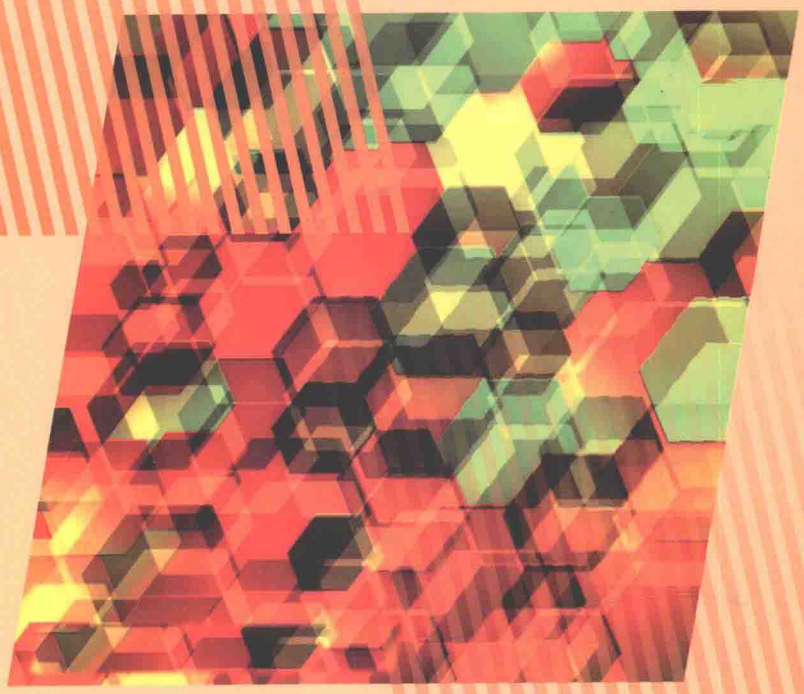
数据结构

李英明　主编



南京大学出版社

数据结构

主　编　李英明
副主编　王　强　冯　超
　　田　杰　祝种谷

南京大学出版社

图书在版编目(CIP)数据

数据结构 / 李英明主编. —南京 : 南京大学出版社, 2016.7
ISBN 978-7-305-17330-1

Ⅰ. ①数… Ⅱ. ①李… Ⅲ. ①数据结构 Ⅳ. ①TP311.12

中国版本图书馆 CIP 数据核字(2016)第 171228 号

出版发行 南京大学出版社
社　　址 南京市汉口路 22 号　　邮　编 210093
出 版 人 金鑫荣

书　　名 数据结构
主　　编 李英明
责任编辑 尤　佳　　编辑热线 025-83592123

照　　排 南京南琳图文制作有限公司
印　　刷 南京新洲印刷有限公司
开　　本 787×1092　1/16　印张 13.5　字数 329 千
版　　次 2016 年 7 月第 1 版　2016 年 7 月第 1 次印刷
ISBN 978-7-305-17330-1
定　　价 32.00 元

网址：http://www.njupco.com
官方微博：http://weibo.com/njupco
官方微信号：njupress
销售咨询热线：(025) 83594756

前　言

数据结构是计算机专业一门重要的专业必修课。用计算机来解决实际问题时，就要涉及数据的表示及数据的处理，而数据表示及数据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。目前数据结构也是全国计算机等级考试的必考内容和多数高校计算机专业专升本入学考试的必考科目。因此，数据结构在计算机及其相关专业中具有举足轻重的地位。

数据结构主要研究数据在计算机中的存储和操作，课程内容丰富、学习量大，其算法又十分抽象。经过我们多年的教学实践，结合了高职高专教学的特色，总结出一些该课程的特点和教学方法。为此，我们编写了这本教材，以满足广大同学的要求和计算机教学的需要。

全书采用C语言作为数据结构和算法的描述语言，概念表达准确，逻辑推理严谨，语言精练，通俗易懂，便于教学和自学。全书共分9章，第1章是绪论，第2章介绍了线性表，第3章介绍了数组和广义表，第4章介绍了栈和队列，第5章介绍了串，第6章介绍了树，第7章介绍了图，第8章介绍了查找，第9章介绍了排序。针对近几年的考试大纲和方向，每章后都精心设计了习题，习题难易适当，题型丰富。

