

数据结构

(第2版)

李伟生 主编



国家开放大学
THE OPEN UNIVERSITY OF CHINA

DATA
STRUCTURE

数据结构

(第2版)

李伟生 主编

中央广播电视大学出版社·北京

图书在版编目 (CIP) 数据

数据结构 / 李伟生主编. —2 版. —北京: 中央广播电视大学出版社, 2014. 12

ISBN 978 - 7 - 304 - 06898 - 1

I. ①数… II. ①李… III. ①数据结构 - 开放大学 - 教材 IV. ①TP311. 12

中国版本图书馆 CIP 数据核字(2015)第 006411 号

版权所有, 翻印必究。

数据结构 (第 2 版)

SHUJU JIEGOU

李伟生 主编

出版·发行: 中央广播电视大学出版社

电话: 营销中心 010 - 66490011 总编室 010 - 68182524

网址: <http://www.crtvup.com.cn>

地址: 北京市海淀区西四环中路 45 号 邮编: 100039

经销: 新华书店北京发行所

策划编辑: 邹伯夏

版式设计: 赵 洋

责任编辑: 邹伯夏

责任校对: 张 娜

责任印制: 赵连生

印刷: 北京密云胶印厂

印数: 0001 ~ 3000

版本: 2015 年 1 月第 2 版

2015 年 1 月第 1 次印刷

开本: 787 × 1092 1/16

印张: 14.75 字数: 326 千字

书号: ISBN 978 - 7 - 304 - 06898 - 1

定价: 27.00 元

(如有缺页或倒装, 本社负责退换)

“数据结构”是计算机专业的专业基础课和主干课程之一。随着计算机技术的发展和广泛应用，本课程也已经成为相关专业的热门限选课或选修课。本书是根据国家开放大学计算机科学与技术专业“数据结构”课程教学大纲的要求编写的。

全书共分9章，第1章介绍了有关数据结构和算法的基本概念，为以下章节的学习做好准备。第2章~第7章由浅入深地讨论了线性表、栈和队列、字符串、数组和广义表、树及图等基本的数据结构，并着重介绍了它们在计算机中的存储方法和相关算法，使读者对以上常用数据结构有一个基本的了解，并为具体应用打下良好的基础。第8章和第9章结合相关的数据结构，针对非数值算法中最常用的“查找”“排序”算法的部分典型算例，介绍了算法的原理和具体实现方法。这两章内容也是数据结构的初步应用。书中以C语言作为数据结构和算法的描述语言，给出了部分程序。读者按照程序中的提示和注释，在相应的运行环境中很容易改写并运行相关程序。

针对国家开放大学学生的特点和实际情况，“数据结构”课程更应该注重应用，内容要突出重点。所以本书在教学内容上遵循少而精和重应用的原则，略去了数据结构的形式定义、抽象数据类型等概念和内容，而对算法分析，则仅仅介绍了一些基本原理并说明如何从直观上对算法进行评估。对现有部分教材中列出但不作为教学要求的章节、带“*”号的内容，本书做了部分删除。另外，本书还删改了部分较烦琐但不影响本课程知识结构的内容。

本书在叙述方法上力求深入浅出、通俗易懂，尽可能以具体实例引出相关概念。书中引入了较多的图例用以描述相应的数据结构、相关操作和算法运行情况。在有关算法的讨论中还充分注意突出其中的关键步骤和要点，并在程序中做了详细注释，这样更利于读者自学。每章的开头简要列举了本章相关内容的应用、学习目的和学习要求。每一章的结束部分概要介绍了该章所学内容，列出了主要知识点，这样对提高学生的学习兴趣，使其把握章节的要点十分有益。

本书是在2008年版本的基础上修订的，在广泛调研了几年来学生学习情况的基础上，结合计算机和信息技术的发展及需求，按照教学改革的新要求，在教学内容取舍、教材编排、知识结构组织、知识点提炼等方面进行了重新审定。作者在撰稿风格上力求符合读者的认知需求，对部分章节进行了重写或改写，增加了部分内容。例如，随着信息技术的发展，

