

全国计算机等级考试

考点解析、例题精解
与实战练习

——二级Java语言 程序设计

NCRE研究组



高等教育出版社
Higher Education Press

全国计算机等级考试考点解析、例题精解与实战练习

二级 Java 语言 程序设计

NCRE 研究组



高等教育出版社

责任编辑 责任校对
封面设计 版式设计

出版发行 高等教育出版社
社址 北京市西城区德胜大街1号
邮政编码 100011
总机 010-28819000

经销 北京白塔山书局有限公司
印刷 北京白塔山书局有限公司

开本 787×1092 1/16
印张 17
字 410 000

内容提要

本书是按照教育部考试中心颁布的全国计算机等级考试最新考试大纲和指定教材编写的。全书共分 17 章,讲解计算机等级考试二级 Java 语言程序设计的相关知识,章节安排与教育部考试中心最新指定教材(2008 年版)同步,主要从考试大纲要求、考试要点、经典题解和同步自测等几个方面对该部分内容进行系统的阐述。本书的内容主要有: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、Java 语言概论、简单数据类型概论、运算符和表达式、流程控制、类、数组和字符串操作、异常处理、输入/输出及文件操作、线程与对象串行化、编写图形用户界面、Applet 程序设计、上机指导、笔试模拟试卷及答案分析、上机模拟试卷及答案分析。

本书具有考点浓缩、例题典型、讲解精当等特点,非常适合参加全国计算机等级考试(二级 Java 语言程序设计)的人员考前复习使用,也适合其他相关人员及等级考试培训班使用。

本书配有光盘。光盘中提供笔试和上机考试两大软件系统,含有 10 套笔试和 10 套上机考试模拟试题。两大软件系统均增加了试题评析的功能。此外,所有上机试题均配有视频演示和关键注解。

图书在版编目(CIP)数据

全国计算机等级考试考点解析、例题精解与实战练习. 二级 Java 语言程序设计/NCRE 研究组. —北京:高等教育出版社,2008.4

ISBN 978-7-04-023858-7

I. 全… II. N… III. ①电子计算机-水平考试-自学参考资料②Java 语言-程序设计-水平考试-自学参考资料
IV. TP3

中国版本图书馆 CIP 数据核字(2008)第 028760 号

策划编辑	何新权	责任编辑	康兆华	封面设计	张志奇	责任绘图	宗小梅
版式设计	陆瑞红	责任校对	金辉	责任印制	陈伟光		

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 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 17
字 数 440 000

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>
畅想教育 <http://www.widedu.com>

版 次 2008 年 4 月第 1 版
印 次 2008 年 4 月第 1 次印刷
定 价 35.00 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 23858-00

前 言

全国计算机等级考试自 1994 年举办以来,得到了社会各界的广泛认可,在推广、普及计算机应用知识和技术,以及为用人单位录用和考核工作人员提供评价标准等方面发挥了重要的作用。全国计算机等级考试是目前国内参加人数最多、影响最大的计算机类考试。

为了适应当前信息技术的飞速发展,教育部考试中心对全国计算机等级考试的考试科目及内容进行了重大的调整,对考试大纲进行全面修订。为了更好地服务于考生,引导考生尽快掌握先进的计算机技术,并能够顺利通过计算机等级考试,配合新考试大纲的推出,我们特别编写了此书。

本书共分 17 章,讲解计算机等级考试二级 Java 语言程序设计的相关知识,章节安排与教育部考试中心最新指定教材(2008 年版)同步,主要从考试大纲要求、考试要点、经典题解和同步自测等几个方面对该部分内容进行系统的阐述。所涉及的内容主要有: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、Java 语言概论、简单数据类型概论、运算符和表达式、流程控制、类、数组和字符串操作、异常处理、输入/输出及文件操作、线程与对象串行化、编写图形用户界面、Applet 程序设计、上机指导、笔试模拟试卷及答案分析、上机模拟试卷及答案分析。

与目前已出版的同类图书相比,本书具有以下一些特色。

(1) 考点浓缩精解,重点突出。本书将指定的考试内容进行浓缩,用言简意赅的语言精讲考试要点、重点和难点,从而使考生更易于理解全国计算机等级考试的要求和考查范围,能够在较短的时间内取得较大的收获。

(2) 例题选取精心,分析到位。书中的例题部分选自近年全国计算机等级考试的真题,另一部分是根据最新考试要求精心设计而成的,具有典型性和针对性。所有例题均给出详尽的分析,便于考生掌握完整的解题思路,以达到举一反三、触类旁通的功效。

(3) 实战练习丰富,附有答案。本书针对考试过关,着重实战,每章均配有练习题,通过这些练习题可对所学内容进行巩固和提高。最后还提供多套模拟试题,便于考生检测自己的总体水平。所有练习题、模拟试题均配有答案,便于自测使用。

