

- ★ 全真模拟实战演练
- ★ 上机考试全程辅导
- ★ 近年考题分类解析
- ★ 考点重点浓缩精解
- ★ 出题方向权威预测



全国计算机等级考试应试辅导丛书（新大纲）

**新
编**

二级公共基础知识

题眼分析与全真训练

计算机等级考试试题研究组 主编



人民邮电出版社
POSTS & TELECOM PRESS

全国计算机等级考试应试辅导丛书（新大纲）

新编二级公共基础知识 题眼分析与全真训练

计算机等级考试试题研究组 主编

人 民 邮 电 出 版 社

图书在版编目(CIP)数据

新编二级公共基础知识题眼分析与全真训练 / 计算机等级考试试题研究组主编.

—北京: 人民邮电出版社, 2005.6

(全国计算机等级考试应试辅导丛书: 新大纲)

ISBN 7-115-13472-3

I. 新... II. 计... III. 电子计算机—水平考试—自学参考资料 IV. TP3

中国版本图书馆 CIP 数据核字(2005)第 055560 号

内 容 提 要

本书根据教育部考试中心颁发的新大纲和指定教程, 以对考生进行综合指导为原则, 综合了历年考试题(常考题), 以及考前培训班教师的实际教学经验编写而成。

全书突出 3 个重点: 考点精讲、题眼分析与全真训练, 目的是让考生在较短时间内能快速提高应试能力, 顺利过关。本书共 5 章, 前 4 章分别介绍了数据结构与算法、程序设计基础、软件工程基础和数据库设计基础, 第 5 章提供了 10 套模拟试题及答案, 供考生自学与提高。

本书适合准备参加全国计算机等级考试(二级)的考生考前自学, 同时也可作为普通高校、成人高等教育及各类培训学校举办的考前二级辅导班的培训教材。

全国计算机等级考试应试辅导丛书(新大纲)

新编二级公共基础知识题眼分析与全真训练

- ◆ 主 编 计算机等级考试试题研究组
责任编辑 马雪伶
- ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号
邮编 100061 电子函件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
北京艺辉印刷有限公司印刷
新华书店总店北京发行所经销
- ◆ 开本: 787×1092 1/16
印张: 8
字数: 192 千字 2005 年 6 月第 1 版
印数: 1—8 000 册 2005 年 6 月北京第 1 次印刷

ISBN 7-115-13472-3/TP · 4690

定价: 14.00 元

读者服务热线: (010)67132692 印装质量热线: (010)67129223

前 言

全国计算机等级考试是目前国内影响最大、参加人数最多的计算机类水平考试。为了帮助广大考生顺利通过计算机等级考试，并全面提高考生的计算机应用水平，我们在深入剖析新考试大纲和历年考题的基础上编写了本丛书。

本丛书具有以下特色。

- ◇ **名师执笔，权威严谨：**丛书由从事全国计算机等级考试试题研究人员及在等级考试第一线从事命题研究、教学、辅导和培训的老师分工编写，层次清晰，结构严谨，导向准确。
- ◇ **一点一练，高效实用：**书的章名、节名与教育部考试中心指定教程同步，每节中分为“考点提炼”和“题眼分析”两个板块。
 - **考点提炼：**将指定的考试内容进行浓缩，精讲考试要点、重点与难点。
 - **题眼分析：**精选历年真题（常考题）进行解析，题型丰富，分析透彻。
- ◇ **全真模拟，实战提高：**根据新大纲、新考点、新题型进行命题。为了加强公共基础知识训练，本书提供 10 套（较丛书中的其他书多 5 套）全真模拟试题，供考生考前实战，感受全真训练。

本套丛书以对考生进行综合指导为原则，具有极强的针对性，特别适合希望在较短时间内取得较大收获的广大应试考生，也可作为全国计算机等级考试各类培训班的教材，以及大、中专院校师生的教学参考书。

丛书由计算机等级考试试题研究组主编，本书由张伍荣编写。另外，参与本书编排和校对的还有卢晓峰、李文龙、郑家琴、叶雪清、黄奕铭、丁院云、俞顺霖、凌明强、李海、丁善祥、许勇、汪伟、许明亚，在此一并致以衷心的感谢！

