53 |
| 2.5 | 线性表典型算法的实现 | 57 |
| | 小结 | 60 |
| | 习题 | 61 |
| 第3章 | 栈和队列 | 63 |
| 3.1 | 栈 | 64 |
| 3.1.1 | 栈的定义和特点 | 64 |
| 3.1.2 | 栈的类型定义 | 64 |
| 3.1.3 | 顺序栈的表示和实现 | 65 |
| 3.1.4 | 链栈的表示和实现 | 69 |
| 3.2 | 栈与递归 | 71 |
| 3.2.1 | 采用递归算法解决的问题 | 71 |
| 3.2.2 | 递归过程与递归工作栈 | 74 |
| 3.3 | 队列 | 75 |
| 3.3.1 | 队列及其特点 | 75 |
| 3.3.2 | 队列的类型定义 | 76 |
| 3.3.3 | 队列的顺序表示和实现 | 77 |
| 3.3.4 | 队列的链式表示和实现 | 80 |
| 3.4 | 栈和队列的应用 | 84 |
| 3.4.1 | 数制的转换 | 84 |
| 3.4.2 | 括号匹配的检验 | 85 |
| 3.4.3 | 表达式求值 | 86 |
| 3.4.4 | 队列的应用 | 89 |
| | 小结 | 90 |
| | 习题 | 90 |
| 第4章 | 串、数组和广义表 | 93 |
| 4.1 | 串的定义与操作 | 94 |
| 4.1.1 | 串的定义与相关概念 | 94 |
| 4.1.2 | 串的抽象类型定义 | 95 |

| | | |
|--------------|--------------|------------|
| 4.2 | 串的实现和表示 | 96 |
| 4.2.1 | 定长顺序存储表示 | 96 |
| 4.2.2 | 堆分配存储表示 | 99 |
| 4.2.3 | 串的链式存储表示 | 100 |
| 4.3 | 串的模式匹配 | 102 |
| 4.3.1 | 简单的模式匹配算法 | 102 |
| 4.3.2 | KMP 算法 | 104 |
| 4.4 | 数组 | 107 |
| 4.4.1 | 数组的类型定义 | 107 |
| 4.4.2 | 数组的顺序存储 | 109 |
| 4.4.3 | 特殊矩阵的压缩存储 | 111 |
| 4.5 | 广义表 | 118 |
| 4.5.1 | 广义表的定义 | 118 |
| 4.5.2 | 广义表的存储结构 | 119 |
| | 小结 | 121 |
| | 习题 | 122 |
| 第 5 章 | 树和二叉树 | 125 |
| 5.1 | 树的基本概念 | 126 |
| 5.1.1 | 树的定义 | 126 |
| 5.1.2 | 树的基本术语 | 127 |
| 5.1.3 | 树的抽象类型定义 | 129 |
| 5.2 | 二叉树基本概念 | 130 |
| 5.2.1 | 二叉树的定义 | 130 |
| 5.2.2 | 二叉树的抽象数据类型定义 | 131 |
| 5.3 | 二叉树的性质和存储结构 | 133 |
| 5.3.1 | 二叉树的性质 | 133 |
| 5.3.2 | 二叉树的存储结构 | 135 |
| 5.4 | 遍历二叉树和线索二叉树 | 137 |
| 5.4.1 | 遍历二叉树 | 138 |
| 5.4.2 | 线索二叉树 | 147 |
| 5.5 | 树和森林 | 152 |
| 5.5.1 | 树的存储结构 | 152 |
| 5.5.2 | 森林(树)与二叉树的转换 | 155 |
| 5.5.3 | 树和森林的遍历 | 157 |
| 5.6 | 哈夫曼树与哈夫曼编码 | 159 |
| 5.6.1 | 哈夫曼树的基本概念 | 159 |