(4) 上机全面辅导,书盘结合。本书介绍了上机考试的环境、要点、考试题型等内容,并对最新的上机考试真题进行分类详解。光盘中提供笔试和上机考试两大软件系统,含有 10 套笔试和 10 套上机考试模拟试题,便于实战演练,感受全真考试环境。两大软件均增设了试题评析功能,便于考生自我检查学习效果,在答题过程中加深对所考核知识要点的理解与掌握。特别地,所有上机试题均配有视频演示及关键注解,如同名师亲临现场,手把手教会考生解题过关。

本书非常适合参加全国计算机等级考试(二级 Java 语言程序设计)的人员考前复习使用,也适合其他相关人员及等级考试培训班使用。

为方便读者复习,书中将重要考点或高频考点用“*”标记。

目 录

第1章 数据结构与算法

1.1 算法	1
考点1 算法的基本概念	1
*考点2 算法复杂度	1
1.2 数据结构的基本概念	2
考点3 什么是数据结构	2
考点4 数据结构的图形化表示	2
*考点5 线性结构与非线性结构	2
1.3 线性表及其顺序存储结构	3
考点6 线性表的基本概念	3
*考点7 线性表的顺序存储结构	3
考点8 顺序表的插入运算	3
考点9 顺序表的删除运算	3
1.4 栈和队列	4
*考点10 栈及其基本运算	4
*考点11 队列及其基本运算	4
1.5 线性链表	4
*考点12 线性链表的基本概念	4
考点13 线性链表的基本运算	4
考点14 循环链表及其基本运算	5
1.6 树与二叉树	5
考点15 树的基本概念	5
*考点16 二叉树及其基本性质	5
考点17 二叉树的存储结构	6
*考点18 二叉树的遍历	6
1.7 查找技术	6
*考点19 顺序查找	6
*考点20 二分法查找	6
1.8 排序技术	7
*考点21 交换类排序法	7
*考点22 插入类排序法	7
*考点23 选择类排序法	7
1.9 经典题解	7
1.10 同步自测	11

1.11 同步自测答案	12
-------------------	----

第2章 程序设计基础

2.1 程序设计	13
*考点1 程序设计方法与风格	13
2.2 结构化程序设计	14
*考点2 结构化程序设计的原则	14
*考点3 结构化程序设计的 基本结构与特点	14
考点4 结构化程序设计原则和 方法的应用	14
2.3 面向对象程序设计	14
考点5 关于面向对象方法	14
*考点6 面向对象方法的基本概念	15
2.4 经典题解	15
2.5 同步自测	16
2.6 同步自测答案	17

第3章 软件工程基础

3.1 软件工程的基本概念	18
考点1 软件工程的定义	18
*考点2 软件生命周期的定义	18
考点3 软件工程的目标与原则	18
3.2 结构化分析方法	19
考点4 关于结构化分析方法	19
*考点5 关于结构化分析的常用工具	19
考点6 软件需求规格说明书	19
3.3 结构化设计方法	20
考点7 有关软件设计的基本内容	20
*考点8 有关结构化设计方法的 基本内容	20
3.4 软件测试	21
*考点9 软件测试的方法与技术	21
*考点10 软件测试的实施	21

3.5 程序调试	22
考点 11 程序调试的基本概念	22
* 考点 12 程序调试方法	22
3.6 经典题解	22
3.7 同步自测	25
3.8 同步自测答案	25

第 4 章 数据库设计基础

4.1 数据库系统的基本概念	26
* 考点 1 数据库	26
* 考点 2 数据库管理系统	26
* 考点 3 数据库系统	27
4.2 数据模型	27
考点 4 数据模型的基本概念	27
* 考点 5 E-R 模型 (实体联系模型)	27
4.3 关系代数	28
考点 6 关系模型的基本操作	28
* 考点 7 关系模型的基本运算	28
* 考点 8 关系代数中的扩充运算	29
4.4 数据库设计与管理	29
考点 9 数据库设计方法与步骤	29
* 考点 10 数据库设计的需求分析	29
* 考点 11 数据库概念设计	29
* 考点 12 数据库逻辑设计	30
考点 13 数据库物理设计	30
4.5 经典题解	30
4.6 同步自测	33
4.7 同步自测答案	34

第 5 章 Java 语言概论

5.1 Java 语言简介	35
* 考点 1 Java 语言的特点及优势	35
考点 2 Java 语言的实现机制	36
5.2 Java 的体系结构	36
* 考点 3 JDK 目录结构	36
* 考点 4 Java 的 API 结构	36
* 考点 5 Java 源程序结构	37
* 考点 6 Java 程序编写及运行的过程	38
5.3 经典题解	39
5.4 同步自测	42
5.5 同步自测答案	43

第 6 章 简单数据类型概论

6.1 概述	44
* 考点 1 Java 标识符的命名规则	44
考点 2 常量与变量	44
6.2 简单数据类型	45
* 考点 3 整型数据	45
* 考点 4 浮点型数据	45
* 考点 5 布尔型数据	45
* 考点 6 字符型数据	45
* 考点 7 各类型数据之间的转换	46
考点 8 Java 类库中对简单数据类型的 类包装	46
6.3 经典题解	46
6.4 同步自测	48
6.5 同步自测答案	49

