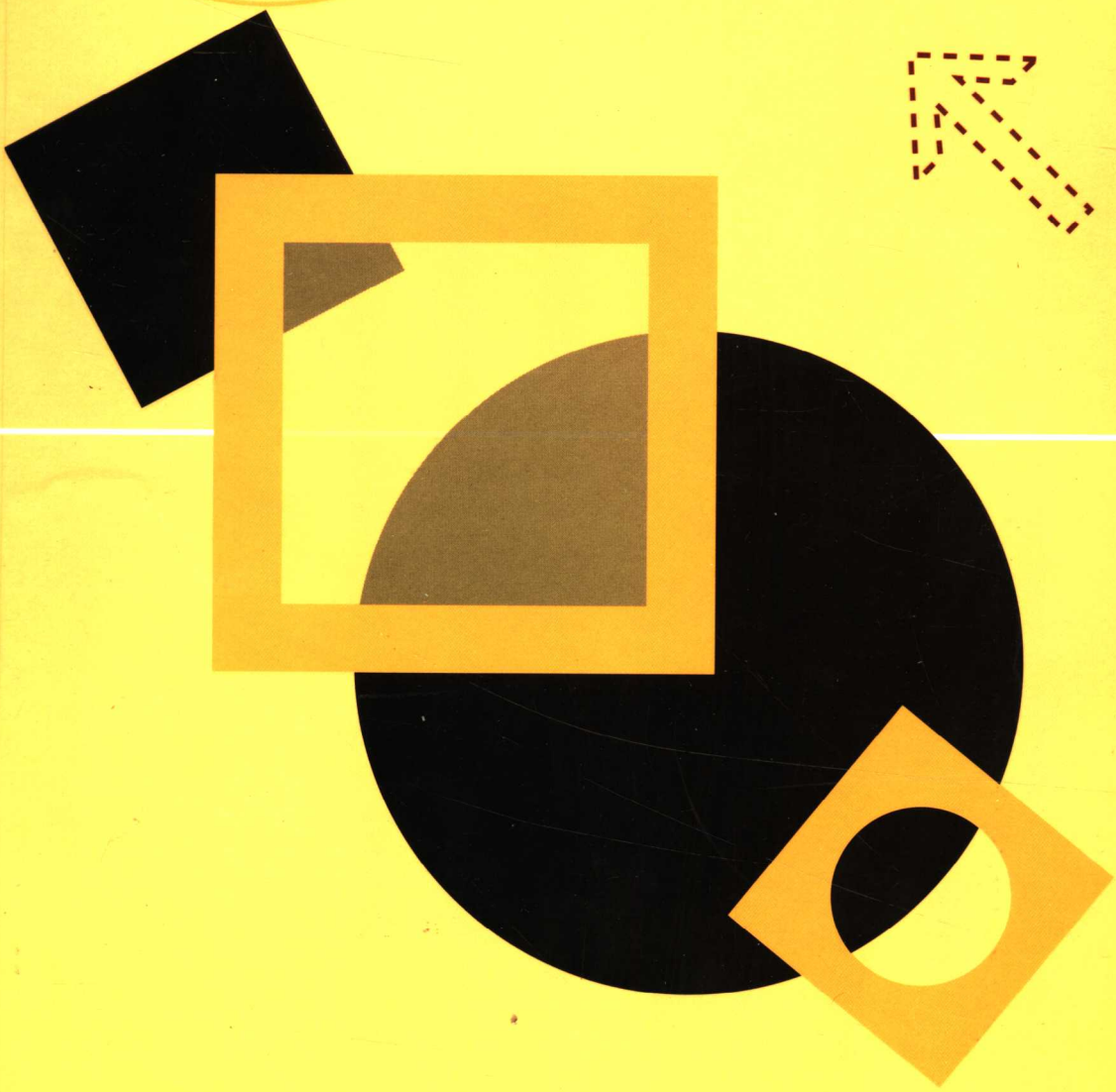
计算机与信息技术专业应用教材

数据结构与算法教程

李春葆　苏光奎　编著



清华大学出版社

▶ 计算机与信息技术专业应用教材

数据结构与算法教程

李春葆　苏光奎　编著

清 华 大 学 出 版 社

北 京

内 容 简 介

数据结构与算法设计是计算机专业的核心课程，主要传授数据组织方法和典型问题求解策略，具有一定的抽象性，不易掌握。

本书作者具有多年授课经验，对教学重点和学习难点有深刻了解。在内容安排上，以教学大纲为指导，充分考虑课程特点，兼顾学习习惯。全书分为 11 章，内容涉及数据结构的基本概念、线性表、栈和队列、串和数组、递归和广义表、树和二叉树、图、查找、内排序、外排序、文件以及算法设计技术。

书中精心设计大量例题，用于演示说明相关概念和方法；各章在课后都给出多个典型练习题，并在附录中提供参考答案。其目的是加深理解，强化应用。

本书适合用作大专院校相关专业“数据结构与算法”课程的教学用书。

图书在版编目（CIP）数据

数据结构与算法教程/李春葆，苏光奎编著.

－北京：清华大学出版社，2005.5

ISBN 7-302-11160-X

Ⅰ. 数…　Ⅱ. ①李…　②苏…　Ⅲ. ①数据结构—高等学校—教材　②算法分析—高等学校—教材　Ⅳ. TP311.12

中国版本图书馆 CIP 数据核字（2005）第 058941 号

出 版 者：清华大学出版社　　**地　　址**：北京清华大学学研大厦

http://www.tup.com.cn　　**邮　　编**：100084

社 总 机：010-62770175　　**客户服务**：010-62776969

组稿编辑：夏非彼

