

21世纪

21世纪高职高专创新精品规划教材

数据结构(用C语言描述)

主 编 王宇川 郭建东
副主编 倪华锦 吴 嵘 罗捷斯
主 审 张 念

创新
精品

“教、学、做”一体化，强化能力培养
“工学结合”原则，提高社会实践能力
“案例教学”方法，增强可读性和可操作性



中国水利水电出版社
www.waterpub.com.cn

内 容 提 要

本书作为高职高专规划教材和校企合作编写教材项目,依据教育部关于“高职高专计算机教育教学基本要求”,结合高职高专的教学特点,全书各章均按照“学习要点→案例引入→提出问题→理论讲解→案例分析”的总体思路编写而成。全书共8章,内容包括:绪论、线性表、栈和队列、其他线性数据结构(串、数组)、树和二叉树、图、查找以及排序。

本书作者从实用的角度出发,以提高学生的算法分析能力和程序设计技能为目标,在多年教学经验的基础上,根据学生的认知规律,对数据结构内容认真提炼,精心组织,并通过大量有现实意义的例题,循序渐进地介绍了数据结构的有关概念、各种基本的数据结构和算法。书中例题都经过了仔细的调试,配有一定数量的上机实训题和课后习题。

本书概念清晰、内容丰富、深入浅出、知识结构及深度合理,可作为高等职业和高等专科学校的教材,也可作为计算机培训班的教材及自学者的参考书。

**本书配有免费电子教案可以从中国水利水电出版社网站下载,网址为:
<http://www.waterpub.com.cn/softdown/>。**

图书在版编目(CIP)数据

数据结构:用C语言描述/王宇川,郭建东主编. —北

京:中国水利水电出版社,2008

21世纪高职高专创新精品规划教材

ISBN 978-7-5084-5542-6

I. 数… II. ①王…②郭… III. ①数据结构—高等学校:技术学校—教材②C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059589号

书 名	数据结构(用C语言描述)
作 者	主 编 王宇川 郭建东 副主编 倪华锦 吴 嵘 罗捷斯 主 审 张 念
出版 发行	中国水利水电出版社(北京市三里河路6号 100044) 网址: www.waterpub.com.cn E-mail: mchannel@263.net (万水) sales@waterpub.com.cn 电话:(010)63202266(总机)、68331835(营销中心)、82562819(万水)
经 售	全国各地新华书店和相关出版物销售网点
排 版	北京万水电子信息有限公司
印 刷	北京蓝空印刷厂
规 格	787mm×1092mm 16开本 15印张 357千字
版 次	2008年6月第1版 2008年6月第1次印刷
印 数	0001—4000册
定 价	26.00元

凡购买我社图书,如有缺页、倒页、脱页的,本社营销中心负责调换

版权所有·侵权必究

本书编委会

主 编： 王宇川 郭建东

副主编： 倪华锦 吴 嵘 罗捷斯

主 审： 张 念

编 委：（排名不分先后）

何曙辉	何受倩	倪华锦
王文莉	罗捷斯	田 夏
吴 嵘	郭建东	王宇川

序

近年来，我国高等职业教育蓬勃发展，为现代化建设培养了大量高素质技能型专门人才，对高等教育大众化作出了重要贡献，顺应了人民群众接受高等教育的强烈需求。高等职业教育作为高等教育发展中的一个类型，肩负着培养面向生产、建设、服务和管理第一线需要的高技能人才的使命，在我国加快推进社会主义现代化建设进程中具有不可替代的作用。随着我国走新型工业化道路、建设社会主义新农村和创新型国家对高技能人才要求的不断提高，高等职业教育既面临着极好的发展机遇，也面临着严峻的挑战。

教材建设是整个高职高专院校教育教学工作的重要组成部分，高质量的教材是培养高质量人才的基本保证，高职高专教材作为体现高职高专教育特色的知识载体和教学的基本工具，直接关系到高职高专教育能否为一线岗位培养符合要求的高技术性人才。中国水利水电出版社本着为高校教育服务，为师生提供高品质教材的原则，按照教育部《关于全面提高高等职业教育教学质量的若干意见》的要求，在全国数百所高职高专院校中遴选了一批具有丰富的教学经验、较高的工程实践能力的学科带头人和骨干教师，成立了高职高专教材建设编委会。编委会成员经过几个月的广泛调研，了解各高职院校教学改革和企业对人才需求的情况，探讨、研究课程体系建设和课程设置，达成共识，组织编写了本套“21世纪高职高专创新精品规划教材”。

