

高等院校信息技术规划教材

数据结构与算法

冯贵良 编著 李忠华 主审



清华大学出版社

高等院校信息技术规划教材

数据结构与算法

冯贵良 编著

清华大学出版社
北京

内 容 简 介

本书把数据结构的原理和算法分析技术有机地结合在一起,系统地介绍各种数据结构及各种数据结构的相关算法,使用C语言作为算法描述语言,通过C语言实现了具体算法,能够更好地让学生理解各种数据结构的基本描述方法,借助抽象数据类型,从逻辑结构的角度系统地介绍线性表、栈、队列、串、数组、矩阵、广义表、二叉树、树和图等各种基本数据结构;从算法的角度讨论查找方法和排序算法;从应用的角度介绍了一些具体的应用在C语言下的代码实现。

全书共分三部分:第1部分(第1章)为基本概念介绍部分,介绍数据结构、抽象数据类型以及算法的基本概念;第2部分(第2~8章)为基本数据结构部分,重点介绍线性表、栈、队列、串、数组、矩阵、广义表、二叉树、树和图等各种基本数据结构,并且附带有具体的算法实现的代码;第3部分(第9章和第10章)为算法应用设计,重点介绍顺序查找、折半查找、插值查找、斐波那契查找、分块查找等静态查找的具体算法,介绍在二叉排序树、平衡二叉树上的动态查找算法以及哈希表查找算法等。全书提供了大量应用实例,每章后均附有习题。

本书适合作为高等院校计算机、软件工程专业高年级本科生、研究生的教材,同时可供对数据结构比较熟悉并且对软件设计有所了解的开发人员、广大科技工作者和研究人员参考

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/冯贵良编著. —北京:清华大学出版社,2016

高等院校信息技术规划教材

ISBN 978-7-302-43216-6

I. ①数… II. ①冯… 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(2016)第041728号

责任编辑:白立军 薛 阳

封面设计:常雪影

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:三河市吉祥印务有限公司

经 销:全国新华书店

开 本:185mm×260mm 印 张:16.75

字 数:387千字

版 次:2016年9月第1版

印 次:2016年9月第1次印刷

印 数:1~2000

定 价:35.00元

产品编号:065886-01

前言

foreword

数据结构的概念最早由 C. A. R. Hoare 于 1966 年提出。在他的经典论文《数据结构笔记》中,首次系统地论述了一组数据结构的构造、表示和操作等问题。1973 年,E. Knuth 在《计算机程序设计技巧》第一卷中给出了关于“信息结构”的系统论述。1976 年,N. Wirth 用“算法+数据结构=程序”这个公式表达了算法与数据结构的联系和它们在程序设计中的地位,从此确立了数据结构在计算机相关专业中的核心基础课程地位。

数据结构的研究不仅涉及计算机硬件(编码理论、存储装置和存取方法)的研究范围,而且和计算机软件的研究有着密切的关系,因此数据结构可以认为是介于数学、计算机硬件和计算机软件之间的一门核心课程。在计算机科学中,它不仅是程序设计的基础,而且是设计和实现操作系统、数据库系统和大型应用系统的重要基础。

“数据结构”课程的主要任务是讨论现实世界中数据的各种逻辑结构、在计算机中的存储结构以及进行各种非数值运算的算法。本书主要内容包括线性表、栈、队列、串、数组、广义表、树和二叉树、图、动态存储管理、查找、排序。

通过本课程的学习,使学生掌握数据组织、存储和处理的常用方法及基于各类常见数据结构的算法设计与分析,进一步强化学生的编程能力,为以后进行软件开发和学习后续专业课程打下基础。

在长期的教学过程中,我们认为,“数据结构”是一门兼具理论性与实践性的课程,在掌握程序设计语言后,本课程还是一门加强与提高学生程序设计能力的重要课程。因此,本书以传统的数据结构的主要内容为主线,在充分讨论结构的逻辑特征与存储表示的基础上,用 C 语言完成数据结构的描述和实现。同时,我们更加强调数据结构的应用,对不同的数据结构类型设计多个应用实例,每一算法或程序的编写力求高效、易读,并遵循程序设计的规范,从而帮助读者将数据结构与工程应用有机地结合起来。

在专业程序员的成长过程中,对于数据结构与算法的学习至关

重要。虽然有许多关于数据结构与算法的书籍,但是本书把数据结构的原理和算法分析技术有机地结合在一起,系统地介绍了各种类型的数据结构和排序、检索的各种算法,以 C 语言作为算法描述语言,通过 C 语言实现具体算法,能够更好地让学生理解各种数据结构的基本描述方法,借助抽象数据类型,从逻辑结构的角度系统地介绍线性表、栈、队列、串、数组、矩阵、广义表、二叉树、树和图等各种基本数据结构;从算法的角度讨论查找方法和排序算法;从应用的角度介绍一些具体的算法在 C 语言下的实现代码。

