

全国计算机等级考试**5合1**通关宝典

专业研究等考**19**年
全新大纲·全新题库



National Computer Rank Examination

全国计算机等级考试

教程 二级Access
公共基础

考点点拨	重要考点分析点拨, 指点考生科学、高效学习
考点分值	科学标示各章节考点在历年考试中所占分值比例
同步训练	精选与各章考点内容对应的历年真题作为同步训练题目



 **北京理工大学出版社**
BEIJING INSTITUTE OF TECHNOLOGY PRESS

全国计算机等级考试命题研究中心 编著
未来教育教学与研究中心

全国计算机等级考试

专业研究等考**19**年
全新大纲·全新题库

NCR
Examination
全国计算机等级考试



National Computer Rank Examination

全国计算机等级考试

教程

二级Access
公共基础



 **北京理工大学出版社**
BEIJING INSTITUTE OF TECHNOLOGY PRESS

全国计算机等级考试命题研究中心 编著
未来教育教学与研究中心

图书在版编目(CIP)数据

全国计算机等级考试五合一通关宝典. 二级 Access 公共基础教程 / 全国计算机等级考试命题研究中心, 未来教育教学与研究中心编著. —北京: 北京理工大学出版社, 2013.12
ISBN 978-7-5640-8550-6

I. ①全… II. ①全… ②未… III. ①电子计算机—水平考试—自学参考资料②关系数据库系统—水平考试—自学参考资料 IV. ①TP3

中国版本图书馆 CIP 数据核字(2013)第 273140 号

出版发行 / 北京理工大学出版社有限责任公司

社 址 / 北京市海淀区中关村南大街 5 号

邮 编 / 100081

电 话 / (010) 68914775 (总编室)

82562903 (教材售后服务热线)

68948351 (其他图书服务热线)

网 址 / <http://www.bitpress.com.cn>

经 销 / 全国各地新华书店

印 刷 / 北京佳艺丰印刷有限公司

开 本 / 787 毫米×1092 毫米 1/16

印 张 / 7.25

字 数 / 216 千字

版 次 / 2014 年 2 月第 1 版 2014 年 2 月第 1 次印刷

定 价 / 94.00 元

责任编辑 / 张慧峰

文案编辑 / 多海鹏

责任校对 / 周瑞红

责任印制 / 边心超

本书使用说明

本系列丛书包括 4 个分册和 1 张学习光盘，分别是针对考试相关基本知识学习的《全国计算机等级考试五合一通关宝典二级 Access·公共基础教程》、《全国计算机等级考试五合一通关宝典二级 Access·教程》，针对考试中选择题部分的《全国计算机等级考试五合一通关宝典二级 Access·选择题题库》和针对考试中操作题部分的《全国计算机等级考试五合一通关宝典二级 Access·上机考试题库》。本书是《全国计算机等级考试五合一通关宝典二级 Access·公共基础教程》。

本书的内容特点如下：

全新升级的教程

根据教育部考试中心文件对计算机等级考试的调整规定，对考试系统的硬件环境和软件环境均进行升级，并发布全新大纲。我们在深入研究最新版大纲、新操作系统及新考试方法的基础上，组织计算机专家编写了本系列图书。书中采用了最新无纸化题库资源，适用于 Windows 7 的系统环境，考生可以通过本书全面掌握最新大纲要求的考试内容。

一学就会的教程

本书的知识体系都经过巧妙设计，力求将复杂问题简单化，将理论难点通俗化，让读者一看就懂，一学就会。

- 针对初学者和考生的学习特点及认知规律，精选内容，分散难点，降低台阶。
- 例题丰富，深入浅出地讲解和分析复杂的概念和理论，力求做到概念清晰、通俗易懂。
- 采用大量插图，并通过生活化的实例，将复杂的理论讲解得生动、易懂。
- 精心为考生设计学习方案，设置各种栏目引导和帮助考生学习。

衔接考试的教程

我们深入分析和研究历年考试真题及全套无纸化真考题库试题，结合考试的命题规律选择内容，安排章节，坚持多考多讲、少考少讲、不考不讲的原则。在讲解各章节的内容之前，都详细介绍了考试的重点和难点，从而帮助考生安排学习计划，做到有的放矢。在每章的最后安排了同步训练，根据考点涉及知识，精心选取各章相关无纸化真考题库试题，进行知识点的巩固练习，考生务必将同步训练中的题目全部练会做熟。

前 言

未来教育教学与研究中心是专门从事全国计算机等级考试研究的资深研究机构。1994年4月第1次全国计算机等级考试开考,未来教育教学与研究中心即开始致力于等考教学培训与图书出版,先后出版了四十多个系列的等考图书,“未来教育”品牌图书在考生中备受推崇。