本套教材的特点如下：

1. 面向高职高专教育，将专业培养目标分解落实于各门课程的技术应用能力要求，建立课程的技术、技能体系，将理论知识贯穿于其中，并融“教、学、做”为一体，强化学生的能力培养。

2. 理论知识的讲解以基础知识和基本理论“必需、够用”为原则，在保证达到高等教育水平的基础上，注重基本概念和基本方法讲解的科学性、准确性和正确性，把重点放在概念、方法和结论的阐释和实际应用上，推导过程力求简洁明了。

3. 在教材中按照技术、技能要求的难易和熟练程度，选择恰当的训练形式和内容，形成训练体系；确定实训项目，并将实训内容体现在教材中。对于单独设置实训的课程，我们将实训分成基础实训和综合实训两个部分。综合实训中重点体现了工学结合的原则，提高学生的社会实践能力。

4. 在编写方式上引入案例教学和启发式教学方法，采用以实际应用引出的问题为背景来设计和组织内容，增强了教材的可读性和可操作性，激发学生的学习兴趣，使知识点更容易理解掌握，从而使学生能够真正地掌握相关技术，为以后的就业打好基础。

5. 教材内容力求体现经济社会发展对应用技术的新要求和新趋势，将新兴的高新技术、复合技术等引进教材，并在教材中提出了一些引导技术发展的新问题，以期引起思考和讨论，有利于培养学生技术应用中的创新精神和能力。

6. 大部分教材都配有电子教案和相关教学资源，以使教材向多元化、多媒体化发展，满足广大教师教学工作的需要。电子教案使用 PowerPoint 制作，教师可根据授课情况任意修改。相关教案和资源可以从中国水利水电出版社网站 www.waterpub.com.cn 下载。

本套教材凝聚了众多奋斗在高等职业教育教学、科研第一线的教师和科研人员多年的教学经验和智慧，教材内容选取新颖、实用，层次清晰，结构合理，概念清晰，通俗易懂，可读性和实用性强。本套教材适用于高职高专院校，也可作为社会各类培训班用书和自学参考用书。

我们期待广大读者对本套教材提出宝贵意见和建议，以便进一步修订，使该套教材不断完善。

21世纪高职高专创新精品规划教材编委会
2008年4月

前 言

数据结构是计算机各专业的—门重要的核心专业基础课程。当用计算机来解决实际问题时,就要涉及到数据(Data)的表示及数据的处理,而数据表示与处理正是数据结构课程的主要研究对象,通过这两方面内容的学习,为后续课程,特别是软件方面的课程打下坚实的基础,同时也提供必要的技能训练。

本书在编写过程中保持了与本系列丛书其他书籍在知识结构上的系统性和编写风格上的一致性,进一步增强了本书的可读性。

本书为高职高专规划教材,依据教育部关于“高职高专计算机教育教学基本要求”,结合高职高专的教学特点,全书各章均按照“学习要点→案例引入→提出问题→理论讲解→案例分析”的总体思路编写而成,力求体现以应用为主体,着重强调知识的理解和运用,实现高职高专教学以应用能力为主要培养目标,做到内容丰富、深入浅出、循序渐进,力求达到先进性、科学性、可读性、通俗性、实用性的有机结合。

为帮助读者学习掌握数据结构知识,每种数据结构的引入不是从抽象的定义出发,而是从实例出发,通过对实例的分析、理解,总结出数据结构的特点及应用,使读者能迅速掌握数据结构的有关概念、存储结构、基本操作和有关算法实现。书中的例题都经过仔细的调试。每章都配有课后习题和实训题,供读者课外巩固所学的内容。

本书共8章,内容包括:绪论、线性表、栈和队列、其他线性数据结构(串、数组)、树和二叉树、图、查找以及排序。

本书第1章、第2章由王宇川、倪华锦编写,第3章、第4章由郭建东、吴嵘编写,第5章由王宇川、何曙辉编写,第6章由郭建东、何受倩编写,第7章由郭建东、罗捷斯编写,第8章由王宇川、田夏编写,全书由王宇川负责,罗捷斯、何曙辉协助统稿;由郭建东、王文莉、吴嵘等编辑校对,全书由张念主审。

