

高等职业院校及应用型本科教材

GAODENG ZHIYE YUANXIAO JI YINGYONGXING BENKE JIAOCAI

数据结构 Java 应用 案例教程

SHUJU JIEGOU **Java** YINGYONG
ANLI JIAOCHENG

主 编 袁开友 郑孝宗
副主编 周龙福 吴平贵 彭 娟



重庆大学出版社

<http://www.cqup.com.cn>

高等职业院校及应用型本科教材
GAODENG ZHIYE YUANXIAO JI YINGYONGXING BENKI

数据结构 Java 应用 案例教程

SHUJU JIEGOU **Java** YINGYONG
ANLI JIAOCHENG

主 编 袁开友 郑孝宗
副主编 周龙福 吴平贵 彭 娟



重庆大学出版社

内容简介

本书通过具体的编程应用案例,系统地介绍了各种类型数据结构的逻辑结构、存储结构及相关的算法。全书共分9章,内容包括绪论、线性表、栈和队列、串、数组和广义表、树与二叉树、图、查找和排序等,此外本书还附录了“应用实践”拓展训练内容及参考代码,供学生参考和练习。全书采用Java应用案例驱动教学,讲解数据结构、算法及应用,内容翔实,层次清晰,实例丰富,讲解深入浅出。

本书作为计算机及相关专业本、专科数据结构课程的教材,也适合各类成人教育相关课程使用,还可以供从事计算机软件开发和应用的工程技术人员阅读、参考。

图书在版编目(CIP)数据

数据结构Java应用案例教程 / 袁开友, 郑孝宗主编. —重庆: 重庆大学出版社, 2014.8

高等职业院校及应用型本科教材

ISBN 978-7-5624-8277-2

I. ①数… II. ①袁… ②郑… III. ①数据结构—高等学校—教材 ②Java语言—程序设计—高等学校—教材 IV. ①TP311.12 ②TP312

中国版本图书馆CIP数据核字(2014)第122396号

高等职业院校及应用型本科教材 数据结构Java应用案例教程

主 编 袁开友 郑孝宗

副主编 周龙福 吴平贵 彭 娟

责任编辑: 章 可 版式设计: 章 可

责任校对: 谢 芳 责任印制: 赵 晟

*

重庆大学出版社出版发行

出版人: 邓晓益

社址: 重庆市沙坪坝区大学城西路21号

邮编: 401331

电话: (023) 88617190 88617185 (中小学)

传真: (023) 88617186 88617166

网址: <http://www.cqup.com.cn>

邮箱: fxk@cqup.com.cn (营销中心)

全国新华书店经销

万州日报印刷厂印刷

*

开本: 787×1092 1/16 印张: 16.5 字数: 412 千

2014年8月第1版 2014年8月第1次印刷

印数: 1—3 000

ISBN 978-7-5624-8277-2 定价: 33.00 元

本书如有印刷、装订等质量问题,本社负责调换

版权所有, 请勿擅自翻印和用本书
制作各类出版物及配套用书, 违者必究

前言

“数据结构”是计算机科学与工程的基础研究之一，是计算机程序设计的重要理论知识，是计算机相关专业必修的一门基础课程和核心课程。它主要介绍和研究数据在计算机中的存储和处理方法，旨在培养学生分析数据、组织数据、应用结构优化的数据进行高效编程的能力；它在计算机课程体系中占据承上启下的地位。掌握和应用数据结构的相关知识能更高效地进行计算机程序开发，是软件开发方向从业人员必备的专业核心技能。

本书以案例驱动教学，除了注重理论体系的完整性外，特别强调应用能力的培养，通过每章应用案例及实践的学习掌握理论知识、建立程序逻辑、提升编程能力。本书主要具有以下特点：

1. 应用教学驱动案例引领理论讲授，帮助学生循序渐进地学习。

本书每一章都通过分析来自学生身边的应用案例，引出体系相对完整的理论知识，通过精心组织设计的主题和应用，由浅入深、循序渐进地讲解和扩展知识，帮助学生逐步学习及理解数据结构相关知识。

2. 注重数据结构算法思想的培养，帮助学生建立程序逻辑。

本书在介绍每种数据结构及其算法时，注重编程思想和程序逻辑的培养，对每种数据结构、算法思想和对应案例都配有大量图表进行详细分析和说明，大部分章节的主要算法配有流程图，帮助学生理解算法思想、建立程序逻辑。