第 7 章 运算符和表达式

7.1 概述	50
考点 1 运算符	50
考点 2 表达式	50
7.2 算术运算符和算术表达式	51
* 考点 3 一元算术运算符	51
* 考点 4 二元算术运算符	51
* 考点 5 算术运算符的优先级	51
7.3 关系运算符和关系运算表达式	51
考点 6 关系运算符及关系运算表达式	51
7.4 布尔逻辑运算符和 布尔逻辑表达式	51
考点 7 逻辑运算符	51
7.5 位运算符和位运算表达式	52
考点 8 位逻辑运算符	52
考点 9 移位运算符	52
考点 10 位运算符的优先级	53
7.6 赋值运算符和赋值表达式	53
考点 11 赋值表达式的类型	53
7.7 条件运算符与条件表达式	53
* 考点 12 条件表达式的值	53
7.8 运算符的优先级和复杂表达式	53
* 考点 13 运算符的优先级	53
7.9 经典题解	53
7.10 同步自测	59

7.11 同步自测答案	61
-------------------	----

第8章 流程控制

8.1 分支语句	62
考点1 条件语句	62
* 考点2 多分支语句	63
8.2 循环语句	63
考点3 while 循环	63
考点4 do-while 循环	64
* 考点5 for 循环	64
8.3 跳转语句	65
考点6 break 语句	65
考点7 continue 语句	65
考点8 return 语句	65
8.4 循环语句与分支语句的嵌套	66
* 考点9 多层循环语句	66
8.5 递归	66
考点10 递归	66
8.6 经典题解	66
8.7 同步自测	69
8.8 同步自测答案	71

第9章 类、数组和字符串操作

9.1 概述	72
考点1 面向对象特性	72
* 考点2 类定义	72
考点3 方法重载与构造方法	74
* 考点4 对象	74
* 考点5 接口	75
考点6 包	76
* 考点7 类及类成员修饰符	76
* 考点8 继承	77
考点9 多态性	78
考点10 内部类	78
* 考点11 类库	78
9.2 一维数组	79
考点12 定义和创建数组	79
* 考点13 初始化数组	79
9.3 多维数组	80
考点14 二维数组的定义	80
* 考点15 二维数组初始化	80
考点16 数组边界	80

考点17 复制数组	80
考点18 调整数组大小	80

9.4 字符串操作	81
-----------------	----

考点19 字符串的表示	81
考点20 字符串的访问	81
* 考点21 字符串的修改	82
* 考点22 字符串的比较	82

9.5 经典题解	82
----------------	----

9.6 同步自测	85
----------------	----

9.7 同步自测答案	88
------------------	----

第10章 异常处理

10.1 概述	89
考点1 异常	89
考点2 异常处理机制	89
考点3 Throwable 类	89
10.2 异常类	90
* 考点4 异常类	90
10.3 处理异常	91
* 考点5 try 和 catch 语句	91
考点6 finally 语句	91
考点7 异常处理的原则	91
10.4 经典题解	91
10.5 同步自测	93
10.6 同步自测答案	94

第11章 输入/输出及文件操作

11.1 概述	95
考点1 计算机数据的输入/输出方向	95
* 考点2 Java 中包含的输入/输出流的类	95
11.2 文件	98
考点3 创建文件	98
* 考点4 File 类提供的方法	98
* 考点5 随机文件流	99
考点6 压缩文件流	99
11.3 字节输入/输出流	99
考点7 字节输入流	99
考点8 字节输出流	100
* 考点9 内存的读/写	100