书中不当之处,恳请广大读者指正。

编 者

2016 年 7 月

目录

Contents

第 1 章	绪论	1
1.1	数据结构的概念	1
1.1.1	为什么要学习数据结构	1
1.1.2	有关概念和术语	4
1.1.3	数据结构课程的内容	6
1.2	数据类型与抽象数据类型	7
1.2.1	数据类型	7
1.2.2	抽象数据类型	7
1.3	算法和算法分析	8
1.3.1	算法特性	8
1.3.2	算法描述	9
1.3.3	算法性能分析与度量	9
第 2 章	线性表	11
2.1	线性表的逻辑结构	11
2.1.1	线性表的定义	11
2.1.2	线性表的基本操作	11
2.2	线性表的顺序存储及运算实现	12
2.2.1	顺序表	12
2.2.2	顺序表上基本运算的实现	14
2.2.3	顺序表应用举例	18
2.3	线性表的链式存储和运算实现	20
2.3.1	单链表	20
2.3.2	单链表上基本运算的实现	22
2.3.3	循环链表	28
2.3.4	双向链表	29
2.3.5	静态链表	30

2.3.6 单链表应用举例	32
2.4 顺序表和链表的比较	34
第3章 栈和队列	36
3.1 栈	36
3.1.1 栈的定义及基本操作	36
3.1.2 栈的存储实现和操作实现	37
3.2 栈的应用举例	40
3.3 队列	50
3.3.1 队列的定义及基本运算	50
3.3.2 队列的存储实现及运算实现	50
3.4 队列应用举例	56
习题	59
第4章 串	60
4.1 串及基本运算	60
4.1.1 串的基本概念	60
4.1.2 串的基本运算	60
4.2 串的定长度顺序存储及基本运算	62
4.2.1 串的定长顺序存储	62
4.2.2 定长顺序串的基本运算	63
4.2.3 模式匹配	64
4.3 串的堆存储结构	69
4.3.1 串名的存储映像	69
4.3.2 堆存储结构	71
4.3.3 基于堆结构的基本运算	71
习题	73
第5章 数组、特殊矩阵和广义表	74
5.1 多维数组	74
5.1.1 数组的逻辑结构	74
5.1.2 数组的内存映像	74
5.2 特殊矩阵的压缩存储	77
5.2.1 对称矩阵	77
5.2.2 三角矩阵	78
5.2.3 带状矩阵	79
5.3 稀疏矩阵	80

5.3.1 稀疏矩阵的三元组表存储	80
5.3.2 稀疏矩阵的十字链表存储	86
5.4 广义表	92
5.4.1 广义表的定义和基本运算	92
5.4.2 广义表的存储	93
5.4.3 广义表的基本操作的实现	95
习题	99
第 6 章 二叉树	101
6.1 定义与性质	101
6.1.1 二叉树的基本概念	101
6.1.2 二叉树的主要性质	103
6.2 基本操作与存储实现	104
6.2.1 二叉树的存储	104
6.2.2 二叉树的基本操作及实现	107
6.3 二叉树的遍历	110
6.3.1 二叉树的遍历方法及递归实现	110
6.3.2 二叉树遍历的非递归实现	112
6.3.3 由遍历序列恢复二叉树	116
6.3.4 不用栈的二叉树遍历的非递归方法	118
6.4 线索二叉树	118
6.4.1 线索二叉树的定义及结构	118
6.4.2 线索二叉树的基本操作实现	120
6.5 二叉树的运用	126
6.5.1 二叉树遍历的运用	126
6.5.2 最优二叉树——哈夫曼树	129
习题	135
第 7 章 树	137
7.1 树的概念与表示	137
7.1.1 树的定义及相关术语	137
7.1.2 树的表示	138
7.2 树的基本操作与存储	139
7.2.1 树的基本操作	139
7.2.2 树的存储结构	140
7.3 树、森林与二叉树的转换	143
7.3.1 树转换为二叉树	143

