



系列规划教材

山东省精品课程配套教材

深入浅出

数据结构与算法

刘晓华 肖进杰 唐焕玲 石艳荣 陈姝颖 编著



清华大学出版社



深入
浅出 系列规划教材

深入 浅出

数据结构与算法

刘晓华 肖进杰 唐焕玲 石艳荣 陈姝颖 编著

清华大学出版社
北京

内 容 简 介

本书阐明了常用的数据结构的内在逻辑关系,讨论了各种结构的物理存储表示方法,通过实例说明各种结构在运算操作时的动态特性,并结合典型应用问题给出算法设计与分析的示例。这样不仅为后续相关课程提供必要的知识准备,更重要的是可以进一步提高读者从事软件分析、设计、编程和数据组织的能力。

全书共 8 章,在内容的组织上遵循由浅入深、循序渐进的原则,按简单的线性结构、树形结构、图结构、查找和排序的次序安排主要教学内容;在内容的叙述上力求做到通俗易懂,算法描述结构清晰、易读易懂,并对每个算法都做了大量注释;全书选取的内容都较好地体现了突出应用的原则,以实例介绍各种数据结构的应用,并在各章都附有相应的习题。

本书可以作为面向应用型的本科院校和高职高专院校计算机类专业的学生的教材,也可以作为大学非计算机专业学生的选修课教材和计算机应用技术人员自学参考书。

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

深入浅出数据结构与算法/刘晓华等编著. —北京:清华大学出版社,2015

深入浅出系列规划教材

ISBN 978-7-302-40722-5

I. ①深… II. ①刘… III. ①数据结构—教材 ②算法分析—教材 IV. ①TP311.12

中国版本图书馆 CIP 数据核字(2015)第 151870 号

责任编辑:白立军 王冰飞

封面设计:傅瑞学

责任校对:焦丽丽

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:三河市少明印务有限公司

经 销:全国新华书店

开 本:185mm×260mm 印 张:17.25 字 数:396千字

版 次:2015年6月第1版 印 次:2015年6月第1次印刷

印 数:1~2000

定 价:35.00元



为什么开发深入浅出系列丛书？

目的是从读者角度写书，开发出高质量的、适合阅读的图书。

“不积跬步，无以至千里；不积小流，无以成江海。”知识的学习是一个逐渐积累的过程，只有坚持系统地学习知识，深入浅出，坚持不懈，持之以恒，才能把一类技术学习好。坚持的动力源于所学内容的趣味性和讲法的新颖性。

计算机课程的学习也有一条隐含的主线，那就是“提出问题→分析问题→建立数学模型→建立计算模型→通过各种平台和工具得到最终正确的结果”，培养计算机专业学生的核心能力是“面向问题求解的能力”。由于目前大学计算机本科生培养计划的特点，以及受教学计划和课程设置的原因，计算机科学与技术专业的本科生很难精通掌握一门程序设计语言或者相关课程。各门课程设置比较孤立，培养的学生综合运用各方面的知识能力方面有欠缺。传统的教学模式以传授知识为主要目的，能力培养没有得到充分的重视。很多教材受教学模式的影响，在编写过程中，偏重概念讲解比较多，而忽略了能力培养。为了突出内容的案例性、解惑性、可读性、自学性，本套书努力在以下方面做好工作。

1. 案例性

所举案例突出与本课程的关系，并且能恰当反映当前知识点。例如，在计算机专业中，很多高校都开设了高等数学、线性代数、概率论，不言而喻，这些课程对于计算机专业的学生来说是非常重要的，但就目前对不少高校而言，这些课程都是由数学系的老师讲授，教材也是由数学系的老师编写，由于学科背景不同和看待问题的角度不同，在这些教材中基本都是纯数学方面的案例，作为计算机系的学生来说，学习这样的教材缺少源动力并且比较乏味，究其原因，很多学生不清楚这些课程与计算机专业的关系是什么。基于此，在编写这方面的教材时，可以把计算机上的案例加入其中，例如，可以把计算机图形学中的三维空间物体图像在屏幕上的伸缩变换、平移变换和旋转变换在矩阵运算中进行举例；可以把双机热备份的案例融入到马尔科夫链的讲解；把密码学的案例融入到大数分解中等。