本书是全国计算机等级考试系列教材之一。全国计算机等级考试(National Computer Rank Examination,简称NCRE)由教育部考试中心主办,每年参加考试的考生超过1000万人,是计算机类考试中规模最大、考试人数最多的考试,也是继高考、英语等级考试之后的全国第三大考试。

计算机等级考试考查考生的实际操作能力及理论基础。因此,为配合社会各类人员参加考试,未来教育组织计算机等级考试命题研究专家和多年从事辅导计算机等级考试的名师,在对近几年的考试深入分析之后,经全国计算机等级考试命题研究中心的反复论证以及未来教育教学与研究中心长期深入的研究,并依据教育部考试中心最新考试大纲的要求,在多位知名一线教师教学讲义的基础上,编写成本套全国计算机等级考试通关宝典系列图书。在系列图书在编写过程中,结合最新大纲和数十套历年考试试卷以及无纸化真题库试题,对经典考题进行了深入剖析,汇集了历年考试的重点、难点,使读者在整个学习过程中少走弯路,可大大提高学习效率。

本套图书体系合理、简明扼要、启迪性强,主要内容包括:教程、上机考试题库和习题集,立体式全方位地对考生进行应试指导,所以这不仅仅是一套书,而且是一套完整的学习方案,使考生能够科学、高效地进行复习,帮助考生顺利通过考试。

本书既可作为参加等级考试的备考用书,也可作为各级、各类学校的教材或者教学参考书,由于计算机等级考试发展迅猛,更要求不断改革创新,我们也迫切期望使用本教材的广大教师和学生提出宝贵的意见和建议,以便我们进一步加以改进。

本书在编写过程中得到了教师和广大学员的大力支持,许多高校的领导和全国计算机等级考试主管部门的老师对本书亦提出了许多宝贵意见,在此一并表示衷心的感谢。

全国计算机等级考试命题研究中心
未来教育教学与研究中心

目 录

第 1 章 数据结构与算法	1	1.8.4 排序方法比较	37
1.1 算法	3	同步训练	37
1.1.1 什么是算法	3	第 2 章 程序设计基础	41
1.1.2 算法复杂度	5	2.1 程序设计方法与风格	43
1.2 数据结构的基本概念	7	2.2 结构化程序设计	44
1.2.1 什么是数据结构	7	2.2.1 结构化程序设计方法的	
1.2.2 数据结构的图形表示	9	重要原则	45
1.2.3 线性结构与非线性结构	10	2.2.2 结构化程序的基本结构与	
1.3 线性表及其顺序存储结构	11	特点	45
1.3.1 线性表的基本概念	11	2.2.3 结构化程序设计的注意	
1.3.2 线性表的顺序存储结构	12	事项	47
1.3.3 线性表的插入运算	13	2.3 面向对象的程序设计	48
1.3.4 线性表的删除运算	14	2.3.1 面向对象方法的基本概念	48
1.4 栈和队列	15	2.3.2 面向对象方法的优点	51
1.4.1 栈及其基本运算	15	同步训练	52
1.4.2 队列及其基本运算	17	第 3 章 软件工程基础	53
1.5 线性链表	19	3.1 软件工程基本概念	55
1.5.1 线性链表的基本概念	19	3.1.1 软件的定义及软件的特点	55
1.5.2 线性链表的基本运算	21	3.1.2 软件危机	56
1.5.3 循环链表及其基本运算	23	3.1.3 软件工程	57
1.6 树与二叉树	23	3.1.4 软件工程过程	58
1.6.1 树的基本概念	24	3.1.5 软件生命周期	59
1.6.2 二叉树及其基本性质	25	3.1.6 软件开发工具与开发环境	59
1.6.3 二叉树的存储结构	28	3.2 结构化分析方法	60
1.6.4 二叉树的遍历	28	3.2.1 需求分析	60
1.7 查找技术	29	3.2.2 需求分析方法	61
1.7.1 顺序查找	30	3.2.3 软件需求规格说明书	63
1.7.2 二分法查找	30	3.3 结构化设计方法	63
1.8 排序技术	31	3.3.1 软件设计概述	63
1.8.1 交换类排序法	31	3.3.2 概要设计	65
1.8.2 插入类排序法	33	3.3.3 详细设计	68
1.8.3 选择类排序法	35	3.4 软件测试	70