11.4 字符类输入/输出流	100	考点 15 串行化对象的方法	112
考点 10 字符类输入流	100	* 考点 16 构造可串行化对象的类	112
考点 11 字符类输出流	100	考点 17 部分定制串行化	112
11.5 过滤流	100	* 考点 18 完全定制串行化	113
* 考点 12 过滤流	100	考点 19 串行化中对敏感信息的保护	113
11.6 管道流	101	考点 20 串行化的注意事项	113
考点 13 管道流	101	12.8 经典题解	114
11.7 J2SE 1.4 的新 I/O 功能	101	12.9 同步自测	119
考点 14 java.nio 包的新特点	101	12.10 同步自测答案	121
考点 15 内存映射文件	101		
考点 16 文件通道	102		
考点 17 CRC32 类	102		
11.8 输入/输出流和正则表达式	102		
考点 18 Pattern 类	102		
考点 19 Matcher 类	102		
11.9 经典题解	103		
11.10 同步自测	106		
11.11 同步自测答案	106		
第 12 章 线程与对象串行化		第 13 章 编写图形用户界面	
12.1 线程的概念	107	13.1 概述	122
* 考点 1 线程与进程	107	考点 1 Java 图形用户界面的开发包	122
* 考点 2 Java 中的线程模型	107	13.2 用 AWT 编写图形用户界面	122
12.2 线程的创建	108	考点 2 java.awt 包	122
* 考点 3 创建线程	108	* 考点 3 构件和容器	122
考点 4 通过实现 Runnable		考点 4 常用容器	123
接口创建线程	108	考点 5 布局管理器	123
考点 5 通过继承 Thread 类创建线程	108	13.3 AWT 事件处理模型	124
12.3 线程调度与线程控制	108	* 考点 6 事件处理机制	124
考点 6 线程优先级与线程调度策略	108	考点 7 事件类	125
* 考点 7 线程的基本控制	109	* 考点 8 事件监听器	125
12.4 线程同步	109	考点 9 AWT 事件及其相应的	
考点 8 对象的加锁及其操作	109	监听器接口	125
考点 9 对象加锁的注意事项	110	考点 10 事件适配器	126
考点 10 死锁的防治	110	13.4 AWT 构件	127
考点 11 线程间的交互	110	* 考点 11 基本构件的应用	127
12.5 线程状态与生命周期	110	考点 12 构件与监听器的对应关系	128
* 考点 12 线程状态与生命周期	110	13.5 用 Swing 编写图形用户界面	128
12.6 线程的其他相关类与方法	111	考点 13 Swing 构件	128
考点 13 线程组	111	* 考点 14 Swing 的类层次结构	128
12.7 对象的串行化	111	考点 15 Swing 的特性	129
考点 14 串行化的概念和目的	111	13.6 Swing 构件和容器	129
		* 考点 16 构件的分类	129
		考点 17 使用 Swing 的基本规则	130
		考点 18 各种容器面板和构件	130
		* 考点 19 布局管理器	132
		考点 20 Swing 的事件处理机制	132
		13.7 经典题解	133
		13.8 同步自测	137
		13.9 同步自测答案	138

第 14 章 Applet 程序设计

14.1 Applet 概述	139
考点 1 Applet 的概念	139
* 考点 2 Applet 生命周期的概念	139
* 考点 3 Applet 类 API 概述	140
考点 4 Applet 的关键方法	141
* 考点 5 Applet 的显示	141
* 考点 6 Graphics 类	142
14.2 Applet 的编写	142
考点 7 Applet 的编写步骤	142
考点 8 用户 Applet 类的定义	142
考点 9 在 HTML 页面中包含 Applet ...	143
* 考点 10 <APPLET> 标记	143
14.3 Applet 中的图形用户界面	144
考点 11 基于 AWT 构件的 Applet 用户界面	144
考点 12 在 Applet 中使用弹出式 窗口	144
考点 13 基于 Swing 的 Applet 用户 界面	144
考点 14 Applet 中的事件处理	145
14.4 Applet 的多媒体支持	145
考点 15 显示图像	145
考点 16 动画制作	146
考点 17 播放声音	146
14.5 Applet 的安全控制	147
* 考点 18 Applet 的安全限制	147
考点 19 Java 中的沙箱模型	147
考点 20 Java 2 安全策略的 定义与实施	147
考点 21 Java 2 中基于数字签名的 安全控制	148
14.6 Applet 与工作环境的通信	148
考点 22 同页面 Applet 之间的通信	148
考点 23 Applet 与浏览器之间的通信	149
考点 24 Applet 的网络通信	149
14.7 Applet 与 Application	150
* 考点 25 Applet 与 Application	150

14.8 经典题解	150
14.9 同步自测	153
14.10 同步自测答案	154

第 15 章 上机指导

15.1 基本数据类型与处理及 对话框的应用	155
15.2 字符串的操作与输出	167
15.3 AWT 与 Swing 构件的应用	175
15.4 Applet 的应用	189
15.5 多线程与异常处理	200
15.6 输入/输出与文件操作	205
15.7 类的基本用法	207
15.8 同步自测	211
15.9 同步自测答案	227

第 16 章 笔试模拟试卷及答案分析

16.1 笔试模拟试卷	228
16.1.1 笔试模拟试卷一	228
16.1.2 笔试模拟试卷二	232
16.1.3 笔试模拟试卷三	237
16.2 笔试模拟试卷答案分析	241
16.2.1 笔试模拟试卷一答案分析	241
16.2.2 笔试模拟试卷二答案分析	244
16.2.3 笔试模拟试卷三答案分析	247

第 17 章 上机模拟试卷及答案分析

17.1 上机模拟试卷	251
17.1.1 上机模拟试卷一	251
17.1.2 上机模拟试卷二	254
17.1.3 上机模拟试卷三	257
17.2 上机模拟试卷答案分析	260
17.2.1 上机模拟试卷一答案分析	260
17.2.2 上机模拟试卷二答案分析	260
17.2.3 上机模拟试卷三答案分析	260

第1章

数据结构与算法

大纲要求

- (1) 算法的基本概念, 算法复杂度 (时间复杂度与空间复杂度) 的概念及意义。
- (2) 数据结构的定义, 数据的逻辑结构和存储结构, 数据结构的图形化表示, 线性结构和非线性结构的概念。
- (3) 线性表的定义, 线性表的顺序存储结构及其插入和删除运算。
- (4) 栈和队列的定义, 栈和队列的顺序存储结构及其基本运算。
- (5) 线性单向链表、双向链表与循环链表的结构及其基本运算。
- (6) 树的基本概念, 二叉树的定义及其存储结构, 二叉树的前序遍历、中序遍历和后序遍历。
- (7) 顺序查找与二分法查找算法, 基本的排序算法 (交换类排序、选择类排序、插入类排序)。