2. 解惑性

很多教材中的知识讲解注重定义的介绍，而忽略因果性、解释性介绍，往往造成知其然而不知其所以然。下面列举两个例子。

(1) 读者可能对 OSI 参考模型与 TCP/IP 参考模型的概念产生混淆，因为两种模型之

间有很多相似之处。其实,OSI参考模型是在其协议开发之前设计出来的,也就是说,它不是针对某个协议族设计的,因而更具有通用性。而TCP/IP模型是在TCP/IP协议栈出现后出现的,也就是说,TCP/IP模型是针对TCP/IP协议栈的,并且与TCP/IP协议栈非常吻合。但是必须注意,TCP/IP模型描述其他协议栈并不合适,因为它具有很强的针对性。说到这里读者可能更迷惑了,既然OSI参考模型没有在数据通信中占有主导地位,那为什么还花费这么大的篇幅来描述它呢?其实,虽然OSI参考模型在协议实现方面存在很多不足,但是,OSI参考模型在计算机网络的发展过程中起到了非常重要的作用,并且,它对未来计算机网络的标准化、规范化的发展有很重要的指导意义。

(2)再例如,在介绍原码、反码和补码时,往往只给出其定义和举例表示,而对最后在计算机中采取补码表示数值?浮点数在计算机中是如何表示的?字节类型、短整型、整型、长整型、浮点数的范围是如何确定的?下面我们来回答这些问题(以8位数为例子),原码不能直接运算,并且0的原码有+0和-0两种形式,即00000000和10000000,这样肯定是不行的,如果根据原码计算设计相应的门电路,由于要判断符号位,设计的复杂度会大大增加,不合算;为了解决原码不能直接运算的缺点,人们提出了反码的概念,但是0的反码还是有+0和-0两种形式,即00000000和11111111,这样是不行的,因为计算机在计算过程中,不能判断遇到0是+0还是-0;而补码解决了0表示的唯一性问题,即不会存在+0和-0,因为+0是00000000,它的补码是00000000,-0是10000000,它的反码是11111111,再加1就得到其补码是10000000,舍去溢出量就是00000000。知道了计算机中数用补码表示和0的唯一性问题后,就可以确定数据类型表示的取值范围了,仍以字节类型为例,一个字节共8位,有00000000~11111111共256种结果,由于1位表示符号位,7位表示数据位,正数的补码好说,其范围从00000000~01111111,即0~127;负数的补码为10000000~11111111,其中,11111111为-1的补码,10000001为-127的补码,那么到底10000000表示什么最合适呢?8位二进制数中,最小数的补码形式为10000000;它的数值绝对值应该是各位取反再加1,即为01111111+1=10000000=128,又因为是负数,所以是-128,即其取值范围是-128~127。

3. 可读性

图书的内容要深入浅出,使人爱看、易懂。一本书要做到可读性好,必须做到“善用比喻,实例为王”。什么是深入浅出?就是把复杂的事物简单地描述明白。把简单事情复杂化的是哲学家,而把复杂的问题简单化的是科学家。编写教材时要以科学家的眼光去编写,把难懂的定义,要通过图形或者举例进行解释,这样能达到事半功倍的效果。例如,在数据库中,第一范式、第二范式、第三范式、BC范式的概念非常抽象,很难理解,但是,如果以一个教务系统中的学生表、课程表、教师表之间的关系为例进行讲解,从而引出范式的概念,学生会比较容易接受。再例如,在生物学中,如果纯粹地讲解各个器官的功能会比较乏味,但是如果提出一个问题,如人的体温为什么是37℃?以此为引子引出各个器官的功能效果要好得多。再例如,在讲解数据结构课程时,由于定义多,表示抽象,这样达不到很好的教学效果,可以考虑在讲解数据结构及其操作时用程序给予实现,让学生看到直接的操作结果,如压栈和出栈操作,可以把PUSH()和POP()操作实现,这样效果会好