尽管我们精益求精，但书中难免存在错漏或不妥之处，敬请读者批评指正。

联系邮箱：NCREservice@126.com。

计算机等级考试试题研究组

目 录

第1章 数据结构与算法	1	2.2 题眼分析	38
1.1 算法	1	2.3 面向对象的程序设计	39
1.1.1 考点提炼	1	2.3.1 考点提炼	39
1.1.2 题眼分析	3	2.3.2 题眼分析	41
1.2 数据结构的基本概念	4	2.4 单元训练及参考答案	42
1.2.1 考点提炼	4	2.4.1 单元训练	42
1.2.2 题眼分析	6	2.4.2 参考答案	43
1.3 线性表及其顺序存储结构	7		
1.3.1 考点提炼	7	第3章 软件工程基础	44
1.3.2 题眼分析	9	3.1 软件工程基本概念	44
1.4 栈和队列	10	3.1.1 考点提炼	44
1.4.1 考点提炼	10	3.1.2 题眼分析	47
1.4.2 题眼分析	12	3.2 结构化分析方法	48
1.5 线性链表	13	3.2.1 考点提炼	48
1.5.1 考点提炼	13	3.2.2 题眼分析	50
1.5.2 题眼分析	16	3.3 结构化设计方法	52
1.6 树与二叉树	18	3.3.1 考点提炼	52
1.6.1 考点提炼	18	3.3.2 题眼分析	56
1.6.2 题眼分析	21	3.4 软件测试	58
1.7 查找技术	24	3.4.1 考点提炼	58
1.7.1 考点提炼	24	3.4.2 题眼分析	60
1.7.2 题眼分析	24	3.5 程序的调试	62
1.8 排序技术	25	3.5.1 考点提炼	62
1.8.1 考点提炼	25	3.5.2 题眼分析	63
1.8.2 题眼分析	29	3.6 单元训练及参考答案	64
1.9 单元训练及参考答案	31	3.6.1 单元训练	64
1.9.1 单元训练	31	3.6.2 参考答案	66
1.9.2 参考答案	34		
		第4章 数据库设计基础	67
第2章 程序设计基础	35	4.1 数据库系统的基本概念	67
2.1 程序设计方法与风格	35	4.1.1 考点提炼	67
2.1.1 考点提炼	35	4.1.2 题眼分析	71
2.1.2 题眼分析	36	4.2 数据模型	74
2.2 结构化程序设计	37	4.2.1 考点提炼	74
2.2.1 考点提炼	37	4.2.2 题眼分析	77

4.3 关系代数	80	5.1.6 模拟试题(六)	98
4.3.1 考点提炼	80	5.1.7 模拟试题(七)	100
4.3.2 题眼分析	83	5.1.8 模拟试题(八)	101
4.4 数据库设计与管理	84	5.1.9 模拟试题(九)	102
4.4.1 考点提炼	84	5.1.10 模拟试题(十)	103
4.4.2 题眼分析	88	5.2 模拟试题答案及解析	104
4.5 单元训练及参考答案	90	5.2.1 模拟试题(一)答案及解析	104
4.5.1 单元训练	90	5.2.2 模拟试题(二)答案及解析	106
4.5.2 参考答案	92	5.2.3 模拟试题(三)答案及解析	108
第5章 模拟试题及答案	93	5.2.4 模拟试题(四)答案及解析	110
5.1 模拟试题	93	5.2.5 模拟试题(五)答案及解析	112
5.1.1 模拟试题(一)	93	5.2.6 模拟试题(六)答案及解析	114
5.1.2 模拟试题(二)	94	5.2.7 模拟试题(七)答案及解析	115
5.1.3 模拟试题(三)	95	5.2.8 模拟试题(八)答案及解析	117
5.1.4 模拟试题(四)	96	5.2.9 模拟试题(九)答案及解析	119
5.1.5 模拟试题(五)	97	5.2.10 模拟试题(十)答案及解析	121

第 1 章 数据结构与算法

大纲要求

- ☑ 算法的基本概念，算法复杂度的概念和意义（时间复杂度与空间复杂度）。
- ☑ 数据结构的定义，数据的逻辑结构与存储结构，数据结构的图形表示，线性结构与非线性结构的概念。
- ☑ 线性表的定义，线性表的顺序存储结构及其插入与删除运算。
- ☑ 栈和队列的定义，栈和队列的顺序存储结构及其基本运算。
- ☑ 线性单链表、双向链表与循环链表的结构及其基本运算。
- ☑ 树的基本概念，二叉树的定义及其存储结构，二叉树的前序、中序和后序遍历。
- ☑ 顺序查找与二分法查找算法，基本排序算法（交换类排序法、选择类排序法和插入类排序法）。