本书可作为普通高等院校、高等专科学校及高等职业技术院校的教材，也可以作为大学非计算机专业的选修课教材和计算机应用技术人员的自学参考书。

本书由李英明、王强 、冯超、田杰、祝种谷等组织编写，由李英明负责全书的统稿。在本书编写过程中，编者参考了大量有关数据结构的书籍和资料，在此对这些参考文献的作者表示感谢。由于编者水平有限，书中难免存在错误和不当之处，恳请广大读者批评指正，以便再版时改进。

编　者

2016年6月

目　录

课件 PPT

第 1 章

绪 论

学习目标

认识数据结构的基本内容。

学习要求

- 了解:数据结构的研究内容。
- 掌握:数据结构的基本概念和术语。
- 了解:数据元素间的结构关系。
- 掌握:算法及算法的描述。

1.1 数据结构的发展

1.1.1 数据结构的发展简史

众所周知,早期的计算机主要应用于科学计算,随着计算机的发展和应用范围的拓宽,计算机需要处理的数据量越来越大,数据的类型越来越多,数据结构越来越复杂,计算机的对象从简单的纯数值型数据发展为非数值型和具有一定结构的数据。要求人们对计算机加工处理的对象进行系统的研究,即研究数据的特性、数据之间存在的关系以及如何有效地组织、管理存储数据,从而提高计算机处理数据的效率。数据结构这门学科就是在此背景上逐渐形成和发展起来的。

最早对这一发展做出杰出贡献的是 D. E. Kunth 教授和 C. A. R. Hoare(霍尔)。D. E. Kunth 的《计算机程序设计技巧》和霍尔的《数据结构札记》对数据结构这门学科的发展做出了重要贡献。随着计算机科学的飞速发展,到 20 世纪 80 年代初期,数据结构的基础研究日臻成熟,成为一门完整的学科。

1.1.2 数据结构的研究内容

用计算机解决一个具体的问题时,大致需要经过以下几个步骤:

(1) 分析问题,确定数据模型。

(2) 设计相应的算法。

(3) 编写程序,运行并调试程序,直至得到正确的结果。

寻求数据模型的实质是分析问题,从中提取操作的对象,并找出这些操作对象之间的关系,然后用数学语言加以描述。有些问题的数据模型可以用具体的代数方程、矩阵等来表示,但更多的实际问题是无法用数学方程来表示的,下面通过几个例子加以说明。