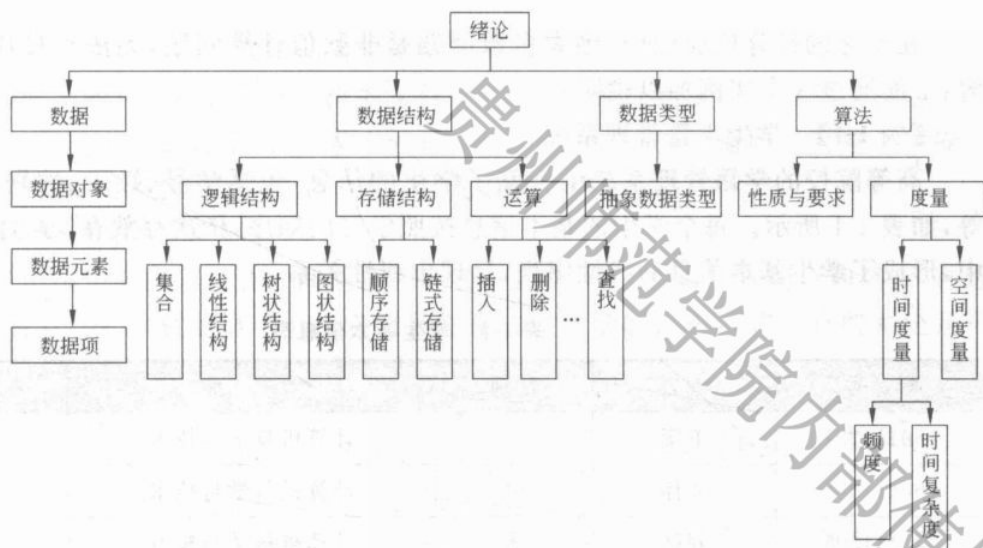
| | | |
|--------------|--------------|------------|
| 5.6.2 | 哈夫曼树的构造算法 | 160 |
| 5.6.3 | 哈夫曼编码 | 164 |
| 小结 | | 168 |
| 习题 | | 169 |
| 第 6 章 | 图 | 173 |
| 6.1 | 图的概述 | 174 |
| 6.1.1 | 图的定义及基本术语 | 174 |
| 6.1.2 | 图的类型定义 | 177 |
| 6.2 | 图的存储结构 | 178 |
| 6.2.1 | 邻接矩阵 | 179 |
| 6.2.2 | 邻接表 | 182 |
| 6.3 | 图的遍历 | 185 |
| 6.3.1 | 深度优先遍历 | 186 |
| 6.3.2 | 广度优先遍历 | 188 |
| 6.4 | 最小生成树 | 190 |
| 6.4.1 | 生成树和最小生成树的概念 | 190 |
| 6.4.2 | Prim 算法 | 191 |
| 6.4.3 | Kruskal 算法 | 194 |
| 6.5 | 最短路径 | 195 |
| 6.5.1 | 单源最短路径 | 195 |
| 6.5.2 | 任意一对顶点间的最短路径 | 200 |
| 6.6 | 拓扑排序与关键路径 | 204 |
| 6.6.1 | 拓扑排序 | 204 |
| 6.6.2 | 关键路径 | 207 |
| 小结 | | 212 |
| 习题 | | 214 |
| 第 7 章 | 查找 | 217 |
| 7.1 | 查找的基本概念 | 218 |
| 7.2 | 静态查找表 | 219 |
| 7.2.1 | 顺序查找 | 219 |
| 7.2.2 | 折半查找 | 221 |
| 7.2.3 | 分块查找 | 225 |
| 7.3 | 动态查找表 | 227 |
| 7.3.1 | 二叉排序树 | 227 |
| 7.3.2 | 平衡二叉树 | 234 |

| | |
|-------------------|------------|
| 7.3.3 B树 | 237 |
| 7.3.4 B+树 | 238 |
| 7.4 哈希表 | 239 |
| 7.4.1 哈希表概述 | 240 |
| 7.4.2 哈希函数的构造方法 | 240 |
| 7.4.3 处理冲突的方法 | 243 |
| 7.4.4 哈希表的查找 | 246 |
| 小结 | 251 |
| 习题 | 251 |
| 第8章 排序 | 255 |
| 8.1 概述 | 256 |
| 8.1.1 排序的基本概念 | 256 |
| 8.1.2 内部排序方法的分类 | 257 |
| 8.1.3 排序记录的存储结构 | 257 |
| 8.1.4 排序算法效率的评价指标 | 258 |
| 8.2 插入排序 | 259 |
| 8.2.1 直接插入排序 | 259 |
| 8.2.2 折半插入排序 | 261 |
| 8.2.3 希尔排序 | 263 |
| 8.3 交换排序 | 265 |
| 8.3.1 冒泡排序 | 265 |
| 8.3.2 快速排序 | 268 |
| 8.4 选择排序 | 271 |
| 8.4.1 简单选择排序 | 271 |
| 8.4.2 堆排序 | 273 |
| 8.5 归并排序 | 278 |
| 8.6 基数排序 | 280 |
| 8.6.1 多关键字的排序 | 280 |
| 8.6.2 链式基数排序 | 281 |
| 8.7 内部排序方法比较 | 285 |
| 小结 | 286 |
| 习题 | 286 |
| 参考文献 | 289 |