1.1 算法

1.1.1 考点提炼

☐ 考点 1：算法的基本概念

所谓算法是指解题方案的准确而完整的描述。对于一个问题，如果可以通过一个计算机程序，在有限的存储空间内运行有限长的时间而得到正确的结果，则称这个问题是算法可解的。算法不等于程序，也不等于计算方法。

1. 算法的基本特征

- (1) 可行性 (effectiveness)：针对实际问题而设计的算法，执行后能够得到满意的结果。
- (2) 确定性 (definiteness)：算法中的每一个步骤都必须有明确的定义，不允许有模棱两可的解释和多义性。
- (3) 有穷性 (finiteness)：算法必须在有限时间内做完，即算法必须能在执行有限个步骤之后终止。

(4) 拥有足够的情报：要使算法有效必须为算法提供足够的情报。当算法拥有足够的情报时，此算法才是有效的；而当提供的情报不够时，算法可能无效。

综上所述，算法是一组严谨地定义运算顺序的规则，并且每一个规则都是有效的，同时是明确的，此顺序将在有限的次数后终止。

2. 算法的基本要素

(1) 算法中对数据的运算和操作：每个算法实际上是按解题要求从环境能进行的所有操作中选择合适的操作所组成的一组指令序列。

基本的运算和操作有以下4类：①算术运算（包括加、减、乘、除等）；②逻辑运算（包括“与”、“或”、“非”等运算）；③关系运算（包括“大于”、“小于”、“等于”、“不等于”等）；④数据传输（包括赋值、输入、输出等操作）。

（2）算法的控制结构：一个算法的功能不仅仅取决于所选用的操作，而且还与各操作之间的执行顺序有关。算法中各操作之间的执行顺序称为算法的控制结构。

算法的控制结构给出了算法的基本框架，它不仅决定了算法中各操作的执行顺序，也直接反映了算法的设计是否符合结构化原则。描述算法的工具通常有传统流程图、N-S结构化流程图和算法描述语言等。一个算法一般都以用顺序、选择和循环3种基本控制结构组合而成。

3. 算法设计的基本方法

计算机算法不同于人工处理的方法，工程上常用的算法有列举法和归纳法。在实际应用时，各种方法之间往往存在着一定的联系。

（1）列举法。

列举法的基本思想是根据提出的问题，列举所有可能的情况，并用问题中给定的条件检测哪些是需要的，哪些是不需要的。列举法常用于解决“是否存在”或“有多少种可能”等类型的问题。

（2）归纳法。

归纳法的基本思想是通过列举少量的特殊情况，经过分析，最后找出一般的关系。从本质上讲，归纳就是通过观察一些简单而特殊的情况，最后总结出一般性的结论。

（3）递推。

递推是指从已知的初始条件出发，逐次推出所要求的各中间结果和最后结果。其中初始条件或是问题本身已经给定，或是通过对问题的分析与化简而确定。递推本质上属于归纳法，工程上许多递推关系式实际上是通过通过对实际问题的分析与归纳而得到的，因此，递推关系式往往是归纳的结果。对于数值型的递推算法必须要注意数值计算的稳定性问题。

（4）递归。

人们在解决一些复杂问题时，为了降低问题的复杂程度（如问题的规模等），一般总是将问题逐层分解，最后归结为一些最简单的问题。这种将问题逐层分解的过程，实际上并没有对问题进行求解，而只是当解决了最后那些最简单的问题后，再沿着原来的分解的逆过程逐步进行综合，这就是递归的基本思想。递归分为直接递归与间接递归两种。

（5）减半递推技术。

实际问题的复杂程度往往与问题的规模有着密切的联系，因此利用分治法解决这类实际问题是有效的。工程上常用的分治法是减半递推技术。

所谓“减半”，是指将问题的规模减半，而问题的性质不变；所谓“递推”，是指重复“减半”的过程。

（6）回溯法。

在工程上有些实际问题很难归纳出一组简单的递推公式或直观的求解步骤，并且也不能进行无限的列举。对于这类问题，一种有效的方法是“试”。通过对问题的分析，找出一个解决问题的线索，然后沿着这个线索逐步试探，若试探成功，就得到问题的解；若试探失败，就逐步回退，换其他路线再逐步试探。

