

王道考研系列

2013年 数据结构 联考复习指导

2013NIAN
SHUJUEGOU
LIANKAO FUXI ZHIDAO

●王道论坛 组编

更多计算机
考研和学习交流
尽在www.cskaoan.com

王道论坛由清华、北大、浙大、上交、哈工大、中科院等名校计算机研究生共同创办，致力于给报考计算机专业研究生的考生提供帮助和指导。王道论坛是唯一一家专注于名校计算机考研的论坛。

王道论坛组编的一系列计算机考研辅导书，融入了论坛众多名校高分选手的智慧，以及论坛交流精华，我们希望将其打造成“书本+在线”的学习方式。对于书中的疑难点，欢迎大家在论坛交流和讨论。

目前已有越来越多的名校采用上机的形式考查考生的动手编程能力，为方便大家练习，我们搭建了九度OJ(ac.jobdu.com)，收录了全国各大高校的复试上机题，希望能给考生复习上机考试提供强有力的支持。



中南大学出版社
www.csupress.com.cn

王道考研系列

2013 年数据结构 联考复习指导

王道论坛 组 编

主要作者介绍

刘莹：

本科毕业于华中科技大学，网名“lanbeiliu”，本科专业金融学，2010 年跨专业考入清华大学计算机科学与技术系，初试状元，初试总分 423 分。

郑家祥：

本科毕业于南昌大学，2011 年考入浙江大学计算机科学与技术学院，初试总分 396 分，数据结构与算法基础扎实。

官水旺：

本科毕业于南京财经大学，2011 年考入南京大学计算机科学与技术系，初试总分 376，专业理论基础扎实。



中南大学出版社
www.csupress.com.cn

内 容 简 介

.....

本书严格按照最新计算机考研大纲的数据结构部分,对大纲所涉及的知识进行集中梳理,力求内容精炼、重点突出、深入浅出。本书精选了名校历年考研真题,并给出详细的解题思路,力求达到讲练结合、灵活掌握、举一反三的功效。创新的“书本+在线”的学习方式,网上答疑,通过本书可大大提高考生的复习效果,达到事半功倍的复习效率。

本书可作为考生参加计算机专业研究生入学考试的备考复习用书,也可作为计算机专业的学生学习数据结构课程的辅导用书。

图书在版编目(CIP)数据

2013 年数据结构联考复习指导/王道论坛组编.

—长沙:中南大学出版社,2012.7

ISBN 978-7-5487-0570-3

I. 2... II. 王... III. 数据结构—研究生—入学考试—
自学参考资料 IV. TP311.12

中国版本图书馆 CIP 数据核字(2012)第 158461 号

2013 年数据结构联考复习指导

王道论坛 组编

☐ 责任编辑 陈海波

☐ 责任印制 文桂武

☐ 出版发行 中南大学出版社

社址:长沙市麓山南路

邮编:410083

发行科电话:0731-88876770

传真:0731-88710482

☐ 印 装 长沙市宏发印刷有限公司

☐ 开 本 787×1092 1/16 ☐ 印张 19 ☐ 字数 474 千字

☐ 版 次 2012 年 7 月第 1 版 ☐ 2012 年 8 月第 3 次印刷

☐ 书 号 ISBN 978-7-5487-0570-3

☐ 定 价 39.00 元

图书出现印装问题,请与经销商调换

序

当前,随着我国经济和科技高速发展,特别是计算机科学突飞猛进的发展,对计算机相关人才,尤其是中高端人才的需求也将不断增长。硕士研究生入学考试可视为人生的第二次大考试,它是改变命运、实现自我理想的又一次机会,而计算机专业一直是高校考研的热门专业之一。

自计算机专业研究生入学考试实行统一命题以来,初试科目包含了最重要的四门基础课程(数据结构、计算机组成原理、操作系统、计算机网络),很多学生普遍反映找不到方向,复习也无从下手。倘若有一套能够指导考生如何复习的好书,必将对考生的帮助匪浅。我的学生风华他们策划和编写了这一系列的计算机专业考研辅导书,重点突出,层次分明。他们结合了自身的复习经验、理解深度以及对大纲把握程度的体会,对考生而言是很有启发和指导意义的。