很多,并且会激发学生的学习兴趣。

4. 自学性

一本书如果适合自学学习,对其语言要求比较高。写作风格不能枯燥无味,让人看一眼就拒人千里之外,而应该是风趣、幽默,重要知识点多举实际应用的案例,说明它们在实际生活中的应用,应该有画龙点睛的说明和知识背景介绍,对其应用需要注意哪些问题等都要有提示等。

一书在手,从第一页开始的起点到最后一页的终点,如何使读者能快乐地阅读下去并获得知识?这是非常重要的问题。在数学上,两点之间的最短距离是直线。但在知识的传播中,使读者感到“阻力最小”的书才是好书。如同自然界中没有直流的河流一样,河水在重力的作用下一定沿着阻力最小的路径向前进。知识的传播与此相同,最有效的传播方式是传播起来损耗最小,阅读起来没有阻力。

是为序。

欢迎老师投稿: bailj@tup.tsinghua.edu.cn。

2014年12月15日



“数据结构与算法”是计算机科学与技术专业的一门重要的专业基础课和核心课程。众所周知,计算机科学是一门研究数据表示和数据处理的科学。数据是计算机可以直接处理的最基本和最重要的对象,无论是进行科学计算、数据处理、过程控制,还是对文件的存储和检索及数据库技术应用等,都是对数据进行加工处理的过程。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、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。

随着现代软件工业的飞速发展,以及面向对象程序设计思想及组件技术的发展,软件开发方式和过去相比已经有了很大的变化。各种开发工具如雨后春笋般不断涌现。这些开发工具大部分都能够自动完成程序中一些机械的代码,并隐藏了其中的实现细节。对于一些简单的应用程序,程序员关注的焦点不再是数据结构和算法,而是用户界面、可重用性、开发效率等。然而,无论软件工程如何发展,开发工具如何进步,再先进的控件、再复杂的类库也需要人来写,必须懂得算法和数据结构才能写出这些类库和控件。学习数据结构和算法并不仅仅要求我们学会如何使用和实现某种数据结构,更重要的是学会分析解决问题的思想和方法。只要我们的计算机体系不变,数据结构和算法仍然是程序的核心,永远不会被淘汰。打好“数据结构与算法”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、数据库管理系统、软件工程、编译原理、人工智能等都是十分有益的。

编者从事计算机专业教学二十多年,多次讲授“数据结构与算法”这门课程,在认真总结这二十多年教学经验和体会的基础上,结合现阶段大学生的素质特点和学习现状,参考了许多相关的教材,面向应用型本科院校的计算机及相关专业的学生编写了这本教材。在编写教材时力求突出学科的理论与实践紧密结合的特征,结合实例讲解理论,使理论来源于实践,又进一步指导实践;同时还做到内容通俗易懂,易教易学,在教给学生计算机专业的基本理论和基本知识的同时,更注重培养学生分析问题、解决问题的能力。

本书共分为8章。第1章综述了数据、数据结构和算法等基本概念;第2章至第4章讨论了线性表,栈和队列,串、数组和广义表等简单的线性结构及应用;第5章、第6章对树、二叉树以及图等复杂的层次结构和网状结构进行了探讨;第7章、第8章讨论了各种查找方法和内部排序方法。每一部分除了介绍各种数据结构和应用之外,还对重要算法从时间和空间上进行了分析和比较。除此之外,每章还配备了各种题型的练习题,供学生对所学知识加以巩固和强化。

本书具有以下特色：

(1) 内容精练。内容的组织和编排以应用为主线。在内容的组织上遵循由浅入深、循序渐进的原则，按简单的线性结构、树形结构、图结构、查找和排序的次序安排主要教学内容。

(2) 算法完整。每个算法都用标准 C 给出了完整描述。算法描述结构清晰、易读易理解，并对每个算法都做了大量注释，易于在主函数中调用运行。

(3) 图文并茂。本教材配合主要内容设计了许多插图，以图例来展现问题的求解过程，从而降低了学生理解问题的复杂性。

(4) 重点突出。每章开始给出本章的主要教学内容，使学生在在学习之前就能明白需要重点掌握的知识；每章结束给出本章小结，对本章的重点内容进行梳理和总结，更有利于学生理解和复习。