考点 2: 算法复杂度

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。同一个算法使用不同的语言实现,或者用不同的编译程序进行编译,或者在不同的计算机上运行,效率均不同。这表明使用绝对的时间单位衡量算法的效率是不合适的。撇开这些与计算机硬件、软件有关的因素,可以认为一个特定算法“运行工作量”的大小,只依赖于问题的规模(通常用整数 n 表示),它是问题的规模函数。即:

$$\text{算法的工作量} = f(n)$$

分析算法的工作量通过采用以下两种方法来分析。

(1) 平均性态 (Average Behavior): 所谓平均性态分析,是指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。

(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。

2. 算法的空间复杂度

算法的空间复杂度是指执行算法所需要的内存空间。一个算法所占用的存储空间包括: ①算法程序所占的空间; ②输入的初始数据所占的存储空间; ③算法执行过程中所要的额外空间。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来说是常数,则称该算法是原地 (in place) 工作的。在许多实际问题中,为了减少算法所占的存储空间,通常采用压缩存储技术,以减少不必要的额外空间。

1.1.2 题眼分析

一、选择题分析

【例 1】在计算机中,算法是指_____。

- | | |
|-------------------|----------|
| (A) 查询方法 | (B) 加工方法 |
| (C) 解题方案的准确而完整的描述 | (D) 排序方法 |

题眼分析 计算机算法是指解题方案的准确而完整的描述,它有以下几个基本特征:可行性、确定性、有穷性和拥有足够的情报。

答案 C

【例 2】在下列选项中,_____不是一个算法一般应该具有的基本特征。

- | | |
|---------|-------------|
| (A) 确定性 | (B) 可行性 |
| (C) 无穷性 | (D) 拥有足够的情报 |

题眼分析 作为一个算法,一般应具有以下几个基本特征:可行性、确定性、有穷性和拥有足够的情报。

答案 C

【例 3】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 |

题眼分析 算法的时间复杂度是指执行算法所需要的计算工作量。为了能够比较客观地反映出一个算法的效率，在度量一个算法的工作量时，不仅应该与所使用的计算机、程序设计语言以及程序编制者无关，而且还应该与算法实现过程中的许多细节无关。为此，可以用算法在执行过程中所需基本运算的执行次数来度量算法的工作量。

答案 C

【例 4】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

题眼分析 一个算法的空间复杂度，一般是指执行这个算法所需的内存空间。一个算法所占用的存储空间包括算法程序所占的空间、输入的初始数据所占的存储空间以及算法执行过程中所需要的额外空间。

答案 D

二、填空题分析

【例 1】问题处理方案的正确而完整的描述称为_____。(2005 年 4 月)

题眼分析 算法是指解题方案的准确而完整的描述，它有以下几个基本特征：可行性、确定性、有穷性和拥有足够的情报。

答案 算法

【例 2】算法是对问题求解过程的一类精确描述，算法中描述的操作都可以通过已经实现的基本操作在限定时间内执行有限次来完成的，这句话说明算法应具有_____特性。

题眼分析 算法应满足有穷性、确定性、可行性、拥有足够的情报等基本特征，而有穷性即为在有限时间内完成算法。

答案 有穷性

1.2 数据结构的基本概念

1.2.1 考点提炼

☐ 考点 1：什么是数据结构

1. 数据结构研究的主要内容

数据结构作为计算机的一门学科，主要研究和讨论以下 3 方面的问题：

- (1) 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；
- (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。

2. 研究数据结构的目

研究数据结构的主要目的是为了提高数据处理的效率。提高数据处理的效率主要包括两个方面：一是提高数据处理的速

3. 数据结构的定义

数据结构是指相互有关联的数据元素的集合。数据元素之间的关系可以用前后件关系（或直接前驱与直接后继关系）来描述。一个数据结构应包含以下两方面信息：

- （1）表示数据元素的信息；
- （2）表示各数据元素之间的前后件关系。

□□考点 2：数据的逻辑结构和数据的存储结构

1. 数据的逻辑结构

数据的逻辑结构是对数据元素之间的逻辑关系的描述，它可以用一个数据元素的集合和定义在此集合中的若干关系来表示。数据的逻辑结构只抽象地反映数据元素之间的逻辑关系，即数据元素之间的前后件关系，而不管它在计算机中的存储表示形式。