计算机这门学科,任何机械式的死记硬背都是收效甚微的。在全面深入复习之后,首先对诸多知识点分清主次,并结合做题,灵活运用所掌握的知识点,再选择一些高质量的模拟试题来检测自己理解和掌握的程度,查漏补缺。这符合我执教40余年来一直坚持“教材-习题集-试题库”的教学体系。

从风华他们策划并组建编写团队到初稿成型,直至最后定稿,我能体会到风华和他的团队确实倾注了大量的精力。这本书的出版一定会受到广大考研学生的欢迎,它会使你在考研的路上得到强有力的帮助。

序

前 言

2011年，由王道论坛组织名校高分选手，编写了4本单科辅导书。单科书是基于王道之前作品的二代作品，不论是编排方式，还是内容质量上都较前一版本的王道书有了较大的提升。这套书也参考了同类优秀的教材和辅导书，更是结合了高分选手们自己的复习经验。无论是对考点的讲解，还是习题的选择和解析，都结合了他们对专业课复习的独特见解。“王道考研系列”单科书，一共4本：

《2013年数据结构联考复习指导》

《2013年计算机组成原理联考复习指导》

《2013年操作系统联考复习指导》

《2013年计算机网络联考复习指导》

2011年的单科书由于是第一年出版，时间较为仓促，小错误相对较多，给读者的复习带来了一些不便。今年，我们不仅修正了去年发现的全部错误，还对考点讲解做出了尽可能的优化，也重新筛选了部分习题，尤其是对习题的解析做出了更好的改进。

综观最近4年的考研试题，难度越来越高、题型越来越灵活，因此考取高分也越来越难。对于报考名校的考生，尤其是跨专业的考生来说，普遍会认为计算机专业课范围广、难度大，考题灵活。而对于一个想继续在计算机专业领域深造的考生来说，认真学习和扎实掌握这4门计算机专业中最基础的专业课，是最基本的前提。

当然，掌握专业课的知识点没有捷径可言，考生也不应怀有任何侥幸心理，扎扎实实地打好基础、踏踏实实地做题巩固，最后灵活致用才是高分的保障。我们只希望这套书能够指导大家复习考研，但学习还是得靠自己，高分不是建立在任何空中楼阁之上的。

“王道考研系列”的特色是“书本+在线”，你在复习中遇到的任何困难，都可以在王道论坛上发帖，论坛的热心道友，以及辅导员都会积极参与并与你交流。你的参与就是对我们最大的鼓舞，任何一个建议，我们都会认真考虑，也会针对大家的意见对本书进行修订。

我们虽然尽最大努力来保证本书质量，但由于编写的时间仓促，以及编者的水平有限，书中如有错误或任何不当之处，望广大读者指正，我们将及时改正。

予人玫瑰，手有余香，王道论坛伴你一路同行！

王道论坛

目 录

第1章 绪 论	(1)
1.1 数据结构的基本概念	(1)
1.1.1 基本概念和术语	(1)
1.1.2 数据结构的三要素	(2)
1.1.3 本节试题精选	(3)
1.1.4 答案与解析	(4)
1.2 算法和算法评价	(5)
1.2.1 算法的基本概念	(5)
1.2.2 算法效率的度量	(5)
1.2.3 本节试题精选	(6)
1.2.4 答案与解析	(8)
第2章 线性表	(10)
2.1 线性表的定义和基本操作	(10)
2.1.1 线性表的定义	(10)
2.1.2 线性表的基本操作	(11)
2.1.3 本节试题精选	(11)
2.1.4 答案与解析	(11)
2.2 线性表的顺序表示	(12)
2.2.1 顺序表的定义	(12)
2.2.2 顺序表上基本操作的实现	(13)
2.2.3 本节试题精选	(15)
2.2.4 答案与解析	(17)
2.3 线性表的链式表示	(24)
2.3.1 单链表的定义	(24)
2.3.2 单链表上基本操作的实现	(25)
2.3.3 双链表	(29)
2.3.4 循环链表	(30)
2.3.5 静态链表	(31)
2.3.6 顺序表和链表的比较	(32)
2.3.7 本节试题精选	(33)
2.3.8 答案与解析	(37)