(5) 案例驱动。每种数据结构给出若干实例，以实例介绍各种数据结构的应用。而且每一章结束都有一个和实际问题相结合的应用实例，从问题描述、问题分析、程序设计到最终的数据测试和运行结果，使学生对如何把所学知识运用到实践中有更充分的了解和认识，从而提高学生分析问题和解决问题的能力。

(6) 启发性强。编者精心编写了具有启发性的例题和各种题型的习题，引导学生的思维过程，加强学生对所学知识的理解和应用。

本书由刘晓华、肖进杰、唐焕玲、石艳荣和陈姝颖编写。其中，第 1 章、第 2 章由陈姝颖执笔；第 3 章、第 4 章由石艳荣执笔；第 5 章由肖进杰执笔；第 6 章由刘晓华执笔，第 7 章、第 8 章由唐焕玲执笔；刘晓华负责全书内容的统筹和修改。

本书的出版获 2014 年山东省普通高校应用型人才培养专业发展支持计划项目和 2011 年山东省高等学校省级精品课程建设项目资助。本书在编写过程中也得到了作者所在单位的同事们和清华大学出版社的大力支持和帮助，在此一并表示衷心的感谢。

由于编者水平有限，书中疏漏与不足之处在所难免，恳请各位专家和读者批评指正。编者的联系方式为 liuxiaohua@sdiibt.edu.cn。

编者

2015 年 5 月



第 1 章	绪论	1
1.1	什么是数据结构	1
1.1.1	学习数据结构的目的是	1
1.1.2	有关概念和术语	4
1.2	数据类型和抽象数据类型	6
1.2.1	数据类型	6
1.2.2	抽象数据类型	6
1.3	算法与算法分析	8
1.3.1	算法的特性	8
1.3.2	算法描述	10
1.3.3	算法效率的度量	10
	本章小结	13
	习题 1	13
第 2 章	线性表	16
2.1	线性表的逻辑结构	16
2.1.1	线性表的定义	16
2.1.2	线性表的抽象数据类型	17
2.2	线性表的顺序存储与实现	18
2.2.1	顺序表	18
2.2.2	顺序表基本操作的实现	19
2.2.3	顺序表应用举例	23
2.3	线性表的链式存储与实现	25
2.3.1	单链表	25
2.3.2	单链表上基本运算的实现	27
2.3.3	单链表的应用	30
2.3.4	循环链表	31
2.3.5	双向链表	32
2.3.6	静态链表	33