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，它反映了数据元素之间的前后件关系，通常记为 R 。一个数据结构 B 可以表示为：

$$B = (D, R)$$

2. 数据的存储结构

数据的存储结构是数据的逻辑结构在计算机存储空间中的存放形式，也称数据的物理结构。一个数据结构中的各数据元素在计算机存储空间的位置与逻辑关系有可能不同。

一种数据结构可根据需要采用不同的存储结构。常用的存储结构有顺序、链接和索引等存储方式。采用不同的存储结构，其数据处理的效率是不同的。

□□考点 3：数据结构的图形表示

一个数据结构除了用二元关系表示外，还可以直观地用图形表示。在数据结构的图形表示中，对于数据集合 D 中的每一个数据元素用中间标有元素值的方框表示，一般称之为数据结点，并简称为结点；为了进一步表示各数据元素之间的前后件关系，对于关系 R 中的每一个二元组，用一条有向线段从前件结点指向后件结点。

在数据结构中，没有前件的结点称为根结点；没有后件的结点称为终端结点（也称为叶子结点）。

一个数据结构中的结点可能是在动态变化的。根据需要或在处理过程中，可以在一个数据结构中增加一个新结点（称为插入运算），也可以删除数据结构中的某个结点（称为删除运算）。除此之外，对数据结构的运算还有查找、分类、合并、分解、复制和修改等。

□□考点 4：线性结构与非线性结构

如果在一个数据结构中一个数据元素都没有，则称该数据结构为空的数据结构。

根据数据结构中各数据元素之间前后件关系的复杂程度，一般将数据结构分为两大类型：线性结构与非线性结构。

如果一个非空数据结构满足下列两个条件：

- （1）有且只有一个根结点；
- （2）每一个结点最多有一个前件，也最多有一个后件，则称该数据结构为线性结构，线性结构又称为线性表。例如，线性表、栈、队列和线性链表都是线性结构。

如果一个数据结构不是线性结构，称之为非线性结构。例如树、二叉树和图都是非线性结构。

线性结构与非线性结构都可以是空的数据结构。对于空的数据结构，如果对该数据结构的运算是按线性结构的规则来处理的，则属于线性结构，否则属于非线性结构。

1.2.2 题眼分析

一、选择题分析

【例 1】数据结构中，与所使用的计算机无关的是数据的_____。

- (A) 存储结构
- (B) 物理结构
- (C) 逻辑结构
- (D) 物理和存储结构

题眼分析 数据结构概念一般包括 3 个方面的内容，数据的逻辑结构、存储结构及数据上的运算集合。数据的逻辑结构只抽象地反映数据元素之间的逻辑关系，即数据元素之间的前后件关系，而不管它在计算机中的存储表示形式。

答案 C

【例 2】数据的存储结构是指_____。(2005 年 4 月)

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

题眼分析 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(也称数据的物理结构)。通常有两类存储结构：一类是顺序存储结构；另一类是链式存储结构。

答案 D

【例 3】在数据结构中，根据各数据元素之间前后件关系的复杂程度，一般将数据结构分成_____两类。

- (A) 动态结构和静态结构
- (B) 紧凑结构和非紧凑结构
- (C) 线性结构和非线性结构
- (D) 内部结构和外部结构

题眼分析 数据的逻辑结构抽象地反映数据元素之间的逻辑关系，即数据元素之间的前后件关系，而不管它在计算机中的存储表示形式。根据数据结构中各数据元素之间前后件关系的复杂程度，一般将数据结构分为两大类型：线性结构与非线性结构。

答案 C

二、填空题分析

【例 1】数据结构包括数据的_____结构和数据的存储结构。

题眼分析 数据结构包括 3 个方面，即数据的逻辑结构、数据的存储结构及对数据的操作运算。数据的逻辑结构抽象地反映数据元素之间的逻辑关系，即数据元素之间的前后件关系，而数据的存储结构是数据的逻辑结构在计算机存储空间中的存放形式。

答案 逻辑

【例 2】如果一个非空数据结构满足下列两个条件：一是有且只有一个根结点；二是每一个结点最多有一个前件，也最多有一个后件，则称该数据结构为_____。

题眼分析 数据的逻辑结构抽象地反映数据元素之间的逻辑关系,即数据元素之间的前后件关系,而不管它在计算机中的存储表示形式。根据数据结构中各数据元素之间前后件关系的复杂程度,一般将数据结构分为两大类型:线性结构与非线性结构。题干中所描述的数据结构是线性结构。