3. 强调应用能力的培养，帮助学生提升编程能力。

本书所有案例，在讲清知识、建立思想、理顺逻辑的基础上，都用完整的 Java 代码加以实现，不但有助于学生深入理解数据结构知识，而且有助于提升学生实际的应用编程能力，增强了学生的编程兴趣和成就感。

4. 强化“应用实践”训练，帮助学生拓展应用编程能力。

本书贯彻分层教学的思想，在附录中增设了“应用实践”拓展训练内容，要求能力较强的学生应用前面重点章节的相关知识实现编程，从而拓展训练学生的应用编程能力和知识迁移能力，同时附有应用实践参考代码，供学生参考和练习。

本书的教学学时以 60 ~ 80 学时为宜,通过一体化教学,让学生以“学中做、做中学”的方式开展学习;如果以“理论+上机”的方式开展教学,上机学时建议不低于 24 学时。教师可根据实际教学学时和学生具体情况等自行调整教学进度和内容。

本书由袁开友主编,并负责设计全书体系及统稿工作,还编写第 2 章、第 3 章、第 8 章、附录 A、附录 C;郑孝宗编写了第 4 章、第 5 章;周龙福编写了第 1 章、第 7 章;吴平贵编写了第 6 章,彭娟编写了第 9 章。在教材的编写过程中,重庆工程学院软件学院的领导和老师对该书的编写提出了很多建设性意见和建议,还得到重庆工程学院教务处的大力支持和关怀,以及广大软件学院学生的积极支持,在此表示衷心的感谢。本书在编写过程中,参考了大量的文献资料,在此对这些文献资料的作者表示诚挚的谢意。

受编者水平和时间方面的限制,书中难免存在疏漏与不足之处,敬请读者和同行专家批评指正。

袁开友
2014 年 5 月



■ 第1章 绪论	1
1.1 引言	1
1.2 基本概念和术语	4
1.3 算法的概念及其特性	6
1.3.1 算法的定义	6
1.3.2 算法的组成要素	7
1.3.3 算法的基本性质	8
1.4 算法设计的要求	8
1.5 算法的描述方法	9
1.6 算法分析和评价	11
1.6.1 算法的时间复杂度	12
1.6.2 算法的空间复杂度	14
本章小结	15
习题	16
■ 第2章 线性表	20
2.1 线性表的概念与基本操作	20
2.2 顺序表	21
2.2.1 顺序表的概念	22
2.2.2 顺序表基本操作及实现	23
2.3 单链表	29
2.3.1 单链表的概念	29
2.3.2 单链表基本操作及实现	30
2.4 循环链表	36
2.4.1 循环链表的概念	36
2.4.2 循环链表基本操作及实现	36
2.5 双链表	39
2.5.1 双链表的概念	39
2.5.2 双链表基本操作及实现	40

本章小结	44
习题	45
■ 第 3 章 栈和队列	47
3.1 栈	47
3.1.1 栈的概念及基本操作	47
3.1.2 顺序栈	48
3.1.3 链栈	52
3.1.4 递归和栈	57
3.2 队列	59
3.2.1 队列概念及基本操作	59
3.2.2 顺序队列	60
3.2.3 循环队列	63
3.2.4 链式队列	65
本章小结	69
习题	70
■ 第 4 章 串	73
4.1 串的概念和基本操作	73
4.2 串的实现	74
4.2.1 顺序定长存储及实现	75
4.2.2 串的堆存储及实现	76
4.2.3 串的链式存储及实现	79
4.3 串的模式匹配算法	81
4.3.1 Brute-Force 模式匹配算法	81
4.3.2 KMP 模式匹配算法	83
本章小结	86
习题	86
■ 第 5 章 数组与广义表	88
5.1 数组	88
5.1.1 数组的概念	88
5.1.2 数组的存储	88
5.1.3 矩阵的压缩	91