字符处理的相关算法显得越来越重要,所以书中增加了字符串中的模式匹配的算法等内容。作者对部分较抽象的内容配以具体实例和图示,特别对部分程序的运行情况,强调用计算机的存储结构变化来描述。结合各章的关键知识点,作者对每章后的习题进行了进一步筛选,在部分章节,尝试以知识点为主线编写。所有这些还有待在教学实践中检验,对于不足之处,我们将与时俱进地改进。

本书由北京交通大学李伟生教授主编,国家开放大学王春风副教授负责组稿。书中第1章、第2章、第4章、第5章、第8章、第9章和附录由李伟生编写,第6章、第7章由国家开放大学徐孝凯教授编写,第3章由王春风编写,北京交通大学研究生张玉宪、邢朝阳等同学参加了教材编写的辅助工作。

本书由北京理工大学吴鹤龄教授担任主审,同时北方工业大学吴洁明教授、北京印刷学院杨树林教授共同参与了审定。各位专家认真审阅了全部书稿,提出了宝贵的修改意见,在此表示衷心的感谢。

由于时间仓促,书中错误和不足之处在所难免,恳请专家和广大读者提出宝贵意见。

编 者
2014.11

第1章 绪论

1

- 1.1 数据结构简介 1
- 1.2 数据结构的基本术语和概念 2
- 1.3 算法和算法分析简介 4
 - 1.3.1 算法 4
 - 1.3.2 时间复杂度 4
 - 1.3.3 空间复杂度 5
- 本章小结 5
- 习题 6

第2章 线性表

7

- 2.1 线性表的定义 7
- 2.2 线性表的逻辑结构和基本操作 8
 - 2.2.1 线性表的逻辑结构 8
 - 2.2.2 线性表的基本操作 8
- 2.3 线性表的顺序存储结构（顺序表）及相关操作 9
 - 2.3.1 顺序存储结构的概念 9
 - 2.3.2 利用数组处理线性表 9
 - 2.3.3 利用指针（变量）处理线性表 9
 - 2.3.4 顺序存储结构的线性表（顺序表）的操作 10
 - 2.3.5 插入、删除操作的时间复杂度分析 12
- 2.4 线性表的链式存储结构（链表）及相关操作 13
 - 2.4.1 线性表的链式存储的基本概念 13
 - 2.4.2 单向链表 13
 - 2.4.3 单向循环链表 21
 - 2.4.4 双向循环链表 21

● 2.5 一元多项式的存储和加法运算	24
2.5.1 一元多项式和线性表	24
2.5.2 使用数组方式	24
2.5.3 使用链表方式	25
● 本章小结	28
● 习题	29

第③章 栈和队列 31

● 3.1 栈	31
3.1.1 栈的定义	31
3.1.2 栈的基本运算	32
3.1.3 栈的顺序存储结构及基本操作	32
3.1.4 栈的链式存储结构及基本操作	35
3.1.5 栈的应用	38
3.1.6 栈与递归	44
● 3.2 队列	47
3.2.1 队列的定义	47
3.2.2 队列的基本运算	47
3.2.3 队列的顺序存储结构及基本操作	48
3.2.4 队列的链式存储结构及基本操作	52
3.2.5 队列的简单应用举例	56
● 本章小结	56
● 习题	57

第④章 字符串 61

● 4.1 字符串的定义和相关概念	61
4.1.1 字符串的定义	61
4.1.2 字符串的相关概念	61
● 4.2 C 语言中字符串的特点、存储结构和访问方式	62
4.2.1 C 语言中字符串的特点	62
4.2.2 C 字符串的存储结构和访问方式	63
4.2.3 程序举例	65
4.2.4 基本函数	69
● 4.3 字符串的模式匹配	72
4.3.1 字符串的模式匹配的概念	72

4.3.2 求子串位置的定位算法	72
4.3.3 模式匹配的 kmp 方法	74
● 本章小结	82
● 习题	82

第5章 数组和广义表 85

● 5.1 数组的定义、逻辑结构和特点	85
5.1.1 一维数组	85
5.1.2 二维数组	86
5.1.3 数组的存储	87
● 5.2 C 语言中数组的定义、存储结构	87
5.2.1 一维数组	87
5.2.2 二维数组	87
● 5.3 特殊矩阵的压缩存储	88
5.3.1 对称矩阵	88
5.3.2 稀疏矩阵	90
● 5.4 广义表	94
5.4.1 广义表的定义	95
5.4.2 广义表的相关概念	95
5.4.3 广义表（列表）的图形表示	95
5.4.4 广义表的存储结构	95
● 本章小结	99
● 习题	100

