

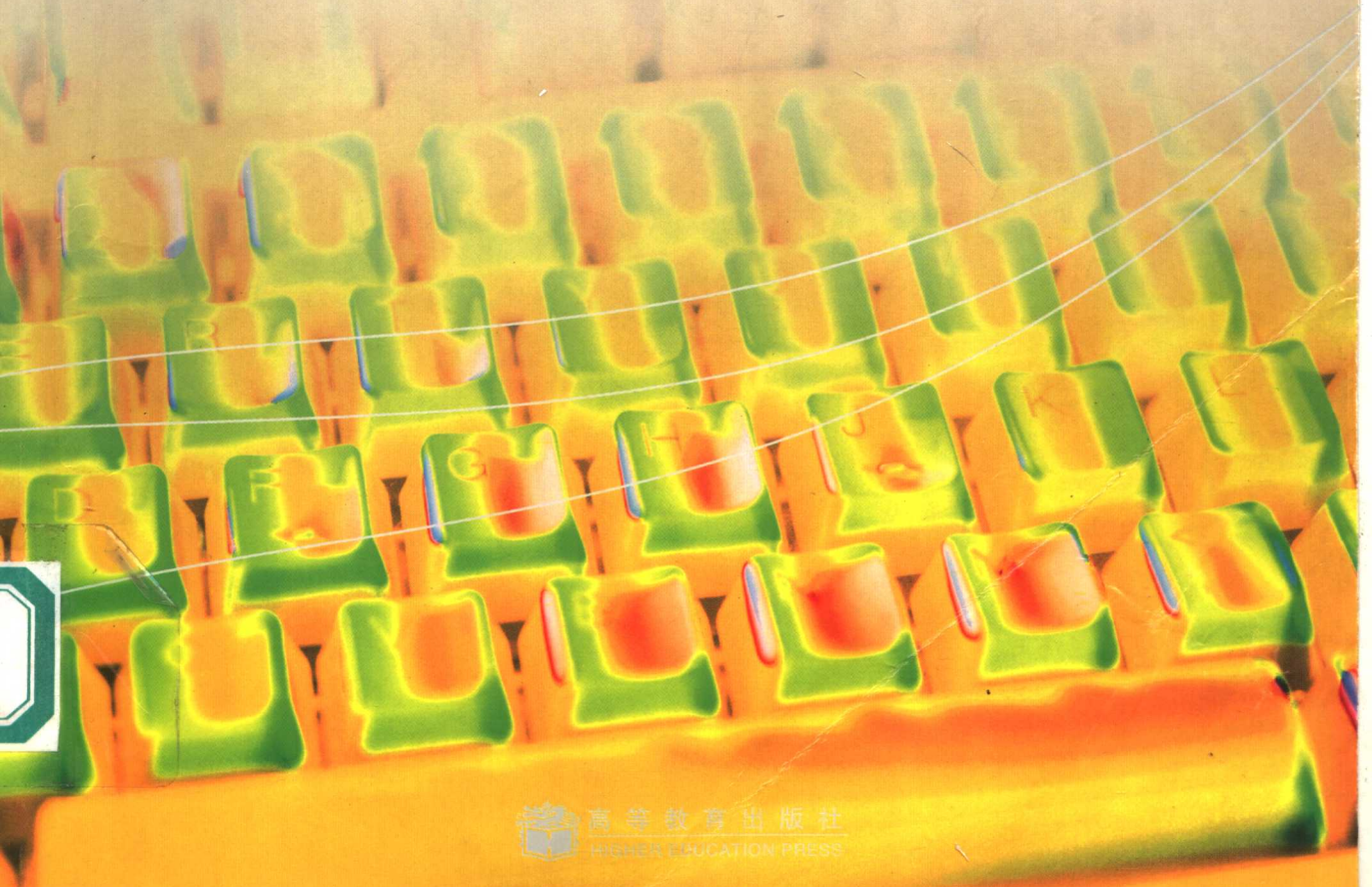
2004年版

全国计算机等级考试

考试要点、真题详解与训练

——二级 Access 数据库程序设计

主 编 欧 阳 副主编 曾令明



高等教育出版社
HIGHER EDUCATION PRESS

全国计算机等级考试(2004 年版)

考试要点、真题详解与训练

——二级 Access 数据库程序设计

主 编 欧 阳

副 主 编 曾令明

编写成员 吴艳玲 梁 伟 杨 勇 于俊乐
蒋维嘉 沙开波 傅 军 马增辉

高等教育出版社

内容提要

本书是根据《全国计算机等级考试考试大纲(2004 年版)》中的二级公共基础知识和 Access 数据库程序设计部分的要求编写的。本书分为文字部分和配套光盘,文字部分包括笔试部分和附录,其中笔试部分主要内容包括:公共基础知识(含数据结构与算法、程序设计基础、软件工程基础、数据库设计基础)、Access 数据库程序设计以及 4 套全真笔试试卷;附录 A 为上机模拟考试软件的安装步骤,附录 B 为相应的考试大纲。其中试题均选自历届考题,具有针对性强、内容不超纲等特点。笔试试卷的题型、题型比例和难易程度都与真题一致。