重要考点提示

根据对历年真题的分析可知, 本章考核内容约占 15%, 主要包括以下几个方面。

- (1) 算法复杂度。
- (2) 数据结构栈、队列、线性链表的基本概念。
- (3) 二叉树的存储结构。
- (4) 线性表、树的结点的计算和遍历。
- (5) 冒泡排序的最坏次数的计算。

1.1 算 法

考点1 算法的基本概念

- (1) 作为算法, 一般应具有 4 个基本特征: 可行性、确定性、有穷性和拥有足够的信息。
- (2) 一个算法通常由两种基本要素组成: 一是对数据对象的运算和操作, 二是算法的控制结构。算法的主要特征是着重于算法的动态执行。一个算法的功能不仅取决于所选用的操作, 而且与各个操作之间的执行顺序有关。算法的控制结构不仅决定了算法中各个操作的执行顺序, 而且直接反映了算法的设计是否符合结构化原则。
- (3) 算法设计的基本方法主要有枚举法、归纳法、递推法、递归法、减半递推技术和回溯法。

*考点2 算法复杂度

算法复杂度主要包括时间复杂度和空间复杂度。

1. 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。算法所执行的基本运算的次数与问题的规模有关。算法的工作量用算法所执行的基本运算的次数来度量。

对于一个固定的规模,算法所执行的基本运算的次数还可能与特定的输入有关。在同一个问题规模下,如果算法执行所需要的基本运算的次数取决于某一特定输入,则可以用两种方法来分析算法的工作量:一种方法是平均性态分析,用 $A(n)$ 表示。另一种方法是讨论算法在最坏情况下的时间复杂度,用 $W(n)$ 表示,它比 $A(n)$ 更具实用价值。

2. 算法的空间复杂度

算法的空间复杂度一般是指执行这个算法所需要的内存空间。

一个算法所占用的存储空间包括算法程序所占用的空间、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。在许多实际问题中,为了减少算法所占用的存储空间,通常采用压缩存储技术,以便尽量减少不必要的额外空间开销。

1.2 数据结构的基本概念

考点3 什么是数据结构

数据结构是指相互关联的数据元素的集合。

1. 数据的逻辑结构

所谓数据的逻辑结构,是指反映数据元素之间逻辑关系的数据结构。数据的逻辑结构有两个要素:一是数据元素的集合,通常记为 D ;二是 D 上的关系,通常记为 R ,即一个数据结构可以表示成

$$B = (D, R)$$

2. 数据的存储结构

数据的逻辑结构在计算机存储空间中的存放形式称为数据的存储结构(也称为数据的物理结构)。在进行数据处理时,被处理的各数据元素总是被存放在计算机的存储空间中,而且各数据元素在计算机存储空间中的位置关系与其逻辑关系可能不同。

一般来说,一种数据的逻辑结构根据需要可以表示成多种存储结构。而采用不同的存储结构,其数据处理的执行效率是不同的。

考点4 数据结构的图形化表示

通常,一个数据结构中的元素结点可能是动态变化的。在对数据结构进行处理的过程中,不仅数据结构中的结点(即数据元素)个数在动态地变化,而且各数据元素之间的关系也有可能在动态地变化。

*考点5 线性结构与非线性结构

数据结构可分为线性结构与非线性结构。如果一个非空的数据结构满足下列两个条件:第一,有且只有一个根结点;第二,每一个结点最多有一个前件,也最多只有一个后件,则称此数据结构为线性结构。线性结构又称为线性表。需要特别说明的是,在一个线性结构中插入或删除任何一个结点后还应该是线性结构。

如果一个数据结构不是线性结构,则称其为非线性结构。对非线性结构的存储和处理比线性结构要复杂得多。线性结构与非线性结构都可以是空的数据结构。一个空的数据结构究竟是属于线性结构还是属于非线性结构,这要根据具体的情况来确定。

1.3 线性表及其顺序存储结构

考点6 线性表的基本概念

线性表由一组数据元素构成。矩阵是一个比较复杂的线性表。线性表是一个线性结构。数据元素在线性表中的位置只取决于它们自己的序号。

非空线性表有以下一些结构特征。

- (1) 有且只有一个根结点，它无前件。
- (2) 有且只有一个终端结点，它无后件。
- (3) 除了根结点和终端结点之外，其他所有结点有且只有一个前件，也有且只有一个后件。

*考点7 线性表的顺序存储结构

线性表的顺序存储结构具有以下两个基本特点。

- (1) 线性表中所有元素所占用的存储空间是连续的。
- (2) 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。

在用一维数组存放线性表时，此一维数组的长度通常要定义得比线性表的实际长度大一些。在一般情况下，如果线性表的长度在处理过程中是动态变化的，则在开辟线性表的存储空间时要考虑到线性表在动态变化的过程中可能达到的最大长度。

