21世纪全国应用型本科计算机案例型规划教材

21st CENTURY
实用规划教材

赠送 电子课件 www.pup6.cn

赠送 素材 www.pup6.cn

数据结构与算法应用实践教程

主　编　李文书

Data structure

- 包含100多种算法并均提供相应代码
- 采用实践案例讲解复杂的理论知识
- 充分运用图形语言来体现抽象内容

北京大学出版社
PEKING UNIVERSITY PRESS

21 世纪全国应用型本科计算机案例型规划教材

数据结构与算法应用实践教程

主　编　李文书

北京大学出版社
PEKING UNIVERSITY PRESS

内 容 简 介

本书和传统同类书籍的区别是除了介绍基本的数据结构知识，如线性表、栈、队列、链表、树、二叉树、AVL 树、红黑树、排序和查找之外，还引进了一些 C 语言中的内存分配、结构数组和结构指针的有关概念及常见问题分析；另外，还介绍了相应知识点的应用实践。总的来说，本书选取的内容均侧重于在实际中有广泛应用的数据结构及算法，有很好的实用价值。本书介绍的所有数据结构及算法都以不同复杂程度给出其编码实现。为了便于读者自学，每章末附有小结及习题与思考。

本书可作为高等院校计算机相关专业的教材，也适合学过一门编程语言的各类读者，包括在读的大中专计算机专业学生、想转行做开发的非专业人员、欲考计算机研究生的应届或在职人员，以及工作后需要补学或温习数据结构及算法的程序员等参考使用。

图书在版编目(CIP)数据

数据结构与算法应用实践教程/李文书主编. —北京：北京大学出版社，2012.2

(21 世纪全国应用型本科计算机案例型规划教材)

ISBN 978-7-301-20052-0

Ⅰ. ①数… Ⅱ. ①李… 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(2012)第 005200 号

书　　名：数据结构与算法应用实践教程

著作责任者：李文书　主编

策 划 编 辑：郑　双

责 任 编 辑：郑　双

标 准 书 号：ISBN 978-7-301-20052-0/TP・1211

出　版　者：北京大学出版社

地　　址：北京市海淀区成府路 205 号　100871

网　　址：http://www.pup.cn　http://www.pup6.cn

电　　话：邮购部 62752015　发行部 62750672　编辑部 62750667　出版部 62754962

电 子 邮 箱：pup_6@163.com

印　刷　者：河北滦县鑫华书刊印刷厂

发　行　者：北京大学出版社

经　销　者：新华书店

787 毫米×1092 毫米　16 开本　19 印张　435 千字

2012 年 2 月第 1 版　2012 年 2 月第 1 次印刷

定　　价：36.00 元

21世纪全国应用型本科计算机案例型规划教材

专家编审委员会

(按姓名拼音顺序)

主　任	刘瑞挺			
副主任	陈　钟	蒋宗礼		
委　员	陈代武	房爱莲	胡巧多	黄贤英
	江　红	李　建	娄国焕	马秀峰
	祁亨年	王联国	汪新民	谢安俊
	解　凯	徐　苏	徐亚平	宣兆成
	姚喜妍	于永彦	张荣梅	

信息技术的案例型教材建设

(代丛书序)

刘瑞挺

北京大学出版社第六事业部在 2005 年组织编写了《21 世纪全国应用型本科计算机系列实用规划教材》，至今已出版了 50 多种。这些教材出版后，在全国高校引起热烈反响，可谓初战告捷。这使北京大学出版社的计算机教材市场规模迅速扩大，编辑队伍茁壮成长，经济效益明显增强，与各类高校师生的关系更加密切。

2008 年 1 月北京大学出版社第六事业部在北京召开了“21 世纪全国应用型本科计算机案例型教材建设和教学研讨会”。这次会议为编写案例型教材做了深入的探讨和具体的部署，制定了详细的编写目的、丛书特色、内容要求和风格规范。在内容上强调面向应用、能力驱动、精选案例、严把质量；在风格上力求文字精练、脉络清晰、图表明快、版式新颖。这次会议吹响了提高教材质量第二战役的进军号。