答案 线性结构

1.3 线性表及其顺序存储结构

1.3.1 考点提炼

□ 考点 1: 线性表的基本概念

线性表是由 n ($n \geq 0$) 个数据元素 $a_1, a_2, \dots, a_n$ 组成的有限序列,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,通常也称其为线性表中的一个结点。

在非空表中的每个数据元素都有一个确定的位置,如 a_1 是第一个元素, a_n 是最后一个数据元素, a_i 是第 i 个数据元素,称 i 为数据元素 a_i 在线性表中的位序。

非空线性表有如下一些结构特征:①有且只有一个根结点 a_1 ,它无前件;②有且只有一个终端结点 a_n ,它无后件;③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。线性表中结点的个数 n 称为线性表的长度。当 $n=0$ 时,称为空表。

□ 考点 2: 线性表的顺序存储结构

线性表的顺序存储结构指的是用一组地址连续的存储单元依次存放线性表中的数据元素。

线性表的顺序存储结构具备如下两个基本特征:①线性表中的所有元素所占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。

假设线性表中的第一个数据元素的存储地址(指第一个字节的地址,即首地址)为 $ADR(a_1)$,每个数据元素占 k 个字节,则线性表中第 i 个元素在计算机存储空间的存储地址为

$$ADR(a_i) = ADR(a_1) + (i-1)k$$

在线性表的顺序存储结构下可以对线性表做以下运算:插入、删除、查找、排序、分解、合并、复制和逆转等。

□ 考点 3: 顺序表的插入运算

线性表的插入运算是指在表的第 i ($1 \leq i \leq n+1$) 个位置元素之前,插入一个新结点 x ,使长度为 n 的线性表 $(a_1, \dots, a_{i-1}, a_i, \dots, a_n)$ 变成长度为 $n+1$ 的线性表 $(a_1, \dots, a_{i-1}, x, a_i, \dots, a_n)$ 。对顺序表而言,需要改变从第 i 个元素起到第 n 个元素的存储位置,即从第 i 到第 n 个元素往后移动一个位置,共需移动 $n-i+1$ 个元素。

令 $E_{is}(n)$ 表示在长度为 n 的顺序表中进行一次插入操作时所需进行“移动”元素个数的

期望值（即平均移动个数），则：

$$E_{is} = \sum_{i=1}^{n+1} p_i(n-i+1)$$

其中， p_i 是在第 i 个元素之前插入一个元素的概率， $n-i+1$ 是在第 i 个元素之前插入一个元素时所需移动的元素个数。由于可能插入的位置 $i=1,2,3,\dots,n+1$ 共 $n+1$ 个，假设在每个位置上插入的机会均等，则：

$$p_i = \frac{1}{n+1}$$

由此，在上述等概率假设的情况下，

$$E_{is} = \frac{1}{n+1} \sum_{i=1}^{n+1} p_i(n-i+1) = \frac{n}{2}$$

□考点 4：顺序表的删除运算

线性表的删除运算是指将表的第 i ($1 \leq i \leq n$) 个结点删去，使长度为 n 的线性表 $(a_1, \dots, a_{i-1}, a_i, a_{i+1}, \dots, a_n)$ 变成长度为 $n-1$ 的线性表 $(a_1, \dots, a_{i-1}, a_{i+1}, \dots, a_n)$ 。对顺序表而言，需要改变从第 $i+1$ 个元素起到第 n 个元素的存储位置，即从第 $i+1$ 到第 n 个元素往前移动一个位置，共需移动 $n-i$ 个元素。

令 $E_{dl}(n)$ 表示在长度为 n 的顺序表中进行一次删除操作时所需进行“移动”元素个数的期望值（即平均移动个数），则

$$E_{dl} = \sum_{i=1}^n q_i(n-i)$$

其中， q_i 是删除第 i 个元素的概率， $n-i$ 是删除第 i 个元素时所需移动元素的个数。同样假设在 n 个可能进行删除的位置 $i=1,2,\dots,n$ 机会均等，则

$$q_i = \frac{1}{n}$$

由此，在上述等概率的假设下，

$$E_{dl} = \frac{1}{n} \sum_{i=1}^n p_i(n-i) = \frac{n-1}{2}$$

由此可见，在以顺序方式存储的线性表中插入或删除一个数据元素，平均约需移动表中一半元素。这个数目在线性表的长度较大时是很可观的。这个缺陷完全是由于顺序存储要求线性表的元素依次紧挨存放造成的。因此，顺序存储表仅适用于不常进行插入和删除操作、表中元素相对稳定的线性表。