7.3.2	森林转化为二叉树	144
7.3.3	二叉树转换为树和森林	145
7.4	树和森林的遍历	146
7.4.1	树的遍历	146
7.4.2	森林的遍历	147
7.5	树的应用	147
7.5.1	判定树	147
7.5.2	集合的表示	149
7.5.3	关系等价求等价类问题	151
习题		152
第8章	图	153
8.1	图的定义和术语	153
8.1.1	图的定义	153
8.1.2	图的相关术语	153
8.1.3	图的基本操作	156
8.2	图的存储表示	157
8.2.1	邻接矩阵	157
8.2.2	邻接表	159
8.2.3	十字链表	161
8.2.4	邻接多重表	163
8.3	图的遍历	165
8.3.1	深度优先搜索	165
8.3.2	广度优先搜索	167
8.4	图的连通性	169
8.4.1	无向图的连通性	169
8.4.2	有向图的连通性	169
8.4.3	生成树和生成森林	170
8.4.4	关结点和重连通分量	172
8.5	最小生成树	175
8.5.1	最小生成树的基本概念	175
8.5.2	构造最小生成树的 Prim 算法	176
8.5.3	构造最小生成树的 Kruskal 算法	178
8.6	最短路径	181
8.6.1	从一个源点到其他各点的最短路径	181
8.6.2	每一对顶点之间的最短路径	183
8.7	有向无环图及其应用	186
8.7.1	有向无环图的概念	186

8.7.2	AOV 网与拓扑排序	187
8.7.3	AOE 网与关键路径	192
习题		196
第 9 章	查找	197
9.1	基本概念与术语	197
9.2	静态查找表	199
9.2.1	静态查找表结构	199
9.2.2	顺序查找	200
9.2.3	有序表的折半查找	201
9.2.4	有序表的插值查找和斐波那契查找	203
9.2.5	分块查找	205
9.3	动态查找表	205
9.3.1	二叉排序树	205
9.3.2	平衡二叉树	210
9.3.3	B-树和 B+树	216
9.4	哈希表查找(杂凑法)	223
9.4.1	哈希表与哈希方法	223
9.4.2	常用的哈希函数	224
9.4.3	处理冲突的方法	225
9.4.4	哈希表的查找分析	229
习题		230
第 10 章	排序	231
10.1	基本概念	231
10.2	插入排序	231
10.2.1	直接插入排序	231
10.2.2	折半插入排序	233
10.2.3	表插入排序	234
10.2.4	希尔排序	236
10.3	交换排序	238
10.3.1	冒泡排序	238
10.3.2	快速排序	239
10.4	选择排序	241
10.4.1	简单选择排序	242
10.4.2	树型选择排序	242
10.4.3	堆排序	243

10.5	二路归并排序	246
10.6	基数排序	248
10.6.1	多关键码排序	248
10.6.2	链式基数排序	248
10.7	外部排序	251
10.7.1	外部排序的方法	251
10.7.2	多路平衡归并的实现	253
习题		255

绪 论

计算机科学是一门研究数据表示和数据处理的科学。数据是计算机化的信息,它是计算机可以直接处理的最基本和最重要的对象。当我们使用计算机进行科学计算、数据处理、过程控制以及对文件的存储和检索时,都需要对数据进行加工处理。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。

1.1 数据结构的概念

数据结构是计算机科学与技术的专业基础课,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实际问题,仅掌握几种计算机程序设计语言难以应付众多复杂的课题。想要有效地使用计算机、充分发挥计算机的性能,必须学好数据结构这门课程,而且学好数据结构课程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 为什么要学习数据结构

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。当人们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:首先要从该具体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据模型的线性方程组,可以使用迭代算法来求解。由于当时所涉及的运算对象是简单的整型、实型或布尔类型数据,所以程序设计者的主要精力集中于程序设计的技巧上,而无须重视数据结构。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,当今处理的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,才能有效地解决问题。下面所列举的就是属于这一类的具体问题。

【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,因此,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、年级顺序的索引表,如图 1.1 所示。由这 4 张表构成的文件便是学生信息检索系统的数学模型,计算机的主要操作便是按照某个特定要求(如给定姓名)对学生信息文件进行查询。

学号	姓名	性别	专业	年级
201242231	李欢	女	计算机科学与技术	2012级
201242115	廖艺洪	女	计算机网络	2012级
201242300	王伟	男	电子信息工程	2012级
201342426	叶春永	男	计算机网络	2013级
201342457	马晓博	女	计算机科学与技术	2013级
201042164	卫晓辉	男	计算机科学与技术	2010级
201042098	陈毅帆	男	电子信息工程	2010级
201042096	王建强	男	计算机网络	2010级
201142188	王伟	男	计算机科学与技术	2011级
201142183	李丽	女	电子信息工程	2011级