案例型教材真能提高教学的质量吗？

是的。著名法国哲学家、数学家勒内·笛卡儿(Rene Descartes，1596—1650)说得好：“由一个例子的考察，我们可以抽出一条规律。(From the consideration of an example we can form a rule.)”事实上，他发明的直角坐标系，正是通过生活实例而得到的灵感。据说是在 1619 年夏天，笛卡儿因病住进医院。中午他躺在病床上，苦苦思索一个数学问题时，忽然看到天花板上有一只苍蝇飞来飞去。当时天花板是用木条做成正方形的格子。笛卡儿发现，要说出这只苍蝇在天花板上的位置，只需说出苍蝇在天花板上的第几行和第几列。当苍蝇落在第四行、第五列的那个正方形时，可以用(4，5)来表示这个位置……由此他联想到可用类似的办法来描述一个点在平面上的位置。他高兴地跳下床，喊着“我找到了，找到了”，然而不小心把国际象棋撒了一地。当他的目光落到棋盘上时，又兴奋地一拍大腿：“对，对，就是这个图”。笛卡儿锲而不舍的毅力，苦思冥想的钻研，使他开创了解析几何的新纪元。千百年来，代数与几何，井水不犯河水。17 世纪后，数学突飞猛进的发展，在很大程度上归功于笛卡儿坐标系和解析几何学的创立。

这个故事，听起来与阿基米德在浴池洗澡而发现浮力原理，牛顿在苹果树下遇到苹果落到头上而发现万有引力定律，确有异曲同工之妙。这就证明，一个好的例子往往能激发灵感，由特殊到一般，联想出普遍的规律，即所谓的“一叶知秋”、“见微知著”的意思。

回顾计算机发明的历史，每一台机器、每一颗芯片、每一种操作系统、每一类编程语言、每一个算法、每一套软件、每一款外部设备，无不像闪光的珍珠串在一起。每个案例都闪烁着智慧的火花，是创新思想不竭的源泉。在计算机科学技术领域，这样的案例就像大海岸边的贝壳，俯拾皆是。

事实上，案例研究(Case Study)是现代科学广泛使用的一种方法。Case 包含的意义很广：包括 Example 例子，Instance 事例、示例，Actual State 实际状况，Circumstance 情况、事件、境遇，甚至 Project 项目、工程等。

我们知道在计算机的科学术语中，很多是直接来自日常生活的。例如 Computer 一词早在 1646 年就出现于古代英文字典中，但当时它的意义不是“计算机”而是“计算工人”，

即专门从事简单计算的工人。同理，Printer 当时也是“印刷工人”而不是“打印机”。正是由于这些“计算工人”和“印刷工人”常出现计算错误和印刷错误，才激发查尔斯·巴贝奇(Charles Babbage，1791—1871)设计了差分机和分析机，这是最早的专用计算机和通用计算机。这位英国剑桥大学数学教授、机械设计专家、经济学家和哲学家是国际公认的“计算机之父”。

20 世纪 40 年代，人们还用 Calculator 表示计算机器。到电子计算机出现后，才用 Computer 表示计算机。此外，硬件(Hardware)和软件(Software)来自销售人员。总线(Bus)就是公共汽车或大巴，故障和排除故障源自格瑞斯·霍普(Grace Hopper，1906—1992)发现的“飞蛾子”(Bug)和“抓蛾子”或“抓虫子”(Debug)。其他如鼠标、菜单……不胜枚举。至于哲学家进餐问题，理发师睡觉问题更是操作系统文化中脍炙人口的经典。

以计算机为核心的信息技术，从一开始就与应用紧密结合。例如，ENIAC 用于弹道曲线的计算，ARPANET 用于资源共享以及核战争时的可靠通信。即使是非常抽象的图灵机模型，也受到二战时图灵博士破译纳粹密码工作的影响。

在信息技术中，既有许多成功的案例，也有不少失败的案例；既有先成功而后失败的案例，也有先失败而后成功的案例。好好研究它们的成功经验和失败教训，对于编写案例型教材有重要的意义。

我国正在实现中华民族的伟大复兴，教育是民族振兴的基石。改革开放以来，我国高等教育在数量上、规模上已有相当的发展。当前的重要任务是提高培养人才的质量，必须从学科知识的灌输转变为素质与能力的培养。应当指出，大学课堂在高新技术的武装下，利用 PPT 进行的“高速灌输”、“翻页宣科”有愈演愈烈的趋势，我们不能容忍用“技术”绑架教学，而是让教学工作乘信息技术的东风自由地飞翔。

