

王道考研系列

2012版 数据结构 联考复习指导

2012BAN
SHUJU JIEGOU LIANKAO
FUXI ZHIDAO


王道论坛 组编 / 王轩 主审

王道论坛由清华大学、北京大学、浙江大学、哈尔滨工业大学、上海交通大学、中科院等国内计算机专业名校的研究生共同创办，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，拥有众多热心的会员。

王道论坛组编的一系列计算机专业考研辅导书，融入了论坛众多名校考研高分考生的智慧，以及论坛交流的精华帖内容，我们希望将其打造成“书本+在线”的学习方式，大家对于书中的疑难点，欢迎在论坛上交流和讨论，我们也会尽最大努力满足每一位读者的需求。

更多计算机
考研和学习交流
尽在www.cskaoan.com



 机械工业出版社
CHINA MACHINE PRESS

王道考研系列

数据结构联考复习指导（2012版）

王道论坛 组编

郑家祥 赵霖 官水旺 编

王轩 主审



机械工业出版社

本书严格按照全国计算机学科专业考研大纲的数据结构部分的内容,对大纲所涉及的知识点进行集中梳理,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,并给出详细解题思路和算法,力求达到讲练结合、灵活掌握、举一反三的功效。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

数据结构联考复习指导:2012版/王道论坛组编.—北京:机械工业出版社,2011.8(2011.9重印)

(王道考研系列)

ISBN 978-7-111-35412-3

I. ①数… II. ①王… III. ①数据结构—研究生—入学考试—自学参考资料 IV. ①TP311.12

中国版本图书馆CIP数据核字(2011)第146723号

机械工业出版社(北京市百万庄大街22号 邮政编码100037)

策划编辑:吉玲 责任编辑:吉玲 郭娟 黄金荣

封面设计:鞠杨 责任印制:李妍

北京振兴源印务有限公司印刷

2011年9月第1版第4次印刷

184mm×260mm·16.75印张·413千字

标准书号:ISBN 978-7-111-35412-3

定价:35.80元

凡购本书,如有缺页、倒页、脱页,由本社发行部调换
电话服务 网络服务

社服务中心:(010) 88361066

销售一部:(010) 68326294

销售二部:(010) 88379649

读者购书热线:(010) 88379203

门户网:<http://www.cmpbook.com>

教材网:<http://www.cmpedu.com>

封面无防伪标均为盗版

前 言

2010年,由王道论坛组编的《2011年计算机专业基础综合考试指导全书》(以下简称《指导全书》)赢得了广大计算机考生的喜爱,让我们倍感欣慰的同时也感觉压力重重。不少读者觉得《指导全书》中的一些讲解不够细致、习题的数量太少。考虑到《指导全书》的内容篇幅不宜太大,为了更好地对每个知识点进行深入地归纳和讲解,今年我们组织了名校高分考生编写了“王道考研系列”的单科辅导书,一共四本:

- 《数据结构联考复习指导(2012版)》
- 《计算机组成原理联考复习指导(2012版)》
- 《计算机网络联考复习指导(2012版)》
- 《计算机操作系统联考复习指导(2012版)》

这套书不仅参考了王道论坛之前的作品《指导全书》和《真题精析》,也参考了同类的优秀教材和辅导书,更结合了高分考生们自己的复习经验。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。

综观最近3年的考研试题,考试难度越来越高、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深造的考生来说,认真学习和扎实掌握这四门计算机专业中最基础的专业课,是最基本的前提。

当然,掌握专业课的学习要点没有捷径可言,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,以及编者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,我们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。

专业课不仅仅用笔试,目前已有越来越多的名校采用上机的形式,来考查考生的实际动手编程能力。为了方便大家练习机试,我们搭建了网络平台——九度OJ(ac.jobdu.com),收集了全国各大高校的复试上机真题,希望能给考生复习上机考试提供强有力的支持。

参加本书编写的人员有:郑家祥,赵霖,官水旺。

予人玫瑰,手有余香,王道论坛伴你一路同行!