本书是作者有关学校与广州黑马软件科技有限公司“校企合作编写教材项目”,广州软件行业协会会长、广州黑马软件科技有限公司董事长兼总经理罗捷斯亲自参加了本书的编写。

本书在编写过程中,得到了广东工贸职业技术学院、广东科贸职业技术学院、广州金桥管理干部学院、广州航海高等专科学校、广东轻工职业技术学院、广州黑马软件科技有限公司以及中国人民解放军后勤部一六一医院等单位的大力支持。

本书适合于高职高专院校计算机各专业学生、成人继续教育和自学人员使用,*号部分为选修内容。

由于时间仓促且作者水平有限,书中难免存在不妥和错误之处,恳请广大读者提出宝贵意见。

编 者

2008年5月

目 录

序

前言

第 1 章 绪论	1
1.1 什么是数据结构	1
1.1.1 概述	1
1.1.2 案例分析	2
1.2 基本概念和术语	4
1.3 数据结构课程的内容	5
1.3.1 数据类型	6
1.3.2 抽象数据类型	7
1.4 算法和算法分析	7
1.5 算法性能分析与度量	8
习题一	10
第 2 章 线性表	12
2.1 线性表的定义及逻辑结构	12
2.2 线性表的基本操作	13
2.3 线性表的顺序存储结构	14
2.3.1 顺序表	14
2.3.2 顺序表上基本运算的实现	15
2.3.3 案例分析	19
2.4 线性表的链式存储结构	23
2.4.1 单链表	24
2.4.2 单链表上的基本运算	25
2.4.3 循环链表	31
*2.4.4 双向链表	32
*2.4.5 静态链表	33
2.4.6 案例分析	34
2.5 顺序表和链表的比较	39
习题二	40
第 3 章 栈和队列	43
3.1 栈	43
3.1.1 栈的定义及基本运算	43
3.1.2 顺序栈	44
3.1.3 链栈	46

3.1.4	案例分析	47
3.2	队列	53
3.2.1	队列的定义及基本运算	53
3.2.2	队列的顺序存储结构	54
3.2.3	队列的链式存储	57
3.2.4	案例分析	59
习题三	63
第4章	其他线性数据结构	67
4.1	串	67
4.1.1	串的定义及基本操作	68
4.1.2	串的定长顺序存储结构及基本运算.....	69
*4.1.3	串的堆存储结构	71
4.1.4	案例分析	73
4.2	多维数组	75
4.2.1	数组的定义及基本操作	75
4.2.2	数组的内存映像	75
4.2.3	案例分析	77
4.3	矩阵的压缩存储	78
4.3.1	稀疏矩阵的压缩存储	78
4.3.2	案例分析	81
4.3.3	特殊矩阵的压缩存储	84
习题四	87
第5章	树和二叉树	90
5.1	树的定义和基本操作	90
5.1.1	树的定义	90
5.1.2	基本术语	92
5.1.3	树的基本操作	92
5.2	二叉树	92
5.2.1	二叉树的定义和基本操作.....	92
5.2.2	二叉树的性质	93
5.2.3	二叉树的存储	95
5.2.4	二叉树的基本操作及实现.....	98
5.2.5	二叉树的遍历方法及递归实现.....	100
5.2.6	二叉树遍历的非递归实现.....	103
5.2.7	由遍历序列恢复二叉树	106
5.3	树和森林	108
5.3.1	树的表示	108
5.3.2	树的存储结构	109
5.3.3	树、森林与二叉树的转换.....	113