本系列教材的编写，以学生就业所需的专业知识和操作技能为着眼点，在适度的基础知识与理论体系覆盖下，突出应用型、技能型教学的实用性和可操作性，强化案例教学。本套教材将会有机融入大量最新的示例、实例以及操作性较强的案例，力求提高教材的趣味性和实用性，打破传统教材自身知识框架的封闭性，强化实际操作的训练，使本系列教材做到“教师易教，学生乐学，技能实用”。有了广阔的应用背景，再造计算机案例型教材就有了基础。

我相信北京大学出版社在全国各地高校教师的积极支持下，精心设计，严格把关，一定能够建设出一批符合计算机应用型人才培养模式的、以案例型为创新点和兴奋点的精品教材，并且通过一体化设计、实现多种媒体有机结合的立体化教材，为各门计算机课程配齐电子教案、学习指导、习题解答、课程设计等辅导资料。让我们用锲而不舍的毅力，勤奋好学的钻研，向着共同的目标努力吧！

刘瑞挺教授　本系列教材编写指导委员会主任、全国高等院校计算机基础教育研究会副会长、中国计算机学会普及工作委员会顾问、教育部考试中心全国计算机应用技术证书考试委员会副主任、全国计算机等级考试顾问。曾任教育部理科计算机科学教学指导委员会委员、中国计算机学会教育培训委员会副主任。PC Magazine《个人电脑》总编辑、CHIP《新电脑》总顾问、清华大学《计算机教育》总策划。

前　言

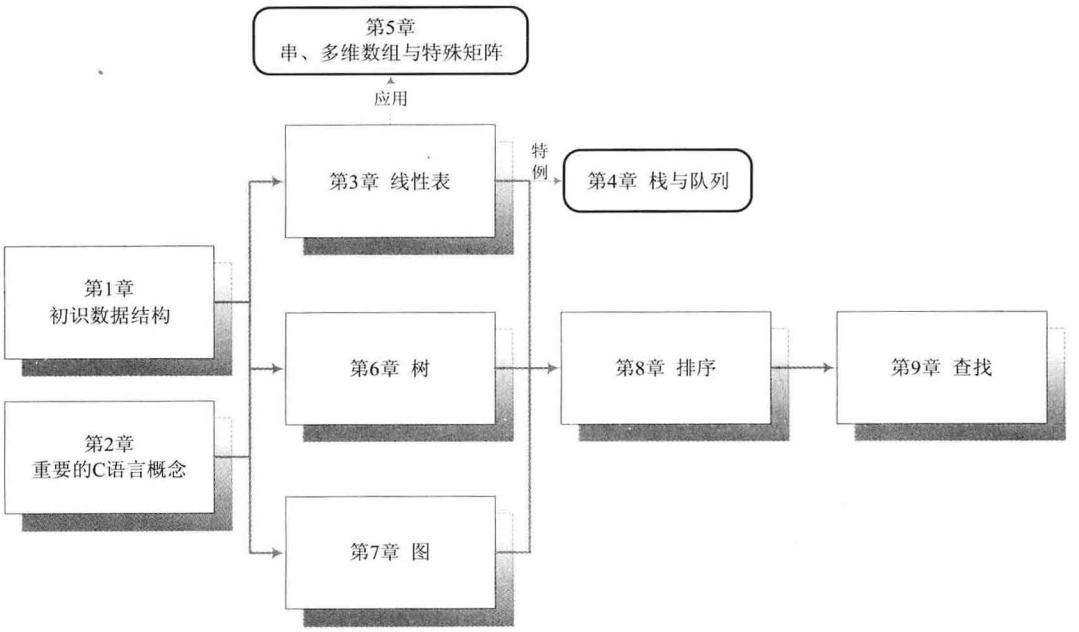
数据结构与算法是计算机科学和相关专业的核心课程，是软件设计的理论基础，是主要研究在非数值计算的程序设计问题中计算机的操作对象(数据元素)以及它们之间的关系和运算，并对算法运行的时间进行分析的学科。本课程的学习将为后续的操作系统、编译原理、软件工程、数据库概论等专业基础课和专业课程的学习，以及软件设计水平的提高打下良好的基础。