配套光盘为上机考试模拟软件,该软件与真考环境完全一致;并且与正式考试共用一个题库,题目命中率高。

本书可作为全国计算机等级考试二级 Access 数据库程序设计的考前指导书,同时也可作为 Access 程序设计课程的学习参考书。

图书在版编目(CIP)数据

全国计算机等级考试(2004 年版)考试要点、真题详解与训练——二级 Access 数据库程序设计/欧阳主编.
北京:高等教育出版社,2005.4

ISBN 7-04-016284-9

I. 全… II. 欧… III. 关系数据库-数据库管理系统, Access-程序设计-水平考试-自学参考资料 IV. TP311.138

中国版本图书馆 CIP 数据核字(2005)第 020830 号

出版发行 高等教育出版社
社 址 北京市西城区德外大街 4 号
邮政编码 100011
总 机 010-58581000

经 销 北京蓝色畅想图书发行有限公司
印 刷 北京市白帆印务有限公司

开 本 787×1092 1/16
印 张 18.75
字 数 450 000

购书热线 010-58581118
免费咨询 800-810-0598
网 址 <http://www.hep.edu.cn>
<http://www.hep.com.cn>
网上订购 <http://www.landaco.com>
<http://www.landaco.com.cn>

版 次 2005 年 4 月第 1 版
印 次 2005 年 4 月第 1 次印刷
定 价 35.40 元(含光盘)

本书如有缺页、倒页、脱页等质量问题,请到所购图书销售部门联系调换。

版权所有 侵权必究

物料号 16284-00

前 言

全国计算机等级考试是由教育部考试中心主办,用于考查应试人员计算机应用知识与能力的水平,可作为用人单位录用和考核工作人员的评价标准。参加全国计算机等级考试的人员主要包括在校大学生、机关公务员、部队官兵等,等级证书的获得与否与他们的就业、晋升有着密切的关系。

全国计算机等级考试自从1994年举办以来,得到了全国各行各业从事计算机工作与学习人士的积极参与及各用人单位的普遍认可,是目前国内影响最大、参加人数最多的计算机类考试。

为了使广大考生能顺利地通过全国计算机等级考试,编者根据《全国计算机等级考试考试大纲(2004年版)》的要求,结合多年从事全国计算机等级考试培训和基础教学工作的实践经验,编写了“全国计算机等级考试 考试要点、真题详解与训练”系列丛书。本套丛书共包括以下13本:

全国计算机等级考试 考试要点、真题详解与训练 —— 一级 MS Office

全国计算机等级考试 考试要点、真题详解与训练 —— 一级 B

全国计算机等级考试 考试要点、真题详解与训练 —— 二级 C 语言程序设计

全国计算机等级考试 考试要点、真题详解与训练 —— 二级 C++ 语言程序设计

全国计算机等级考试 考试要点、真题详解与训练 —— 二级 Java 语言程序设计

全国计算机等级考试 考试要点、真题详解与训练 —— 二级 Visual Basic 语言程序设计

全国计算机等级考试 考试要点、真题详解与训练 —— 二级 Visual FoxPro 数据库程序设计

全国计算机等级考试 考试要点、真题详解与训练 —— 二级 Access 数据库程序设计

全国计算机等级考试 考试要点、真题详解与训练 —— 三级 PC 技术

全国计算机等级考试 考试要点、真题详解与训练 —— 三级信息管理技术

全国计算机等级考试 考试要点、真题详解与训练 —— 三级数据库技术

全国计算机等级考试 考试要点、真题详解与训练 —— 三级网络技术

全国计算机等级考试 考试要点、真题详解与训练 —— 四级

丛书具有以下特色:

① 按考试大纲要求列举了考试的必备知识。

② 丛书中所列举的试题都选自历届全国计算机等级考试试题,按考试时间的顺序(由近及远)将历届考题归类。

③ 丛书中的试题都是真题,具有典型性和针对性。此外,还给出了试题分析和参考答案。

④ 丛书配有上机考试光盘。光盘中的上机考试系统与真考系统一致,其登录、抽题、答题、提交与正式上机考试一模一样。自动生成试卷,自动计时,自动评分,与正式考试共用一个题库,命中率高。

本套丛书将会使考生更易于理解全国计算机等级考试的基本要求和解答试题的思路,使读者在较短的时间内取得较大的收获,为参加等级考试和使用计算机打下良好的基础,为考生通过

考试增添一分把握。

下面介绍一下如何准备二级公共基础知识和 Access 数据库程序设计考试。

① 公共基础知识在笔试试卷中占 30 分,多数考题要求记忆。在本书的第 1 章中,根据考试大纲要求,将考试中心教材中的要点进行了归纳,并按考点设计了许多真题。考生只要掌握本书第 1 章的要点并将试题看懂、记住,就应该能满足公共基础知识部分的测试要求。

② Access 程序设计在笔试试卷中占 70 分,要求考生掌握 Access 的语法和简单算法。在此,通过对最新考试大纲和教材的分析和总结,设计了许多针对考试的真题,同时在相应章节还给出了考试要点。

③ Access 上机考试题型有基本操作题、简单应用题和综合应用题,考试时间是 90 分钟。其中基本操作题 1 道,占 30 分;简单应用题 1 道,占 40 分;综合应用题 1 道,占 30 分。上机考试有这样一个特点:有些考生很快就能交卷,而有些考生思考很久也做不出来。这是因为,有的考生平时重视上机操作,操作熟练,而有些考生平时很少上机练习,对上机考试也采取背的方式,所以很难通过考试,针对这种情况,我们开发了“全国计算机等级考试上机考试系统”并制作成光盘。以配合读者学习。

合理分配复习时间也很重要,一般认为笔试用 60% 的时间、上机用 40% 的时间比较合适。

总之,全国计算机等级考试其实不难,只要按照上面介绍的方法去做,通过率就会很高。最后祝广大考生顺利通过考试,获得证书。

本书第 1 章由欧阳编写,第 2 章由吴艳玲编写,第 3 章由梁伟编写,第 4 章由于俊乐编写,第 5 章由沙开波编写,第 6 章由蒋维嘉编写,第 7 章由傅军编写,第 8 章由马增辉编写,第 9 章由曾令明编写,第 10 章由杨勇编写。