(a) 学生信息表

王建强	8
卫晓辉	6
廖艺洪	2
王伟	3,9
马晓博	5
李丽	10
李欢	1
陈毅帆	7
叶春永	4

(b) 姓名索引表

计算机科学与技术	1,5,6,9
计算机网络	2,4,8
电子信息工程	3,7,10

(c) 专业索引表

2010级	6,7,8
2011级	9,10
2012级	1,2,3
2013级	4,5

(d) 年级索引表

图 1.1 学生信息查询系统中的数据结构

诸如此类的还有电话自动查号系统、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】 八皇后问题。在八皇后问题中,处理过程不是根据某种确定的计算法则,而是利用试探和回溯的探索技术求解。为了求得合理布局,在计算机中要存储布局的当前状态。从最初的布局状态开始,一步步地进行试探,每试探一步形成一个新的状态,整个试探过程形成了一棵隐含的状态树,如图 1.2 所示(为了描述方便,将八皇后问题简化为四皇后问题)。回溯法求解过程实质上就是一个遍历状态树的过程。在这个问题中所出现的树也是一种数据结构,它可以应用在许多非数值计算的问题中。

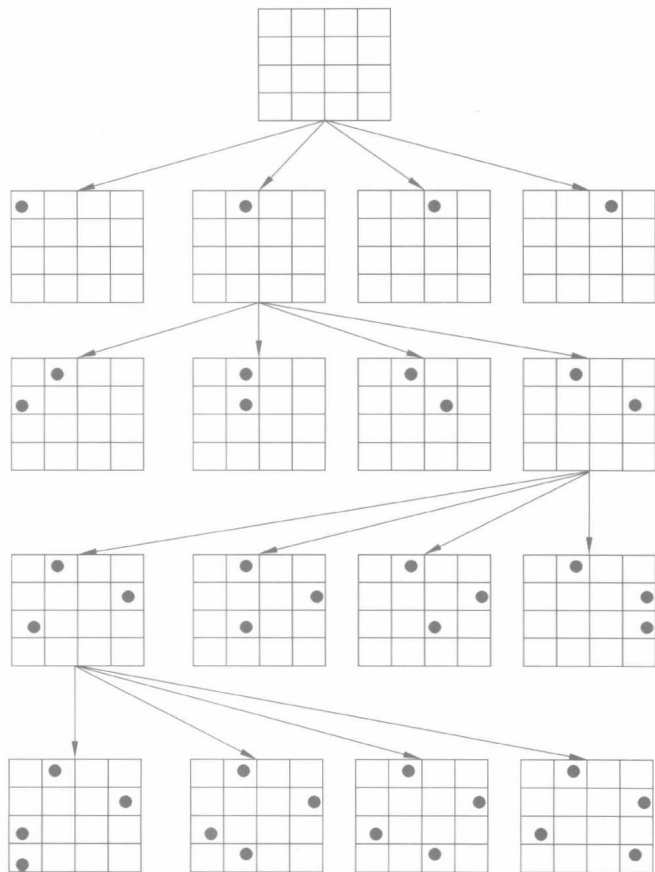


图 1.2 四皇后问题中隐含的状态树

【例 1.3】 教学计划编排问题。一个教学计划包含许多课程,在教学计划包含的许多课程之间,有些必须按规定的先后次序进行,有些则没有次序要求。即有些课程之间有先修和后续的关系,有些课程可以任意安排次序。这里各个课程之间的次序关系可用一个称作图的数据结构来表示,如图 1.3 所示。有向图中的每个顶点表示一门课程,如果从顶点 C_i 到 C_j 之间存在有向边 $\langle C_i, C_j \rangle$,则表示课程 i 必须先于课程 j 进行。

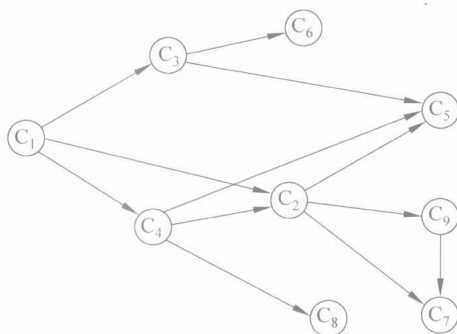
由以上三个例子可见,描述这类非数值计算问题的数学模型不再是数学方程,而是诸如表、树、图之类的数据结构。因此,可以说数据结构课程主要是研究非数值计算的程序设计问题中所出现的计算机操作对象以及它们之间的关系和操作的学科。