作为一门理论与实践并重的课程，不仅要掌握数据结构的基础知识，还要掌握编程的基本技巧。随着海量数据的增加，对于能够处理这些数据程序的需求变得日益迫切。可是，由于在数据输入量很大的时候，程序的低效率现象变得非常明显。因此，这又要求对效率问题给予更大的关注。例如，一些特定问题通过精心的实现方法是可以把对大量数据处理的时间限制从十几年减至不到 1 秒。因此，在某些情况下，对于影响算法实现运行时间的一些微小细节都需要认真的探究。

如今，数据结构和算法已经成为软件开发工程师必备的基础知识之一。社会上大多数公司在招聘软件开发人员时都会考察应聘人员数据结构与算法的熟练程度，并以此作为衡量应聘者水平的重要依据。本书可以作为高等院校计算机学科和信息类学科本、专科的教材，也可以作为其他理工专业的选修教材；对于从事计算机工程与应用工作的科技工作者也是一本实用的参考用书。

本书旨在让读者了解这门课程，掌握所含内容的内在规律，最终能够灵活运用，甚至有所发展。正如学写字先描红，不先模仿怎么创新呢？如果连现有的结论和成果都不知道来龙去脉，谈何再有新的突破呢？哲学上讲的，只有遵从人类的认知规律，人们才能更容易地认识新事物，这也是为什么越来越多的人选用国外的教科书，即使他们的英文并不好。我们知道，教科书为了达到其传授知识的目的，就必须遵从人的认知规律。从数据结构的发展来看，它是应问题的需要而出现的，并为解决问题而服务。换而言之，对于数据结构的讲解，应当把重点放在算法上，在各种典型问题上提出新的数据结构。最终得出的认识是，为了特定的问题和算法而选用特定的数据结构；为了改进算法而改进数据结构；为了新的问题和算法而创造出新的数据结构。而决不是像现在这样的认识：我学了数据结构能解决什么问题呢？乍一看，好像没什么差别，但现在这个认识就只能说明你不是数据结构的主人，只是数据结构的奴隶。本书系统介绍数据结构基础理论知识及算法设计方法，它在内容选取上符合计算机学科和信息类学科人才培养目标的要求及教学规律和认知规律，在组织编排上体现“先理论、后应用、理论与应用相结合”的原则，并兼顾学科的广度和深度，力求适用面广。本书的编写参考了国内外数据结构的最新教材和研究成果，全书共分为 9 章。第 1 章介绍数据结构的讨论范畴、基本概念、数据的逻辑结构、数据的物理结构及算法的描述与分析；第 2 章介绍一些重要的 C 语言概念，同时我们还对 C 语言常见问题进行分析。由于本书代码都使用 C 语言来实现，因此具有 C 语言基础知识对本书的学习是必不可少的；第 3 章介绍线性表各种存储结构及相关应用；第 4 章介绍栈与队列的基本

概念、各种存储结构及相关应用；第 5 章介绍字符串、多维数组和特殊矩阵的存储结构及相关应用；第 6 章介绍树、二叉树和森林的基本概念、各种存储结构及遍历、线索二叉树、二叉排序树及相关应用；第 7 章介绍图的概念、各种存储结构及遍历、最小生成树、关键路径、拓扑排序、最短路径及相关应用；第 8 章介绍 5 种基本的排序算法，插入排序、交换排序、选择排序、归并排序和基数排序及相关应用；第 9 章介绍查找的基本概念、静态查找表及动态查找表的实现算法及相关应用。为了使读者对相关知识有一个更清晰的脉络，我们给出了各章之间的依赖关系如下图所示。



每一章都精心设计有经典的应用实践问题，并且附有一定数量难度适宜的课后习题与思考，旨在引导读者不断深入地学习，学以致用，灵活处理一些实际问题，提高程序设计的能力。本书中的所有算法，均在 Visual C++下调试通过，无须任何修改就可直接上机运行、验证这些算法。与本书配套的还有《数据结构重点难点问题剖析》(C 语言版)，由浙江大学出版社出版。书中提供配套的习题和实习题，并可作为学习指导手册。