目 录

前言

第1章 绪论	1
1.1 数据结构的基本概念	1
1.1.1 考点精析	1
1.1.2 试题精选	3
1.2 算法和算法评价	4
1.2.1 考点精析	4
1.2.2 试题精选	5
答案与解析	7
1.1.2 试题精选	7
1.2.2 试题精选	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 考点精析	10
2.1.2 试题精选	11
2.2 线性表的顺序表示	11
2.2.1 考点精析	11
2.2.2 试题精选	14
2.3 线性表的链式表示	16
2.3.1 考点精析	16
2.3.2 试题精选	24
答案与解析	28
2.1.2 试题精选	28
2.2.2 试题精选	28
2.3.2 线性表的链式表示	34
第3章 栈和队列	52
3.1 栈	52
3.1.1 考点精析	52
3.1.2 试题精选	55
3.2 队列	58
3.2.1 考点精析	58
3.2.2 试题精选	62
3.3 栈和队列的应用	64
3.3.1 考点精析	64
3.3.2 试题精选	67

3.4 特殊矩阵的压缩存储	68
3.4.1 考点精析	68
3.4.2 试题精选	72
答案与解析	72
3.1.2 试题精选	72
3.2.2 试题精选	78
3.3.2 试题精选	81
3.4.2 试题精选	85
第4章 树与二叉树	86
4.1 树的基本概念	86
4.1.1 考点精析	86
4.1.2 试题精选	88
4.2 二叉树的概念	88
4.2.1 考点精析	88
4.2.2 试题精选	91
4.3 二叉树的遍历和线索二叉树	93
4.3.1 考点精析	93
4.3.2 试题精选	99
4.4 树、森林	103
4.4.1 考点精析	103
4.4.2 试题精选	107
4.5 树与二叉树的应用	108
4.5.1 考点精析	108
4.5.2 试题精选	116
答案与解析	119
4.1.2 试题精选	119
4.2.2 试题精选	120
4.3.2 试题精选	123
4.4.2 试题精选	139
4.5.2 试题精选	142
第5章 图	149
5.1 图的基本概念	149
5.1.1 考点精析	149
5.1.2 试题精选	152
5.2 图的存储及基本操作	154
5.2.1 考点精析	154
5.2.2 试题精选	157
5.3 图的遍历	159
5.3.1 考点精析	159
5.3.2 试题精选	163

5.4	图的应用	165
5.4.1	考点精析	165
5.4.2	试题精选	175
	答案与解析	178
5.1.2	试题精选	178
5.2.2	试题精选	180
5.3.2	试题精选	182
5.4.2	试题精选	186
第 6 章	查 找	194
6.1	查找的基本概念	194
6.1.1	考点精析	194
6.2	顺序查找和折半查找	195
6.2.1	考点精析	195
6.2.2	试题精选	199
6.3	B 树和 B+树	202
6.3.1	考点精析	202
6.3.2	试题精选	206
6.4	散列表	207
6.4.1	考点精析	207
6.4.2	试题精选	210
	答案与解析	212
6.2.2	试题精选	212
6.3.2	B 树和 B+树	218
6.4.2	试题精选	221
第 7 章	排 序	225
7.1	排序的基本概念	225
7.1.1	考点精析	225
7.1.2	试题精选	226
7.2	插入排序	226
7.2.1	考点精析	226
7.2.2	试题精选	229
7.3	交换排序	231
7.3.1	考点精析	231
7.3.2	试题精选	233
7.4	选择排序	236
7.4.1	考点精析	236
7.4.2	试题精选	238
7.5	归并排序和基数排序	240
7.5.1	考点精析	240
7.5.2	试题精选	242

7.6 各种内部排序算法的比较及应用.....	243
7.6.1 考点精析.....	243
7.6.2 试题精选.....	245
答案与解析.....	246
7.1.2 试题精选.....	246
7.2.2 试题精选.....	246
7.3.2 试题精选.....	248
7.4.2 试题精选.....	253
7.5.2 试题精选.....	256
7.6.2 试题精选.....	257
参考文献.....	260

绪论

- (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和计算。

年份	单选题/分	综合题/分	考查内容
2011年	2题×1	0	计算给定程序段的时间复杂度

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 考点精析

1. 基本概念和术语

(1) 数据

数据是对客观事物的符号表示。在计算机科学中是指所有能输入到计算机中并被计算机程序处理的符号的总称。

(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) 数据结构

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。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**逻辑结构**。数据结构在计算机中的表示（又称映像）称为数据的**物理结构**，又称**存储结构**。数据的逻辑结构和物理结构是密不可分两个方面，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。