文稿编辑：科海

封面设计：付剑飞

版式设计：科海

印 刷 者：北京市耀华印刷有限公司

发 行 者：新华书店总店北京发行所

开　　本：787×1092　1/16　　**印张**：18.625　　**字数**：444 千字

版　　次：2005 年 6 月第 1 版　　2005 年 6 月第 1 次印刷

书　　号：ISBN 7-302-11160-X/TP・7376

印　　数：1~5000

定　　价：25.00 元

丛 书 序

为适应信息社会高速发展的需求，目前全国各类高等院校都在进行计算机教学的全方位改革，目的是规划出一整套面向计算机与信息技术专业、具有中国高校计算机教育特色的课程计划和教材体系，本丛书就是在这一背景下应运而生的。我们组织了由全国高校计算机专业的专家教授组成的“计算机与信息技术专业应用教材”课题研究组，通过对计算机和信息技术专业全方位的研讨，并结合我国当前的实际情况，编写了这套系统性、科学性和实践性都很强的丛书。

丛书特色

☑ 先进性：力求介绍最新的技术和方法

先进性和时代性是教材的生命，计算机与信息技术专业的教学具有更新快、内容多的特点，本丛书在体例安排和实际讲述过程中都力求介绍最新的技术和方法，并注重拓宽学生的知识面，激发他们学习的热情和创新的欲望。

☑ 理论与实践并重：阐明基础理论，强调实践应用

理论是实践的基础，实践是理论的升华；不能有效指导实践的理论是空头理论，没有理论指导的实践是盲目的实践。对于时代呼唤的信息化人才而言，二者缺一不可。本丛书以知识点为主线，穿插演示性案例于理论讲解之中，使枯燥的理论变得更易于理解、易于接受；此外，还在每一章的末尾提供大量的实习题和综合练习题，目的是提高学生综合利用所学知识解决实际问题的能力。