学习数据结构的目的是为了了解计算机处理对象的特性,将实际问题所涉及的处理

对象在计算机中表示出来并对它们进行处理。与此同时,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。

课程编号	课程名称	先修课程
C ₁	计算机导论	无
C ₂	数据结构	C ₁ , C ₄
C ₃	汇编语言	C ₁
C ₄	C程序设计	C ₁
C ₅	计算机图形学	C ₂ , C ₃ , C ₄
C ₆	接口技术	C ₃
C ₇	数据库原理	C ₂ , C ₉
C ₈	编译原理	C ₄
C ₉	操作系统	C ₂

(a) 计算机专业的课程设置



(b) 表示课程之间优先关系的有向图

图 1.3 教学计划编排问题的数据结构

1.1.2 有关概念和术语

在系统地学习数据结构知识之前,先对一些基本概念和术语赋予确切的含义。

数据(Data)是信息的载体,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,通过程序的运行来处理各种各样的数据。计算机科学中,所谓的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、科学计算和商务处理;非数值数据包括字符、文字、图形、图像、语音等。

数据元素(Data Element)是数据的基本单位。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、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。

有时一个数据元素可由若干个数据项(Data Item)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。通常在解决实际应用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)或数据元素类(Data Element Class)是具有相同性质的数据元素的集合。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(数据元素类),数据元素是数据元素类的一个实例。例如,在交通咨询系统的交通网中,所有的顶点是一个数据元素类,顶点 A 和顶点 B 各自代表一个城市,是该数据元素类中的两个实例,其数据元素的值分别为 A 和 B。

数据结构(Data Structure)是指相互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这样或那样的关系,这种数据元素之间的关系称为结构。根据数据元素之间的关系的不同特性,通常有下列 4 类基本的结构。

1. 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。

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. 树型结构

该结构的数据元素之间存在着一对多的关系。

4. 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 1.4 为表示上述 4 类基本结构的示意图。

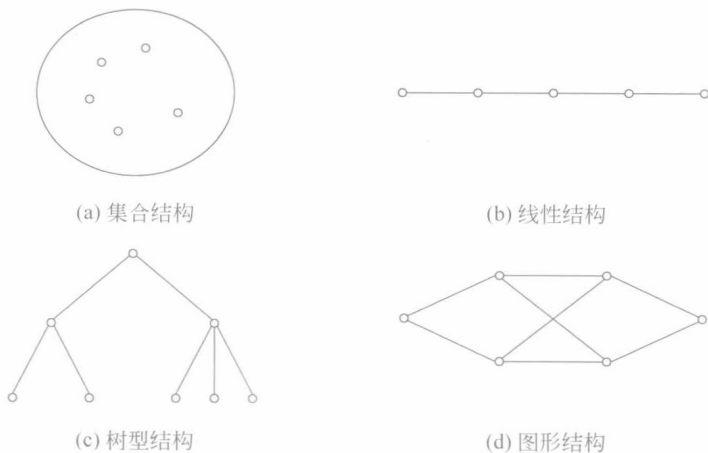


图 1.4 4 类基本结构的示意图

由于集合的数据元素之间关系极为松散,因此也可用其他结构来表示它。

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。

数据结构的定义形式为:

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

其中, D 是数据元素的有限集合; R 是 D 上关系的有限集。

数据结构包括数据的逻辑结构和数据的物理结构。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,它与数据的存储无关。研究数据结构的目的是为了在计算机中实现对它的操作,为此还需要研究如何在计算机中表示一个数据结构。数据结构在计算机中的表示(又称映像)称为数据的物理结构,或称存储结构。它所研究的是数据结构在计算机中的实现方法,包括数据结构的表示及元素间关系的表示。

数据的存储结构可采用顺序存储或链式存储的方法。

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。

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相邻的指针字段来表示,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。

除了通常采用的顺序存储方法和链式存储方法外,有时为了查找方便还采用索引存储方法和散列存储方法。

1.1.3 数据结构课程的内容

数据结构与数学、计算机硬件和软件有十分密切的关系。数据结构是介于数学、计算机硬件和计算机软件之间的一门计算机相关专业的核心课程,是高级程序设计语言、编译原理、操作系统、数据库、人工智能等课程的基础。同时,数据结构技术也广泛应用于信息科学、系统工程、应用数学以及各种工程技术领域。

数据结构课程集中讨论软件开发过程中的设计阶段、设计编码和分析阶段的若干基本问题。此外,为了构造出好的数据结构及其实现,还需考虑数据结构及其实现的评价与选择。因此,数据结构的内容包括三个层次的 5 个“要素”,如表 1.1 所示。

表 1.1 数据结构课程内容体系

方面 层次	数据表示	数据处理
抽象	逻辑结构	基本运算
实现	存储结构	算法
评价	不同数据结构的比较及算法	