考点8 顺序表的插入运算

在线性表采用顺序存储结构时，如果插入运算在线性表的末尾进行，即在第 n 个元素之后（可以认为是在第 $n+1$ 个元素之前）插入新元素，则只要在表的末尾增加一个元素即可，无须移动表中的元素；如果要在线性表的第 1 个元素之前插入一个新元素，则需要移动表中的所有元素。在一般情况下，如果插入运算在第 i ($1 \leq i \leq n$) 个元素之前进行，则原来第 i 个元素之后（包括第 i 个元素）的所有元素都必须移动。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。因此，在线性表顺序存储的情况下，要插入一个新元素，其效率是很低的，特别是在线性表比较大的情况下尤为突出。

考点9 顺序表的删除运算

在一般情况下，要删除第 i ($1 \leq i \leq n$) 个元素时，则要从第 $i+1$ 个元素开始，直到第 n 个元素之间共 $n-i$ 个元素依次向前移动一个位置。删除操作结束后，线性表的长度就减小了 1。

在线性表采用顺序存储结构时，如果删除运算在线性表的末尾进行，即删除第 n 个元素，则不需要移动表中的其他元素；如果要删除线性表中的第 1 个元素，则需要移动表中所有的元素。在一般情况下，如果要删除第 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依次向前移动一个位置。在平均情况下，要在线性表中删除一个元素，需要移动表中一半的元素。因此，在线性表顺序存储的情况下，要删除一个元素，其效率也是很低的，特别是在线性表比较大的情况下尤为突出。

1.4 栈和队列

*考点10 栈及其基本运算

栈是限定仅在一端进行插入和删除运算的线性表。栈顶元素总是最后被插入的元素，从而也是最先被删除的元素；栈底元素总是最先被插入的元素，从而也是最后才能被删除的元素。即栈是按照“先进后出”或“后进先出”的原则来组织数据的。

栈的基本运算有3种：入栈、退栈及读栈顶元素。

*考点11 队列及其基本运算

队列是指允许在一端进行插入而在另一端进行删除运算的线性表。队列又称为“先进先出”或“后进后出”的线性表，它体现了“先来先服务”的原则。在队列中，队尾指针与队首指针共同反映了队列中元素的动态变化情况。

在循环队列结构中，当存储空间最后一个位置已被使用且要进行入队运算时，只要存储空间的第一个位置空闲，便可将元素加入到第一个位置，即将存储空间的第一个位置作为队尾。循环队列的初始状态为空。循环队列主要有两种基本运算：入队运算和退队运算。每进行一次入队运算，队尾指针就进一；每进行一次退队运算，队首指针就进一。

1.5 线性链表

*考点12 线性链表的基本概念

在链式存储结构中，存储数据结构的存储空间可以不连续，各数据结点的存储顺序与数据元素之间的逻辑关系可以不一致，而数据元素之间的逻辑关系是由指针域来确定的。链式存储方式既可用于表示线性结构，也可用于表示非线性结构。

一般来说，在线性表的链式存储结构中，各数据结点的存储序号是不连续的，并且各结点在存储空间中的位置关系与逻辑关系也不一致。对于线性链表，可以从头指针开始，沿各结点的指针依次访问链表中的所有结点。

考点13 线性链表的基本运算

1. 在线性链表中查找指定元素

在非空线性链表中寻找包含指定元素值 x 的前一个结点 p 的基本方法如下：从头指针所指向的结点开始向后沿指针依次扫描，直到后面已没有结点或下一个结点的数据域为 x 时为止。因此，由这种方法所找到的结点 p 有两种可能：当线性链表中存在包含元素 x 的结点时，则所找到的 p 为第一次遇到的包含元素 x 的前一个结点序号；当线性链表中不存在包含元素 x 的结点时，则所找到的 p 为线性链表中的最后一个结点序号。

2. 线性链表的插入

为了在线性链表中插入一个新元素，首先要给此元素分配一个新结点，以便用于存储这个元素的值。新结点可以从可用栈中取得，然后将存放新元素值的结点链接到线性链表

中的指定位置。多个线性链表可以共享栈，从而很方便地实现存储空间的动态分配。线性链表在插入过程中不发生数据元素移动的现象，只需改变有关结点的指针即可，从而提高了插入效率。

3. 线性链表的删除

为了在线性链表中删除包含指定元素的结点，首先要在线性链表中找到这个结点，然后将要删除的结点放回可用栈。在线性链表中删除一个元素后，无须移动表的数据元素，只需改变被删除元素所在结点的前一个结点的指针域即可。当从线性链表中删除一个元素后，此元素的存储结点就变为空闲，应将这个空闲结点送回可用栈。