5.2 广义表	103
5.2.1 广义表的定义	103
5.2.2 广义表的特性	104
5.2.3 广义表的存储结构	104
本章小结	106
习题	107
 ■ 第6章 树和二叉树	108
6.1 树	109
6.1.1 树的概念及基本术语	109
6.1.2 树的逻辑表示方法	110
6.1.3 树的存储结构	110
6.2 二叉树	112
6.2.1 二叉树的概念	112
6.2.2 二叉树的性质	113
6.2.3 二叉树存储结构	115
6.2.4 二叉树的遍历	116
6.3 线索二叉树	119
6.3.1 线索二叉树的概念	119
6.3.2 线索化二叉树	119
6.3.3 遍历线索化二叉树	122
6.4 哈夫曼树	123
6.4.1 哈夫曼树概述	123
6.4.2 哈夫曼树的构造算法	124
6.4.3 哈夫曼编码	125
6.5 树、森林和二叉树	126
6.5.1 二叉树与树、森林之间的转换	126
6.5.2 树、森林的遍历	128
本章小结	130
习题	131
 ■ 第7章 图	134
7.1 图的概念及基本术语	135
7.2 图的存储结构	136

7.2.1 邻接矩阵表示法	137
7.2.1 邻接表表示法	138
7.3 图的遍历	139
7.3.1 深度优先搜索法	140
7.3.2 广度优先搜索法	142
7.4 图的应用	144
7.4.1 最小生成树	145
7.4.2 最短路径	152
7.4.3 AOV 网与拓扑排序	156
7.4.4 AOE 网与关键路径	160
本章小结	165
习题	166

■ 第8章 查找

8.1 查找的概念及基本术语	168
8.2 线性表查找	169
8.2.1 顺序查找	169
8.2.2 二分查找	171
8.2.3 分块查找	173
8.2.4 顺序表三种查找方法的比较	178
8.3 树表的查找	178
8.4 哈希表的查找	185
本章小结	190
习题	191

■ 第9章 排序

9.1 排序的概念及基本术语	193
9.2 插入排序	194
9.2.1 直接插入排序	194
9.2.2 二分插入排序	197
9.2.3 希尔排序	199
9.3 交换排序	202
9.3.1 冒泡排序	202
9.3.2 快速排序	204

9.4 选择排序	206
9.4.1 直接选择排序	206
9.4.2 堆排序	208
9.5 归并排序	210
9.6 基数排序	212
9.7 各种内排序方法的比较和选择	213
本章小结	214
习题	215
■ 附录 A 应用实践	218
实践 1 顺序表的建立与基本操作	218
实践 2 单链表的建立与基本操作	219
实践 3 顺序栈的建立与基本操作	219
实践 4 顺序队列的建立与基本操作	219
实践 5 线性表查找	220
实践 6 内排序	220
■ 附录 B 各章习题参考答案	222
■ 附录 C 应用实践参考代码	234
实践 1 顺序表的建立与基本操作	234
实践 2 单链表的建立与基本操作	237
实践 3 顺序栈的建立与基本操作	240
实践 4 顺序队列的建立与基本操作	242
实践 5 线性表查找	245
实践 6 内排序	247
■ 参考文献	251

第1章 | 绪 论

1.1 引 言

“数据结构”在计算机科学中是一门综合性的专业基础课。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。数据结构是介于数学、计算机硬件和软件三者之间的一门核心课程。数据结构这一门课的内容不仅是一般程序设计（特别是非数值性程序设计）的基础，而且是设计和实现编译程序、操作系统、数据库系统及其他系统程序的重要基础。

计算机解决一个具体问题时，大致需要经过下列几个步骤：首先要从具体问题中抽象出一个适当的数学模型，然后设计一个解此数学模型的算法（Algorithm），最后编出程序、进行测试、调整直至得到最终解答。寻求数学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，然后用数学的语言加以描述。计算机算法与数据的结构密切相关，算法无不依附于具体的数据结构，数据结构直接关系到算法的选择和效率。

由于早期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，非数值计算显得越来越重要。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。下面所列举的就是属于这一类的具体问题。

【例 1.1】学生信息检索系统。

当我们需要查询某个学生的有关情况时；或者想查询某个专业或年级的学生的有关情况时，只要建立了相关的数据结构，按照某种算法编写相关程序，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（见图 1.1（a））以及分别按姓名（见图 1.1（b）），专业（见图 1.1（c）），年级（见图 1.1（d））顺序排列的索引表。由这 4 张表构成的文件便是学生信息检索的数学模型，计算机的主要操作就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。