2.4 一元多项式的表示及加法实现..... 36

2.5 应用实例——约瑟夫环问题..... 38

本章小结 42

习题 2 42

第 3 章 限定性线性表——栈和队列..... 46

3.1 栈..... 46

 3.1.1 栈的定义 46

 3.1.2 栈的表示和实现 47

3.2 栈的应用举例..... 52

3.3 队列..... 56

 3.3.1 队列的定义 56

 3.3.2 队列的表示和实现 57

3.4 队列的应用举例..... 62

3.5 应用实例——银行排队服务模拟..... 63

本章小结 68

习题 3 68

第 4 章 串、数组和广义表 72

4.1 串的定义..... 72

4.2 串的表示和实现..... 73

 4.2.1 定长顺序存储表示 74

 4.2.2 堆分配存储表示 75

 4.2.3 串的块链存储表示 77

4.3 模式匹配..... 77

 4.3.1 简单模式匹配 78

 4.3.2 一种改进的模式匹配 79

4.4 数组..... 81

 4.4.1 数组的定义 81

 4.4.2 数组的顺序存储与实现 82

 4.4.3 矩阵的压缩存储 83

4.5 广义表..... 91

 4.5.1 广义表的定义 91

 4.5.2 广义表的存储结构 92

4.6 应用实例——投票选举..... 93

本章小结 96

习题 4 97

第 5 章	树和二叉树	100
5.1	树的基本概念	100
5.1.1	树的定义	100
5.1.2	树的基本术语	102
5.2	二叉树	104
5.2.1	二叉树的定义	104
5.2.2	二叉树的性质	105
5.2.3	二叉树的存储结构	107
5.3	二叉树的遍历	110
5.3.1	二叉树的遍历算法	110
5.3.2	二叉树遍历算法的应用	112
5.4	线索二叉树	114
5.4.1	线索二叉树的定义	114
5.4.2	二叉树的线索化	115
5.4.3	线索二叉树的遍历	117
5.5	树和森林	118
5.5.1	树的存储结构	118
5.5.2	森林与二叉树的转换	121
5.5.3	树和森林的遍历	122
5.6	哈夫曼树及其应用	123
5.6.1	基本术语	123
5.6.2	构造哈夫曼树	124
5.6.3	哈夫曼树的应用	127
5.7	应用实例——并查集	129
	本章小结	132
	习题 5	133
第 6 章	图	136
6.1	图的基本概念	136
6.1.1	图的定义	136
6.1.2	图的基本术语	137
6.2	图的存储结构	141
6.2.1	邻接矩阵	141
6.2.2	邻接表	143
6.2.3	有向图的十字链表	146
6.2.4	无向图的邻接多重表	147
6.3	图的遍历	149



6.3.1	深度优先搜索	149
6.3.2	广度优先搜索	151
6.4	无向图的连通分量和生成树	152
6.5	图的应用	153
6.5.1	最小生成树	153
6.5.2	有向无环图与拓扑排序	157
6.5.3	关键路径	161
6.5.4	最短路径	166
6.6	应用实例——畅通工程	171
	本章小结	175
	习题 6	175
第 7 章	查找	179
7.1	查找的基本概念	179
7.2	静态查找	180
7.2.1	顺序查找	181
7.2.2	折半查找	183
7.2.3	分块查找	185
7.3	动态查找	187
7.3.1	二叉排序树	187
7.3.2	平衡二叉树	194
7.3.3	B 树	203
7.4	哈希表	204
7.4.1	哈希表的概念	205
7.4.2	哈希函数的构造	205
7.4.3	处理冲突的方法	207
7.4.4	哈希表查找及其分析	209
7.5	应用实例——通讯录查询系统	210
	本章小结	217
	习题 7	218
第 8 章	排序	221
8.1	排序的基本概念	221
8.2	插入排序	222
8.2.1	直接插入排序	223
8.2.2	折半插入排序	224
8.2.3	希尔排序	225
8.3	交换排序	227

8.3.1	冒泡排序	227
8.3.2	快速排序	229
8.4	选择排序	232
8.4.1	简单选择排序	232
8.4.2	堆排序	233
8.5	归并排序	237
8.6	基数排序	239
8.6.1	多关键字排序	239
8.6.2	链式基数排序	240
8.7	内部排序方法比较	244
8.8	应用实例——内部排序算法比较	246
	本章小结	255
	习题 8	256
	参考文献	260

本章主要内容:

- (1) 数据结构中的名词、术语及基本概念;
- (2) 数据的逻辑结构与存储结构;
- (3) 算法的 5 个重要特性及设计算法的原则;
- (4) 算法的时间复杂度和空间复杂度分析。

随着计算机的广泛应用,计算机已不再局限于科学计算,而更多地用于数据处理、控制、信息管理及音频、视频等非数值计算的处理工作。用计算机求解任何问题都离不开程序设计,而程序设计的实质是数据表示和数据处理。数据要能被计算机处理,首先必须能够存储在计算机的内存中,这项任务称为数据表示,数据表示的核心任务是数据结构的设计;一个实际问题的求解必须满足各项处理要求,这项任务称为数据处理,数据处理的的核心任务是算法设计。数据结构课程主要讨论数据表示和数据处理的基本问题。本章将概括地介绍数据结构的基本概念、基本思想和基本方法。

1.1 什么是数据结构

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,要想有效地使用计算机、充分发挥计算机的性能,除了掌握几种计算机程序设计语言外,还必须学习和掌握好数据结构的有关知识。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 学习数据结构的目的是