3.4.1 软件测试的目的和准则	70	4.2.3 层次模型	92
3.4.2 软件测试方法	71	4.2.4 网状模型	93
3.4.3 白盒测试的测试用例设计	71	4.2.5 关系模型	93
3.4.4 黑盒测试的测试用例设计	74	4.3 关系代数	96
3.4.5 软件测试的实施	75	4.3.1 关系代数的基本运算	96
3.5 程序调试	76	4.3.2 关系代数的扩充运算	97
3.5.1 程序调试的基本概念	76	4.3.3 关系代数的应用实例	100
3.5.2 软件调试方法	77	4.4 数据库设计与管理	100
同步训练	78	4.4.1 数据库设计概述	101
第4章 数据库设计基础	80	4.4.2 需求分析	101
4.1 数据库系统的基本概念	82	4.4.3 概念设计	102
4.1.1 数据库、数据库管理系统与 数据库系统	82	4.4.4 逻辑设计	104
4.1.2 数据库技术的发展	85	4.4.5 物理设计	105
4.1.3 数据库系统的基本特点	86	4.4.6 数据库管理	105
4.1.4 数据库系统体系结构	87	同步训练	106
4.2 数据模型	88	附录	109
4.2.1 数据模型的基本概念	88	附录 A 全国计算机等级考试二级公共基础 知识考试大纲(最新版)	109
4.2.2 E-R 模型	89	附录 B 同步训练参考答案	110

第1章

数据结构与算法



视频课堂

第1课	什么是数据结构 ●数据的逻辑结构 ●数据的存储结构	第2课	栈及其基本运算 ●栈的定义 ●栈的特点 ●栈的基本运算
第3课	二叉树及其基本性质 ●二叉树的定义 ●满二叉树和完全二叉树 ●二叉树的基本性质 ●二叉树的遍历		