由于时间仓促和作者水平有限,错误和不足之处在所难免,敬请读者指正。联系方式: guoxm3@vip.163.com。

编 者

2004 年 12 月

目 录

第1章 公共基础知识	(1)	2.3.1 考试必备知识	(81)
1.1 数据结构与算法	(1)	2.3.2 全真考题解答	(82)
1.1.1 考试必备知识	(1)	2.3.3 全真试题训练	(82)
1.1.2 全真考题解答	(7)	2.3.4 全真试题参考答案	(85)
1.1.3 全真试题训练	(14)	2.4 SQL 基本命令	(85)
1.1.4 全真试题参考答案	(18)	2.4.1 考试必备知识	(85)
1.2 程序设计基础	(19)	2.4.2 全真考题解答	(86)
1.2.1 考试必备知识	(19)	2.4.3 全真试题训练	(87)
1.2.2 全真考题解答	(21)	2.4.4 全真试题参考答案	(91)
1.2.3 全真试题训练	(26)	2.5 Access 简介及启动和关闭 Access	(91)
1.2.4 全真试题参考答案	(27)	2.5.1 考试必备知识	(91)
1.3 软件工程基础	(28)	2.5.2 全真考题解答	(93)
1.3.1 考试必备知识	(28)	2.5.3 全真试题训练	(93)
1.3.2 全真考题解答	(34)	2.5.4 全真试题参考答案	(95)
1.3.3 全真试题训练	(39)	第3章 数据库和表	(96)
1.3.4 全真试题参考答案	(43)	3.1 创建数据库	(96)
1.4 数据库设计基础	(43)	3.1.1 考试必备知识	(96)
1.4.1 考试必备知识	(43)	3.1.2 全真考题解答	(98)
1.4.2 全真考题解答	(47)	3.1.3 全真试题训练	(98)
1.4.3 全真试题训练	(53)	3.1.4 全真试题参考答案	(100)
1.4.4 全真试题参考答案	(56)	3.2 建立表	(100)
第2章 数据库基础知识	(57)	3.2.1 考试必备知识	(100)
2.1 数据库基本概念	(57)	3.2.2 全真考题解答	(102)
2.1.1 考试必备知识	(57)	3.2.3 全真试题训练	(105)
2.1.2 全真考题解答	(59)	3.2.4 全真试题参考答案	(110)
2.1.3 全真试题训练	(61)	3.3 维护表	(110)
2.1.4 全真试题参考答案	(68)	3.3.1 考试必备知识	(110)
2.2 关系数据库	(69)	3.3.2 全真考题解答	(112)
2.2.1 考试必备知识	(69)	3.3.3 全真试题训练	(112)
2.2.2 全真考题解答	(70)	3.3.4 全真试题参考答案	(113)
2.2.3 全真试题训练	(72)	3.4 操作表	(113)
2.2.4 全真试题参考答案	(80)	3.4.1 考试必备知识	(113)
2.3 数据库设计基础	(81)	3.4.2 全真考题解答	(115)

3.4.3 全真试题训练	(116)	5.1.3 全真试题训练	(141)
3.4.4 全真试题参考答案	(118)	5.1.4 全真试题参考答案	(143)
第4章 查询	(119)	5.2 创建窗体	(143)
4.1 认识查询	(119)	5.2.1 考试必备知识	(143)
4.1.1 考试必备知识	(119)	5.2.2 全真考题解答	(144)
4.1.2 全真考题解答	(120)	5.2.3 全真试题训练	(145)
4.1.3 全真试题训练	(121)	5.2.4 全真试题参考答案	(147)
4.1.4 全真试题参考答案	(124)	5.3 自定义窗体及美化窗体	(147)
4.2 创建选择查询及在查询中进行 计算	(124)	5.3.1 考试必备知识	(147)
4.2.1 考试必备知识	(124)	5.3.2 全真考题解答	(151)
4.2.2 全真考题解答	(125)	5.3.3 全真试题训练	(152)
4.2.3 全真试题训练	(125)	5.3.4 全真试题参考答案	(157)
4.2.4 全真试题参考答案	(127)	第6章 报表	(158)
4.3 创建交叉表查询和参数查询	(127)	6.1 报表的定义、组成和分类	(158)
4.3.1 考试必备知识	(127)	6.1.1 考试必备知识	(158)
4.3.2 全真考题解答	(127)	6.1.2 全真考题解答	(159)
4.3.3 全真试题训练	(128)	6.1.3 全真试题训练	(160)
4.3.4 全真试题参考答案	(129)	6.1.4 全真试题参考答案	(163)
4.4 创建操作查询	(129)	6.2 创建报表	(163)
4.4.1 考试必备知识	(129)	6.2.1 考试必备知识	(163)
4.4.2 全真考题解答	(130)	6.2.2 全真考题解答	(164)
4.4.3 全真试题训练	(131)	6.2.3 全真试题训练	(165)
4.4.4 全真试题参考答案	(132)	6.2.4 全真试题参考答案	(166)
4.5 创建 SQL 查询	(132)	6.3 编辑报表	(166)
4.5.1 考试必备知识	(132)	6.3.1 考试必备知识	(166)
4.5.2 全真考题解答	(133)	6.3.2 全真考题解答	(167)
4.5.3 全真试题训练	(135)	6.3.3 全真试题训练	(168)
4.5.4 全真试题参考答案	(137)	6.3.4 全真试题参考答案	(169)
4.6 操作已创建的查询	(137)	6.4 报表排序、分组及计算控件的使用 ...	(169)
4.6.1 考试必备知识	(137)	6.4.1 考试必备知识	(169)
4.6.2 全真考题解答	(138)	6.4.2 全真考题解答	(170)
4.6.3 全真试题训练	(139)	6.4.3 全真试题训练	(171)
4.6.4 全真试题参考答案	(139)	6.4.4 全真试题参考答案	(172)
第5章 窗体	(140)	6.5 创建子报表和多列报表	(172)
5.1 认识窗体	(140)	6.5.1 考试必备知识	(172)
5.1.1 考试必备知识	(140)	6.5.2 全真考题解答	(173)
5.1.2 全真考题解答	(141)	6.5.3 全真试题训练	(173)
		6.5.4 全真试题参考答案	(174)