1.3.2 题眼分析

一、选择题分析

【例1】下列叙述中，错误的是_____。

- (A) 线性表是由 n 个数据元素组成的一个有限序列
- (B) 线性表是一种线性结构
- (C) 线性表的所有结点有且只有一个前件和一个后件
- (D) 线性表可以是空表

题眼分析 线性表是由 n ($n \geq 0$) 个数据元素 $a_1, a_2, \dots, a_n$ 组成的有限序列，线性表是一个线性结构，线性表也可以是空表，因此选项 A、B、D 都是正确说法。在线性表中第一个结点没有前件，最后一个结点没有后件，其他所有结点有且只有一个前件和一个后件，因此选项 C 的说法是错误的，是答案。

答案 C

【例2】某线性表采用顺序存储结构，每个元素占 4 个存储单元，首地址为 200，则第 12 个元素的存储地址为_____。

- (A) 248
- (B) 247
- (C) 246
- (D) 244

题眼分析 设线性表中的第一个数据元素的存储地址（指第一个字节的地址，即首地址）为 $ADR(a_1)$ ，每个数据元素占 k 个字节，则线性表中第 i 个元素在计算机存储空间中的存储地址为

$$ADR(a_i) = ADR(a_1) + (i-1)k$$

因此， $ADR(a_{12}) = 200 + (12-1)4 = 244$ 。

答案 D

【例3】在长度为 n 的顺序表的第 i ($1 \leq i \leq n+1$) 个位置上插入一个元素，元素的移动次数为_____。

- (A) $n-i+1$
- (B) $n-i$
- (C) i
- (D) $i-1$

题眼分析 在长度为 n 的顺序表中插入一个元素，有 $n+1$ 个可供插入位置。设插入位置为 i ，当 $i=n+1$ 时，不需要移动元素，即移动次数为 0；当 $i=n$ 时，需将位置 n 上的元素即表尾元素向后移动一个位置，即移动次数为 1；当 $i=n-1$ 时，需将 n 和 $n-1$ 的元素依次向后移动一个位置，即移动次数为 2……当 $i=1$ 时，需将全部元素依次向后移动一个位置，即移动次数为 n 。由此可知，在第 i ($1 \leq i \leq n+1$) 个位置上插入一个元素，元素的移动次数为 $n-i+1$ 。

答案 A

【例4】在一个长度为 n 的顺序表中，删除第 i ($1 \leq i \leq n$) 个元素时，需要移动的元素个数为_____。

- (A) $n-i+1$
- (B) $n-i$
- (C) i
- (D) $i-1$

题眼分析 分析方法同【例3】，只不过在删除一个元素时，有 n 个位置可供选择。需要改变从第 $i+1$ 个元素起到第 n 个元素的存储位置，即从第 $i+1$ 到第 n 个元素往前移动一个位置，共需移动 $n-i$ 个元素。

答案 B

【例 5】以下描述中，不是线性表的顺序存储结构的特征的是_____。

(A) 不便于插入和删除

(B) 需要连续的存储空间

(C) 可随机访问

(D) 需另外开辟空间来保存元素之间的关系

题眼分析 线性表的顺序存储是用一片连续空间来存放数据元素，其特点是逻辑上相邻的元素在物理位置上也相邻，数据元素之间逻辑上的先后关系自动隐含在物理位置的相邻之中，因此不需另外开辟空间来保存元素之间的关系。

答案 D

二、填空题分析

【例 1】在线性表的顺序存储结构中，若一个元素的下标为 i ，则它的前件下标为 (1)，后件的下标为 (2)。

题眼分析 在线性表的顺序存储结构中，元素之间的先后关系是通过下标来表示的。下标为 i 的元素，它的前件的下标为 $i-1$ ，后件的下标为 $i+1$ 。

答案 (1) $i-1$

(2) $i+1$

【例 2】对于长度为 n 的顺序表进行插入运算和删除运算，可能会引起元素的移动。在等概率情况下，插入操作平均需要移动 (1) 个元素；删除操作平均需要移动 (2) 个元素。