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

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

(b) 姓名索引表

(d) 年级索引表

图 1.1 学生信息查询系统中的数据

【例 1.2】人机对弈问题。

人机对弈是一个古老的人工智能问题，其解题思路是将对弈的策略事先存入计算机，包括对弈过程中所有可能出现的情况和响应的对策。在决定对策时，根据当前局势发展的趋势作出最有利的选择。因此计算机操作的对象（数据元素）是对弈过程中的每一步棋盘状态（格局），元素之间的关系由比赛规则决定。通常这个关系不是线性的，从一个格局可能派生出多个格局，因此常用树形结构来表示，如图 1.2 所示。

【例 1.3】教学计划编排问题。

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为图的数据结构来表示，如图 1.3

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。

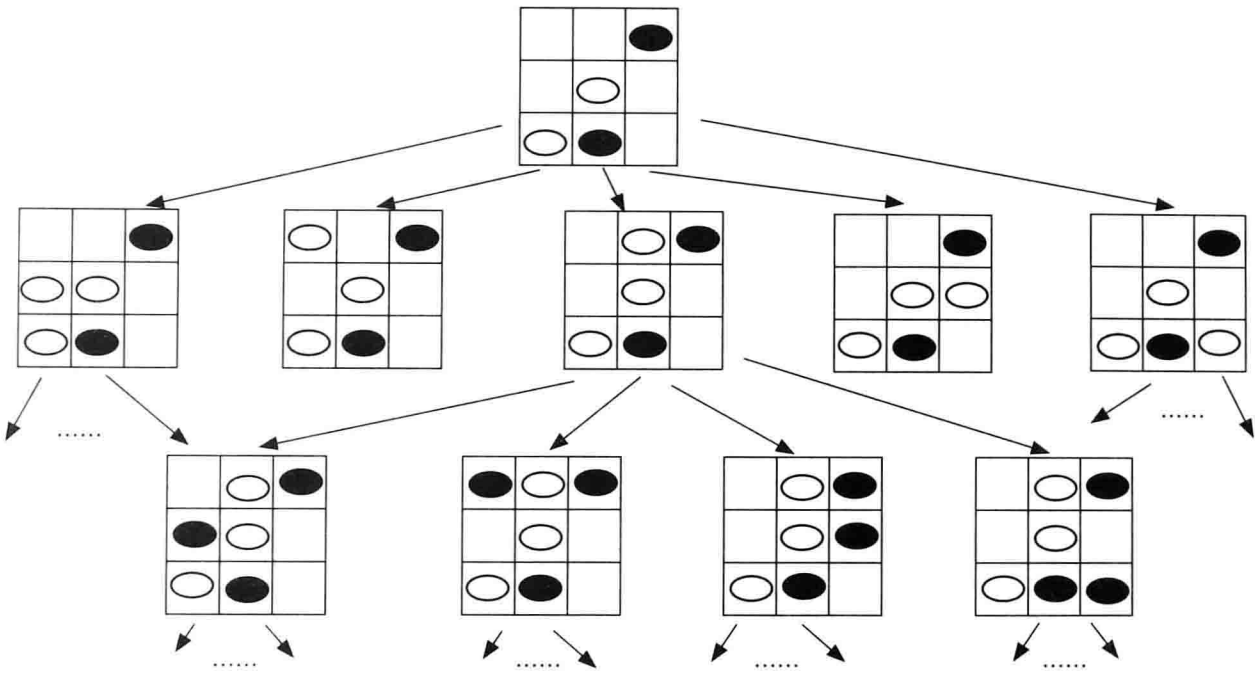
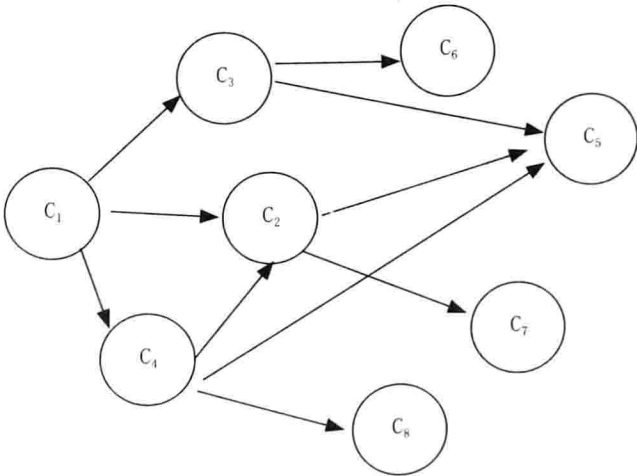