本书由李文书主编，王松、贾宇波、安利新、董建民、王根岭、吴雅萱和马国兵为副主编，在写作过程中编者得到了许多老教授的帮助和支持，提出了许多宝贵意见，在此表示衷心的感谢！

由于编者水平有限，加之时间仓促，书中难免存在不妥之处，敬请读者指正。欢迎读者与编者联系，其 E-mail 地址：wshlee@163.com。

李文书
2011 年 10 月于畅园

目　录

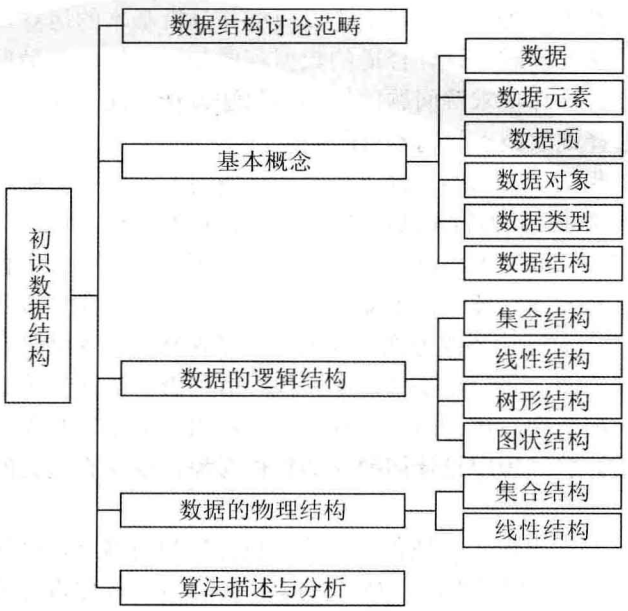
第1章

初识数据结构

学习目标

(1) 掌握数据结构的基本概念。
(2) 掌握抽象数据类型相关概念和软件构造方法。
(3) 掌握算法的含义及其时间复杂度的计算。

知识结构图



重点和难点

本章讨论的都是一些基本概念，因此没有难点，重点在于了解有关数据结构的各个名词和术语的含义，以及语句频度和时间复杂度、空间复杂度的估算。

学习指南

(1) 掌握各名词、术语的含义，以及数据的逻辑结构和存储结构之间的关系。
(2) 了解抽象数据类型的定义、表示和实现方法。
(3) 理解算法 5 个要素的确切含义。
(4) 掌握计算语句频度和估算算法时间复杂度的方法。

1.1 数据结构讨论范畴

人们利用计算机的目的是解决实际的应用问题。在明确所要解决问题的基础上，经过对问题的深入分析和抽象，为其建立一个逻辑模型并分析基本的运算，然后确定恰当的数据结构表示该模型，在此基础上设计合适的数据存储和相关算法，最后完成具体的程序来模拟和解决实际问题。计算机求解问题的核心是**算法设计**，而算法设计又高度依赖于数据结构，数据结构的选择则取决于问题本身的需求。

在现实生活中，我们更多的不是解决数值计算的问题，而是需要一些更科学有效的手段(比如表、树和图等数据结构)的帮助，才能更好地处理问题。例如，在 Web 信息处理方面，我们需要图、字符、散列表、排序、索引、检索等知识；在人工智能方面，我们需要了解广义表、集合、有向图、搜索树等知识；在数据库方面，我们需要线性表、链表、排序、B+索引树等知识；在操作系统方面，我们需要掌握队列、存储管理表、排序、目录树等知识；在编译原理方面，我们需要字符串、栈、散列表、语法树等知识；在图形图像方面，我们同样需要掌握队列、栈、图、矩阵、空间索引、检索等知识。总的来说，**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，以及它们之间的关系和操作等相关问题的学科。**

20 世纪 70 年代初，出现了大型程序，软件也开始相对独立，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”，认为程序设计的实质是对确定的问题选择一种好的结构，加上设计一种好的算法。可见，数据结构在程序设计当中占据了重要的地位。

程序=数据结构+算法。

1.2　基 本 概 念

数据(data)是利用文字符号、数字符号以及其他规定的符号对现实世界的事物及其活动所作的抽象描述。它是信息的载体，能被计算机识别、存储和加工处理。随着计算机技术的发展，数据这一概念的含义越来越广泛。不仅整数、实数、复数等是数据，字符、表格、声音、图形、图像等也都能够由计算机接收和处理，也都是数据。