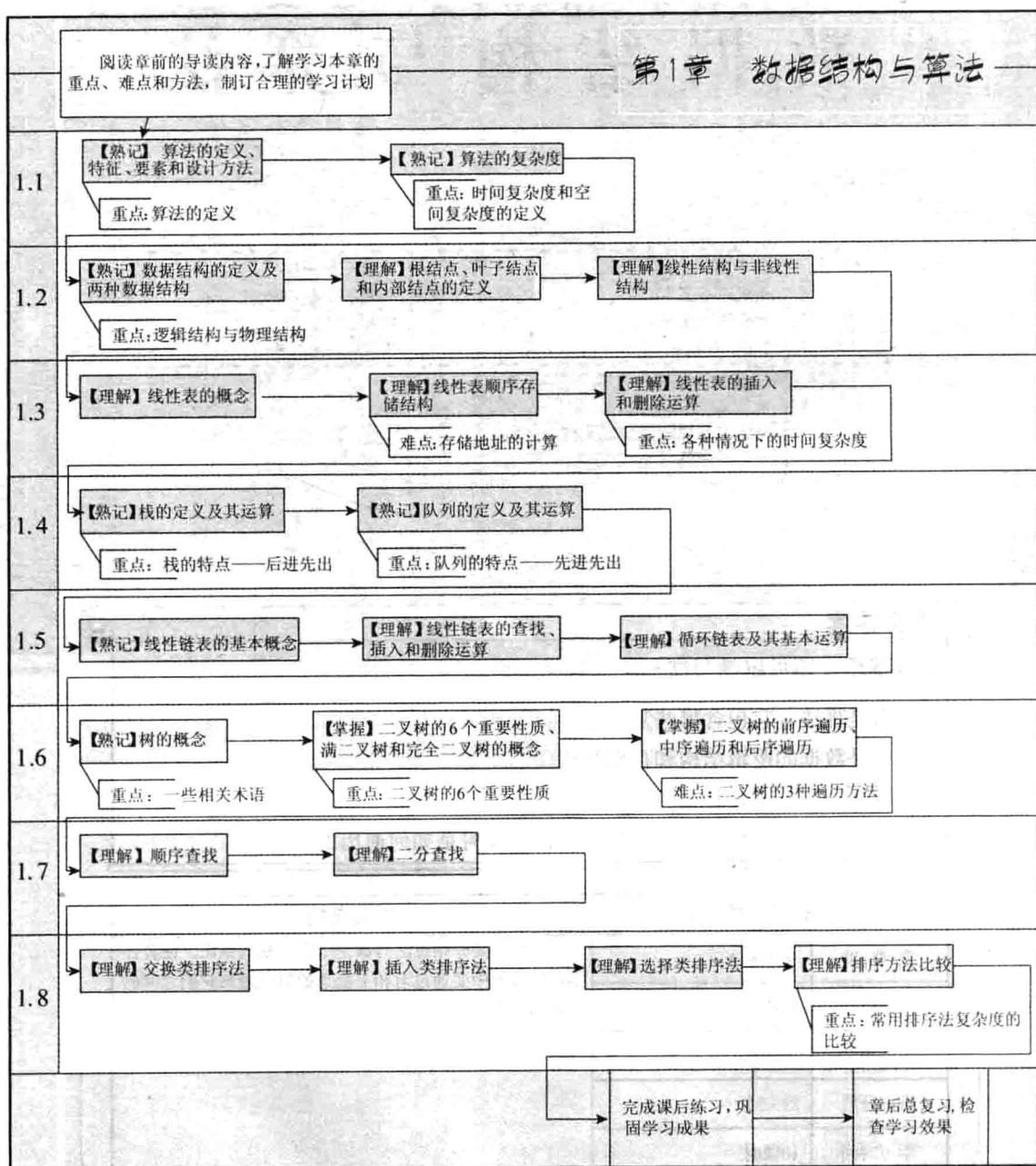
章前导读

通过本章，你可以学习到：

- 什么是算法，它包含哪些复杂度
- 什么是数据的逻辑结构和存储结构
- 栈和队列的定义是什么
- 二叉树的定义是什么，有哪些性质，二叉树是如何遍历的

本章评估		学习点拨
重要度	★★★★	本章主要介绍算法与数据结构的基础知识。读者在学习的过程中要通过对相关概念的对比理解它们之间的区别和联系。
知识类型	理论	
考核类型	选择题	
所占分值	约3.5分	
学习时间	10课时	

本章学习流程图



1.1 算 法

本节从算法的基本概念展开,阐述算法的基本特征、基本要素、设计方法以及设计准则,进而详细讲解算法的时间复杂度和空间复杂度。

1.1.1 什么是算法

1 算法的定义

有的学者认为,算法是程序的灵魂。实际上,对于算法的研究已经有数千年的历史了。计算机的出现,使得用机器自动解题的梦想成为现实,人们可以将算法编写成程序交给计算机执行,使许多原来认为不可能完成的算法变得实际可行。



学习提示

【熟记】算法的定义及2个基本要求
【理解】算法基本设计的6种方法和4个准则

算法 是指对解决方案的准确而完整的描述,简单地说,就是解决问题的操作步骤。

值得注意的是,算法不等于数学上的计算方法,也不等于程序。在用计算机解决实际问题时,往往先设计算法,用某种表达方式(如流程图)描述,然后再用具体的程序设计语言描述此算法(即编程)。在编程时由于要受到计算机系统运行环境的限制,因此,程序的编制通常不可能优于算法的设计。

2 算法的基本特征

(1) 可行性

算法在特定的执行环境中执行应当能够得出满意的结果,即必须有一个或多个输出。一个算法,即使在数学理论上是正确的,但如果在实际的计算工具上不能执行,则该算法也是不具有可行性的。

例如,在进行数值计算时,如果某计算工具具有7位有效数字(如程序设计语言中的单精度运算),则在计算下列3个量的和时:

$$A=10^{12}, B=1, C=-10^{12}$$

如果采用不同的运算顺序,就会得到不同的结果,例如:

$$A+B+C=10^{12}+1+(-10^{12})=0$$

$$A+C+B=10^{12}+(-10^{12})+1=1$$

而在数学上, $A+B+C$ 与 $A+C+B$ 是完全等价的。因此,算法与计算公式是有差别的。在设计一个算法时,必须考虑它的可行性。

(2) 确定性

算法的确定性表现在对算法中每一步的描述都是明确的,没有多义性,只要输入和初始状态相同,则无论执行多少遍,所得的结果都应该相同。如果算法的某个步骤有多义性,则该算法将无法执行。

例如,在进行汉字读音辨认时,汉字“解”在“解放”中读作jiě,但它作为姓氏时却

读作 xiè, 这就是多义性, 如果算法中存在多义性, 计算机将无法正确地执行。

(3) 有穷性

算法中的操作步骤为有限个, 且每个步骤都能在有限时间内完成。这包括合理的执行时间的含义, 如果一个算法执行耗费的时间太长, 即使最终得出了正确结果, 也是没有意义的。

例如, 数学中的无穷级数, 当 n 趋向于无穷大时, 求 $2n \times n!$, 显然, 这是无终止的计算, 这样的算法是没有意义的。

(4) 拥有足够的情报

一般来说, 算法在拥有足够的输入信息和初始化信息时, 才是有效的; 当提供的情报不够时, 算法可能无效。

例如, $A=3, B=5$, 求 $A+B+C$ 的值, 显然由于对 C 没有进行初始化, 无法计算出正确的答案, 所以算法在拥有足够的输入信息和初始化信息时, 才是有效的。

在特殊情况下, 算法也可以没有输入。因此, 一个算法有 0 个或多个输入。

总之, 算法是一个动态的概念, 是指一组严谨地定义运算顺序或操作步骤的规则, 并且, 每一个规则都是有效的、明确的, 此顺序将在有限的次数下终止。

3 算法的基本要素

算法的功能取决于两方面因素: 选用的操作和各个操作之间的顺序。因此, 一个算法通常由两种基本要素组成:

- 对数据对象的运算和操作;
- 算法的控制结构, 即运算或操作间的顺序。

(1) 算法中对数据对象的运算和操作

前面介绍了算法的一般定义和基本特征。实际上讨论的算法, 主要是指计算机算法。在计算机上, 可以直接执行的基本操作通常都是用指令来描述的, 每个指令代表一种或几种操作。

指令系统 指一个计算机系统能执行的所有指令的集合。

指令系统是软件与硬件分界的一个主要标志, 是软件与硬件之间沟通的桥梁。指令系统在计算机系统中的地位如图 1-1 所示。

算法就是按解题要求从指令系统中选择合适的指令组成的指令序列。因此, 计算机算法就是计算机能执行的操作所组成的指令序列。不同计算机系统, 指令系统是有差异的, 但一般的计算机系统, 都包括以下 4 类基本的运算和操作, 如表 1-1 所示。

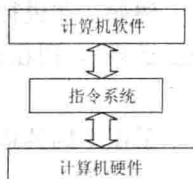


图 1-1 计算机的体系结构

表 1-1 4 类基本的运算和操作

运算类型	操 作	例 子
算术运算	+, -, ×, ÷	$a+b, 3-1\cdots$
逻辑运算	与 (&), 或 (), 非 (!)	$!1, 1 1, 0, 1&1\cdots$
关系运算	>, <, =, ≠	$a>b, a=c, b\neq c\cdots$
数据传输	赋值、输入、输出	$a=0, b=3\cdots$

(2) 算法的控制结构

算法的控制结构是算法中各个操作之间的执行顺序。

算法一般由顺序、选择（又称分支）和循环（又称重复）3种基本结构组合而成。描述算法的工具具有传统的流程图、N-S结构化流程图和算法描述语言等。

图1-2所示是用流程图方式表示的选择结构的两种类型。

图1-2(a)的执行步骤如下所述。

●步骤1 X赋值为2。

●步骤2 判断X的值是否小于3，条件成立。

●步骤3 X的值减少2。

●步骤4 输出X的值，最后结果为0。

图1-2(b)的执行步骤如下所述。

●步骤1 X赋值为2。

●步骤2 X的值增加2。

●步骤3 判断X的值是否小于3，条件不成立。

●步骤4 输出X的值，最后结果为4。

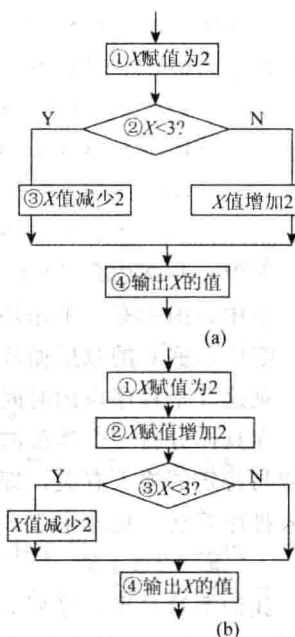
图1-2(a)执行的是先判断X的值是否小于3，如果条件成立，则X的值减2，最终结果为0，而图1-2(b)先将X的值增加2，然后再判断X的值是否小于3，最终结果为4。

从中可以看出，选用的基本操作虽然相同，但由于存在执行顺序的差异，得到的结果完全不同。

4 算法基本设计方法

虽然设计算法是一件非常困难的工作，但是算法设计也不是无章可循的，人们经过实践，总结和积累了许多行之有效的方法。常用的几种算法设计方法有列举法、归纳法、递推法、递归法、减半递推技术和回溯法。

图1-2 算法的控制结构



1.1.2 算法复杂度

一个算法的复杂度高低体现在运行该算法所需要的计算机资源的多少，所需的资源越多，就说明该算法的复杂度越高；反之，所需的资源越少，则该算法的复杂度越低。计算机的资源，最重要的是时间和空间（即存储器）资源。

因此，算法复杂度包括算法的时间复杂度和算法的空间复杂度。

1 算法的时间复杂度

算法的时间复杂度是指执行算法所需要的计算工作量。

值得注意的是：算法程序执行的具体时间和算法的时间复杂度并不是一致的。算法程序执行的具体时间受到所使用的计算机、程序设计语言以及算法实现过程中的许多细节所影响，而算法的时间复杂度与这些因素无关。