数据结构的三要素：逻辑结构、存储结构和数据运算。

(5) 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。

(6) 抽象数据类型

抽象数据类型是指一个数学模型以及定义在该模型上的一组操作，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，即不论其内容结构如何变化，只要它数学特性不变，都不影响其外部的使用。

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图见图 1-1。

- 1) **线性结构** 结构中的数据元素之间存在一对一的关系。
- 2) **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。
- 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。

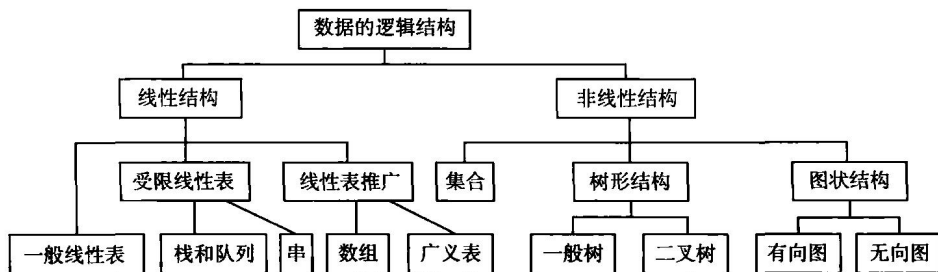


图 1-1 数据的逻辑结构分类图

(2) 数据的物理结构

物理结构又称存储结构，是数据结构在计算机中的实际表示方式，它包括对数据元素的表示和对关系的表示。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**：把逻辑上相邻的结点存储在物理位置上也相邻的存储单元里，结点之间的关系由存储单元的邻接关系来体现。其优点是可实现随机存取，每个结点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结点在物理位置上也相邻，结点间的逻辑关系是由附

加的指针字段表示。其优点是不会出现碎片现象，充分利用所有存储单元；缺点是每个结点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

数据的运算是在数据的逻辑结构上定义的操作算法，如检索、插入、删除、更新和排序等。

1.1.2 试题精选

一、单项选择题

- 以下与数据的存储结构无关的术语是 ()。
 - 循环队列
 - 链表
 - 哈希表
 - 栈
- 以下属于逻辑结构的是 ()。
 - 顺序表
 - 哈希表
 - 有序表
 - 单链表
- 以下数据结构中，() 是非线性数据结构。
 - 树
 - 字符串
 - 队列
 - 栈
- 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 ()。
 - 数据的操作方法
 - 数据元素的类型
 - 数据元素之间的关系
 - 数据的存取方法
- 连续存储设计时，存储单元的地址 ()。
 - 一定连续
 - 一定不连续
 - 不一定连续
 - 部分连续，部分不连续
- 链式存储设计时，结点内的存储单元地址 ()。
 - 一定连续
 - 一定不连续
 - 不一定连续
 - 部分连续，部分不连续
- 可以用 () 定义一个完整的数据结构。
 - 数据元素
 - 数据对象
 - 数据关系
 - 抽象数据类型

二、综合应用题

- 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。
- 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？

1.2 算法和算法评价

1.2.1 考点精析

1. 算法的基本概念

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。

算法具有以下五个重要特性：

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

确定性：算法中每一条指令必须有确切的含义，确保不会产生二义性。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。

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实现的基本运算执行有限次来实现的。

输入性：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。

输出性：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。

评价好的算法应考虑达到以下四个目标：

正确性：算法应当能够正确地解决求解问题。

可读性：算法应当具有良好的可读性，便于人们使用。

健壮性：当输入非法数据时，算法也能适当地做出反应或处理，而不会出现莫名其妙的输出结果。

效率与低存储量的要求：效率是指算法的运行时间，存储量要求是指算法运行过程中所需要的最大存储空间。

2. 算法的评价

算法的效率评价通过时间复杂度和空间复杂度来描述。

(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 $T(n)$ ，它是该算法所求解问题规模 n 的函数。当问题的规模趋向无穷大时， $T(n)$ 的数量级（阶）称为渐近时间复杂度，简称时间复杂度，记作 $T(n)=O(f(n))$ 。

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

例如 在数组 $A[0..n-1]$ 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。

上述表达式中“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 $0 \leq T(n) \leq C \times f(n)$ 。