第3章 栈和队列	(54)
3.1 栈	(54)
3.1.1 栈的基本概念	(54)
3.1.2 栈的顺序存储结构	(55)
3.1.3 栈的链式存储结构	(57)
3.1.4 本节试题精选	(58)
3.1.5 答案与解析	(60)
3.2 队列	(67)
3.2.1 队列的基本概念	(67)
3.2.2 队列的顺序存储结构	(68)
3.2.3 队列的链式存储结构	(70)
3.2.4 双端队列	(72)
3.2.5 本节试题精选	(73)
3.2.6 答案与解析	(75)
3.3 栈和队列的应用	(79)
3.3.1 栈在括号匹配中的应用	(79)
3.3.2 栈在表达式求值中的应用	(79)
3.3.3 栈在递归中的应用	(80)
3.3.4 队列在层次遍历中的应用	(81)
3.3.5 队列在计算机系统中的应用	(82)
3.3.6 本节试题精选	(83)
3.3.7 答案与解析	(84)
3.4 特殊矩阵的压缩存储	(88)
3.4.1 数组的定义	(88)
3.4.2 数组的存储结构	(89)
3.4.3 矩阵的压缩存储	(89)
3.4.4 稀疏矩阵	(92)
3.4.5 本节试题精选	(92)
3.4.6 答案与解析	(93)
第4章 树与二叉树	(94)
4.1 树的基本概念	(95)
4.1.1 树的定义	(95)
4.1.2 基本术语	(95)
4.1.3 树的性质	(96)
4.1.4 本节试题精选	(96)
4.1.5 答案与解析	(97)
4.2 二叉树的概念	(98)

4.2.1	二叉树的定义及其主要特性	(98)
4.2.2	二叉树的存储结构	(100)
4.2.3	本节试题精选	(102)
4.2.4	答案与解析	(103)
4.3	二叉树的遍历和线索二叉树	(107)
4.3.1	二叉树的遍历	(107)
4.3.2	线索二叉树	(110)
4.3.3	本节试题精选	(113)
4.3.4	答案与解析	(116)
4.4	树、森林	(133)
4.4.1	树的存储结构	(133)
4.4.2	树、森林与二叉树的转换	(135)
4.4.3	树和森林的遍历	(136)
4.4.4	树的应用——并查集	(136)
4.4.5	本节试题精选	(138)
4.4.6	答案与解析	(140)
4.5	树与二叉树的应用	(144)
4.5.1	二叉排序树	(144)
4.5.2	平衡二叉树	(148)
4.5.3	哈夫曼(Huffman)树和哈夫曼编码	(151)
4.5.4	本节试题精选	(153)
4.5.5	答案与解析	(156)
第5章	图	(164)
5.1	图的基本概念	(164)
5.1.1	图的定义	(164)
5.1.2	本节试题精选	(168)
5.1.3	答案与解析	(169)
5.2	图的存储及基本操作	(171)
5.2.1	邻接矩阵法	(171)
5.2.2	邻接表法	(173)
5.2.3	图的基本操作	(174)
5.2.4	本节试题精选	(175)
5.2.5	答案与解析	(177)
5.3	图的遍历	(180)
5.3.1	广度优先搜索(Breadth First Search, BFS)	(180)
5.3.2	深度优先搜索(Depth First Search, DFS)	(182)
5.3.3	图的遍历与图的连通性	(183)
5.3.4	本节试题精选	(184)

5.3.5 答案与解析	(186)
5.4 图的应用	(191)
5.4.1 最小生成树(Minimum Spanning Tree, MST)	(191)
5.4.2 最短路径	(194)
5.4.3 拓扑排序	(197)
5.4.4 关键路径	(199)
5.4.5 本节试题精选	(200)
5.4.6 答案与解析	(204)
第6章 查找	(214)
6.1 查找的基本概念	(214)
6.2 顺序查找和折半查找	(215)
6.2.1 顺序查找	(215)
6.2.2 折半查找	(217)
6.2.3 分块查找	(219)
6.2.4 本节试题精选	(220)
6.2.5 答案与解析	(222)
6.3 B树和B+树	(227)
6.3.1 B树及其基本操作	(227)
6.3.2 B+树基本概念	(230)
6.3.3 本节试题精选	(231)
6.3.4 答案与解析	(232)
6.4 散列(Hash)表	(236)
6.4.1 散列表的基本概念	(236)
6.4.2 散列函数的构造方法	(236)
6.4.3 处理冲突的方法	(237)
6.4.4 散列查找及性能分析	(238)
6.4.5 本节试题精选	(239)
6.4.6 答案与解析	(242)
第7章 排序	(248)
7.1 排序的基本概念	(249)
7.1.1 排序的定义	(249)
7.1.2 本节试题精选	(249)
7.1.3 答案与解析	(250)
7.2 插入排序	(250)
7.2.1 直接插入排序	(250)
7.2.2 折半插入排序	(251)
7.2.3 希尔排序	(252)