考点 14 循环链表及其基本运算

循环链表的结构与前面所讨论的线性链表相比，具有以下两个特点。

(1) 在循环链表中增加了一个表头结点，其数据域为任意值或者根据需要来设置，其指针域指向线性表的第一个元素的结点。循环链表的头指针指向表头结点。

(2) 循环链表中最后一个结点的指针域不为空，而是指向表头结点，即循环链表中所有结点的指针构成了一个环状链。

在循环链表中，只要指出表中任何一个结点的位置，就可以由它出发访问表中其他所有的结点，而线性单向链表做不到这一点。

1.6 树与二叉树

考点 15 树的基本概念

树是一种简单的非线性数据结构。在这种数据结构中，所有数据元素之间具有明显的层次关系。

树在计算机中通常用多重链表来表示。多重链表中各结点的链域个数不同，这将导致对树进行处理的算法会很复杂。如果用定长的结点来表示树中的每个结点，即取树的度作为每个结点的链域个数，就可以使得对树的各种处理算法大大简化。但是在这种情况下，仍会造成存储空间的浪费，因为有可能在很多结点中存在空链域。

*考点 16 二叉树及其基本性质

二叉树具有以下两个特点。

- (1) 非空二叉树只有一个根结点。
- (2) 每个结点最多有两棵子树，且分别称为此结点的左子树和右子树。

二叉树具有以下几个性质。

- (1) 在二叉树的第 k 层上，最多有 2^{k-1} ($k \geq 1$) 个结点。
- (2) 深度为 m 的二叉树最多有 $2^m - 1$ 个结点。
- (3) 在任意一棵二叉树中，度为 0 的结点总是比度为 2 的结点多一个。
- (4) 具有 n 个结点的二叉树，其深度至少为 $\lfloor \log_2 n \rfloor + 1$ ，其中 $\lfloor \log_2 n \rfloor$ 表示取 $\log_2 n$ 的整数部分。

在满二叉树中，每一层上的结点数都达到最大值，即在满二叉树的第 k 层上有 2^{k-1} 个结

点,且深度为 m 的满二叉树共有 $2^m - 1$ 个结点。

满二叉树一定是完全二叉树,而完全二叉树不一定是满二叉树。

考点17 二叉树的存储结构

在计算机中,二叉树通常采用链式存储结构。用于存储二叉树结点的存储地址的指针域有两个:一个用于指向此结点的左子结点的存储地址,称为左指针域;另一个用于指向此结点的右子结点的存储地址,称为右指针域。

*考点18 二叉树的遍历

二叉树的遍历是指不重复地访问二叉树中的所有结点。

在遍历二叉树的过程中,一般先遍历左子树,然后再遍历右子树。在先左后右的原则下,根据访问根结点的次序,二叉树的遍历可以分为3种:前序遍历、中序遍历、后序遍历。下面分别介绍这3种遍历方法。

1. 前序遍历

若二叉树为空,则直接结束并返回。否则:(1)访问根结点;(2)前序遍历左子树;(3)前序遍历右子树。

2. 中序遍历

若二叉树为空,则直接结束并返回。否则:(1)中序遍历左子树;(2)访问根结点;(3)中序遍历右子树。

3. 后序遍历

若二叉树为空,则直接结束并返回。否则:(1)后序遍历左子树;(2)后序遍历右子树;(3)访问根结点。

1.7 查找技术

*考点19 顺序查找

在进行顺序查找的过程中,如果线性表中的第一个元素就是所要查找的元素,则只需进行一次比较就查找成功,查找效率最高;但如果被查找的元素是线性表中的最后一个元素,或者被查找元素根本不在线性表中,则为了查找这个元素,需要与线性表中的所有元素进行比较,这是顺序查找的最坏情况。在平均情况下,利用顺序查找法在线性表中查找一个元素,大约要与线性表中一半的元素进行比较。

*考点20 二分法查找

二分法查找只适用于顺序存储的有序表。所谓的有序表是指线性表中的元素按值非递减的顺序排列(即从小到大,但允许相邻元素的值相等)。二分法查找的效率要比顺序查找法高得多。可以证明,对于长度为 n 的有序线性表,在最坏情况下,二分法查找只需比较 $\log_2 n$ 次,而顺序查找法则需比较 n 次。

1.8 排序技术

*考点 21 交换类排序法

假设线性表的长度为 n ，则在最坏情况下，冒泡排序需要经过 $n/2$ 遍的从前向后的扫描和 $n/2$ 遍的从后向前的扫描，所需要的比较次数为 $n(n-1)/2$ 。但这个工作量并不是必需的，一般情况下都要小于这个工作量。

快速排序法的关键在于对线性表进行分割，并对所分割出的子表再进行分割。随着对各子表的不断分割，划分出来的子表会越来越多，但每次只能对一个子表进行再分割处理，需要将暂时不分割的子表记忆起来，这就要通过一个栈来实现。

*考点 22 插入类排序法

在简单插入排序法中，每一次比较后最多移掉一个逆序，因此，这种排序方法的效率与冒泡排序法相同。在最坏情况下，简单插入排序法需要 $n(n-1)/2$ 次比较。

希尔排序的效率与所选取的增量序列有关。

*考点 23 选择类排序法

简单选择排序法在最坏情况下需要比较 $n(n-1)/2$ 次。

堆排序法对于规模较小的线性表并不适合，但对于规模较大的线性表来说是很有效的。在最坏情况下，堆排序法所需要的比较次数为 $O(n\log_2 n)$ 。

1.9 经典题解

一、选择题

1. 下列叙述中，正确的是_____。(2007 年 9 月)