图 1.2 人机对弈图

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_8	编译原理	C_4

(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 1.3 教学计划编排问题的数据结构

由以上 3 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。

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

在系统地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。

1. 数据（Data）

数据是人们利用文字符号、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做的描述。在计算机科学中，数据的含义非常广泛，将一切能够输入计算机，并被计算机程序处理的信息，包括文字、表格、图像等，都称为数据。例如，一个学生信息的表格。

2. 数据元素（Data Element）

数据元素是组成数据的基本单位。在不同的条件下，数据元素又可称为元素、结点、顶点、记录等。在程序中通常把结点作为一个整体进行考虑和处理。例如，在如图 1.4 所示的学生信息中，为了便于处理，把其中的每一行（代表一个学生）作为一个基本单位来考虑，故该数据由 30 个结点构成。一般情况下，一个结点中含有若干个字段（也称为数据项）。例如，在图 1.4 中，每个结点都由学号、姓名、分组、年龄和住址 5 个字段构成。字段是构成数据的最小单位。

3. 数据对象（Data Object）或数据元素类（Data Element Class）

数据对象是性质相同的数据元素的集合，是数据的一个子集。例如：整数数据对象是集合 $N=\{0, \pm 1, \pm 2, \cdots\}$ ，字母字符数据对象是集合 $C=\{'A', 'B', \cdots, 'Z'\}$ ，如图 1.4 所示的表也可看成一个数据对象。由此可以看出，不论数据元素集合是无限集（如整数集）、有限集（如字符集），还是由多个数据项组成的复合数据元素（如学籍表），只要性质相同，都是同一个数据对象。

学号（ID）	姓名（Name）	分组（Group）	年龄（Age）	住址（Addr）
120010101	李华	100	16	四川成都
120010102	王丽	010	15	重庆万州
120010103	张阳	011	19	陕西西安
120010104	赵斌	012	16	重庆云阳
120010105	孙琪	020	18	四川广安
120010106	马丹	021	19	陕西宝鸡
120010107	刘畅	030	20	重庆黔江
⋮	⋮	⋮	⋮	⋮
120010130	黄凯	032	17	江苏南京

图 1.4 学生信息表

4. 数据结构（Data Structure）

数据结构是研究数据元素（Data Element）之间抽象化的相互关系和这种关系在计算

机中的存储表示（即所谓数据的逻辑结构和物理结构），并对这种结构定义相适应的运算，设计出相应的算法，而且确保经过这些运算后所得到的新结构仍然是原来的结构类型。根据数据元素间关系的不同特性，通常有下列4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。
- 树形结构：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，图形结构也称为网状结构。

图 1.5 为表示上述 4 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。

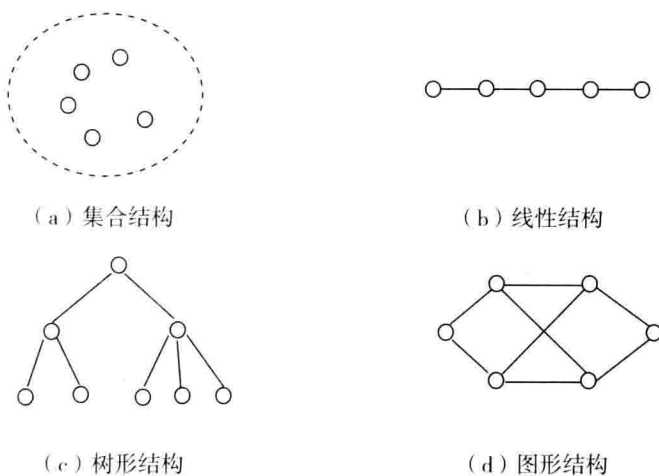


图 1.5 4 类基本结构的示意图

从上述的数据结构的概念中可知，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。

5. 数据的逻辑结构

数据的逻辑结构（Logical Structure）是指数据结构中元素之间的逻辑关系，它是从具体问题中抽象出来的数学模型，是独立于计算机存储器的（与具体的计算机无关）。数据的逻辑结构可分为 4 种基本类型：集合结构、线性结构、树形结构和图形结构，表和树是最常用的两种高效数据结构，许多高效的算法可以用这两种数据结构来设计实现。表是线性结构（一对一关系），树（一对多）和图（多对多）是非线性结构。