7.2.4	本节试题精选	(253)
7.2.5	答案与解析	(255)
7.3	交换排序	(256)
7.3.1	冒泡排序	(256)
7.3.2	快速排序	(257)
7.3.3	本节试题精选	(259)
7.3.4	答案与解析	(261)
7.4	选择排序	(266)
7.4.1	简单选择排序	(266)
7.4.2	堆排序	(267)
7.4.3	本节试题精选	(270)
7.4.4	本节试题精选	(271)
7.5	归并排序和基数排序	(275)
7.5.1	归并排序	(275)
7.5.2	基数排序	(277)
7.5.3	本节试题精选	(277)
7.5.4	答案与解析	(278)
7.6	各种内部排序算法的比较及应用	(279)
7.6.1	内部排序算法的比较	(279)
7.6.2	内部排序算法的应用	(280)
7.6.3	本节试题精选	(281)
7.6.4	答案与解析	(282)
7.7	外部排序	(284)
7.7.1	外部排序的基本概念	(285)
7.7.2	外部排序的方法	(285)
7.7.3	多路平衡归并与败者树	(286)
7.7.4	置换-选择排序(生成初始归并段)	(288)
7.7.5	最佳归并树	(289)
7.7.6	本节试题精选	(290)
7.7.7	答案与解析	(291)
参考文献		(294)

第1章 绪 论

【复习要点】

- (一) 数据结构相关的概念和术语
- (二) 数据结构的三要素：逻辑结构、物理结构和数据运算
- (三) 算法时间复杂度和空间复杂度的分析与计算

【考题分析】

年份	单选题/分	综合题/分	考查内容
2010 年	0	√ ^①	算法题中分析所写算法的时间复杂度和空间复杂度
2011 年	1 题 × 2	√	分析给定程序段的时间复杂度； 算法题中分析所写算法的时空复杂度
2012 年	1 题 × 2	√	分析给定程序段的时间复杂度； 算法题中分析所写算法的时间复杂度

① 打“√”表示有综合应用题部分涉及本章的知识点。

本章内容并不在考研大纲中，它是数据结构的一个概述，读者应通过对本章的学习初步了解数据结构的基本内容。时间复杂度和空间复杂度是本章重点，属必考内容，一定要熟练掌握，每年都会结合算法设计题而出现。特别是，2011 年和 2012 年还直接以选择题的形式考查了时间复杂度的计算。掌握本章的基本概念对后续章节的学习是非常重要的。

1.1 数据结构的基本概念

1.1.1 基本概念和术语

1. 数据

数据是信息的载体，是描述客观事物属性的数、字符以及所有能输入到计算机中并被计算机程序识别和处理的符号的集合。

2. 数据元素

数据元素是数据的基本单位，在计算机程序中通常作为一个整体进行考虑和处理。一个数据元素可由若干个数据项组成。数据项是构成数据元素的不可分割的最小单位。数据

项是对客观事物某一方面特性的数据描述。例如,学生记录就是一个数据元素,它由学号、姓名、性别等数据项组成。

注意:不要混淆数据、数据元素、数据项之间的概念,也要注意和数据库中的相关术语进行区别:如数据记录、数据字段等概念。

3. 数据对象

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。

4. 数据类型

数据类型是一个值的集合和定义在此集合上一组操作的总称。

- 1) 原子类型:其值不可再分的数据类型。
- 2) 结构类型:其值可以再分解为若干成分(分量)的数据类型。
- 3) 抽象数据类型:抽象数据组织和与之相关的操作。