表示一个事物的一组数据称为一个**数据元素**(data element)，它是数据的基本单位，在程序中作为一个整体加以考虑和处理。在数据结构中，根据需要，数据元素又被称为元素、顶点或记录。

数据项(data item)是具有独立含义的最小标识单位。在有些场合下，数据项又称为字段或域。例如，将一个学生的自然情况信息作为一个数据元素，而学生信息中的每一项(如学号、姓名、出生年月等)为一个数据项。

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。既然数据对象是数据的子集，在实际应用中，处理的数据元素通常具有相同性质，在不产生混淆的情况下，我们都将数据对象简称为数据。

数据类型(data type)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语言中，用以描述(程序)操作对象的特性。在用高级程序语言编写的程序中，每个变量、常量或表达式都有一个它所属的确定的数据类型。类型明显或隐含地规定了在程序执行期间变量或表达式所有可能取值的范围，以及在这些值上允许进行的操作。因此，**数据类型是一个值的集合和定义在这个值集上的一组操作的总称**。例如，C 语言中的整型变量，其值集为某个区间上的整数(区间大小依赖于不同的机器)，定义在其上的操作为加、减、乘、除和取模等算术运算。

在 C 语言中，按照取值的不同，数据类型可以分为以下两类：

(1) **原子类型**：是不可以再分解的基本类型，包括整型、实型、字符型等；

(2) **结构类型**：由若干个类型组合而成，是可以再分解的。例如，整型数组是由若干整型数据组成的。

抽象数据类型(abstract data type)是指一个数学模型及该模型上定义的一组操作的集合。抽象数据类型的定义仅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。

事实上，抽象数据类型体现了程序设计中问题分解、抽象和信息隐藏的特性。抽象数据类型把实际生活中的问题分解为多个规模小且容易处理的问题，然后建立一个计算机能处理的数据类型，并把每个功能模块的实现细节作为一个独立的单元，从而使具体实现过程隐藏起来。

为了便于在之后的讲解中对抽象数据类型进行规范的描述，我们给出了描述抽象数据类型的标准格式：

```
ADT 抽象数据类型名
    Data
        数据元素之间逻辑关系的定义
```

```
    Operation
        操作 1
            初始条件
            操作结果描述
        操作 2
            ……
        操作 n
            ……
endADT
```

数据结构(data structure)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。其主要研究的是数据的逻辑结构、物理结构及数据的运算。在计算机中，数据元素并不是孤立、杂乱无章的，而是具有内在联系的数据集合。数据元素之间存在的一种或多种特定关系，也就是数据的组织形式。为编写出一个“好”的程序，必须分析待处理对象的特性及各处理对象之间存在的关系。这也就是研究数据结构的意义所在。

一般来说，我们把数据结构分为逻辑结构和物理结构。

1.3 数据的逻辑结构

逻辑结构是指数据对象中数据元素之间的相互关系。也是我们今后学习数据结构最需关注的问题。逻辑结构可分为以下四种：**集合结构、线性结构、树形结构和图状结构**。

(1) 集合结构

集合结构中的数据元素除了同属于一个集合外，它们之间没有其他关系。各个数据元素是“平等”的，它们的共同属性是“同属于一个集合”。数据结构中的集合关系就类似于数学中的集合，如图 1.1 所示。