6.6 设计复杂的报表以及预览、打印和保存 报表	(174)	8.2.3 全真试题训练	(186)
6.6.1 考试必备知识	(174)	8.2.4 全真试题参考答案	(187)
6.6.2 全真考题解答	(175)	第9章 模块	(188)
6.6.3 全真试题训练	(176)	9.1 模块的基本概念及模块创建	(188)
6.6.4 全真试题参考答案	(176)	9.1.1 考试必备知识	(188)
第7章 数据访问页	(177)	9.1.2 全真考题解答	(189)
7.1 数据访问页视图	(177)	9.1.3 全真试题训练	(189)
7.1.1 考试必备知识	(177)	9.1.4 全真试题参考答案	(191)
7.1.2 全真考题解答	(177)	9.2 VBA 程序设计基础	(191)
7.1.3 全真试题训练	(178)	9.2.1 考试必备知识	(191)
7.1.4 全真试题参考答案	(178)	9.2.2 全真考题解答	(209)
7.2 创建数据访问页	(178)	9.2.3 全真试题训练	(222)
7.2.1 考试必备知识	(178)	9.2.4 全真试题参考答案	(249)
7.2.2 全真考题解答	(179)	第10章 全真笔试试卷	(252)
7.2.3 全真试题训练	(179)	10.1 笔试试卷(一)	(252)
7.2.4 全真试题参考答案	(180)	10.2 笔试试卷(一)参考答案及评分 标准	(258)
7.3 编辑数据访问页	(180)	10.3 笔试试卷(二)	(258)
7.3.1 考试必备知识	(180)	10.4 笔试试卷(二)参考答案及评分 标准	(263)
7.3.2 全真考题解答	(181)	10.5 笔试试卷(三)	(264)
7.3.3 全真试题训练	(182)	10.6 笔试试卷(三)参考答案及评分 标准	(270)
7.3.4 全真试题参考答案	(182)	10.7 笔试试卷(四)	(271)
第8章 宏	(183)	10.8 笔试试卷(四)参考答案及评分 标准	(276)
8.1 宏的概念	(183)	附录A 如何安装和使用本书配套光盘	(277)
8.1.1 考试必备知识	(183)	附录B 二级考试大纲(2004年版)	(285)
8.1.2 全真考题解答	(183)		
8.2 宏的操作	(184)		
8.2.1 考试必备知识	(184)		
8.2.2 全真考题解答	(185)		

第1章 公共基础知识

1.1 数据结构与算法

1.1.1 考试必备知识

1. 算法

(1) 算法的基本概念

所谓算法是关于解题方案的准确而完善的描述。

算法的基本特征：可行性 (effectiveness)、确定性 (definiteness)、有穷性 (finiteness)、拥有足够的信息 (necessary information)。

算法的基本要素如下：

① 算法中对数据的运算和操作：包括算术运算、逻辑运算、关系运算和数据传输。

② 算法的控制结构：基本的控制结构包括顺序结构、选择结构、循环结构。

算法设计的基本方法：列举法、归纳法、递推、递归、减半递推技术、回溯法。

(2) 算法的复杂度

算法的复杂度主要包括时间复杂度和空间复杂度。

算法的时间复杂度：所谓算法的时间复杂度，是指执行算法所需要的计算工作量。算法的工作量可以用算法所执行的基本运算次数来度量，而算法所执行的基本运算次数是问题规模的函数。即

$$\text{算法的工作量} = f(n)$$

其中 n 是问题的规模。

在同一个问题规模下，如果算法执行所需的基本运算次数取决于某一特定的输入时，可以用以下两种方法分析算法的工作量。

① 平均性态 (average behavior)：所谓平均性态分析，是指用各种特定输入下的基本运算次数的加权平均值来度量算法的工作量，可以用以下公式表示：

$$A(n) = \sum_{x \in D_n} P(x) t(x)$$

其中， x 是所有可能输入中的某个特定输入， $P(x)$ 是 x 出现的概率， $t(x)$ 是算法在输入为 x 时所执行的基本运算次数， D_n 表示当问题规模为 n 时，算法执行时所有可能输入的集合。

② 最坏情况复杂性 (worst-case complexity)：所谓最坏情况分析，是指在规模为 n 时，算法所执行的基本运算的最大次数。它定义为

$$W(n) = \max_{x \in D_n} \{t(x)\}$$

算法的空间复杂度：一般是指执行这个算法所需要的存储空间大小。一个算法所占用的存

存储空间包括算法程序所占的空间、输入的初始数据所占用的存储空间以及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。

2. 数据结构的基本概念

(1) 数据结构学科的研究对象

数据结构学科主要研究如下3个方面的内容:

- ① 数据集中各数据元素之间所固有的逻辑关系,即数据的逻辑结构。
- ② 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。
- ③ 对各种数据结构进行的运算。

研究数据结构学科的目的是提高数据处理的效率。主要包括:

- ① 数据处理速度。
- ② 尽量节省在数据处理过程中所占用的计算机存储空间。

(2) 数据结构的定义

数据处理:数据处理是指对数据集中的各元素以各种方式进行运算。

数据元素:在数据处理领域中,每一个需要处理的对象都可以抽象为数据元素。

数据结构:数据结构是指相互有关联的数据元素的集合。

① 数据的逻辑结构:是指反映数据元素之间逻辑关系的数据结构。它包含两个要素:一是数据元素的集合,通常记为 D ;二是 D 上的关系,它反映了 D 中各数据元素之间的前、后件关系,通常记为 R 。形式表示如下:

$$B = (D, R)$$

其中 B 表示数据结构。

② 数据的存储结构:是指数据的逻辑结构在计算机存储空间中的存放形式。

一般来说,一种数据结构的逻辑结构根据需要可以表示成多种存储结构。常用的存储结构有顺序、链接、索引等。

(3) 数据结构的图形表示

图形表示方法:对于数据集合 D 中的每一个数据元素都用中间标有元素值的方框表示,一般称之为数据结点,简称为结点。对于关系 R 中的每一个二元组,都用一条有向线段从前件结点指向后件结点,以表示数据元素之间的前、后件关系。

例如,用图形表示数据结构 $B = (D, R)$,其中:

$$D = \{d_i \mid 1 \leq i \leq 6\} = \{d_1, d_2, d_3, d_4, d_5, d_6\}$$

$$R = \{(d_1, d_2), (d_1, d_3), (d_3, d_4), (d_5, d_4), (d_5, d_6)\}$$

这个数据结构的图形如图 1.1 所示。

(4) 线性结构和非线性结构

数据结构分为两大类:线性结构和非线性结构。

线性结构满足如下条件:

- ◆ 有且仅有一个根结点。
- ◆ 每一个结点最多有一个前件,也最多有一个后件。
- ◆ 在一个线性结构中插入或删除任何一个结点后还是线性结构。

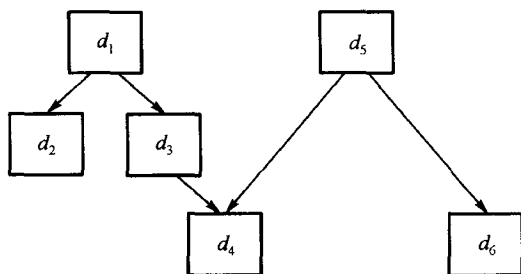


图 1.1 数据结构的图形

如果一个数据结构不是线性结构,则称之为非线性结构。

3. 线性表及其顺序存储结构

(1) 线性表的基本概念

线性表定义:线性表是由 $n(n \geq 0)$ 个数据元素 $a_1, a_2, \dots, a_n$ 组成的一个有限序列,表中的每一个数据元素,除了第一个外,有且只有一个前件,除了最后一个外,有且只有一个后件,即线性表或是一个空表,或者可以表示为

$$(a_1, a_2, \dots, a_i, \dots, a_n)$$

其中 $a_i (i = 1, 2, \dots, n)$ 是属于数据对象的元素,通常也称为线性表中的一个结点。

非空线性表有如下结构特征:

- ◆ 有且只有一个根结点 a_1 , 无前件。
- ◆ 有且只有一个终端结点 a_n , 无后件。
- ◆ 除根结点与终端结点外,其他所有结点有且只有一个前件,也有且只有一个后件。

线性表的长度:线性表中结点的个数 n 称为线性表的长度。当 $n = 0$ 时,称为空表。

(2) 线性表的顺序存储结构

线性表的顺序存储结构的基本特点:

- ◆ 线性表中所有元素所占的存储空间是连续的。
- ◆ 线性表中各数据元素在存储空间中是按逻辑顺序依次存放的。

线性表的随机存取地址的计算公式如下:

$$\text{ADD}(a_i) = \text{ADD}(a_1) + (i - 1)k$$

线性表的主要操作:插入、删除、查找、排序、分解、合并、复制和逆转。

(3) 线性表的插入运算

线性表插入操作前、后元素的对应关系:

设长度为 n 的线性表为

$$(a_1, a_2, \dots, a_i, \dots, a_n)$$

在线性表的第 i 个元素 a_i 之前插入一个新元素 b 后得到的长度为 $n + 1$ 的线性表为

$$(a'_1, a'_2, \dots, a'_j, a'_{j+1}, \dots, a'_n, a'_{n+1})$$

插入前、后两线性表中的元素满足如下关系:

$$a'_j = \begin{cases} a_j & 1 \leq j \leq i-1 \\ b & j = i \\ a_{j-1} & i+1 \leq j \leq n+1 \end{cases}$$

线性表插入操作的实现方法:

一般来说,要在第 $i (1 \leq i \leq n)$ 个元素之前插入一个新元素时,首先要从最后一个(即第 n 个)元素开始,直到第 i 个元素之间的 $n-i+1$ 个元素依次向后移动一个位置。移动结束后,第 i 个位置就被空出,然后将新元素插入到第 i 项。插入结束后,线性表的长度就增加了 1。

线性表插入操作算法的复杂度分析:

由线性表插入算法可知:如果插入的位置在第 $i (1 \leq i \leq n)$ 个元素之前,则原来第 i 个元素之后的所有元素都必须向后移动。在平均情况下,要在线性表中插入一个新元素,需要移动表中一半的元素。