[例 1.1] 学生成绩表

如图 1-1 所示是一个学生成绩表,表中的每一行称为一条记录,并按学号升序排列,它们之间存在"一对一"的关系,是一种线性结构,它构成了学生成绩表的逻辑结构。

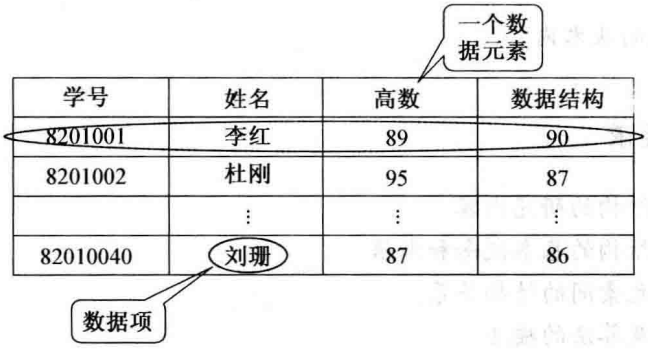


图 1-1 学生成绩表

学生成绩表在计算机外存中的存储方式构成该表的存储结构,在该表中查找记录、插入记录、删除记录以及对记录进行排序等操作又构成了数据的运算。

[例 1.2] 组织示意图

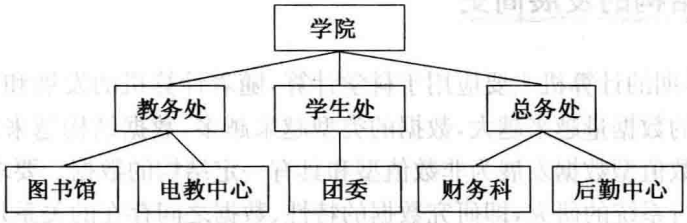


图 1-2 树形结构示意图

如图 1-2 所示是某高校组织示意图,其中高校名称是树根,把下设处室看成它的树枝中间结点,把处室下级单位看成树叶,这就构成了树形结构,树形结构通常用来表示结点的分层组织,结点之间是"一对多"的关系,除根结点之外,每个结点有且只有一个父结点。这种结构也是一种数据结构,其主要操作是遍历、查找、插入或删除等。

[例 1.3] 七桥问题

Euler 在 1736 年访问俄罗斯的哥尼斯堡时,他发现当地的居民正从事一项非常有趣的消遣活动。哥尼斯堡城中有一条名叫普莱格尔的河流,在河上建有七座桥,如图 1-3 所示。

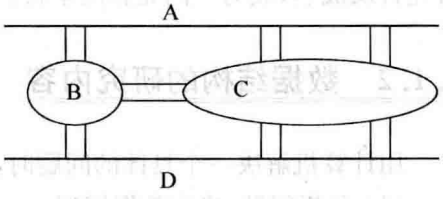


图 1-3 七桥图

这项有趣的消遣活动是在星期六做一次走过

所有7座桥的散步，每座桥只能经过一次而且起点与终点必须是同一地点。

设4块陆地分别为A、B、C、D，Euler把每一块陆地看成一个点，连接两块陆地的桥以线表示，如图1-4所示。

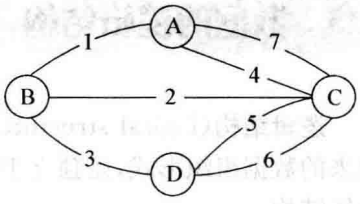


图1-4 欧拉回路

后来推论出此种走法是不可能的。他的论点是这样的，除了起点以外，每一次当一个人由一座桥进入一块陆地(或点)时，他同时也由另一座桥离开此点。即每个点如果有进去的边就必须有出来的边，因此每一个陆地与其他陆地连接的桥数必为偶数。7座桥组成的图形中，没有一点含有偶数条数，因此上述的任务是不可能实现的。

和上述问题相似，生活中还有不少的实例，如通信网和公路网都是“多对多”的关系，具有这种关系的结构称为图形结构。其主要的操作有遍历、求最短路径等。

类似的还有工资管理系统、棋类对弈问题等。对于这些非数值问题的描述，都是上述的表、树和图之类的数据结构，并且这些数据结构的元素和元素之间都存在着相互关系。因此，数据结构是一门抽象地研究数据之间的关系的学科。

1.2 数据结构的基本概念和术语

数据(Data)：是指在计算机科学中能够被计算机输入、存储、处理和输出的一切信息，是计算机处理的信息的某种特定的符号表示形式。包括数字、英文、汉字以及表示图形、声音、光和电的符号等。

数据项(Data Item)：是数据的最小单位，有时也称为域(field)，即数据表中的字段，如图1-1所示。数据项是具有独立含义且不可分割的最小标识单位。

数据元素(Data Element)：是数据的基本单位，在计算机信息处理中通常作为一个整体来考虑。一个数据元素可以由若干个数据项组成，数据元素也称为元素、结点、顶点、记录。如图1-1所示。

数据对象(Data Object)：具有性质相同的数据元素的集合，是数据的一个子集。例如大写字母字符数据对象是集合$C=\{'A','B','C',\cdots,'Z'\}$；整数数据对象是集合$N=\{0,\pm1,\pm2,\cdots\}$。

数据类型(Data Type)：是一个值的集合和定义在这个值集合上的一组操作的总称。数据类型中定义了两个集合：值集合和操作集合。其中值集合定义了该类型数据元素的取值，操作集合定义了该类型数据允许参加的运算。例如C语言中的int类型，取值范围是[−32 768,32 767]，主要的运算为加、减、乘、除、取模、乘方等。