题眼分析 参见 1.3.1 节考点 3 和考点 4。

答案 (1) $n/2$

(2) $(n-1)/2$

1.4 栈和队列

1.4.1 考点提炼

□ 考点 1：栈及其基本运算

1. 栈的定义

栈 (Stack) 是限定只能在表的一端进行插入和删除操作的线性表。在表中，允许插入和删除的一端称做“栈顶 (top)”，不允许插入和删除的另一端称做“栈底 (bottom)”。不含元素的空表称为空栈。栈的修改是按后进先出的原则进行的。因此，栈又称为先进后出 (FILO, First In Last Out) 或后进先出 (LIFO, Last In First Out) 的线性表，如图 1-1 所示。

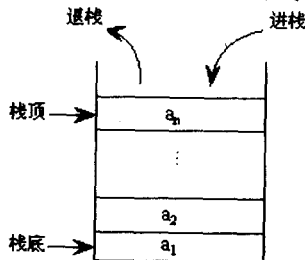


图 1-1 栈

2. 栈的顺序存储及其运算

在程序设计语言中，用一维数 $S(1:m)$ 作为栈的顺序存储空间，其中 m 为栈的最大容量。

$S(\text{bottom})$ 通常为栈底元素(栈非空情况下), $S(\text{top})$ 为栈顶元素。 $\text{top}=0$ 时表示栈空, $\text{top}=m$ 时表示栈满。

(1) 入栈运算: 入栈运算是指在栈顶位置插入一个新元素。首先将栈顶指针加 1 (即 top 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, 这种情况称为栈“上溢”错误。

(2) 退栈运算: 退栈是指取出栈顶元素并赋予一个指定的变量。首先将栈顶元素(栈顶指针指向的元素)赋予一个指定的变量, 然后将栈顶指针减 1 (即 top 减 1)。当栈顶指针为 0 时, 说明栈空, 不可进行退栈操作, 这种情况称为栈的“下溢”错误。

(3) 读栈顶元素: 读栈顶元素是指将栈顶元素赋予一个指定的变量。这个运算不删除栈顶元素, 只是将它赋予一个变量, 因此栈顶指针不会改变。当栈顶指针为 0 时, 说明栈空, 读不到栈顶元素。

□考点 2: 队列及其基本运算

队列 (Queue) 是限定只能在表的一端进行插入和在另一端进行删除操作的线性表。在表中, 允许插入的一端称做“队尾 (rear)”, 允许删除的另一端称做“队头 (front)”。队列的修改是依“先进先出”的原则进行的, 因此队列又称先进先出 (FIFO, First In First Out) 表或后进后出 (LILO, Last In Last Out) 线性表, 如图 1-2 所示。

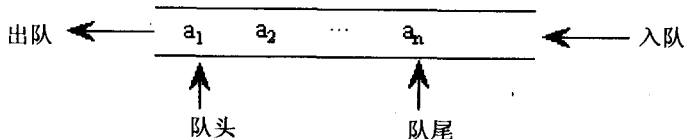


图 1-2 队列

往队列的队尾插入一个元素称为入队运算, 从队列的排头删除一个元素称为退队运算。入队运算只涉及队尾指针 rear 的变化, 而退队运算只涉及到队头指针 front 的变化。

□考点 3: 循环队列及其基本运算

在实际应用中, 队列的顺序存储结构一般采用循环队列的形式。所谓循环队列, 就是将队列存储空间最后一个位置绕到第一个位置, 形成逻辑上的环状空间, 供队列循环使用。

在循环队列中, 用队尾指针 rear 指向队列中的队尾元素, 用排头指针 front 指向排头元素的前一个位置, 因此, 从排头指针 front 指向的后一个位置直到队尾指针 rear 指向的位置之间所有的元素均为队列中的元素。

循环队列的初始状态为空, 即 $\text{rear}=\text{front}=m$ 。

当循环队列满时有 $\text{rear}=\text{front}$, 而当循环队列空时也有 $\text{rear}=\text{front}$, 即在循环队列中当 $\text{rear}=\text{front}$ 时, 不能确定是队列满还是队列空。在实际使用循环队列时, 为了能区分队列满还是队列空, 通常还需增加一个标志 s , $s=0$ 表示队列空, $s=1$ 表示队列非空。

循环队列主要有两种基本运算: 入队运算与退队运算。

(1) 入队运算。

入队运算是指在循环队列的队尾加入一个新元素。这个运算有两个基本操作: 首先将队尾