- A) 程序执行的效率与数据的存储结构密切相关 B) 程序执行的效率只取决于程序的控制结构
C) 程序执行的效率只取决于所处理的数据量 D) 以上 3 种说法都不对

解析：采用不同的存储结构，其数据处理的效率是不同的，程序执行的效率也就不同，因此选项 A 是正确的。程序执行的效率不仅与程序的控制结构有关，而且与所处理的数据量有关，因此选项 B、C、D 都不正确。

答案：A

2. 下列叙述中，正确的是_____。(2007 年 9 月)

- A) 数据的逻辑结构与存储结构必定是一一对应的
B) 由于计算机存储空间是向量式的存储结构，因此，数据的存储结构一定是线性结构
C) 程序设计语言中的数组一般是顺序存储结构，因此，利用数组只能处理线性结构
D) 以上 3 种说法都不对

解析：数据元素在计算机存储空间中的位置关系与其逻辑关系不一定相同，即数据的逻辑结构与存储结构不一定一一对应。

答案：D

3. 冒泡排序在最坏情况下的比较次数是_____。(2007 年 9 月)

- A) $n(n+1)/2$ B) $n\log_2 n$ C) $n(n-1)/2$ D) $n/2$

解析：假设线性表的长度为 n ，则在最坏情况下，冒泡排序需要经过 $n/2$ 遍的从前向后的扫描和 $n/2$ 遍的

从后向前的扫描, 所需要的比较次数为 $n(n-1)/2$ 。

答案: C

4. 一棵二叉树中共有 70 个叶子结点和 80 个度为 1 的结点, 则此二叉树中的总结点数为_____。(2007 年 9 月)

A) 219

B) 221

C) 229

D) 231

解析: 在任意一棵二叉树中, 度为 0 的结点总是比度为 2 的结点多一个, 故度为 2 的结点数为 $70 - 1 = 69$, 故总结点数为 $69 + 70 + 80 = 219$ 。

答案: A

5. 下列叙述中, 正确的是_____。(2007 年 4 月)

A) 算法的效率只与问题的规模有关, 而与数据的存储结构无关

B) 算法的时间复杂度是指执行算法所需要的计算工作量

C) 数据的逻辑结构与存储结构是一一对应的

D) 算法的时间复杂度与空间复杂度一定相关

解析: 算法的效率不仅与问题的规模有关, 而且与数据的存储结构有关。算法的时间复杂度是指执行算法所需要的计算工作量。各数据元素在计算机存储空间中的位置关系与其逻辑关系可能不同。算法的时间复杂度与空间复杂度不一定相关。

答案: B

6. 下列对队列的叙述中, 正确的是_____。(2007 年 4 月)

A) 队列属于非线性表

B) 队列按“先进后出”原则组织数据

C) 队列在队尾删除数据

D) 队列按“先进先出”原则组织数据

解析: 队列是指允许在一端进行插入而在另一端进行删除的线性表。在队列的这种数据结构中, 最先插入的元素将最先被删除, 反之, 最后插入的元素在最后才能被删除。因此, 队列又称为“先进先出”或“后进后出”的线性表, 它体现了“先来先服务”的原则。

答案: D

7. 对右图二叉树进行前序遍历的结果为_____。(2007 年 4 月)

A) DYBEAFCZX

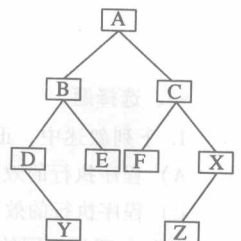
B) YDEBFZXCA

C) ABDYECFXZ

D) ABCDEFXYZ

解析: 前序遍历首先访问根结点, 然后遍历左子树, 最后遍历右子树; 并且在遍历左、右子树时, 仍然先访问根结点, 然后遍历其左子树, 最后遍历其右子树。因此, 前序遍历的结果是 ABDYECFXZ。

答案: C



8. 某二叉树中有 n 个度为 2 的结点, 则此二叉树中的叶子结点数为_____。(2007 年 4 月)

A) $n+1$

B) $n-1$

C) $2n$

D) $n/2$

解析: 在任意一棵二叉树中, 度为 0 的结点总是比度为 2 的结点多一个, 故度为 0 的结点数为 $n+1$ 。

答案: A

9. 下列叙述中, 正确的是_____。(2006 年 9 月)

A) 一个算法的空间复杂度大, 则其时间复杂度必定大

B) 一个算法的空间复杂度大, 则其时间复杂度必定小

C) 一个算法的时间复杂度大, 则其空间复杂度必定小

D) 上述 3 种说法都不对

解析: 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。一个算法的空间复杂度一般是指执行这个算法所需要的内存空间。在一个算法的空间复杂度大的情况下, 其时间复杂度只是有可能很大, 要视具体情况而定; 反之亦然。

答案: D