算法的计算工作量是用算法所执行的基本运算次数来度量的，而算法所执行的基本运算

学习提示

【熟记】算法的时间复杂度和空间复杂度各自的概念及度量

次数是问题规模（通常用整数 n 表示）的函数，即

$$\text{算法的工作量} = f(n)$$

其中， n 为问题的规模。

所谓问题的规模就是问题的计算量的大小。如 $1+2$ ，这是规模比较小的问题，但 $1+2+3+\dots+10\,000$ ，这就是规模比较大的问题。

例如，在下列 3 个程序段中：

① $\{x++; s=0\}$

② $\text{for } (i=1; i \leq n; i++)$

$\{x++; s+=x;\}$ /* 一个简单的 for 循环，循环体内操作执行了 n 次 */

③ $\text{for}(i=1; i \leq n; i++)$

$\text{for}(j=1; j \leq n; j++)$

$\{x++; s+=x;\}$ /* 嵌套的双层 for 循环，循环体内操作执行了 n^2 次 */

①中，基本运算“ $x++$ ”只执行一次，重复执行次数分别为 1；

②中，由于有一个循环，所以基本运算“ $x++$ ”执行了 n 次；

③中，嵌套的双层循环，所以基本运算“ $x++$ ”执行了 n^2 次。

则这 3 个程序段的时间复杂度分别为 $O(1)$ 、 $O(n)$ 和 $O(n^2)$ 。

在具体分析一个算法的工作量时，在同一个问题规模下，算法所执行的基本运算次数还可能与特定的输入有关，即输入不同时，算法所执行的基本运算次数不同。例如，使用简单插入排序算法（见本书 1.8 节），对输入序列进行从小到大排序。输入序列为：

a. 1 2 3 4 5

b. 1 3 2 5 4

c. 5 4 3 2 1

我们不难看出，序列 a 所需的计算工作量最少，因为它已经是非递减顺序排列，而序列 c 将耗费的基本运算次数最多，因为它完全是递减顺序排列的。

在这种情况下，可以用以下两种方法来分析算法的工作量：

- 平均性态；
- 最坏情况复杂性。

请思考



算法的复杂度是以什么来度量的？

2 算法的空间复杂度

算法的空间复杂度	是指执行这个算法所需要的内存空间。
----------	-------------------

算法执行期间所需的存储空间包括 3 个部分：

- 输入数据所占的存储空间；
- 程序本身所占的存储空间；
- 算法执行过程中所需要的额外空间。

其中，额外空间包括算法程序执行过程中的工作单元以及某种数据结构所需要的附加存储空间。

如果额外空间量相对于问题规模（即输入数据所占的存储空间）来说是常数，即额外空间量不随问题规模的变化而变化，则称该算法是原地（In Place）工作的。

为了降低算法的空间复杂度，主要应减少输入数据所占的存储空间以及额外空间，通常采用压缩存储技术。

1.2 数据结构的基本概念

在进行数据处理时，实际需要处理的数据一般有很多，而这些大量的数据都需要存放在计算机中，那么大量的数据在计算机中如何组织才能提高数据处理的效率，节省计算机的存储空间呢？

通过本节的学习，可以了解什么是数据结构，它们是如何用图形表示的，以及线性结构与非线性结构的区别。



学习提示

【熟记】数据结构研究的3方面内容：数据逻辑结构与物理结构的概念

1.2.1 什么是数据结构

数据结构研究的内容包括3个方面：

- 数据集合中各数据元素之间所固有的逻辑关系，即数据的逻辑结构；
- 在对数据进行处理时，各数据元素在计算机中的存储关系，即数据的存储结构；
- 对各种数据结构进行的运算。

其中，数据元素是一个含义很广泛的概念。它是数据的“基本单位”，在计算机中通常作为一个整体进行考虑和处理。在数据处理领域中，每一个需要处理的对象，甚至于客观事物的一切个体，都可以抽象成数据元素，简称为元素。

例如：

- 日常生活中一日三餐的名称——早餐、午餐、晚餐，可以作为一日三餐的数据元素；
- 在地理学中表示方向的方向名称——东、南、西、北，可以作为方向的数据元素；
- 在军队中表示军职的名称——连长、排长、班长、战士，可以作为军职的数据元素。

如果要给数据结构（Data Structure）下一个完整而准确的定义，那将是一件非常困难的事情。对数据结构的概念，在不同的书中，有不同的提法，顾名思义，所谓数据结构，包含两个要素，即“数据”和“结构”。

数据	是需要处理的数据元素的集合，一般来说，这些数据元素具有某个共同的特征。
----	-------------------------------------

例如，东、南、西、北这4个数据元素都有一个共同的特征，它们都是地理方向名，分别表示二维地理空间中的4个方向，这4个数据元素构成了地理方向名的集合。

又例如，早餐、午餐、晚餐这3个数据元素也有一个共同的特征，即它们都是一日三餐的名称，从而构成了一日三餐名的集合。

结构	所谓“结构”，就是关系，是集合中各个数据元素之间存在的某种关系（或联系）。
----	---------------------------------------

“结构”是数据结构研究的重点。数据元素根据其之间的不同特性关系，通常可以分为以下4类：线性结构 [见图 1-3(a)]、树形结构 [见图 1-3(b)]、网状结构 [见图 1-3(c)] 和集合 [见图 1-3(d)]。