5.3.4	树和森林的遍历	115
5.4	哈夫曼树和判定树	117
5.4.1	哈夫曼树的定义与构造方法	117
5.4.2	哈夫曼编码	118
5.4.3	判定树	119
5.5	案例分析	120
	习题五	123
第6章	图	127
6.1	图的定义和术语	127
6.2	图的存储结构	131
6.2.1	邻接矩阵	131
6.2.2	邻接表表示法	133
6.3	图的遍历	136
6.3.1	深度优先搜索	136
6.3.2	广度优先搜索	139
6.4	图的应用	141
6.4.1	图的连通性	141
*6.4.2	生成树和生成森林	143
*6.4.3	关节点和重连通分量	145
6.4.4	最小生成树	147
6.4.5	最短路径	153
*6.4.6	AOV网与拓扑排序	157
6.5	案例分析	161
	习题六	166
第7章	查找	168
7.1	基本概念与术语	168
7.2	静态查找表	171
7.2.1	静态查找表结构	171
7.2.2	顺序查找	171
7.2.3	有序表的折半查找	172
7.2.4	斐波那契查找	174
7.2.5	分块查找	175
7.2.6	案例分析	176
7.3	动态查找表	179
7.3.1	二叉排序树	179
7.3.2	案例分析	182
7.4	哈希表查找(杂凑法)	186
7.4.1	哈希表与哈希方法	186
7.4.2	常用的哈希函数	187

7.4.3	处理冲突的方法	189
7.4.4	哈希表的查找分析	191
习题七	192
第 8 章	排序	195
8.1	基本概念	195
8.2	插入排序	196
8.2.1	直接插入排序	196
8.2.2	希尔排序	196
8.2.3	案例分析	197
8.3	交换排序	200
8.3.1	冒泡排序	200
8.3.2	快速排序	202
8.3.3	案例分析	202
8.4	选择排序	205
8.4.1	简单选择排序	205
*8.4.2	堆排序	206
8.5	二路归并排序	208
8.6	基数排序	210
8.6.1	多关键码排序	210
8.6.2	链式基数排序	211
8.7	外排序	214
8.7.1	外部排序的方法	214
*8.7.2	多路平衡归并的实现	216
习题八	218
上机实训练习	221
第 2 章	上机实训题	221
第 3 章	上机实训题	222
第 4 章	上机实训题	223
第 5 章	上机实训题	224
第 6 章	上机实训题	225
第 7 章	上机实训题	226
第 8 章	上机实训题	226

第 1 章 绪论

学习要点

- 什么是数据结构
- 数据结构的基本概念
- 数据结构的基本术语
- 数据类型和抽象的数据类型
- 算法的基本概念和算法的分析
- 算法的特性
- 算法时间复杂度的分析计算

引言

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。无论是进行科学计算或数据处理、过程控制以及对文件的存储和检索及数据库技术等计算机应用领域中，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。

案例：如何通过计算机查找某个学生的有关情况或者查询某个专业或年级的学生的有关情况？如何通过计算机编排教学计划？

问题：线性关系的数学模型和图状关系的数学模型有什么区别？针对这两种数学模型，可采用哪种数据结构来描述？

1.1 什么是数据结构

1.1.1 概述

数据结构 (Data Structure) 是计算机及相关专业的技术基础课，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。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。

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。使用计算机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学模型的线性方程组,该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是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。这类问题涉及到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。以下是这类问题的实例。

1.1.2 案例分析

案例1 学生信息检索系统。

问题:

通过计算机查找某个学生的有关情况或者想查询某个专业或年级的学生的有关情况。

算法分析:

只要我们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。由这四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)对学生信息文件进行查询。

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着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。

案例2 教学计划编排问题

问题:

如何通过计算机编排教学计划?

算法分析:

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。这种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1.2所示。有向图中的每个顶点表示一门课程,如果从顶点 v_i 到 v_j 之间存在有向边 $\langle v_i, v_j \rangle$,则表示课程 i 必须先于课程 j 进行。

由以上两个例子可见,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。



学号	姓名	性别	专业	年级
980001	吴承志	男	计算机科学与技术	98级
980002	李淑芳	女	信息与计算科学	98级
990301	刘丽	女	数学与应用数学	99级
990302	张会友	男	信息与计算科学	99级
990303	石宝国	男	计算机科学与技术	99级
000801	何文颖	女	计算机科学与技术	2000级
000802	赵胜利	男	数学与应用数学	2000级
000803	崔文靖	男	信息与计算科学	2000级
010601	刘丽	女	计算机科学与技术	2001级
010602	魏永鸣	男	数学与应用数学	2001级