在计算机发展的初期,人们使用计算机的目的主要是处理数值计算问题。当使用计算机来解决一个具体问题时,一般需要经过下列几个步骤:首先要从该问题抽象出一个适当的数学模型,然后设计或选择一个解此数学模型的算法,最后编制出程序进行调试、测试,直至得到最终的解答。例如,求解梁架结构中应力的数学模型为线性方程组,该方程组可以使用迭代算法来求解;预报人口增长情况的数学模型为微分方程。

随着计算机应用领域的不断扩大和软、硬件技术的发展,非数值计算问题越来越显得

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结构更为复杂,数据元素之间的相互关系一般无法用数学方程式加以描述。因此,解决这类问题的关键不再是数学分析和计算方法,而是要设计出合适的数据结构。下面列举几个例子来说明什么是数据结构。

例 1.1 学生信息检索系统。当需要查找某个学生的有关情况的时候,或者想查询某个专业或年级的学生的有关情况的时候,只要建立了相关的数据结构,按照某种算法编写了相关程序,就可以实现计算机自动检索。由此,可以在学生信息检索系统中建立一张按学号顺序排列的学生信息表和分别按姓名、专业、年级顺序排列的索引表,如图 1.1 所示。

学号	姓名	性别	专业	年级
100001	安金正	男	计算机科学与技术	10级
100002	付林	女	信息与计算科学	10级
110301	刘丽	女	数学与应用数学	11级
110302	曲超	男	信息与计算科学	11级
110303	孙明	男	计算机科学与技术	11级
120801	赵科	女	计算机科学与技术	12级
120802	朱琳	男	数学与应用数学	12级
130803	崔文靖	男	信息与计算科学	13级
130601	刘丽	女	计算机科学与技术	13级
130602	修云飞	男	数学与应用数学	13级

(a) 学生信息表

安金正	1
付林	2
刘丽	3, 9
曲超	4
孙明	5
赵科	6
朱琳	7
崔文靖	8
修云飞	10

(b) 姓名索引表

计算机科学与技术	1, 5, 6, 9
信息与计算科学	2, 4, 8
数学与应用数学	3, 7, 10

(c) 专业索引表

10级	1, 2
11级	3, 4, 5
12级	6, 7
13级	8, 9, 10

(d) 年级索引表

图 1.1 学生信息查询系统中的数据结构

由这 4 张表构成的文件便是学生信息检索的数学模型,计算机的主要操作便是按照某个特定要求(如给定姓名或学号)对学生信息文件进行查询。

诸如此类的还有图书馆的书目自动检索系统、电话自动查号系统、考试查分系统、仓库库存管理系统等。在这类文档管理的数学模型中,计算机处理的对象之间通常存在的是一种最简单的线性关系,这类数学模型可称为线性的数据结构。

例 1.2 计算机和人对弈问题。计算机之所以能和对弈,是因为有人将对弈的策略事先存储在计算机中。由于对弈的过程是在一定规则下随机进行的,所以为使计算机能灵活对弈就必须对对弈过程中所有可能发生的情况以及相应的策略都考虑周全,并且,

一个“好”的棋手在对弈时不仅要看棋盘当时的状态,还应该能预测棋局发展的趋势,甚至最后结局。因此,在对弈问题中,计算机操作的对象是对弈过程中可能出现的棋盘状态,称为格局。例如,图 1.2(a)所示为井字棋的一个格局,而格局之间的关系式由比赛规则决定。通常,这个关系不是线性的,因为从一个棋盘格局可以派生出几个格局,例如,从图 1.2(a)可以派生出 5 个格局,如图 1.2(b)所示,而从每一个新的格局又可能派生出 4 个可能出现的格局。因此,若将从对弈开始到结束的过程中所有可能出现的格局都画在一张图上,则可得到一棵倒长的“树”。“树根”是对弈开始之前的棋盘格局,而所有的“叶子”就是可能出现的结局,对弈的过程就是从树根沿树杈到某个叶子的过程。“树”可以是某些非数值计算问题的数学模型,在这种模型中,数据间的关系是一对多的非线性关系,称之为树形结构。