学习目标

1. 掌握数据、数据对象、数据结构等基本概念,深刻理解数据结构包含的内容。
2. 掌握抽象数据类型的相关概念,理解抽象数据类型与数据类型的差异。
3. 掌握算法的含义及其时间复杂度的计算。

知识结构图



现在,计算机已经广泛而深入地应用到了人类社会的各个领域,计算机处理的对象也由单纯的数值对象发展到字符、声音、图像等各种各样具有不同结构的数据,为了有效地组织和管理数据,设计出高质量的程序,只有深入分析和研究数据对象自身的特性,以及各数据对象之间的关系,才能对它们进行有效的处理,设计出高效的算法。如何合理地组织数据、高效地处理数据,是“数据结构”主要研究的问题。本章简要介绍有关数据结构的基本概念和算法分析方法。

1.1 概 述

1.1.1 数据结构的研究内容

1. 数值计算解决问题的方法

最初用计算机解决的是数值计算问题,一般需要经过以下几个步骤:

- (1) 建立实际问题的数学模型,也就是求解问题的公式或者方程,同时包括涉及的对象和对象间的关系;
- (2) 设计解决此数学模型的算法;
- (3) 使用某种程序设计语言编写程序;
- (4) 上机调试,直到得到最终结果。



例如,已知半径求圆面积。这是一个数值计算问题,涉及的数据对象为半径 r 、面积 $area$,数学模型是 $area = \pi r^2$,解决问题的基本步骤就是输入半径 r ,根据公式计算,最后输出结果。可以选用某种程序设计语言,例如 C 语言、Java 语言编写出相应的程序,并且上机调试,直到能求出正确的解。

2. 非数值计算问题的实例

在实际的计算机应用中,绝大多数问题是非数值计算问题,无法直接用数学方程求解,下面通过 3 个实例加以说明。

【例 1-1】 学生学籍管理系统。

高等院校的学籍管理系统中存储了学生的信息,包括学号、姓名、性别、专业和籍贯等,如表 1-1 所示。每个学生的基本信息按照学号的顺序,依次存放在“学生基本信息表”中,形成了学生基本信息的线性序列,呈现出线性关系。

表 1-1 学生基本信息表

| 学 号 | 姓 名 | 性 别 | 专 业 | 籍 贯 |
|-----------|-----|-----|----------|-----|
| 160414201 | 王 阳 | 男 | 计算机科学与技术 | 广东 |
| 160414202 | 关 伟 | 男 | 计算机科学与技术 | 辽宁 |
| 160414203 | 刘 鹏 | 男 | 计算机科学与技术 | 吉林 |
| 160414204 | 冯瑞雪 | 女 | 计算机科学与技术 | 河北 |

诸如此类的线性表结构还有图书管理系统、酒店管理系统等。在这类问题中,计算机处理的对象是各种表,数据元素之间存在着简单的一对一线性关系,因此这类问题的数学模型就是各种线性表,施加于对象上的操作有查找、插入和删除等。这类数学模型称为“线性”的数据结构。

【例 1-2】 人机对弈问题。

计算机之所以能和人弈是因为已经在计算机中存储了对弈的策略。由于对弈的过

程是在一定规则下随机进行的,为使计算机能灵活对弈,必须把对弈过程中所有可能发生的情况及相应的对策都加以考虑。此时,把计算机操作的对象——对弈过程中可能出现的棋盘状态称为格局。格局之间的关系由不同的对弈规则决定,这个关系往往是一种层次结构:从一个格局可以派生出若干新的格局。从这个新格局又可以派生出多个更新的格局,将对弈开始到结束整个过程可能派生的所有格局表示出来,就像一棵倒挂的“树”,从初始状态(根)到某一最终格局(叶子)的一条路径,就是一次具体的对弈过程,如图 1-1 所示。