5. 抽象数据类型

抽象数据类型(ADT)是指一个数学模型以及定义在该模型上的一组操作。抽象数据类型的定义仅取决于它的一组逻辑特性,而与其在计算机内部如何表示和实现无关,即不论其内部结构如何变化,只要它的数学特性不变,都不影响其外部的使用。通常用(数据对象、数据关系、基本操作集)这样的三元组来表示抽象数据类型。

6. 数据结构

在任何问题中,数据元素都不是孤立存在的,而是在它们之间存在着某种关系,这种数据元素相互之间的关系称为**结构(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。数据结构包括三方面的内容:数据的逻辑结构、数据的存储结构和数据的运算。数据的逻辑结构和存储结构是密不可分的两个方面,一个算法的设计取决于所选定的逻辑结构,而算法的实现依赖于所采用的存储结构^①。

1.1.2 数据结构的三要素

1. 数据的逻辑结构

逻辑结构是指数据元素之间的逻辑关系,即从逻辑关系上描述数据。它与数据的存储无关,是独立于计算机的。数据的逻辑结构分为线性结构和非线性结构,线性表是典型的线性结构;集合、树和图是典型的非线性结构。数据的逻辑结构分类见图1-1。

- 1) 集合:结构中的数据元素之间除了“同属于一个集合”的关系外,别无其他关系。
- 2) 线性结构:结构中的数据元素之间存在一对一的关系。
- 3) 树形结构:结构中的数据元素之间存在一对多的关系。
- 4) 图状结构或网状结构:结构中的数据元素之间存在多对多的关系。

2. 数据的存储结构

存储结构是指数据结构在计算机中的表示(又称映像),也称物理结构。它包括数据元素的表示和关系的表示。数据的存储结构是逻辑结构用计算机语言的实现,它依赖于计算机语言。数据的存储结构主要有:顺序存储、链式存储、索引存储和散列存储。

- 1) 顺序存储:把逻辑上相邻的元素存储在物理位置上也相邻的存储单元里,元素之间

^① 读者应通过后续章节的学习,逐步理解设计与实现的概念与区别。

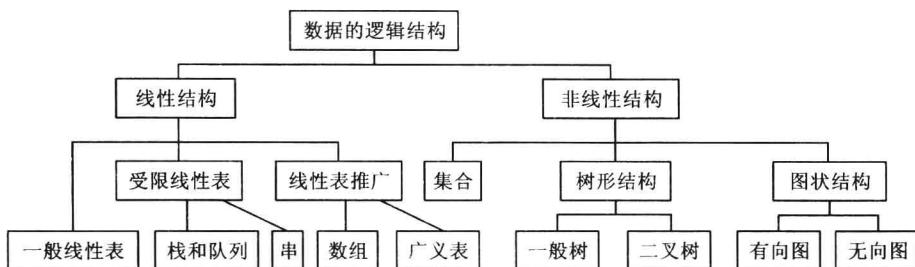


图 1-1 数据的逻辑结构分类图

的关系由存储单元的邻接关系来体现。其优点是可实现随机存取，每个元素占用最少的存储空间；缺点是只能使用相邻的一整块存储单元，因此可能产生较多的碎片现象。

2) 链接存储：不要求逻辑上相邻的元素在物理位置上也相邻，借助指示元素存储地址的指针表示元素之间的逻辑关系。其优点是不会出现碎片现象，充分利用所有存储单元；缺点是每个元素占用较多的存储空间，并且只能实现顺序存取。

3) 索引存储：通常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)。其优点是检索速度快；缺点是增加了附加的索引表，会占用较多的存储空间。另外，在增加和删除数据时由于要修改索引表，因而会花费较多时间。

4) 散列存储：根据结点的关键字直接计算出该结点的存储地址，又称为 Hash 存储。其优点是检索、增加和删除结点的操作都很快；缺点是如果散列函数不好可能出现结点存储单元的冲突，而解决冲突会增加时间和空间开销。