第6章 树和二叉树 102

● 6.1 树的概念	102
6.1.1 树的定义	102
6.1.2 树的日常应用举例	103
6.1.3 树的表示	103
6.1.4 树的基本术语	104
6.1.5 树的性质	105
● 6.2 二叉树的概念	107
6.2.1 二叉树的定义	107
6.2.2 二叉树的性质	107

6.3	二叉树的存储结构	109
6.3.1	顺序存储结构	109
6.3.2	链接存储结构	110
6.4	二叉树遍历	111
6.4.1	二叉树遍历的概念	111
6.4.2	二叉树的递归遍历算法	112
6.4.3	二叉树的非递归遍历算法	115
6.4.4	二叉树的按层遍历算法	116
6.5	二叉树的其他运算	117
6.6	二叉树运算的程序调试	122
6.7	哈夫曼树	124
6.7.1	基本术语	124
6.7.2	构造哈夫曼树	126
6.7.3	哈夫曼编码	128
6.7.4	哈夫曼树运算的程序调试	131
	本章小结	133
	习题	134

第7章 图

137

7.1	图的概念	137
7.1.1	图的定义	137
7.1.2	图的基本术语	138
7.2	图的存储结构	141
7.2.1	邻接矩阵	141
7.2.2	邻接表	144
7.2.3	边集数组	147
7.3	图的遍历	148
7.3.1	深度优先搜索遍历	149
7.3.2	广度优先搜索遍历	152
7.3.3	非连通图的遍历	155
7.3.4	图的遍历算法的上机调试	155
7.4	图的生成树和最小生成树	160
7.4.1	图的生成树和最小生成树的概念	160
7.4.2	克鲁斯卡尔算法	162

● 7.5 最短路径	167
7.5.1 最短路径的概念	167
7.5.2 从一个顶点到其余各顶点的最短路径	168
● 7.6 拓扑排序	170
● 本章小结	173
● 习题	174

第8章 查找

178

● 8.1 查找的基本概念	178
● 8.2 线性表的查找	179
8.2.1 顺序查找	179
8.2.2 折半查找	181
8.2.3 分块查找	183
● 8.3 树表的查找	184
8.3.1 二叉排序树的定义	184
8.3.2 二叉排序树的查找	184
8.3.3 二叉排序树的插入和删除	185
● 8.4 哈希表及其查找	189
8.4.1 哈希表的基本概念	189
8.4.2 哈希函数的构造方法	190
8.4.3 处理冲突的方法	191
● 本章小结	193
● 习题	194

第9章 排序

196

● 9.1 排序的基本概念	196
● 9.2 插入排序	198
9.2.1 直接插入排序	198
9.2.2 折半插入排序	199
● 9.3 交换排序	201
9.3.1 冒泡排序	201
9.3.2 快速排序	202
● 9.4 选择排序	204
9.4.1 直接选择排序	204
9.4.2 堆排序	205

- 9.5 归并和归并排序 209
 - 9.5.1 归并两个有序的序列 209
 - 9.5.2 归并排序 209
- 本章小结 212
- 习题 213

附录 实验	215
-------	-----

参考文献	223
------	-----

“数据结构”是计算机科学与技术专业的核心专业基础课，课程着重介绍和研究数据的逻辑结构、存储结构及其处理方式等。其相关理论和设计技术在软件设计、系统开发、科学计算等领域有广阔的应用和发展前景。数据结构课程也是学习后续课程，如操作系统、数据库原理、算法的设计与分析等课程的重要基础。

本章的主要内容包括：数据结构简介，数据结构的基本概念，对算法、与算法分析相关的概念的简单介绍。

通过本章的学习，要求：

- (1) 了解“数据结构”课程的主要学习内容、学习目的和学习方法。
- (2) 掌握数据结构的基本概念，了解常用的4种基本结构。
- (3) 掌握逻辑结构、物理结构的含义，理解它们之间的关系。
- (4) 了解算法和算法分析的基本概念。