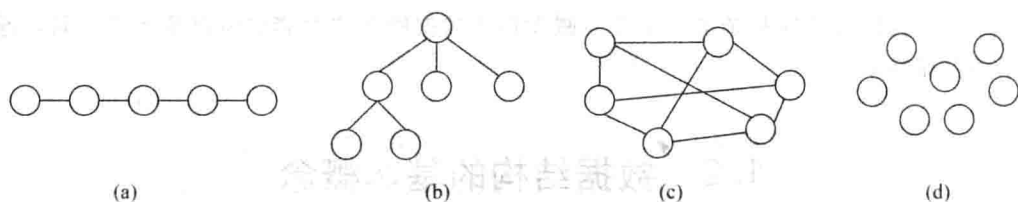


图 1-3 4 类基本结构

在数据处理领域中，通常把两两数据元素之间的关系用前后件关系（或直接前驱与直接后继关系）来描述。实际上，数据元素之间的任何关系都可以用前后件关系来描述。

例如，在考虑一日三餐的时间顺序关系时，“早餐”是“午餐”的前件（或直接前驱），而“午餐”是“早餐”的后件（或直接后继）；同样，“午餐”是“晚餐”的前件，“晚餐”是“午餐”的后件。

又例如，在考虑军队中的上下级关系时，“连长”是“排长”的前件，“排长”是“连长”的后件；“排长”是“班长”的前件，“班长”是“排长”的后件；“班长”是“战士”的前件，“战士”是“班长”的后件。

前后件关系是数据元素之间最基本的关系。

综上所述，数据结构是指相互有关联的数据元素的集合。换句话说，如果各个数据元素之间是有关联的，我们就说，这个数据元素的集合是有“结构”的。

数据结构的两个要素——“数据”和“结构”是紧密联系在一起的，“数据”是有结构的数据，而不是无关联的、松散的数据；而“结构”，就是数据元素间的关系，是由数据的特性所决定的。

1 数据的逻辑结构

前面提到“结构”这个词时，我们解释为关系。数据元素之间的关系，可以分为逻辑关系和在计算机中存储时产生的位置关系两种。相应的，数据结构分为数据的逻辑结构和数据的存储结构。

由数据结构的定义得知，一个数据结构应包含以下两方面的信息：

- 表示数据元素的信息；
- 表示各数据元素之间的前后件关系的信息。

在此定义中，并没有考虑数据元素的存储，所以上述的数据结构实际上是数据的逻辑结构。

数据的逻辑结构 指反映数据元素之间逻辑关系（即前后件关系）的数据结构。

数据的逻辑结构的数学形式定义——数据结构是一个二元组：

$$B = (D, R)$$

其中， B 表示数据结构， D 是数据元素的集合， R 是 D 上关系的集合，它反映了 D 中各数据元素之间的前后件关系，前后件关系也可以用一个二元组来表示。

例如，如果把一日三餐看作一个数据结构，则可表示成

$$B = (D, R)$$

$$D = \{\text{早餐, 午餐, 晚餐}\}$$

$$R = \{(\text{早餐, 午餐}), (\text{午餐, 晚餐})\}$$

又例如, 部队军职的数据结构, 可表示成

$$B=(D, R)$$

$$D=\{\text{连长, 排长, 班长, 战士}\}$$

$$R=\{(\text{连长, 排长}), (\text{排长, 班长}), (\text{班长, 战士})\}$$

2 数据的存储结构

数据的存储结构 又称为数据的物理结构, 是数据的逻辑结构在计算机存储空间中的存放方式。

由于数据元素在计算机存储空间中的位置关系可能与逻辑关系不同, 因此, 为了表示存储在计算机存储空间中的各数据之间的逻辑关系 (即前后件关系), 在数据的存储结构中, 不仅要存放各数据元素的信息, 还需要存入各数据元素之间的前后件关系的信息。

各数据元素在计算机存储空间中的位置关系与它们的逻辑关系不一定是相同的。

例如, 在前面提到的一日三餐的数据结构中, “早餐”是“午餐”的前件, “午餐”是“早餐”的后件, 但在对它们进行处理时, 在计算机存储空间中, “早餐”这个数据元素的信息不一定被存储在“午餐”这个数据元素信息的前面, 可能在后面, 也可能不是紧邻在前面, 而是中间被其他的信息所隔开。

下面介绍两种最主要的数据存储方式。

(1) 顺序存储结构

这种存储方式主要用于线性的数据结构, 它把逻辑上相邻的数据元素存储在物理上相邻的存储单元里, 结点之间的关系由存储单元的邻接关系来体现。

例如, 线性表 $(K_1, K_2, K_3, K_4, K_5)$, 假定每个结点占一个存储单元, 结点 K_1 存放在 200 号单元中, 则顺序存储实现如图 1-4(a) 所示。从图中可以看到, 逻辑上相邻的结点在物理存储中也互相邻接。如结点 K_3 逻辑上紧跟在结点 K_2 的后面, 而在物理 202, 也是紧跟在 K_2 的存放单元 201 的后面。