(a) 学生信息表

崔文靖	8
何文颖	6
李淑芳	2
刘丽	3, 9
石宝国	5
魏永鸣	10
吴承志	1
赵胜利	7
张会有	4

(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

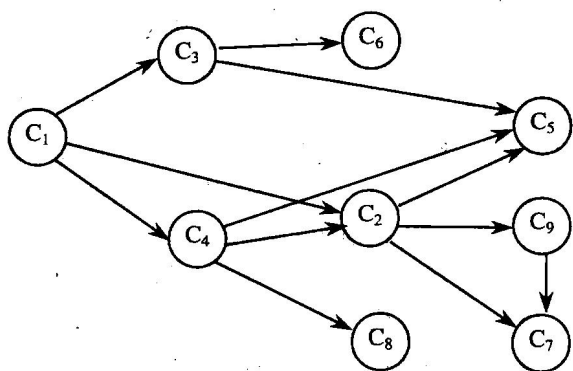
2000级	6, 7, 8
2001级	9, 10
98级	1, 2, 3
99级	4, 5

(d) 年级索引表

图 1.1 学生信息查询系统中的数据结构

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 程序设计语言	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 1.2 教学计划编排问题的数据结构

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。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。

1.2 基本概念和术语

在系统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): 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,应用程序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、科学计算和商务处理等;非数值数据包括字符、文字、图形、图像、语音等。

数据元素 (Data Element): 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) 或数据元素类 (Data Element Class): 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。根据数据元素间关系的不同特性,通常有下列四类基本的结构:

- 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。
- 树型结构。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 1.3 为表示上述四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: 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。

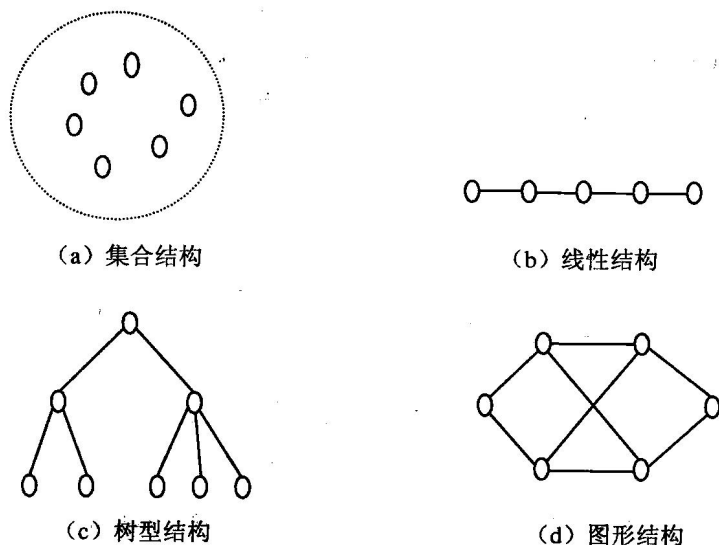


图 1.3 四类基本结构的示意图

数据结构包括数据的逻辑结构 (Logical Structure) 和数据的物理结构 (Physical Structure)。数据的逻辑结构可以看成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标识 (又称映像) 称为数据的物理结构或者存储结构。它所研究的是数据结构在计算机中的实现方法, 包括数据结构中元素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 (Sequential Storage) 或链式存储 (Linked Storage) 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。

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

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及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。

软件工程过程主要包括开发过程、运作过程、维护过程。而数据结构课程主要涉及软件

开发过程中的问题分析、需求定义、各模块功能规约以及数据结构描述(伪代码:一种算法描述语言)等若干基本问题。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,数据结构课程体系的内容包括3个层次的5个“要素”,如图1.4所示。

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分析	

图 1.4 数据结构课程体系的内容

数据结构的核⼼技术是分解与抽象。通过分解可以划分出数据的3个层次;再通过抽象,舍弃数据元素的具体内容,就得到逻辑结构。类似地,通过分解将处理要求划分成各种功能,再通过抽象舍弃实现细节,就得到运算的定义。上述两个方面的结合使我们将问题变换为数据结构。这是一个从具体(即具体问题)到抽象(即数据结构)的过程。然后,通过增加对实现细节的考虑进一步得到存储结构和运算实现,从而完成设计任务。这是一个从抽象(即数据结构)到具体(即具体实现)的过程。熟练掌握这两个过程是数据结构课程在专业技能培养方面的基本目标。