☑ 易教易学：创新体例，合理布局，通俗易懂

本丛书结构清晰，内容系统详实，布局合理，体例较好；力求把握各门课程的核心，通俗易懂，便于教学的展开，也便于学生学习。

丛书组成

首批推出的计算机与信息技术专业应用教材涵盖计算机基础、程序设计和数据库三大领域，共 15 本：

- 操作系统教程
- 计算机系统结构教程
- 数据结构与算法教程
- Java 语言程序设计
- Access 数据库程序设计

- C 程序设计教程（基于 Visual C++平台）
- C 程序设计教程学习与上机指导（基于 Visual C++平台）
- C++程序设计
- C++程序设计学习与上机实验指导
- Visual FoxPro 程序设计
- Visual Basic 程序设计
- SQL Server 2000 应用系统开发教程
- SQL Server 2000 学习与上机实验指导
- 数据库原理与应用——基于 Access
- 数据库原理与应用——基于 Visual FoxPro

服务之窗

本丛书的出版者和作者谒诚为读者提供服务。

本丛书的网络支持与服务网址为 http://www.khp.com.cn/xxjsjc/。在这里提供了实用的相关资源与最新信息，读者可以方便地下载本丛书的实例源代码，便捷地参与讨论，并可同作者进行交流，得到作者和专家的帮助。我们将努力有效、快捷地解决读者在图书使用和学习中遇到的疑难问题，并致力于提供更多的增值服务。

丛书编委会

编者寄语

如果说科学技术的飞速发展是 21 世纪的一个重要特征的话，那么教学改革将是 21 世纪教育工作不变的主题。要紧跟教学改革，不断创新，真正编写出满足新形势下教学需求的教材，还需要我们不断地努力实践、探索和完善。本丛书虽然经过细致的编写与校订，仍难免有疏漏和不足，需要不断地补充、修订和完善。我们热情欢迎使用本丛书的教师、学生和读者朋友提出宝贵意见和建议，使之更臻成熟。

本丛书作者的电子邮件：licb@public.wh.hb.cn

本丛书出版者的电子邮件：feedback@khp.com.cn

前　言

数据结构与算法设计是计算机专业的核心课程。主要传授组织数据方法和各种典型的问题求解策略。本书是作者针对该课程的特点，在总结长期教学经验的基础上编写的。全书分为11章，第1章为概论，介绍数据结构的基本概念，特别强调算法分析的方法；第2章为线性表，介绍线性表的两种存储结构即顺序表和链表的逻辑结构与基本运算的实现过程；第3章为栈和队列，介绍这两种特殊的线性结构的概念与应用；第4章为串和数组，介绍串的概念、模式匹配算法、多维数组和稀疏矩阵的基本运算；第5章为递归和广义表，介绍广义表的概念与递归运算算法；第6章为树和二叉树，介绍树和二叉树的概念与各种运算，特别突出递归算法的实现过程；第7章为图，介绍图的概念树与图的各种运算的实现过程；第8章为查找，介绍各种查找算法的实现过程；第9章为内排序，介绍各种内排序算法的实现过程；第10章为文件，介绍各种文件组织方式和外排序算法；第11章为算法设计技术，介绍常用的算法及其在数据结构中的应用。

本书适合于作为计算机及相关专业本科生“数据结构与算法”课程的教材，也适合于计算机水平考试人员参考。

由于水平所限，尽管编者不遗余力，仍可能存在错误和不足之处，敬请读者批评指正。

编　者

2005年1月

目　录

第1章 概 论

计算机的功能主要是处理数据，这些数据绝不是杂乱无章的，而是有着某种内在联系。只有分清楚数据的内在联系，合理地组织数据，才能对它们进行有效的处理。如何合理地组织数据、高效率地处理数据，正是本书的目的。本章简要介绍有关数据结构的基本概念和算法分析方法。

1.1 什么是数据结构

数据就是指能够被计算机识别、存储和加工处理的信息，包括文字、表格、图像等。例如，一个班的全部学生记录、a～z的字母集合、1～1000之间的所有素数等都是数据。

数据元素（也称为结点）是数据的基本单位，在程序中通常把它作为一个整体进行处理。有时一个数据元素可以由若干个数据项组成。数据项是具有独立意义的不可分割的最小标识单位。如整数这个集合中，10这个数就可称是一个数据元素。又比如在一个数据库（关系数据库）中，一个记录可称为一个数据元素，而这个元素中的某一字段就是一个数据项。