1.1 数据结构简介

随着计算机技术和信息技术的发展，不同的研究领域和相关行业，包括日常生活的方方面面，越来越多的信息需要利用计算机加工、处理。纷繁复杂的信息包括：数字、字符、文字、表格、图像、声音等。对这些信息首先需要进行数字化，同时必须对这些具有不同实际背景的数据进行合理分类，研究它们的特性，找出数据间的共性关系，构建出若干类典型、通用、经济且易于对数据操作的模型来组织数据，然后研究如何有效地把这些数据存储到计算机中，设计相关算法对数据进行分析、处理、加工，如图1-1所示。

以上所述的核心内容可归结为：如何依据数据间的逻辑关系把数据合理、有效地存储到计算机中，应用相应的算法对其进行处理，而这正是数据结构要研究和解决的问题。由此可见，数据结构是理论和实践密切结合的课程。在学习中既要充分注意综合分析、逻辑思维能力的培养，更要注重算法设计和编程等能力的提高。

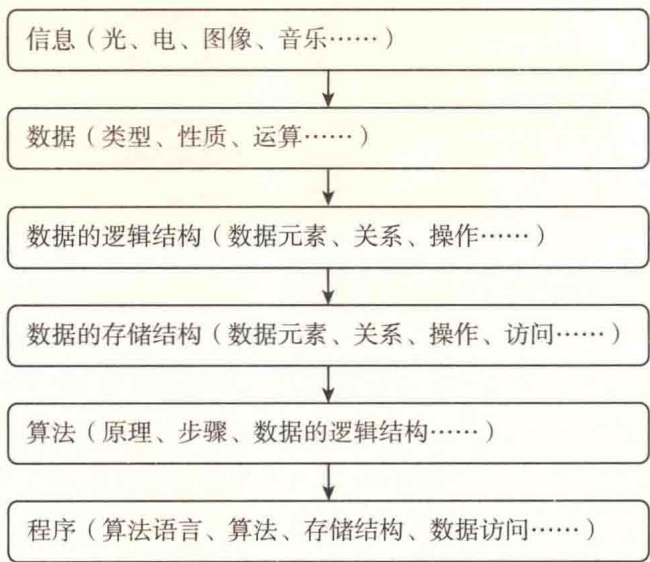


图 1-1 数据结构的简单描述

1.2 数据结构的基本术语和概念

1. 数据（data）

数据是描述和量化客观事物和信息等的符号。在计算机领域，它是指所有能输入计算机并能被计算机系统和程序识别、存储、加工和处理的符号的总称。不同的信息，如图像、图形、声音、光、电、月球表面信息、行星的运动轨迹、原子核的裂变过程等都能通过数字化归于数据的范畴。

2. 数据元素（data element）

数据元素是数据的基本单位，在计算机程序中通常把数据元素作为一个整体来存储和处理。数据元素可以只是单个的数据项，如学生的年龄就是一个数据元素。其也可以由多个数据项复合组成，如根据需要可以把学生的相关信息（学号、姓名、年龄、性别、电话号码等）多个数据项组成一个数据元素统一处理。数据项是数据不可分割的最小单位。数据元素在许多应用中又被称为记录。

3. 数据结构（data structure）

数据结构是相互之间存在一种或多种特定关系的数据元素的集合。数据元素间的关系称为结构。客观事物之间存在着各种不同的联系，但将其抽象为数据以后再来研究它们所具有的共性关系就简单得多。数据结构研究这种关系的目的是把数据合理、有效地存储到计算机上进行处理，所以我们的着眼点是数据间的位置关系、数据间是否存在直接或间接的联系等方面。例如，一个班的学生名单表中，学生是一个接着一个排列的，可以将其抽象为“一对

一的线性结构”，而把它们随机地记录在笔记本上时，从位置上看并没有任何关系，只能看出这些人同属于一个班级。又如，某单位的上级单位与各个下级单位的关系、祖辈与后辈的关系就可以被抽象为“一对多的树形结构”。而诸如某城市中各个公交站点之间的关系、通信线路上各用户之间的关系则可以用“多对多的图状结构”描述。根据数据间的不同特性，通常有下列4类基本结构：

- (1) 集合：结构中的数据除了“同属于一个集合”的关系外，不存在其他关系。
- (2) 线性结构：结构中的数据元素的位置之间存在一对一的关系。
- (3) 树形结构：结构中的元素之间存在一对多的关系。
- (4) 图状结构：结构中的数据元素存在多对多的关系。图状结构又称网状结构。