按数据元素取值的不同特性，高级语言中的数据类型一般都包括两部分：原子类型和结构类型。原子类型的值是不可分的，例如，C语言中的int类型、char类型、float类型；结构类型是通过若干成分(可以是原子类型或结构类型)构造而成的。高级语言一般都提供用户自定义结构类型的机制。

数据结构(Data Structure)：带结构的数据元素的集合，描述了一组数据元素及元素间的相互关系。数据元素间的关系包括3个方面：数据的逻辑结构、存储结构和操作集合。

1.3 数据的逻辑结构

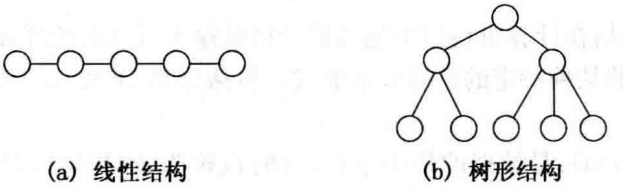
逻辑结构(logical structure)：是指数据元素之间的逻辑关系，是用户根据使用需要建立起来的数据组织形式，是独立于计算机的。根据数据元素之间的不同关系，有以下四种基本逻辑结构。

(1) 线性结构：其中的数据元素之间是“一对一”的关系。在线性结构中，有且仅有一个开始结点和一个终端结点，除了开始结点和终端结点，其余结点都有且仅有一个直接前驱和一个直接后继，如图 1-5(a)所示。

(2) 树形结构或层次结构：其中的数据元素之间存在着“一对多”的关系。比如，部门之间的隶属关系、人类社会的父子关系、上下级关系等。在树形结构中，除根结点之外，每个结点都有唯一的直接前驱，所有的结点都可以有 0 个或多个直接后继，如图 1-5(b)所示。

(3) 图形结构或网状结构：其中的数据元素之间存在着“多对多”的关系。在图状结构中，每个结点都可以有多个直接前驱和多个直接后继，如图 1-5(c)所示。

(4) 集合结构：数据元素间除了“同属于一个集合”的关系外，无任何其他关系。由于集合关系非常松散，因此可以用其他的结构代替，如图 1-5(d)所示。



(a) 线性结构　(b) 树形结构

(c) 图形结构　(d) 集合结构

图 1-5　数据结构的示意图

数据的逻辑结构可概括为两大类：线性结构和非线性结构。线性结构包括线性表、栈、队列、字符串、数组、广义表；非线性结构包括树、二叉树和图。

一个数据结构的逻辑结构 G 可以用二元组来表示：

$G=(D,R)$

其中：D 是数据元素的集合，R 是 D 上所有数据元素之间关系的集合(表示各元素的前驱、后继关系)。R 关系中圆括号表示双向，尖括号表示单向。

[例 1-4]　一种数据结构 Graph$=(D,R)$

其中：

$$\begin{cases} D=\{A,B,C,D,E\} \\ R=\{r\} \\ r=\{(A,B),(A,C),(B,C),(B,D),(B,E),(C,E)\} \end{cases}$$

r 中的(A,B)表示顶点 A 到顶点 B 之间的边是双向的，上述的结构关系如图 1-6 所示。

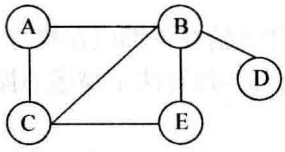


图 1-6 图形结构

1.4 数据的存储结构

数据的存储结构(Storage Structure)又称物理结构，是数据的逻辑结构在计算机存储器中的存储形式(或称映象)。对机器语言来说，这种存储形式是具体的，高级语言可以借助它的数据类型来描述存储形式的具体细节。依据数据元素在计算机中的表示，主要有以下 4 种不同的存储结构。