3. 数据的运算

施加在数据上的运算包括运算的定义和实现。运算定义是针对逻辑结构的，指出运算的功能；运算的实现是针对存储结构的，指出运算的具体操作步骤。

1.1.3 本节试题精选

一、单项选择题

1. 可以用()定义一个完整的数据结构。
A. 数据元素 B. 数据对象 C. 数据关系 D. 抽象数据类型
2. 以下数据结构中，()是非线性数据结构。
A. 树 B. 字符串 C. 队列 D. 栈
3. 以下属于逻辑结构的是()。
A. 顺序表 B. 哈希表 C. 有序表 D. 单链表
4. 以下与数据的存储结构无关的术语是()。
A. 循环队列 B. 链表 C. 哈希表 D. 栈
5. 连续存储分配设计时，存储单元的地址()。
A. 一定连续 B. 一定不连续
C. 不一定连续 D. 部分连续，部分不连续

6. 在存储数据时,通常不仅要存储各数据元素的值,而且还要存储()。

- A. 数据的操作方法
- B. 数据元素的类型
- C. 数据元素之间的关系
- D. 数据的存取方法

7. 链式存储设计时,结点内的存储单元地址()。

- A. 一定连续
- B. 一定不连续
- C. 不一定连续
- D. 部分连续,部分不连续

二、综合应用题

1. 对于两种不同的数据结构,逻辑结构或物理结构一定不相同吗?

2. 试举一例,说明对相同的逻辑结构,同一种运算在不同的存储方式下实现,其运算效率不同。

1.1.4 答案与解析

一、单项选择题

1. D

抽象数据类型描述了数据的逻辑结构和抽象运算,从而构成了一个完整的数据结构定义。

2. A

树和图是典型的非线性数据结构,其他选项都属于线性数据结构。

3. C

顺序表、哈希表和单链表都是存储结构。而有序表指关键字有序的线性表,仅描述结点之间的逻辑关系,既可以链式存储也可以顺序存储。

4. D

存储结构有顺序存储、链式存储、索引存储和散列(Hash)存储。栈是一种抽象数据类型,可采用顺序存储或链式存储。而循环队列是用顺序表表示的队列。

5. A

连续存储分配不一定是顺序存储,也可能是静态链表,但地址的分配一定是连续的。

6. C

存储数据时,不仅要存储数据元素的值,还要存储数据元素之间的关系。

7. A

链式存储设计时,各个不同结点的存储空间可以不连续,但是结点内的存储单元地址则必须连续。

二、综合应用题

1. 解答:

应该注意到,数据的运算也是数据结构的一个重要方面。

对于两种不同的数据结构,它们的逻辑结构和物理结构完全有可能相同。比如二叉树和二叉排序树,二叉排序树可以采用二叉树的逻辑表示和存储方式,前者通常用来表示层次关系,而后者通常用于排序和查找。虽然它们的运算都有建立树、插入结点、删除结点和查找结点等,但是对于二叉树和二叉排序树,这些运算的定义是不同的,以查找结点为

例, 二叉树的时间复杂度为 $O(n)$, 而二叉排序树的时间复杂度为 $O(\log_2 n)$ 。

2. 解答:

线性表既可以用顺序存储方式实现, 也可以用链式存储方式实现。在顺序存储方式下, 线性表中插入和删除操作, 平均要移动近一半的元素, 时间复杂度为 $O(n)$; 而在链式存储方式下, 插入和删除的时间复杂度都是 $O(1)$ 。

1.2 算法和算法评价

1.2.1 算法的基本概念

算法(algorithm)是对特定问题求解步骤的一种描述, 它是指令的有限序列, 其中每一条指令表示一个或多个操作。此外, 一个算法还具有下列 5 个重要特性:

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, 读者理解时不会产生二义性。并且, 在任何条件下, 算法只有唯一的一条执行路径, 即对于相同的输入只能得出相同的输出。

3) 可行性: 一个算法是可行的, 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。

4) 输入: 一个算法有零个或多个的输入, 这些输入取自于某个特定的对象的集合。

5) 输出: 一个算法有一个或多个的输出, 这些输出是同输入有着某种特定关系的量。

通常设计一个“好”的算法应考虑达到以下目标:

1) 正确性: 算法应当能够正确地解决求解问题。

2) 可读性: 算法应当具有良好的可读性, 以助于人们理解。

3) 健壮性: 当输入非法数据时, 算法也能适当地做出反应或进行处理, 而不会产生莫名其妙的输出结果。

4) 效率与低存储量需求: 效率是指算法执行的时间, 存储量需求是指算法执行过程中所需要的最大存储空间, 这两者都与问题的规模有关。

1.2.2 算法效率的度量

算法效率的度量是通过时间复杂度和空间复杂度来描述的。

1. 时间复杂度

一个语句的频度是指该语句在算法中被重复执行的次数。算法中所有语句的频度之和记作 $T(n)$, 它是该算法问题规模 n 的函数, 时间复杂度主要分析 $T(n)$ 的数量级。算法中的基本运算(最深层循环内的语句)的频度与 $T(n)$ 同数量级, 所以通常采用算法中基本运算的频度 $f(n)$ 来分析算法的时间复杂度。因此, 算法的时间复杂度也记为:

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

上式中“ O ”的含义是 $T(n)$ 的数量级, 其严格的数学定义是: 若 $T(n)$ 和 $f(n)$ 是定义在正整数集合上的两个函数, 则存在正常数 C 和 n_0 , 使得当 $n \geq n_0$ 时, 都满足 $0 \leq T(n) \leq C \times f(n)$ 。

算法的时间复杂度不仅依赖于问题的规模 n ，也取决于待输入数据的性质（如输入数据元素的初始状态）。

例如 在数组 $A[0 \dots n-1]$ 中，查找给定值 K 的算法大致如下：

```
(1) i = n - 1;
(2) while(i >= 0 && (A[i] != k))
(3)     i --;
(4) return i;
```

此算法中的语句(3)的频度不仅与问题规模 n 有关，还与输入实例中 A 的各元素取值及 K 的取值有关：

- ① 若 A 中没有与 K 相等的元素，则语句(3)的频度 $f(n) = n$ 。
- ② 若 A 的最后一个元素等于 K ，则语句(3)的频度 $f(n)$ 是常数 0。

最坏时间复杂度是指在最坏情况下，算法的时间复杂度。

平均时间复杂度是指所有可能输入实例在等概率出现的情况下，算法的期望运行时间。

最好时间复杂度是指在最好情况下，算法的时间复杂度。

一般总是考虑在最坏的情况下的时间复杂度，以保证算法的运行时间不会比它更长。在分析一个程序的时间复杂性时，有以下两条规则：

① 加法规则

$$T(n) = T_1(n) + T_2(n) = O(f(n)) + O(g(n)) = O(\max(f(n), g(n)))$$

② 乘法规则

$$T(n) = T_1(n) \times T_2(n) = O(f(n)) \times O(g(n)) = O(f(n) \times g(n))$$

常见的渐近时间复杂度有：

$$O(1) < O(\log_2 n) < O(n) < O(n \log_2 n) < O(n^2) < O(n^3) < O(2^n) < O(n!) < O(n^n)$$

2. 空间复杂度

算法的空间复杂度 $S(n)$ ，定义为该算法所耗费的存储空间，它是问题规模 n 的函数。渐近空间复杂度也常简称为空间复杂度，记作 $S(n) = O(g(n))$ 。

一个上机程序除了需要存储空间来存放本身所用指令、常数、变量和输入数据外，也需要一些对数据进行操作的工作单元和存储一些为实现计算所需信息的辅助空间，若输入数据所占空间只取决于问题本身，和算法无关，则只需分析除输入和程序之外的额外空间。

算法原地工作是指算法所需辅助空间是常量，即 $O(1)$ 。

1.2.3 本节试题精选

一、单项选择题

1. 一个算法应该是()。
 - A. 程序
 - B. 问题求解步骤的描述
 - C. 要满足五个基本特性
 - D. A 和 C
2. 某算法的时间复杂度为 $O(n^2)$ ，表明该算法的()。

4. 以下算法的时间复杂度为()。

}

- 5.【2011 年计算机联考真题】设 n 是描述问题规模的非负整数, 下面程序片段的时间复杂度是()。

A. $O(\log_2 n)$ B. $O(n)$ C. $O(n \log_2 n)$ D. $O(n^2)$

- 1

- ## 7. 程序段

其中 n 为正整数, 则最后一行的语句频度在最坏情况下是()。

8. 以下算法中加下划线语句的执行次数为()。

A. $n(n+1)$ B. n C. $n+1$ D. n^2

9. 下面说法错误的是()。