图1-2列出了4种数据基本结构的示意图。

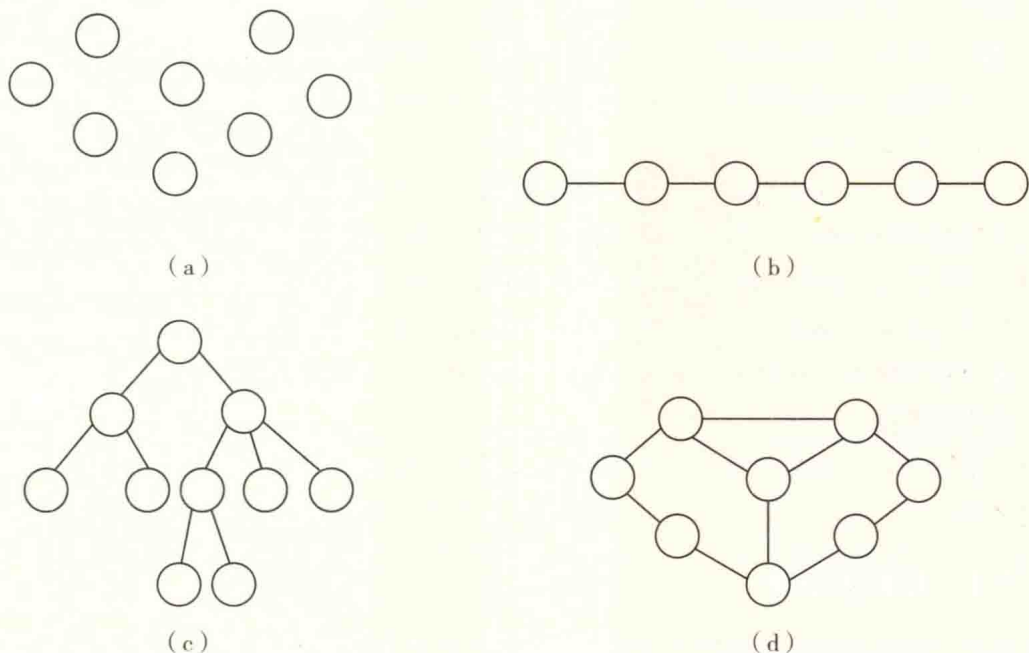


图1-2 基本结构示意图

(a) 集合；(b) 线性结构；(c) 树形结构；(d) 图状结构

4. 逻辑结构 (logic structure)

通常把数据元素之间的逻辑（抽象）关系称为逻辑结构，如上面列举的4种结构。

5. 存储（物理）结构 (storage structure)

数据的逻辑结构在计算机中的表示称为物理结构或存储结构。它包括数据元素的表示和关系的表示。同一种逻辑结构可以根据实际应用的需要采用不同的存储方式，也就是可以有不同的存储结构。

6. 数据对象 (data object)

数据对象是数据的一个子集，是性质相同的数据元素的集合。计算机在处理数据时，总

是根据问题的需要，针对某一种或几种数据对象。例如，要对学生的学号进行排序、查找等处理时，涉及“整数集合”；而对学生的学习成绩进行统计、计算时，涉及“实数集合”。

1.3 算法和算法分析简介

程序设计的要素包括：算法语言、算法、数据结构和程序设计技术。其中，算法和数据结构密切相关，算法必须要有恰当的数据结构才能有效地发挥作用，而数据结构的设计必须充分考虑其是否能适应相应算法的要求。本书在介绍典型数据结构的同时，还要针对性地介绍相关算法。

1.3.1 算法

简而言之，算法就是解决特定问题的方法，是对特定问题求解步骤的一种描述。算法是计算机学科中最具方法论性质的核心概念，是计算机科学的灵魂。算法具有以下5个特性：

(1) 有穷性。在合法输入下，一个算法必须在执行有穷步之后结束，而其中每一个步骤都能在有限时间内完成。

(2) 确定性。算法中每一个步骤、每一条指令都有确切的含义，对符合算法要求的任何输入数据都能正确执行，而对于相同的输入只能得到相同的输出。

(3) 可行性。算法中描述的操作都可以通过将已经实现的基本操作执行有限次来实现。

(4) 算法有零个或多个输入。算法要有处理的对象，也就是数据。根据需要，可以在执行时输入某些数据，并通过变量接受它们。有些数据可能已被嵌入算法中，或通过赋值方法使变量获得数据。