(4) 线性表的删除运算

线性表删除操作前、后元素的对应关系:

设长度为 n 的线性表为

$$(a_1, a_2, \dots, a_i, \dots, a_n)$$

删除线性表的第 i 个元素 a_i 后得到的长度为 $n-1$ 的线性表为

$$(a'_1, a'_2, \dots, a'_j, \dots, a'_{n-1})$$

删除后两线性表中的元素满足如下关系:

$$a'_j = \begin{cases} a_j & 1 \leq j \leq i-1 \\ a_{j+1} & i \leq j \leq n-1 \end{cases}$$

线性表删除操作的实现方法:

一般来说,要删除第 $i (1 \leq i \leq n)$ 个元素时,首先要从第 $i+1$ 个元素开始,直到最后一个(即第 n 个)元素之间的 $n-i$ 个元素依次向前移动一个位置。删除结束后,线性表的长度就减少了 1。

线性表删除操作算法的复杂度分析:

由线性表插入算法可知:如果删除的是第 $i (1 \leq i \leq n)$ 个元素,则原来第 i 个元素之后的所有元素都必须向前移动一个位置。在平均情况下,要在线性表中删除一个已有的元素,需要移动表中一半的元素。

(5) 线性表顺序存储结构的适用场合

线性表的顺序存储结构对于小线性表或者其中元素不常变动的线性表来说比较合适,因为顺序存储的结构比较简单。但这种顺序存储的方式对于元素需要变动的大线性表就不太合适了,因为插入和删除的效率比较低。

4. 栈和队列

(1) 栈及其基本运算

栈的定义:栈(stack)是限定在一端进行插入和删除的线性表。它按照“后进先出”的原则组织数据。

栈的顺序存储:在程序设计语言中,一般是用一维数组 $S(1:m)$ 作为栈的顺序存储空间,其

中 m 为栈的最大容量。

栈的基本运算：

① 入栈运算：首先将栈顶指针加 1，然后将新元素插入到栈顶指针指向的位置。当栈顶指针已经指向存储空间的最后一个位置时，说明栈空间已满，不可能再进行入栈操作，否则，将发生栈“上溢”错误。

② 退栈运算：首先将栈顶元素赋给一个指定的变量，然后将栈顶指针减 1。当栈顶指针为 0 时，说明栈是空的，不可能再进行退栈操作，否则，将发生栈“下溢”错误。

③ 读栈顶元素：将栈顶元素赋给一个指定的变量，栈指针不变。当栈顶指针为 0 时，说明栈是空的，读栈顶元素操作失败。

(2) 队列及其基本运算

队列的定义：队列 (queue) 是指允许在一端进行插入、而在另一端进行删除的线性表。它按照“先进先出”的原则组织数据。

循环队列空的状态： $s = 0$ 且 $front = rear = m$ 。

循环队列满的状态： $s = 1$ 且 $front = rear$ 。

循环队列的基本运算：

① 入队运算：首先将队尾指针加 1 ($rear = rear + 1$) 并当 $rear = m + 1$ 时置 $rear = 1$ ；然后将新元素插入到指针指向的位置。当循环队列满时 ($s = 1$ 且 $front = rear$) 入队操作将引起“上溢”错误。

② 退队运算：首先将队首指针加 1 ($front = front + 1$) 并当 $front = m + 1$ 时置 $front = 1$ ；然后将队首指针指向的元素赋给指定的变量。当循环队列为空时 ($s = 0, front = rear$)，退队操作将引起“下溢”错误。

5. 线性链表

(1) 线性链表的基本概念

① 线性链表：线性链表的链式存储结构。线性链表的每个结点中的数据域存放数据元素的值，指针域存放后件结点的存储地址。

双向链表的链式存储结构比线性链表的链式存储结构多出一个指针域，它用来存放前件结点的存储地址。

② 带链的栈：栈的链式结构。栈的链式结构基本上和线性链表的链式存储结构相同，只是线性链表的链式存储结构的头指针变成了栈的链式结构的栈顶指针。

③ 带链的队列：队列的链式结构。队列的链式结构和线性链表的链式存储结构基本相同，只不过队列的链式结构保持有两个指针，一个指向队列头的头指针和一个指向队列尾的尾指针。

(2) 线性链表的基本运算

① 线性链表的主要运算：线性链表的主要运算有插入、删除、合并、分解、逆转、复制、排序和查找。

② 线性链表的查找：线性链表的查找过程是从头指针指向的结点开始往后沿指针进行扫描，直到后面已没有结点或下一个结点的数据域为搜索值 x 为止。

③ 线性链表的插入：线性链表的插入是先在栈中为新元素分配一个新的结点 p 并赋值，然后利用线性链表的查找算法找到待插入位置的前一个结点的指针 q 。先将 p 指向 q 的后件，然后将 p 挂接在 q 结点后面。

④ 线性链表的删除:利用线性链表的查找算法找到待删除元素的前一个结点 p , 用另一个指针 q 暂时保存 p 的后续结点(即待删除结点), 然后把 q 结点的后续链直接挂接在 p 的后面, 最后归还 q 结点所分配的栈空间。

(3) 循环链表及其基本运算

循环链表有如下几个特点:

- ◆ 在循环链表中增加一个表头结点, 使得循环链表对空表和非空表的操作实现了统一。
- ◆ 循环链表中最后一个结点的指针域不是空, 而是指向表头结点。
- ◆ 判断循环链表是否为空的办法不是看表头指针为空, 而是看表头结点的后续结点是否还是表头结点。
- ◆ 在循环链表中, 从任何一个结点出发都可以访问到表中其他所有的结点。