6. 数据存储结构

数据存储结构（Storage Structure）是数据的逻辑结构在计算机内存中的存储方式，

又称为物理结构。数据存储结构要用计算机编程语言来实现，因而依赖于具体的计算机语言，其可分为顺序存储或链式存储两种方式。

- 顺序存储：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，由此得到的存储方式称为顺序存储结构。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：是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，由此得到的存储方式称为链式存储结构。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式和链式存储方式外，有时为了查找的方便还采用索引存储方式和散列存储方式。

索引存储：在数据文件的基础上增加一个索引文件，通过索引表建立索引，可以将一个顺序表分成几个顺序子表，其目的是在查询时提高查找效率，避免盲目查找。

散列存储：是通过数据元素与存储地址之间建立某种映射关系，使每个数据元素与每个存储地址之间尽量达到一对一的关系。

7. 数据处理

数据处理是指对数据进行查找、插入、删除、合并、排序、统计以及简单计算等的操作过程。在早期，计算机主要用于科学和工程计算，进入 20 世纪 80 年代以后，计算机主要用于数据处理。据有关统计资料表明，现在计算机用于数据处理的时间比例达到 80% 以上，随着时间的推移和计算机应用的进一步普及，计算机用于数据处理的时间比例必将进一步增大。

8. 数据类型

数据类型 (Data Type) 是和数据结构密切相关的一个概念，在用高级程序设计语言编写的程序中，每个变量、常量或表达式都对应一个确定的数据类型。数据类型可分为两类：一类是非结构的原子类型，如基本数据类型 (整型、实型、字符型等)；另一类是结构类型，它可以由多个结构类型组成，并可以分解。结构类型的分解量可以是结构的，也可以是非结构的。例如数组的值由若干分量组成，每个分量可以是整型，也可以是数组等结构类型。

1.3 算法的概念及其特性

1.3.1 算法的定义

算法 (algorithm) 是指在解决问题时，按照某种机械的步骤一定可以得到问题的结果 (有解时给出问题的解，无解时给出无解的结论) 的处理过程。当面临某个问题时，需要找到用计算机解决这个问题的方法和步骤，算法就是解决这个问题的方法和步骤的描述。

所谓机械步骤，是指算法中有待执行的运算和操作，必须是相当基本的。换言之，它们都是能够精确地被计算机运行的算法，执行者（计算机）甚至不需要掌握算法的含义，即可根据该算法的每一个步骤进行操作并最终得出正确的结果。

“算法”其实并不是一个陌生的词，因为从小学大家就开始接触算法。例如运行四则运算，必须按照一定的算法步骤一步一步地做。“先运算括号内再运算括号外，先乘除后加减”可以说是四则运算的算法。以后学习的指数运算、矩阵运算和其他代数运算的运算规则都是一种算法。

就本课程而言，算法就是计算机解决问题的过程。在这个过程中，无论是形成解决问题的思路还是编写算法，都是在实施某种算法。前者是推理实现的算法，后者是操作实现的算法。

1.3.2 算法的组成要素

算法由操作、控制结构、数据结构 3 要素组成。

1. 操作

算法实现平台尽管有许多种类，它们的函数库、类库也有较大差异，但必须具备的基本操作功能是相同的。这些操作包括以下几个方面：

算术运算：加、减、乘、除。

关系比较：大于、小于、等于、不等于。

逻辑运算：与、或、非。

数据传送：输入、输出、赋值（计算）。

2. 算法的控制结构

一个算法功能的实现不仅取决于所选用的操作，还取决于各操作之间的执行顺序，即控制结构。算法的控制结构给出了算法的框架，决定了各操作的执行次序。这些结构包括以下几个方面：

顺序结构：各操作是依次进行的。

选择结构：由条件是否成立来决定选择执行。

循环结构：有些操作要重复执行，直到满足某个条件时才结束，这种控制结构也称为重复或迭代结构。

3. 数据结构

算法操作的对象是数据，数据间的逻辑关系、数据的存储方式及处理方式即是数据结构。它与算法设计是紧密相关的。在后面的具体案例分析讲解中会进行描述。