(5) 算法有一个或多个输出。输出是一组与输入量有着某种特定关系的量，是算法执行后的结果。

1.3.2 时间复杂度

算法的时间复杂度是评估算法的重要标准之一，它能较好地体现算法本身的时间效率，而与具体实现算法的计算机软件、硬件无关。时间复杂度在“算法的设计与分析”理论研究中占有重要地位，而对于应用人员，只要求了解时间复杂度分析的一般原理，并能将它作为选用算法的一个标准参照就可以了。

分析一个算法的时间复杂度首先要从中选取一种能在很大程度上体现该算法执行时间的基本操作，以该基本操作重复执行的次数作为度量。例如，在矩阵相乘的算法中，乘法就是基本操作。在从一批整数中求最大数的算法中，对两个数据的比较就是基本操作，一批整数中数据的个数称为问题的规模。

一般情况下，算法基本操作重复执行的次数是问题规模 n 的某个函数 $f(n)$ 。而算法的时间复杂度简单来说是指算法中某种基本操作执行次数的数量级。通常用 $T(n) = O(f(n))$ 表示，其中 O 表示 $f(n)$ 的数量级。

例 1.1 用 C 语言求两个 n 阶矩阵的乘积的算法。

```
for(i=1;i<=n;i++)
    for(j=1;j<=n;j++)
    {
        c[i][j]=0;
        for(k=1;k<=n;k++)
            c[i][j]=c[i][j]+a[i][k]*b[k][j];
    }
```

上述算法问题的规模为 n ，基本操作乘法的执行次数 $f(n) \approx n \times n \times n$ ，所以 $T(n) = O(n^3)$ 。有关时间复杂度的分析，在以下章节中还会多次遇到。

1.3.3 空间复杂度

类似于算法的时间复杂度，空间复杂度作为算法所需存储空间的量度，记作：

$$S(n) = O(f(n))$$

其中， n 为问题的规模，算法所需的存储空间是问题规模 n 的函数 $f(n)$ 。空间复杂度对于“算法设计与分析”的研究也是必要的，但它的重要性远不如时间复杂度，故本书不做重点讨论。但程序设计人员在设计算法时始终要有“时、空”这两个概念，要尽量避免存储空间的浪费。

本章小结

本章介绍了数据结构和算法的基本概念，主要知识点如下：

- (1) 数据元素是数据的基本单位，它可以是一个数据项，也可以由若干个数据项复合组成。
- (2) 数据结构研究的是数据的逻辑结构和物理结构，并在相应结构上定义有关操作，设计并实现相关算法。
- (3) 数据结构是相互间存在一种或多种特定关系的数据集合，数据元素之间的关系称为结构。
- (4) 数据元素之间的逻辑关系称为逻辑结构。把数据存储到计算机中，并具体体现数据元素间的逻辑结构称为物理结构（存储结构）。
- (5) 同一种逻辑结构可以有不同的存储结构。
- (6) 算法是解决特定问题的方法，它有 5 个特性。
- (7) 算法的时间复杂度为 $T(n) = O(f(n))$ ，其中 n 是问题的规模。

习题

一、单项选择题

1. 在数据结构中, 与所使用的计算机无关的是数据的 () 结构。
A. 逻辑 B. 存储 C. 逻辑与存储 D. 物理
2. 数据结构在计算机内存中的表示是 ()。
A. 给相关变量分配存储单元 B. 数据的存储结构
C. 数据的逻辑结构 D. 算法的具体体现
3. 算法的时间复杂度与 () 有关。
A. 算法本身 B. 所使用的计算机
C. 算法的程序设计 D. 数据结构
4. 数据的存储结构包括数据元素的表示和 ()。
A. 数据处理的方法 B. 数据元素间的关系的表示
C. 相关算法 D. 数据元素的类型

二、简答题

1. 请说出4类基本数据结构, 它们各有什么特点, 请各举出一例。
2. 数据结构研究的主要问题是什么?
3. 要求在 n 个数据元素中找其中值最大的元素。说出你使用的算法, 其基本操作是什么? 它的执行次数是多少? 求算法的时间复杂度。