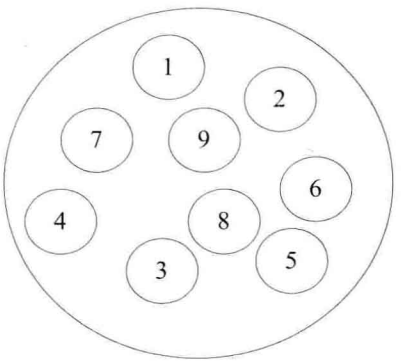


图 1.1 集合结构示意图

(2) 线性结构

线性结构中的数据元素之间是一对一的关系，如图 1.2 所示。

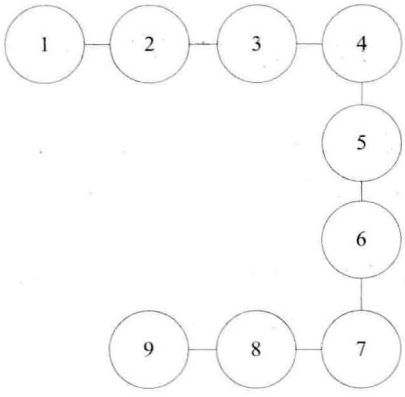


图 1.2　线性结构示意图

(3) 树形结构

树形结构中的数据元素之间存在一种一对多的层次关系，如图 1.3 所示。

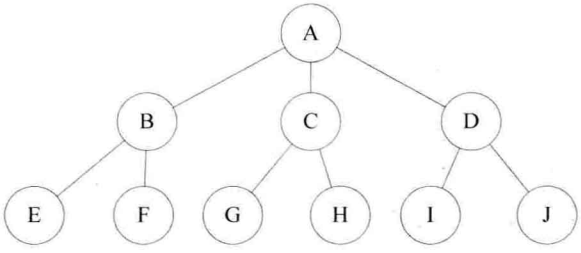


图 1.3　树形结构示意图

(4) 图状结构

图状结构的数据元素是多对多的关系，如图 1.4 所示。

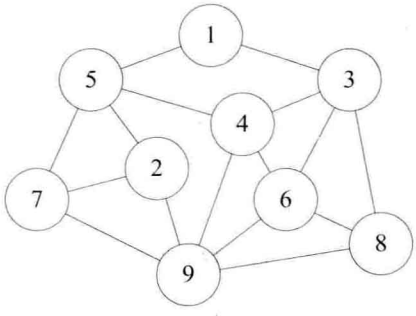


图 1.4　树状结构示意图

注意

在上述图例中，我们将每一个数据元素看做一个结点，用圆圈表示。元素之间的逻辑关系用结点之间的连线表示。如果这个关系是有方向的，那么用带箭头的连线表示。逻辑结构是针对具体问题的，是为了解决某个问题，在对问题理解的基础上，选择一个合适的数据结构表示数据元素之间的关系。

1.4 数据的物理结构

数据结构在计算机中的表示(又称为映像)称为数据的**物理结构**，也称为**存储结构**，是指数据的逻辑结构在计算机中的存储形式。数据的存储结构应正确反映数据元素之间的逻辑关系，如何存储数据元素之间的逻辑关系，是实现物理结构的重点和难点。

元素之间的关系在计算机中有两种不同的表示方法：**顺序映像**和**非顺序映像**，并由此得到数据元素两种不同的存储结构：**顺序存储**和**链式存储**。

1. 顺序存储结构

把数据元素存放在地址连续的存储单元里，其数据间的逻辑关系和物理关系是一致的，借助元素在存储器中的相对位置来表示数据元素之间的逻辑关系，如图 1.5 所示。

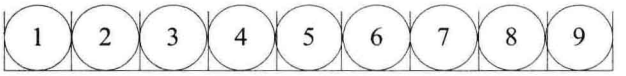


图 1.5 顺序存储结构示意图

2. 链式存储结构

把数据元素存放在任意的存储单元里，这组存储单元可以是连续的，也可以是不连续的。数据元素的存储关系并不能反映其逻辑关系，因此需要借助指示元素存储地址的**指针**(pointer)表示数据元素之间的逻辑关系，如图 1.6 所示。

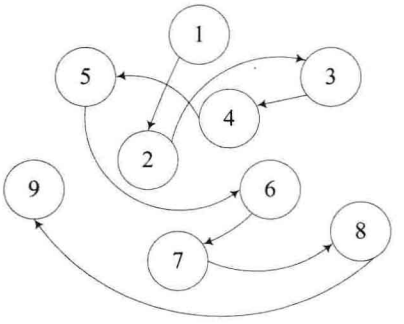


图 1.6 链式存储结构示意图

数据的逻辑结构和物理结构是密切相关的两个方面，同一逻辑结构可以对应不同的物理结构。以后读者会看到，任何一个算法的设计取决于选定的**数据(逻辑)结构**，而算法的实现依赖于采用的**存储结构**。逻辑结构是面向问题的，而物理结构就是面向计算机的，其基本的目标就是将数据及其逻辑关系存储到计算机的内存中。