(1) 顺序存储结构：是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，可用一维数组描述。

(2) 链式存储结构：是借助指示元素存储地址的指针来表示数据元素之间的逻辑关系。可用指针类型描述，数据元素不一定存在地址的存储单元，存储处理的灵活性较大。

(3) 索引存储：是在原有存储数据结构的基础上，附加建立一个索引表，索引表中的每一项都由关键字和地址组成。采取索引存储结构的主要作用是提高数据的检索速度。

(4) 散列存储：是通过构造散列函数来确定数据存储地址或查找地址。

1.5 算法和算法的描述

算法和数据结构的关系紧密，任何一个算法的设计都取决于选定的数据的逻辑结构，而算法的实现则依赖于数据所采用的数据结构。

1.5.1 什么是算法

1. 算法的概念

算法是为了解决某类问题而规定的一个有限长的操作序列，是对解题过程的准确而完整的描述。

2. 算法的特性

一个算法一般具有以下特性：

(1) 输入：一个算法必须有0个或多个输入，这些输入取自于特定的对象集合。可以使用输入语句由外部提供，也可以使用赋值语句在算法内给定。

(2) 确定性：算法的每一步都应确切地、无歧义地定义。对于需要执行的动作都应严格地、清晰地规定。

(3) 有穷性：一个算法无论在什么情况下都应在执行有穷步后结束。

(4) 可行性：一个算法是可执行的，即算法中描述的操作都是可以通过已经实现的基本运算执行有限次来实现的。

(5) 输出：一个算法应有一个或多个输出，输出的量是算法计算的结果。

3. 算法与程序的区别

算法与程序的区别主要表现在以下几个方面。

(1) 算法代表了对问题的求解过程，而程序则是算法在计算机上的实现。算法用特定的程序设计语言来描述，就成了程序。

(2) 程序中的指令必须是机器可执行的，而算法中的指令则无此限制。

(3) 一个算法必须在有穷步之后结束，一个程序不一定满足有穷性。

1.5.2 算法设计的要求

通常设计一个好的算法应考虑达到如下目标：

(1) 正确性：在合理的数据输入下，能在有限的运行时间内得到正确的结果。

(2) 可读性：程序可读性好，有助于对算法的理解。

(3) 健壮性：当输入非法的数据时，算法应能恰当地做出反应或进行相应处理，而不是产生莫名其妙的输出结果。并且，处理出错的方法不应是中断程序的执行，而是返回一个表示错误或错误性质的值，以便在更高的抽象层次上进行处理。

(4) 高效性：对同一个问题，执行时间越短，算法的效率越高。

(5) 低存储量：完成相同的功能，执行算法所占用的存储空间应尽可能地少。

1.5.3 算法的描述

算法可以用流程图、自然语言、计算机程序语言或其他语言来描述，但描述必须精确地说明计算过程。为了便于理解和掌握算法的思想和实质，本书采用类C语言进行算法描述。类C语言实际上是对C语言的一种简化，保留了C语言的精华，忽略了C语言语法规则中的一些细节，这样描述出的算法清晰、直观、便于阅读和分析。在本书中算法是以函数形式描述，描述如下：

```
类型标识符    函数名(形式参数表)
/*  算法说明  */
{语句序列}
```

算法说明是不可缺少的部分，是对算法的功能、数据存储结构、形式参数的含义等的说明。

1.5.4 算法效率的评价

对于一个给定的问题求解，往往可以设计出若干个算法。如何评价这些算法的优劣呢？一个正确的算法效率通常用时间复杂度与空间复杂度来评价。

1. 时间复杂度(Time Complexity)

一个算法的执行时间等于其所有语句执行时间的总和，而任一语句的执行时间为该语句的执行次数与该语句执行一次所需时间的乘积。当算法转换成程序之后，每条语句的执行时间取决于机器的硬件速度、指令类型及编译的代码质量，而这些是很难确定的。因此，将算法中基本操作重复执行的次数作为算法执行时间的量度。

一般情况下，算法中基本操作重复执行的次数是问题规模 n 的某个函数 $f(n)$，算法的时间量度记作：

$$T(n)=O(f(n))$$

它表示随问题规模 n 的增大，算法执行时间的增长率和 $f(n)$ 的增长率相同，称作算法的渐近时间复杂度，简称时间复杂度。时间复杂度不是精确的执行次数，而是估算的数量级，它主要体现的是随着问题规模 n 的增大，算法执行时间的变化趋势。