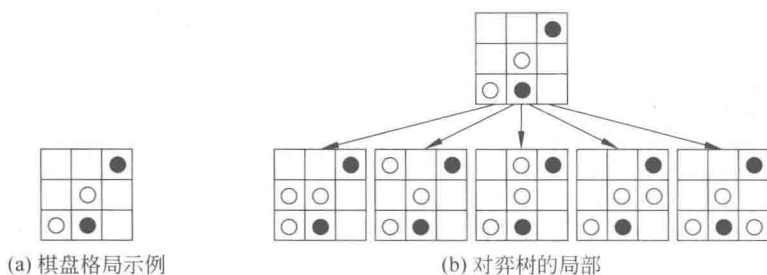


图 1.2 井字棋对弈树

例 1.3 城市之间的公路网问题。假设在 n 个城市之间想建一条公路网,则连通 n 个城市只需要 $n-1$ 条道路。设公路的建设费用与公路的长度成正比,那么如何在最节省经费的前提下建立这个公路网?

如图 1.3 所示,在城市公路网中,每个顶点代表一个城市,边表示城市之间的道路。在这种数据结构中,数据之间的关系是多对多的非线性关系,称这种数据结构为图形结构



图 1.3 城镇之间的公路网

构。类似结构的还有多岔路口信号灯的设置、地图的染色问题及教学计划编排问题等。

由以上3个例子可见,描述这类非数值计算问题的数学模型不再是数学方程,而是诸如表、树、图之类的数据结构。因此,可以说数据结构课程主要是研究非数值计算的程序设计问题中所出现的计算机操作对象以及它们之间的关系和操作的学科。

学习数据结构的目的是了解和掌握计算机处理对象的特性,将实际问题中所涉及的处理对象在计算机中表示出来并对它们进行处理。同时,通过算法训练来提高学生的思维能力,通过程序设计的技能训练来促进学生的综合应用能力和专业素质的提高。

1.1.2 有关概念和术语

在系统地学习数据结构知识之前,先理解一些基本概念和术语的确切含义。

数据(data)是信息的载体,是对自然界客观事物的符号表示,它能够被计算机识别、存储和加工处理。它是计算机程序加工的原料,应用程序能处理各种各样的数据。计算机科学中,所谓数据就是计算机加工处理的对象,它可以是数值数据,也可以是非数值数据。数值数据是一些整数、实数或复数,主要用于工程计算、科学计算等;非数值数据包括图形、图像、语音、视频等。

数据元素(data element)是数据的基本单位,在计算机程序中通常作为一个整体进行考虑和处理。在不同的条件下,数据元素又可称为元素、结点、顶点、记录等。例如,学生信息检索系统中学生信息表中的一个记录、人机对弈问题中状态树的一个状态、城镇公路网问题中的一个顶点等,都被称为一个数据元素。

数据项(data item)是数据处理中的不可分割的、具有独立意义的最小标识单位。一个数据元素可由若干个数据项组成。例如,学籍管理系统中学生信息表的每一个数据元素就是一个学生记录。它包括学生的学号、姓名、性别、籍贯、出生年月、成绩等数据项。这些数据项可以分为两种:一种叫做初等项,如学生的性别、籍贯等,这些数据项是在数据处理时不能再分割的最小单位;另一种叫做组合项,如学生的成绩,它可以再划分为数学、物理、化学等更小的项。通常,在解决实际问题时是把每个学生记录当作一个基本单位进行访问和处理的。

数据对象(data object)是具有相同性质的数据元素的集合,是数据的一个子集。在某个具体问题中,数据元素都具有相同的性质(元素值不一定相等),属于同一数据对象。例如,整数数据对象是集合 $N = \{0, \pm 1, \pm 2, \dots\}$, 字母字符数据对象是集合 $C = \{'A', 'B', \dots\}$ 。

数据结构(data structure)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,在它们之间都存在着某种关系,这种数据元素之间的相互关系称为结构。根据数据元素之间关系的不同特性,通常有下列四类基本结构。

(1) 集合结构:在集合结构中,数据元素之间的关系是“属于同一个集合”。集合是元素关系极为松散的一种结构。

(2) 线性结构:该结构的数据元素之间存在着一对一的关系。

(3) 树形结构:该结构的数据元素之间存在着一对多的层次关系。