(2) 链式存储结构

链式存储结构就是在每个结点中至少包含一个指针域, 用指针来体现数据元素之间在逻辑上的联系。

例如, 线性表 $(K_1, K_2, K_3, K_4, K_5)$ 可以用链式存储, 如图 1-4(b) 所示。从图中可以看到, 结点之间的逻辑关系是通过指针来体现的。如结点 K_3 的存放单元并不紧跟在结点 K_2 的存放单元后面, 但 K_2 中有一个指针指示 K_3 的存放地址。

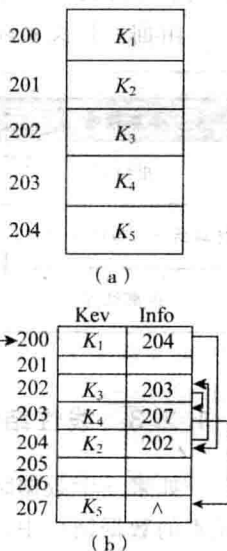


图 1-4 线性表

(a) 顺序存储的线性表;

(b) 链式存储的线性表

请思考



在哪一类存储结构中, 数据的逻辑结构与物理结构相一致?

1.2.2 数据结构的图形表示

数据元素之间最基本的关系是前后件关系。前后件关系, 即每一个二元组, 都可以用图

形来表示。用中间标有元素值的方框表示数据元素，一般称之为数据结点，简称为结点。对于每一个二元组，用一条有向线段从前件指向后件。

例如，一日三餐的数据结构可以用如图 1-5(a)所示的图形来表示。

又例如，军职数据结构可以用如图 1-5(b)所示的图形来表示。

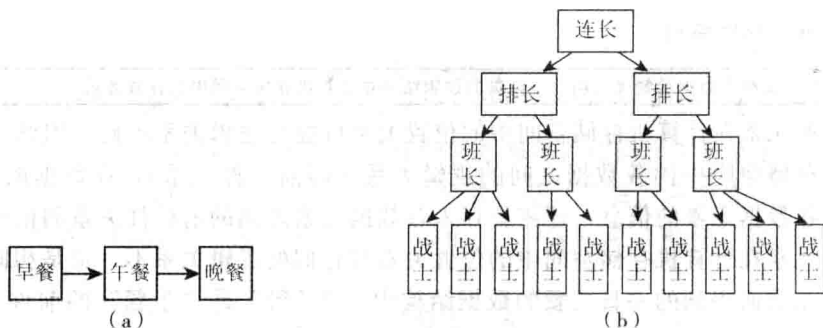


图 1-5 数据结构的图形表示

(a) 一日三餐数据结构；(b) 军职数据结构

用图形表示数据结构具有直观易懂的特点，在不引起歧义的情况下，前件结点到后件结点连线上的箭头可以省去。例如，在树形结构中，通常都是用无向线段来表示前后件关系的。

由前后件关系还可引出以下 3 个基本概念，如表 1-2 所示。

表 1-2 结点基本概念

基本概念	含 义	例 子
根结点	数据结构中，没有前件的结点	在图 1-5(a)中，“早餐”是根结点；在图 1-5(b)中，“连长”是根结点
终端结点（或叶子结点）	没有后件的结点	在图 1-5(a)中，“晚餐”是终端结点；在图 1-5(b)中，“战士”是终端结点
内部结点	数据结构中，除了根结点和终端结点以外的结点，统称为内部结点	在图 1-5(a)中，“午餐”是内部结点；在图 1-5(b)中，“排长”和“班长”是内部结点

1.2.3 线性结构与非线性结构

如果一个数据结构中没有数据元素，则称该数据结构为空的数据结构。在只有一个数据元素的数据结构中，删除该数据元素，就得到一个空的数据结构。

根据数据结构中各数据元素之间前后件关系的复杂程度，一般将数据结构划分为两大类型：线性结构和非线性结构，如表 1-3 所示。

表 1-3 线性结构与非线性结构

基本概念	含 义	例 子
线性结构	一个非空的数据结构如果满足以下两个条件： ● 有且只有一个根结点； ● 每一个结点最多有一个前件，也最多有一个后件	图 1-5(a) 一日三餐数据结构
非线性结构	不满足以上两个条件的数据结构就称为非线性结构，非线性结构主要是指树形结构和网状结构	图 1-5(b) 军职数据结构

学习提示

【理解】区分线性结构与非线性结构