6. 树与二叉树

(1) 树的基本概念

- ① 基本术语: 根结点、父结点、子结点、叶子结点、结点的度、树的度、树的深度、子树。
- ② 要求考生掌握用树表示算术表达式的方法。
- ③ 要求考生了解树链表中的结点结构。

(2) 二叉树及其基本性质

① 二叉树的特点: 二叉树具有以下两个特点, 一是非空二叉树只有一个根结点, 二是每一个结点最多有两棵子树。

② 二叉树的基本性质:

性质 1 在二叉树的第 k 层上, 最多有 2^{k-1} ($k \geq 1$) 个结点。

性质 2 深度为 m 的二叉树是指二叉树最多有 $2^m - 1$ 个结点。

性质 3 在任意一棵二叉树中, 度为 0 的结点(即叶子结点)总是比度为 2 的结点多一个。

性质 4 具有 n 个结点的二叉树, 其深度至少为 $\lceil \log_2 n \rceil + 1$, 其中 $\lceil \log_2 n \rceil$ 表示 $\log_2 n$ 的整数部分。

(3) 满二叉树和完全二叉树

性质 5 满二叉树的第 k 层上有 2^{k-1} 个结点, 且其深度为 m 的满二叉树有 $2^m - 1$ 个结点。

性质 6 具有 n 个结点的完全二叉树的深度为 $\lceil \log_2 n \rceil + 1$ 。

性质 7 设完全二叉树共有 n 个结点。如果从根结点开始, 按层序(每一层从左到右)用自然数 $1, 2, \dots, n$ 给结点进行编号, 则对于编号为 k ($k = 1, 2, \dots, n$) 的结点有以下结论:

① 若 $k = 1$, 则该结点为根结点, 它没有父结点; 若 $k > 1$, 则该结点的父结点编号为 $\lceil k/2 \rceil$ 。

② 若 $2k \leq n$, 则编号为 k 的结点的左子结点编号为 $2k$; 否则该结点无左子结点(显然也没有右子结点)。

③ 若 $2k + 1 \leq n$, 则编号为 k 的结点的右子结点编号为 $2k + 1$; 否则该结点无右子结点。

(4) 二叉树的存储结构

$L(i)$	$V(i)$	$R(i)$
--------	--------	--------

$L(i)$: 左指针域, 指向该结点的左子结点(存放左子结点的存储地址)。

$R(i)$: 右指针域, 指向该结点的右子结点(存放右子结点的存储地址)。

$V(i)$: 数据域, 存放结点的值。

(5) 二叉树的遍历

二叉树的遍历集中用到了递归的思想, 它主要有以下 3 种遍历方式:

① 前序遍历(DLR): 先访问根结点, 然后前序遍历左子树, 再前序遍历右子树。

② 中序遍历(LDR): 先中序遍历左子树, 然后访问根结点, 再中序遍历右子树。

③ 后序遍历(LRD): 先后序遍历左子树, 然后后序遍历右子树, 再访问根结点。

7. 查找技术

(1) 顺序查找

① 查找方法: 从表头到表尾逐个比较, 若相同, 则结束查找, 否则继续比较表中的下一个元素, 直到整个表都遍历完。

② 适用场合: 无序表或链式存储的有序表。

(2) 二分查找

① 查找方法: 每次把待查找值与表的中间元素进行比较, 以减半的方式缩小搜索范围。

② 适用场合: 顺序存储的有序表。

8. 排序技术

(1) 交换类排序

常用的交换类排序有: 冒泡排序法、快速排序法。

(2) 插入类排序

常用的插入类排序有: 简单插入排序法、希尔排序法。

(3) 选择类排序

常用的选择类排序有: 简单选择排序法、堆排序法。

1.1.2 全真考题解答

一、选择题

(1) 算法的时间复杂度是指

A. 执行算法程序所需要的时间

B. 算法程序的长度

C. 算法执行过程中所需要的基本运算次数

D. 算法程序中的指令条数

答案: C

分析: 本题考查考生对算法的时间复杂度概念的理解。如果考生注意到了“所谓算法的时间复杂度, 是指执行算法所需要的计算工作量。算法的工作量可以用算法所执行的基本运算次数来度量”这句话, 就会很快对 C 项做出选择。A 项的迷惑性比较大, 考生容易望文生义地把它选上去。B 项和 D 项的意思差不多都是想误导考生以算法程序的长短论它的时间复杂度。实际上, 算法程序长的算法不一定比算法程序短的算法的时间复杂度高。比如, 一个 10 000 行的顺序执行的算法和一个只有 2 000 行但其中有 1 000 行循环执行了 50 次的算法比较, 前者的时间复杂度是 10 000, 后者的则是 51 000, 显然后者的时间复杂度更高。

(2) 算法的空间复杂度是指

- A. 算法程序的长度
B. 算法程序中的指令条数
C. 算法程序所占的存储空间
D. 算法执行过程中所需要的存储空间

答案: D

分析: 本题考查考生对算法的空间复杂度概念的理解, 算法的空间复杂度是指算法在执行过程中所需要的存储空间。也包括静态地存储算法程序指令的存储空间, 也包括算法程序动态的执行过程中为中间结果和数据结构开辟的数据存储空间。题中 A、B、C 3 项都只考虑到了影响算法程序静态存储空间大小的因素。