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。这些数据元素不是孤立存在的，而是有着某种关系，这种关系称为结构。

1.1.1 逻辑结构

数据元素和数据元素之间的逻辑关系称为数据的逻辑结构。根据数据元素之间逻辑关系的不同特性，分为下列4类基本结构。

- 集合：结构中的数据元素同属于一个集合（集合类型由于元素之间的关系过于松散，数据结构课程中较少讨论）。
- 线性结构：结构中的数据元素存在一对一的关系。
- 树形结构：结构中的数据元素存在一对多的关系。
- 图形结构：结构中的数据元素存在多对多的关系。也称为网状结构。

在大多数情况下，数据的逻辑结构可以用二元组

```
S=(D,R)
```

来表示。其中D是结点（即数据元素）的有穷集合，即D是由有限个结点所构成的集合，R是D上的关系的有穷集合，即R是由有限个关系所构成的集合，而每个关系都是从D到D的关系。在表示每个关系时，用尖括号表示有向关系，如<a，b>表示存在结点a到结点b之间的关系；用圆括号表示无向关系，如（a，b）表示既存在结点a到结点b之间的关系，又存在结点b到结点a之间的关系。设r是一个D到D的关系，r∈R，若d，d'∈K，且<d，d'>∈r，则称d'是d的后继结点，d是d'的前趋结点，这时d和d'是相邻的结点（都是相对r而言的）；如果不存在一个d'使<d，d'>∈r，则称d为r的终端结点；如果不存在一个d'使<d'，d>∈r，则称d为r的开始结点；如果d既不是终端结点也不是开始结点，则称d是内部结点。

例如一个城市表，如表1.1所示，就是一个数据结构，它由很多记录（这里的数据元素就是记录）组成，每个元素又包括多个字段（数据项）。那么这个表的逻辑结构是怎么样的呢？通常分析数据结构都是从数据元素之间的关系来分析，对于这个表中的任一个记录，它只有一个前趋结点，也只有一个后继结点，而且整个表只有一个开始结点和一个终端结点。由此可知，这个表的逻辑结构为线性结构。其逻辑结构表示如下：

```
City=(D,R)
D={Beijing,Shanghai,Wuhan,Xi'an,Nanjing}
R={r}
r={<Beijing,Shanghai>,<Shanghai,Wuhan>,<Wuhan,Xi'an>,<Xi'an,Nanjing>}
```

表 1.1　城市表

城市	区号	说明
Beijing	010	首都
Shanghai	021	直辖市
Wuhan	027	湖北省省城
Xi' an	029	陕西省省城
Nanjing	025	江苏省省城

数据的逻辑结构可以用相应的关系图来表示，称之为逻辑结构图。

【例1.1】　设数据的逻辑结构如下：

```
B1=(D,R)
D={1,2,3,4,5,6,7,8,9}
R={r}
r={<1,2>,<1,3>,<3,4>,<3,5>,<4,6> ,<4,7>,<5,8>,<7,9>}
```

试画出对应的逻辑结构图，并指出哪些是开始结点，哪些是终端结点，说明是何种数据结构。

解：B1对应的逻辑结构图如图1.1所示。其中，1是开始结点；2，6，8，9是终端结点。它是一种树形结构。

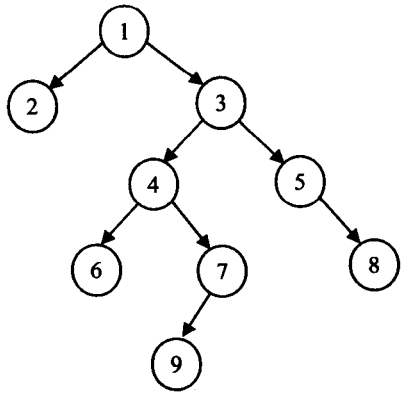


图 1.1 逻辑结构图

1.1.2 存储结构

数据在计算机中的存储表示称为数据的存储结构，也称为物理结构。

把数据存储到计算机中时，通常要求既存储各结点的数值，又存储结点与结点之间的逻辑关系。在实际应用中，数据的存储方法是灵活多样的，可根据问题规模（通常是指结点数目的多少）和运算种类等因素适当选择。本书将介绍4种基本的存储结构，它们是：顺序存储结构、链式存储结构、索引存储结构和散列存储结构。下面简要介绍这些存储结构的特点。