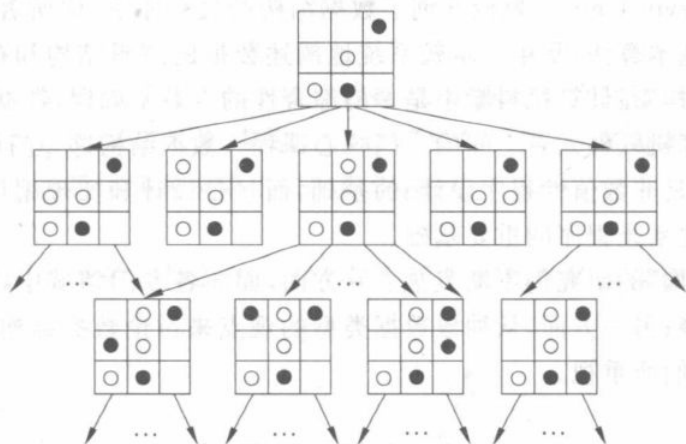


图 1-1 对弈树局部

人机对弈问题的数学模型就是如何用树结构表示棋盘和棋子等,算法是博弈的规则和策略。诸如此类的树结构还有计算机的文件系统、一个单位的组织机构等。在这类问题中,计算机处理的对象是树结构,元素之间是一对多的层次关系,施加于对象上的操作有查找、插入和删除等,这类数学模型称为“树”的数据结构。

【例 1-3】 交通导游图问题。

在计算机网络中有许多交通查询系统。一般只需要按照提示,输入起点和终点,并选取交通方式(自驾、公交)就能显示出相应的自驾路线或详细的换乘方式。有些系统还可以根据不同的要求给出多种不同的方案。

以设计一个高校的校内导游系统为例,假设校内的标志性建筑有 7 个,用 A~G 字母表示,字母间的连线表示两个景点之间的路径,连线上的权值表示景点间的距离,如图 1-2 所示。如何选取一条路径,能够游玩所有的景点并使得行走的路线最短,这类问题就是最短路径问题,最短路径问题的数学模型就是图结构。

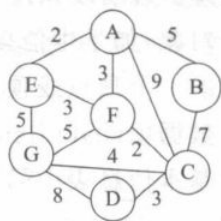


图 1-2 最短路径问题

以图结构作为数学模型的应用有很多,如教学计划、交通运输、网络布线、通信网等,在这类问题中,元素之间是多对多的网状关系,施加于对象上的操作依然有查找、插入和删除,此外还有求关键路径、最小生成树、最短路径等。

3. 数据结构的研究范畴

从上面 3 个实例可以看出,非数值计算问题的数学模型不再是数学方程,而是诸如线

性表、树和图等数据结构。因此,简单地说,数据结构是一门研究非数值计算的程序设计问题中的操作对象,以及这些对象之间的关系和操作的学科。具体的内容在后面的课程中将会进一步讲解。

1.1.2 数据结构的发展过程

“数据结构”是从1968年才开始在国外作为一门独立的课程。1968年美国唐·欧·克努特(Donald Ervin Knuth)教授开创了数据结构的最初体系,他所著的《计算机程序设计艺术》第一卷《基本算法》是第一本较系统地阐述数据的逻辑结构和存储结构及其操作的著作。“数据结构”在计算机科学中是一门综合性的专业基础课,数据结构是介于数学、计算机硬件和计算机软件三者之间的一门核心课程。数据结构这一门课的内容不仅是一般程序设计(特别是非数值性程序设计)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的重要基础。

有关“数据结构”的研究仍不断发展:一方面,面向各专门领域中特殊问题的数据结构正在研究和发展的;另一方面,从抽象数据类型的观点来讨论数据结构,已成为一种新的趋势,越来越被人们所重视。

1.2 基本概念和术语



1.2.1 数据、数据元素、数据项和数据对象

数据(data)是客观事物的符号表示,是所有能输入到计算机中并被计算机程序处理的符号的总称。计算机能处理多种形式的数,例如,科学计算软件处理的是整数、实数等数值数据;文字处理软件处理的是字符数据;多媒体软件处理的是图形、图像、声音、动画等多媒体数据。

数据元素(data element)是数据的基本单位,在计算机中通常作为一个整体进行考虑和处理。在有些情况下,数据元素也称为元素、记录、结点等。数据元素用于完整地描述一个对象,如学生信息表中的学生记录,人机对弈棋盘的一个格局(状态),以及交通旅游图中的一个顶点或顶点间的关系等。

数据项(data item)是组成数据元素的、有独立含义的、不可分割的最小单位。例如,学生基本信息表中的学号、姓名、籍贯等都是数据项。一个数据元素可以由若干数据项组成。