(3) 下列叙述中正确的是

- A. 线性表是线性结构
B. 栈和队列是非线性结构
C. 线性链表是非线性结构
D. 二叉树是线性结构

答案: A

分析: 线性结构和非线性结构是一对互斥的概念。一般来说, 一个数据结构不是线性结构, 就一定是非线性结构。在已学过的数据结构中线性表、线性链表、栈和队列都是线性结构; 树和图是非线性结构。

(4) 数据的存储结构是指

- A. 数据所占的存储空间量
B. 数据的逻辑结构在计算机中的表示方式
C. 数据在计算机中的顺序存储方式
D. 存储在外存中的数据

答案: B

分析: 本题考查考生对数据的存储结构和逻辑结构概念的理解、区别和联系。数据的逻辑结构是指数据元素之间的逻辑关系的数据结构。数据的存储结构则是数据的逻辑结构在计算机中的物理实现, 有时也称做数据的物理结构。两者的区别是, 数据的逻辑结构只涉及数据之间抽象的数学关系; 存储结构则涉及如何在计算机中通过对数据的物理存储进行组织来表达数据元素之间的逻辑关系。比如, 在线性表的顺序存储中是利用物理存储空间上的连续性来表达线性表中数据的前、后件关系; 在线性表的链式存储中, 是通过指针域构成的逻辑链条来表达数据的前、后件关系。一般来说, 一种数据的逻辑结构对应的物理实现, 即数据的存储结构不止一种。

(5) 长度为 10 的顺序表的首地址是从 1023 开始的, 顺序表中每个元素的长度为 2, 在第 4 个元素前面插入一个元素和删除第 7 个元素后, 顺序表的总长度不变。问在执行插入和删除操作前, 顺序表中第 5 个元素在执行插入和删除操作后的顺序表中的存储地址是

- A. 1028
B. 1029
C. 1031
D. 1033

答案: D

分析: 本题考查考生对线性表随机存取地址的计算公式以及线性表中的插入和删除操作前后元素的对应关系的掌握。由于问的是原来顺序表中的第 5 个元素, 它在插入操作后变成了第 6 个元素 (因为插入的元素在它前面)。由于删除的第 7 个元素在它后面, 不会影响它在顺序表中的排位, 因此在执行插入和删除操作后, 原先顺序表中的第 5 个元素变成了新的顺序表中的第 6 个元素。再按照线性表随机存取地址的计算公式 $ADD(a_i) = ADD(a_1) + (i-1) \times k$ 计算 $ADD(a_6) = ADD(a_1) + (6-1) \times 2 = 1023 + 5 \times 2 = 1033$, 因此答案选 D。

(6) 下列关于线性表的两种存储结构叙述正确的是

- A. 存储相同数目的元素线性链表比顺序表要节省存储空间
- B. 对无序表的查找,顺序表和线性链表的效率是一样的
- C. 顺序表适用于插入、删除等更新操作频繁的场所
- D. 线性链表适用于查询操作比较频繁的场所

答案: B

分析: 对线性表的两种存储结构的比较能够加深对顺序表和线性链表这两种存储结构的理解。顺序表是通过在内存中分配一块连续的空间,用物理上的相邻关系表达数据间的逻辑相邻关系的;而线性链表则是通过增加一个指针域来把相邻的数据元素链接成一个线性序列。因此,存储相同数目的元素线性链表比顺序表要多消耗一些存储空间来存放后继指针。但是线性链表的这种结构使得它存储数据的空间可以是离散的,并不像顺序表那样一定要求物理上的连续空间。因此,在其中增加和删除元素的操作很简单,只需要分配一个数据结点,然后把它挂接到正确的地方即可。而在顺序表中要增加和删除一个数据元素,为了保证数据在物理空间上的连续性,不得不大量移动数据元素。因此,C、D两项也可以排除了。至于顺序表和线性链表在查询效率上的比较问题,对于无序表的查找,不管是用顺序表还是线性链表都只能用顺序查找的方法把线性表遍历一遍,因此效率大致是相同的。顺序表只是在数据组织成一种有序的结构时采用二分查找方法要比线性链表的速度快。

(7) 下列关于栈的叙述中不正确的是

- A. 在栈中只能在一端插入、删除数据
- B. 在栈中只能在一端插入数据,在另一端删除数据
- C. 栈是先进后出的线性表
- D. 栈是后进先出的线性表

答案: B

分析: 栈是限制在一端对数据进行插入和删除的线性表,它对数据的操作顺序是先进后出,也就是后进先出。考生在做这道题时如果不明白先进后出和后进先出指的是同一回事,很可能会在C和D中选择。至于B项说的是队列的特性。

(8) 在线性链表的插入算法中,若要把结点 q 插在结点 p 后面,下列操作正确的是

- A. 使结点 p 指向结点 q ,再使结点 q 指向结点 p 的后件结点
- B. 使结点 q 指向 p 的后件结点,再使结点 p 指向结点 q
- C. 使结点 q 指向结点 p ,再使结点 p 指向结点 q 的后件结点
- D. 使结点 p 指向 q 的后件结点,再使结点 q 指向结点 p



答案: B

分析: 在修改结点指针域的操作中,有一个操作顺序的问题。比较A和B只是操作顺序颠倒了一下。A中先使结点 p 指向 q 后, q 就成为 p 的新后件结点,原先通过结点 p 指向的后件结点与结点 p 脱节了。那么,后面的一步操作没有任何意义的:使结点 q 指向 p 的后件结点即使结点 q 成为自己的后件结点。按照B指定的顺序操作就不会出现在引用结点 p 的指针域之前已经把它值修改了的情形。至于C和D项是命题者设计的干扰项,想让考生把 p 和 q 的顺序搞混。

(9) 下列叙述中错误的是