数据结构作为一门独立的课程在国外是从1968年才开始的,但在此之前其有关内容已散见于编译原理及操作系统之中。20世纪60年代中期,美国的一些大学开始设立有关课程,但当时的课程名称并不叫数据结构。1968年美国唐·欧·克努特教授开创了数据结构的最初体系,他所著的《计算机程序设计技巧》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。从20世纪60年代末到70年代初,出现了大型程序,软件也相对独立,结构程序设计成为程序设计方法学的主要内容,人们越来越重视数据结构。从20世纪70年代中期到80年代,各种版本的数据结构著作相继出现。目前,数据结构的发展并未终结,一方面,面向各专门领域中特殊问题的数据结构得到研究和发展,如多维图形数据结构等;另一方面,从抽象数据类型和面向对象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。

首先,本文介绍程序设计语言中出现的各种数据类型。

1.3.1 数据类型

数据类型是和数据结构密切相关的⼀个概念。它最早出现在高级程序设计语言中,用以刻画程序中操作对象的特性。在用高级语言编写的程序中,每个变量、常量或表达式属于一个确定的数据类型。数据类型显式地或隐含地规定了在程序执行期间变量或表达式所有可能的取值范围,以及在这些值上允许进行的操作。因此,数据类型(Data Type)是一个值的集合和定义在这个值集上的一组操作的总称。

在高级程序设计语言中,数据类型可分为两类:一类是原子类型,另一类则是结构类型。

原子类型的值是不可分解的。如 C 语言中整型、字符型、浮点型、双精度型等基本类型，分别用保留字 int、char、float、double 标识。而结构类型的值是由若干成分按某种结构组成的，因此是可分解的，并且它的成分可以是非结构的，也可以是结构的。例如，数组的值由若干分量组成，每个分量可以是整数，也可以是数组等。在某种意义上，数据结构可以看成是“一组具有相同结构的值”，而数据类型则可被看成是由一种数据结构和定义在其上的一组操作所组成的。

1.3.2 抽象数据类型

抽象数据类型 (Abstract Data Type, ADT) 是指一个数学模型以及定义在该模型上的一组操作。抽象数据类型的定义取决于它的一组逻辑特性，而与其在计算机内部如何表示和实现无关。即不论其内部结构如何变化，只要它的数学特性不变，都不影响其外部的使用。

抽象数据类型和数据类型实质上是一个概念。例如，各种计算机都拥有的整数类型就是一个抽象数据类型，尽管它们在不同处理器上的实现方法可以不同，但由于其定义的数学特性相同，在用户看来都是相同的。因此，“抽象”的意义在于数据类型的数学抽象特性。

但在另一方面，抽象数据类型的范畴更广，它不再局限于前述各处理器中已定义并实现的数据类型，还包括用户在设计软件系统时自己定义的数据类型。为了提高软件的重用性，在近代程序设计方法学中，要求在构成软件系统的每个相对独立的模块上，定义一组数据和施于这些数据上的一组操作，并在模块的内部给出这些数据的表示及其操作的细节，而在模块的外部使用的只是抽象的数据及抽象的操作。这也就是面向对象的程序设计方法。

抽象数据类型的定义可以由一种数据结构和定义在其上的一组操作组成，而数据结构又包括数据元素及元素间的关系，因此抽象数据类型一般可以由元素、关系及操作 3 个要素来定义。

抽象数据类型的特征是使用与实现相分离，实行封装和信息隐蔽。就是说，在抽象数据类型设计时，把类型的定义与其实现分离开来。

1.4 算法和算法分析

算法与数据结构的关系紧密，在算法设计时先要确定相应的数据结构，而在讨论某一种数据结构时也必然会涉及相应的算法。下面就从算法特性、算法描述、算法性能分析与度量等 3 个方面对算法进行介绍。

1. 算法特性

算法 (Algorithm) 是对特定问题求解步骤的一种描述，是指令的有限序列。其中每一条指令表示一个或多个操作。一个算法应该具有下列特性：

- 有穷性。一个算法必须在有穷步之后结束，即必须在有限时间内完成。
- 确定性。算法的每一步必须有确切的定义，无二义性。算法的执行对应着的相同的输