数据对象(data object)是性质相同的数据元素的集合,是数据的一个子集。例如,整数数据对象是集合 $N = \{0, \pm 1, \pm 2, \dots\}$,字符数据对象是集合 $C = \{'A', 'B', \dots, 'Z', 'a', 'b', \dots, 'z'\}$,学生基本信息表也是一个数据对象。数据对象可以是有限的,也可以是无限的。数据元素则是数据对象集合中的数据成员。

1.2.2 逻辑结构和存储结构

利用计算机解决实际问题时涉及两个方面的内容:信息的表示和信息的处理。在大

多数情况下,信息以及信息中的各个元素往往不是孤立的,而是存在着一定的结构关系。同时,这些信息的表示方法又直接关系到处理信息的程序的效率。数据结构就是指相互之间存在一种或多种特定关系的数据元素的集合,或者简称为带结构的数据对象,它研究这些数据在计算机中的存储方式以及处理这些数据的方法。数据结构包括逻辑结构和存储结构两个层次。

1. 逻辑结构

1) 逻辑结构含义

数据的逻辑结构是指从逻辑关系上描述数据,是数据元素之间逻辑关系的整体。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。

2) 逻辑结构种类

根据数据元素间关系的不同特性,通常有下列4类基本的逻辑结构,如图1-3所示。

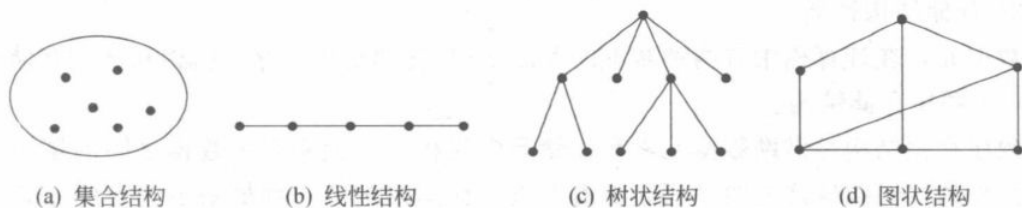


图 1-3 4类基本逻辑结构关系图

(1) 集合结构。在集合中,数据元素间的关系是属于同一个集合,别无其他关系。集合是元素关系极为松散的一种结构。

(2) 线性结构。在线性结构中,数据元素的直接前驱和直接后继之间存在着一对一的关系。

(3) 树状结构。在树状结构中,数据元素的前驱和后继之间存在着一对多的关系。

(4) 图状结构。在图状结构中,数据元素的前驱和后继之间存在着多对多的关系,图状结构也称作网状结构。

其中集合结构、树状结构和图状结构都属于非线性结构。

3) 逻辑结构的表示

数据的逻辑结构有两个要素:一个是数据元素的集合,另一个是关系的集合。因此在形式上,数据结构通常可以采用二元组来表示,定义形式如下:

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

其中, D 是数据元素的有限集, R 是 D 上关系的有限集。

【例 1-4】 设数据的逻辑结构定义为 $T=(D, R)$ 其中:

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

$$R = \{(a, b), (a, c), (b, e), (b, f), (b, g), (c, d)\}$$

请画出该结构的逻辑图。

【问题分析】 本题主要考查数据结构中逻辑结构的概念, D 是数据元素的集合, 每个元素分别用小圆圈表示; R 是元素间关系的集合, 当元素间的关系用 $\langle x, y \rangle$ 表示时, 表示元素间的关系具有方向性, 用带箭头的短线表示, 当元素间的关系用 (x, y) 表示时, 表示元素间的关系为无序, 直接用短线表示。

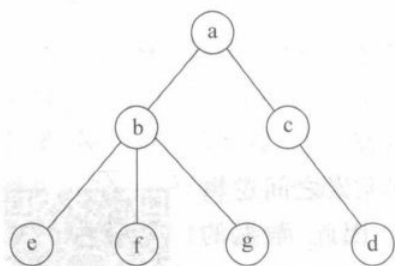


图 1-4 逻辑结构图

【解】 逻辑结构图如图 1-4 所示。

2. 存储结构

1) 存储结构含义

为解决实际问题, 上述逻辑结构的信息应该在计算机中标识(又称映像)出来, 此时体现的是数据结构在计算机中的实现方法, 数据对象在计算机中的存储表示称为数据的存储结构, 也称为物理结构。把数据对象存储到计算机时, 通常要求既要存储各数据元素的数据, 又要存储数据元素之间的逻辑关系, 数据元素在计算机内用结点来表示。