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。

另外, 由于算法的时间复杂度主要分析 $T(n)$ 的数量级, 而算法中的基本运算的频度与 $T(n)$ 同数量级, 所以通常采用算法中基本运算的频度来分析算法的时间复杂度, 一般是**最深层循环内**的语句被视为算法基本运算。

在分析一个程序的时间复杂性时, 有以下两条规则:

a) 加法规则

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

b) 乘法规则

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

常见的渐进时间复杂度有:

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

(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。渐进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 $O(1)$ 。

1.2.2 试题精选

一、单项选择题

- 一个算法应该是 ()。
 - 程序
 - 问题求解步骤的描述
 - 要满足五个基本特性
 - A 和 C
- 下面关于算法说法错误的是 ()。
 - 算法最终必须由计算机程序实现
 - 为解决某问题的算法与为该问题编写的程序含义是相同的
 - 算法的可行性是指指令不能有二义性
 - 以上几个都是错误的
- 算法的时间复杂度取决于 ()。
 - 问题的规模
 - 待处理数据的初态
 - 执行的次数
 - A 和 B
- 某算法的时间复杂度为 $O(n^2)$, 表明该算法的 ()。

- A. 问题规模是 n^2 B. 执行时间等于 n^2
 C. 执行时间与 n^2 成正比 D. 问题规模与 n^2 成正比
 5. 下面说法错误的是 ()。

- (1) 算法原地工作的含义是指不需要任何额外的辅助空间
 (2) 在相同的规模 n 下, 复杂度 $O(n)$ 的算法在时间上总是优于复杂度 $O(2^n)$ 的算法
 (3) 所谓时间复杂度是指最坏情况下, 估算算法执行时间的一个上界
 (4) 同一个算法, 实现语言的级别越高, 执行效率就越低

- A. (1) B. (1), (2)
 C. (1), (4) D. (3)

6. 【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 ()。

```
x=2;
while(x<n/2)
```

```
    x=2*x;
```

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

7. 以下算法的时间复杂度为 ()。

```
void fun(int n){
    int i=1;
    while(i<=n)
        i=i*2;
}
```

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

8. 程序段

```
for(i=n-1;i>1;i--){
    for(j=1;j<i;j++)
        if(A[j]>A[j+1])
```

A[j]与A[j+1]对换;

其中 n 为正整数, 则最后一行的语句频度在最坏情况下是 ()。

- A. $O(n)$ B. $O(n \log n)$
 C. $O(n^3)$ D. $O(n^2)$

9. 有以下算法, 其时间复杂度为 ()。

```
void fun(int n){
    int i=0;
    while(i+i+i<=n)
        i++;
}
```

- A. $O(n)$ B. $O(n \log n)$
 C. $O(\sqrt[3]{n})$ D. $O(\sqrt{n})$

10. 以下算法中加下划线语句的执行次数为 ()。

```
int m=0,i,j;
for(i=1;i<=n;i++)
    for(j=1;j<=2*i;j++)
        m++;
```

A. $n(n+1)$

B. n

C. $n+1$

D. n^2

二、综合应用题

1. 分析以下各程序段，求出算法的时间复杂度。

① $i=1;k=0;$

```
while(i<n-1){
    k=k+10*i;
    i++;
}
```

② $y=0;$

```
while((y+1)*(y+1)<=n)
    y=y+1;
```

③ $for(i=1;i<=n;i++)$

```
    for(j=1;j<=i;j++)
        for(k=1;k<=j;k++)
            x++;
```

④ $for(i=0;i<n;i++)$

```
    for(j=0;j<m;j++)
        a[i][j]=0;
```

2. 一个算法所需时间由下述递归方程表示，试求出该算法的时间复杂度的级别（或阶）。

$$T(n) = \begin{cases} 1 & \text{若 } n=1 \\ 2T(n/2)+n & \text{若 } n>1 \end{cases}$$

式中， n 是问题的规模，为简单起见，设 n 是 2 的整数幂。

答案与解析

1.1.2 试题精选

一、单项选择题

1. 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，而循环队列是首尾相连的顺序表。

2. 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。

3. A

树是典型的非线性数据结构，其他选项都属于线性数据结构。

4. 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。

5. A

连续存储不一定是顺序存储，也可能是静态链表，但地址的分配一定是连续的。

6. A

链式存储设计时, 各个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。

7. 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。

二、综合应用题

1. 解答:

线性表中的插入、删除操作, 在顺序存储方式下平均移动近一半的元素, 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 $O(1)$ 。

2. 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 $O(\log_2 n)$ 。

1.2.2 试题精选

一、单项选择题

1. B

程序不一定满足有穷性, 如死循环、操作系统等, 而算法必须有穷。算法代表了对问题的解, 而程序则是算法在计算机上的特定的实现。

2. D

A, 对算法没有此强制要求。B, 算法不等于程序。C, 可行性的解释错误。

3. D

算法的时间复杂度不仅取决于问题的规模, 还和待处理数据的初态有关。

4. C

时间复杂度是问题规模 n 的函数, 记为 $T(n)=O(f(n))$, $T(n)$ 的增长率与 $f(n)$ 的增长率相同。 $T(n)=O(n^2)$ 表示 $T(n)=m \times n^2$ (m 为常量), 其问题规模仍为 n 而不是 n^2 。

5. A

(1) 算法原地工作是指算法所需辅助空间是常量。

(2) 时间复杂度为 $O(n)$ 的算法, 必然总是优于复杂度为 $O(2^n)$ 的算法, 文中并非指程序具体的执行时间。

(3) 时间复杂度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。

(4) 对于同一个算法, 实现语言的级别越高, 执行效率就越低。

6. A

在程序中, 执行频率最高的语句为 “ $x=2*x$ ”。设该语句共执行了 t 次, 则 $2^{t+1}=n/2$, 故 $t=\log_2(n/2)-1=\log_2 n-2$, 得 $T(n)=O(\log_2 n)$ 。

7. D

基本运算是语句 $i=i*2$, 设其执行时间为 $T(n)$, 则有, $2^{T(n)} \leq n$, 即 $T(n) \leq \log_2 n \leq O(\log_2 n)$

8. D

当所有相邻元素都为逆序时, 则最后一行的语句每次都会执行。此时,

$$T(n) = \sum_{i=2}^{n-1} \sum_{j=1}^i 1 = \sum_{i=2}^{n-1} i = (n-2)(n+1)/2 = O(n^2)$$

所以在最坏情况下的该语句频度是 $O(n^2)$ 。

9. C

算法的基本运算是语句 $i++$, 设其执行时间为 $T(n)$, 则有, $T(n) \times T(n) \times T(n) \leq n$, 即 $T(n)^3 \leq n$ 。故有, $T(n) \leq \sqrt[3]{n} = O(\sqrt[3]{n})$ 。

10. A

$m++$ 语句的执行次数为

$$\sum_{i=1}^n \sum_{j=1}^{2i} 1 = \sum_{i=1}^n 2i = 2 \sum_{i=1}^n i = n(n+1)$$

二、综合应用题

1. 解答:

① 基本语句是 $k=k+10*i$, 共执行了 $n-2$ 次, 所以 $T(n)=O(n)$ 。

② 设循环体共执行 $T(n)$ 次, 每循环一次, 循环变量 y 加 1, 最终 $T(n)=y$ 。故 $(T(n))^2 \leq n$, 解得 $T(n)=O(n^{1/2})$ 。

③ $x++$ 是基本语句, $T(n)=O(\sum_{i=1}^n \sum_{j=1}^i \sum_{k=1}^j 1)=O(\frac{1}{6}n^3)=O(n^3)$ 。

④ $a[i][j]=0$ 是基本语句, 执行了 $m*n$ 次, 所以 $T(m, n)=O(m*n)$ 。

2. 解答:

时间复杂度为 $O(n \log_2 n)$ 。

设 $n=2^k (k \geq 0)$, 根据题目所给定义, 有 $T(2^k)=2T(2^{k-1})+2^k=2^2T(2^{k-2})+2 \times 2^k$, 由此, 可得一般递推公式 $T(2^k)=2^i T(2^{k-i})+i \times 2^k$, 进而, 可得 $T(2^k)=2^k T(2^0)+k \times 2^k=(k+1)2^k$, 即 $T(n)=n(\log_2 n+1)=n(\log_2(2n))=O(n \log_2 n)$ 。