1．顺序存储结构

顺序存储结构是把逻辑上相邻的元素存储在一组连续的存储单元中，其元素之间的逻辑关系由存储单元地址间的关系隐含表示。

对于前面的逻辑结构City，假定每个元素占用30个存储单元，数据从100号单元开始由低地址向高地址方向存放，对应的顺序存储结构如图1.2所示。

地址	城市名	区号	说明
100	Beijing	010	首都
130	Shanghai	021	直辖市
160	Wuhan	027	湖北省省城
190	Xi’an	029	陕西省省城
210	Nanjing	025	江苏省省城

图 1.2 顺序存储结构

顺序存储结构的主要优点是节省存储空间。因为分配给数据的存储单元全用于存放结点的数据值，结点之间的逻辑关系没有占用额外的存储空间。用这种方法来存储线性结构的数据元素时，可实现对各数据元素的随机访问。这是因为线性结构中每个数据元素都对应一个序号（开始结点的序号为1，它的后继结点的序号为2……），可以根据结点的序号i，计算出结点的存储地址：

```
Loc(i)=q+(i-1)×p
```

其中，p是每个元素所占的单元数；q是第一个元素结点所占单元的首地址。

顺序存储结构的主要缺点是不便于修改，在对结点进行插入、删除运算时，可能要移动一系列的结点。

2. 链式存储结构

链式存储结构是给每个结点附加指针字段，用于存放相邻结点的存储地址。假定给前面的逻辑结构City中的每个结点附加一个“下一个结点地址”即后继指针字段，用于存放后继结点的首地址，则可得到如图1.3所示的City的链式存储表示。从图中可以看出，每个结点占用两个连续的存储单元，一个存放结点的数值，另一个存放后继结点的首地址。

地址	城市名	区号	说明	下一个结点地址
100	Beijing	010	首都	210
130	Nanjing	025	江苏省省城	∧
160	Xi’an	029	陕西省省城	130
190	Wuhan	027	湖北省省城	160
210	Shanghai	021	直辖市	190

图 1.3　City 的链式存储表示

链式存储结构的主要优点是便于修改，在进行插入、删除运算时，仅需修改结点的指针字段值，不必移动结点。

与顺序存储结构相比，链式存储结构的主要缺点是存储空间的利用率较低，因为分配给数据的存储单元有一部分被用来存放结点之间的逻辑关系了。另外，由于逻辑上相邻的结点在存储器中不一定相邻，因此，在用这种方法存储的线性结构中不能对结点进行随机访问。

3. 索引存储结构

索引存储结构是在存储结点信息的同时，还建立附加的索引表。索引表中的每一项称为索引项，索引项的一般形式是：（关键字，地址），关键字惟一标识一个结点，地址作为指向结点的指针。对于线性结构来说，各结点的地址在索引表中是按结点的序号依次排列的。图1.4是City的一种索引存储表示。

索引表

地址	关键字	指针
300	010	100
310	021	130
320	025	210
330	027	160
340	029	190

结点表

地址	城市名	区号	说明
100	Beijing	010	首都
130	Shanghai	021	直辖市
160	Wuhan	027	湖北省省城
190	Xi’an	029	陕西省省城
210	Nanjing	025	江苏省省城