2) 存储结构种类

数据元素在计算机中有两种基本的存储结构, 分别是顺序存储结构和链式存储结构。

(1) 顺序存储结构。

顺序存储结构是借助数据元素在存储器中的相对位置来表示数据之间的逻辑关系, 由此得到的存储结构称为顺序存储结构。顺序存储结构是一种最基本的存储方式, 特别是线性结构。如果采用顺序存储方法, 那么逻辑相邻的数据元素, 物理位置也相邻。顺序存储结构通常借助于程序设计语言中的数组实现。

对于前面的“学生基本信息表”, 假定每个结点(学生记录)占用 50 个存储单元, 数据从 0 号单元开始由低地址向高地址方向存储, 对应的顺序存储结构如表 1-2 所示。

表 1-2 顺序存储结构

| 地址 | 学号 | 姓名 | 性别 | 专业 | 籍贯 |
|-----|-----------|-----|----|----------|----|
| 0 | 160414201 | 王阳 | 男 | 计算机科学与技术 | 广东 |
| 50 | 160414202 | 关伟 | 男 | 计算机科学与技术 | 辽宁 |
| 100 | 160414203 | 刘鹏 | 男 | 计算机科学与技术 | 吉林 |
| 150 | 160414204 | 冯瑞雪 | 女 | 计算机科学与技术 | 河北 |

(2) 链式存储结构。

顺序存储结构要求所有的元素依次存放在一片连续的存储空间中, 而链式存储结构无须占用一整块存储空间, 它是借助指示元素存储地址的指针来表示数据元素之间的逻辑关系, 因此除了存储数据所必需的空间外, 需要附设空间开辟一个指针字段, 用于存放后继元素的存储地址, 由此得到的存储结构称为链式存储结构。在链式存储结构中, 即使逻辑相邻的元素也不要求其物理位置相邻。显然, 链式存储结构通常借助于程序设计语言中的指针类型实现。

假定前面的“学生基本信息表”中, 每个结点(学生记录)依然占用 50 个存储单元, 但

顺序乱了,为了表示出之前的逻辑关系,需要在每个结点中附加一个数据项存储“下一个结点地址”,即后继指针字段,用于存放后继结点的地址,因此,每个结点分成了两部分,一部分存放结点的数据信息,与顺序存储结构的数据元素相同,另一部分存放后继结点的首地址,而且为了找到第一个结点,需要设置头指针,指向第一个结点,最后一个结点的后继地址为 NULL。如表 1-3 所示,其中头指针为 3150。

表 1-3 “学生基本信息表”的链式存储结构

| 地址 | 学号 | 姓名 | 性别 | 专业 | 籍贯 | 后继结点的首地址 |
|------|-----------|-----|----|----------|----|----------|
| 2000 | 160414203 | 刘鹏 | 男 | 计算机科学与技术 | 吉林 | 2900 |
| 2250 | 160414202 | 关伟 | 男 | 计算机科学与技术 | 辽宁 | 2000 |
| 2900 | 160414204 | 冯瑞雪 | 女 | 计算机科学与技术 | 河北 | NULL |
| 3150 | 160414201 | 王阳 | 男 | 计算机科学与技术 | 广东 | 2250 |

为了更清晰地反映链式存储结构,可采用更直观的图示来表示,表 1-3 的“学生基本信息表”的链式存储结构可用如图 1-5 所示的方式表示。



图 1-5 链式存储结构示意图

3. 数据的运算

通常情况下,不同类型的应用有不同的解决方法,精心选择数据的组织形式可以带来更高的运行或者存储效率。数据的运算通常也称为数据的操作,是定义在数据的逻辑结构上的,但运算的具体实现要借助于存储结构,如同上一节所描述的。常用的运算如下。

- (1) 查找:也称为检索,即在数据结构里查找满足一定条件的结点。
- (2) 插入:往数据结构里增加新的结点。
- (3) 删除:把指定的结点从数据结构里去掉。
- (4) 更新:改变指定结点的一个或多个字段的值。

插入、删除、更新运算都包含着查找运算,以确定插入、删除、更新的确切位置。

4. 数据结构包含的三方面内容的关系

数据的逻辑结构、数据的存储结构及数据的运算这 3 个方面构成一个数据结构的整体。数据的逻辑结构属于用户视图,是面向问题的,反映了数据内部的构成方式;数据的存储结构属于具体实现的视图,是面向计算机的,是数据及其关系在计算机内的存储表