[例 1-5] 有下列 3 条语句

```
(a) x=0
(b) for (i=1;i<=n;i ++) x=x+1
(c) for (i=1;i<=n;i ++)
      for(j=1;j<=n;j ++) x=x+i*j
```

上述 3 条语句的频度分别为 $1, n, n^2$，(a) 中语句执行了一次，时间复杂度为 $O(1)$；(b) 中语句 $x=x+1$ 执行了 n 次，时间复杂度为 $O(n)$；(c) 中赋值语句要执行 n^2 次，时间复杂度为 $O(n^2)$。不同数量级的时间复杂度增长率是不同的，当问题规模 n 越大时，其关系如下：$O(1)<O(lgn)<O(n)<O(nlgn)<O(n^2)<O(n^3)<O(2^n)$。

2. 空间复杂度(Space Complexity)

一个程序的空间复杂度是指程序运行从开始到结束所需要的存储空间。包括算法本身所占用的存储空间、输入/输出数据占用的存储空间以及算法在运行过程中的工作单元和实现算法所需辅助空间。类似于算法的时间复杂度，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。在进行空间复杂度分析时，若输入数据所占空间只取决于问题本身，和算法无关，则只需要分析除输入和程序之外的额外空间，否则应同时考虑本身所需空间。

复习思考题

一、名词解释

1. 数据 2. 数据结构 3. 数据的逻辑结构 4. 数据的存储结构 5. 算法

二、选择题

1. 研究数据结构就是研究(　　)。

A. 数据的逻辑结构

B. 数据的存储结构

C. 数据的逻辑结构和存储结构

D. 数据的逻辑结构和存储结构以及其数据在运算上的实现

2. 组成数据的基本单位是(　　)。

A. 数据项　　B. 数据类型　　C. 数据元素　　D. 数据变量

3. 数据结构是一门研究计算机中(　　)对象及其关系的学科。

A. 数值运算　　B. 非数值运算

C. 集合　　D. 非集合

4. 在数据结构中,从逻辑上可以把数据结构分成(　　)。

A. 动态结构和静态结构　　B. 紧凑结构和非紧凑结构

C. 线性结构和非线性结构　　D. 内部结构和外部结构

5. 数据的存储结构包括顺序、链接、散列和(　　)4 种基本类型。

A. 向量　　B. 数组　　C. 集合　　D. 索引

6. 算法分析的两个主要方面是(　　)。

A. 正确性和简单性　　B. 可读性和文档性

C. 数据复杂性和程序复杂性　　D. 时间复杂度和空间复杂度

7. 算法分析的目的是(　　)。

A. 找出数据结构的合理性　　B. 研究算法中的输入和输出的关系

C. 分析算法的效率以求改进　　D. 分析算法的易懂性和文档性

三、填空题

1. 数据逻辑结构包括_____、_____、_____和_____四种类型。

2. 在线性结构中,第一个结点_____前驱结点,其余每个结点有且只有_____个前驱结点;最后一个结点_____后继结点,其余每个结点有且只有_____个后继结点。

3. 在树形结构中,树根结点没有_____结点,其余每个结点有且只有_____个前驱结点;叶子结点没有_____结点,其余每个结点的后继结点可以_____。

4. 数据结构形式的定义为(在 D,R),其中 D 是_____的有限集,R 是 D 上的_____有限集。

5. 线性结构中元素之间存在_____关系,树形结构中元素之间存在_____关系,图形结构中元素之间存在_____关系。

6. 算法的五个重要特性是_____、_____、_____、_____、_____。

7. 设有一批数据元素,为了最快的存储某元素,数据结构宜用_____结构,为了方便插入一个元素,数据结构宜用_____结构。

8. 程序段 $i=1$;while($i<=n$)　$i=i*4$ 的时间复杂度为_____。

四、判断题

1. 数据元素是数据的最小单位。　　(　　)