图 1.4　City 的索引存储表示

在进行关键字查找时，可以先在索引表中快速查找到相应的关键字（因为索引表中按关键字有序排序，可以采用二分查找），然后通过地址找到结点表中对应的记录。

线性结构采用索引存储后，可以对结点进行随机访问。在进行插入、删除运算时，由于只需移动存储在索引表中的结点的存储地址，而不必移动存储在结点表中的结点的数值，所以仍可保持较高的运算效率（这是因为，在一般情况下，结点中总包含有多个字段，移动一个结点的数值要比移动一个结点的地址花费更多的时间）。

索引存储结构的缺点是，为建立索引表而增加了时间和空间的开销。

4．散列存储结构

散列存储结构是根据结点的值确定结点的存储地址。具体做法是：以结点中某个字段的值为自变量，通过某个函数（称为散列函数）计算出对应的函数值i，再把i当作结点的存储地址。

对于City结构，假设以城市名的值作为自变量key，选用函数

```
H(key)=ASC(LEFT(key,1)) mod 8
```

来计算结点的存储地址，其中，ASC(LEFT(key,1))表示取字符串key中第一个字符的ASCII码，mod是取模运算，计算结果如下：

key	Beijing	Shanghai	Wuhan	Xi'an	Nanjing
ASC(key)	66	83	87	88	78
H(key)	2	3	7	0	6

于是，可以得到图1.5所示的City的散列存储表示。

地址	城市名	区号	说明
0	Xi'an	029	陕西省省城
1			
2	Beijing	010	首都
3	Shanghai	021	直辖市
4			
5			
6	Nanjing	025	江苏省省城
7	Wuhan	027	湖北省省城

图 1.5 City 的散列存储表示

散列存储的优点是查找速度快，只要给出待查找结点的数值，就有可能立即算出结点的存储地址。

与前3种存储方法不同的是，散列存储方法只存储结点的数值，不存储结点与结点之间的逻辑关系。散列存储方法一般只用于要求对数据能够进行快速查找、插入的场合。采用散列存储的关键是要选择一个好的散列函数和处理“冲突”的办法。本书第8章将详细介绍这种存储方法。

在用高级语言编程时，可以用编程语言所提供的数据类型来描述数据的存储结构。例如，用“一维数组”表示一组连续的存储单元，来实现顺序存储结构、索引存储结构和散列存储结构；用C/C++语言中的“指针”来实现链式存储结构。

1.1.3 数据运算

数据运算就是施加于数据的操作，如查找、添加、修改、删除等。在数据结构中，运算不仅仅是加减乘除这些算术运算，还常常涉及算法问题。算法的实现与数据的存储结构密切相关。

1.1.4 数据结构和数据类型

将按某种逻辑关系组织起来的一组数据元素，按一定的存储方式存储于计算机中，并在其上定义了一个运算的集合，称为一个数据结构。

数据类型是高级程序设计语言中的一个基本概念，它和数据结构的概念密切相关。

一方面，在程序设计语言中，每一个数据都属于某种数据类型。类型明显或隐含地规定了数据的取值范围、存储方式以及允许进行的运算。因此可以认为，数据类型是在程序设计语言中已经实现了的数据结构。

另一方面，在程序设计过程中，当需要引入某种新的数据结构时，必须借助编程语言所提供的数据类型来描述数据的存储结构。

1.2 算法和算法分析

1.2.1 算法及其表示

算法是对特定问题求解步骤的一种描述，它是指令的有限序列，其中每条指令表示一个或多个操作。算法有以下5个重要特征。

① 有穷性：一个算法必须总是（对任何合法的输入值）在执行有穷步之后结束，且每一步都可在有穷时间内完成。

② 确定性：算法中每一条指令必须有确切的含义，不会产生二义性。

③ 可行性：一个算法是能执行的，即算法中描述的操作都是可以通过已经实现的基本运算的有限次执行来实现。

④ 输入性：一个算法有一个或多个输入。

⑤ 输出性：一个算法有一个或多个输出。

描述算法的方法很多，有的采用类PASCAL语言，有的采用自然语言。本书采用C/C++语言来描述算法的实现过程。