

飞思考试中心
Fecit Examination Center



全国计算机等级考试命题研究组 编著
飞思教育产品研发中心 联合监制
未来教育教学与研究中心

全国计算机等级考试指定教材配套辅导
National Computer Rank Examination

全国计算机等级考试

教程同步辅导

(二级公共基础知识)



多媒体视频课堂

以“例题+讲解+链接”的形式对考试知识点进行互动性的再现，犹如老师实时亲临指导。



电子工业出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
<http://www.phei.com.cn>

飞思考试中心

全国计算机等级考试教程同步辅导 (二级公共基础知识)

全国计算机等级考试命题研究组 编著

飞思教育产品研发中心

联合监制

未来教育教学与研究中心

电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

内 容 简 介

计算机等级考试在新大纲的标准下实施,为了向考生提供更专业、更实用、更具针对性的服务,全国计算机等级考试命题研究组和未来教育教学与研究中心联合设计、开发了本系列图书。

在对新大纲与教程进行了深入研究之后,精心设计了这本符合考生需要的图书。书中内容包括:“重点、难点透析”、“易错题型讲解”、“同步训练”、“综合测试题”和“教程配套习题精解”等,科学搭配,名师指导,为考生提供全方位的辅导,同时还配以多媒体教学光盘,辅助学习,增加过关机会。

本书既可以作为计算机等级考试考生的自学用书,也可以作为计算机等级考试培训班的教学参考书和辅导用书。

未经许可,不得以任何方式复制或抄袭本书的部分或全部内容。

版权所有,侵权必究。

图书在版编目(CIP)数据

全国计算机等级考试教程同步辅导(二级公共基础知识)/全国计算机等级考试命题研究组编著. —北京:电子工业出版社,2006.1

(飞思考试中心)

ISBN 7-121-01901-9

I. 全... II. 全... III. ①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 126158 号

责任编辑:李泽才

印 刷:北京中科印刷有限公司

出版发行:电子工业出版社

北京海淀区万寿路 173 信箱 邮编:100036

经 销:各地新华书店

开 本:880×1230 1/16 印张:13.75 字数:396 千字

印 次:2006 年 1 月第 1 次印刷

定 价:27.00 元(含光盘 1 张)

凡购买电子工业出版社的图书,如有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系,联系电话:010-68279077。质量投诉请发邮件至 zltz@phei.com.cn,盗版侵权举报请发邮件至 dbqq@phei.com.cn。

本书是根据《全国计算机等级考试教程》的教学要求以及最新修订的《全国计算机等级考试考试大纲》的要求，由命题专家集3年的心血编写而成，知识覆盖面广，考点命中率高。本书的特色如下：

特色一 教程配套辅导

本书是根据最新考试大纲和教程进行编写的。书中每章每节都与教程严格对应，既对教程的内容进行了重点讲解，又对教程中相应章节后的习题进行了详细的分析，以弥补教程中只有题目没有答案，或只有答案没有解析的缺憾。

特色二 不是教程而等价教程

如果你没有购买教程，本书可以作为你的等级考试教程使用。本书将考试中要求的知识点全部讲到，只要你将本书中的全部知识点真正掌握，对所有的题目能够真正理解，考试过关应该是没问题的。所以本书可以与教程等价使用。

特色三 巨量练习伴随你

在每节结束时，均有“同步训练”题目进行练习，从而对该节中的知识点进行巩固。在每章结束时，通过“综合测试题”对该章的知识点进行加深和巩固，真正达到以学代练、以练促学的目的。

特色四 赠送多媒体光盘

随书赠送多媒体光盘。该部分是以“多媒体课堂”的形式对综合例题进行讲解。例题中包括多个知识点，对每个知识点都有相应的链接（对知识点的详细说明），举一反三，综合进行练习。每题都有详细的解题步骤、源程序和解析等，让你知其所以然。

本书既可以作为计算机等级考试的考生自学之用，也可以作为计算机等级考试培训班的教学参考书和辅导用书。由于时间仓促，水平有限，书中的疏漏或错误之处在所难免，在此恳请广大读者不吝赐教。

全国计算机等级考试命题研究组
未来教育教学与研究中心

目 录

CONTENTS

第1章 数据结构与算法

1.1 算法	3
重点、难点透析	3
易错题型讲解	5
同步训练	5
同步训练答案及解析	6
1.2 数据结构的基本概念	6
重点、难点透析	6
易错题型讲解	7
同步训练	8
同步训练答案及解析	8
1.3 线性表及其顺序存储结构	9
重点、难点透析	9
易错题型讲解	11
同步训练	11
同步训练答案及解析	12
1.4 栈和队列	13
重点、难点透析	13
易错题型讲解	14
同步训练	16
同步训练答案及解析	16
1.5 线性链表	17
重点、难点透析	17
易错题型讲解	19
同步训练	20
同步训练答案及解析	21
1.6 树与二叉树	22
重点、难点透析	22
易错题型讲解	24
同步训练	28
同步训练答案及解析	30
1.7 查找技术	32
重点、难点透析	32
易错题型讲解	32
同步训练	33
同步训练答案及解析	34
1.8 排序技术	35

重点、难点透析	35
易错题型讲解	37
同步训练	40
同步训练答案及解析	41

综合测试题一	43
--------	----

第2章 程序设计基础

2.1 程序设计基础	57
重点、难点透析	57
易错题型讲解	58
同步训练	59
同步训练答案及解析	59
2.2 结构化程序设计	60
重点、难点透析	60
易错题型讲解	61
同步训练	62
同步训练答案及解析	63
2.3 面向对象的程序设计	64
重点、难点透析	64
易错题型讲解	66
同步训练	68
同步训练答案及解析	69
综合测试题二	71

第3章 软件工程基础

3.1 软件工程基本概念	77
重点、难点透析	77
易错题型讲解	80
同步训练	81
同步训练答案及解析	82
3.2 结构化分析方法	83
重点、难点透析	83
易错题型讲解	86
同步训练	88
同步训练答案及解析	88
3.3 结构化设计方法	90
重点、难点透析	90
易错题型讲解	96

同步训练	99
同步训练答案及解析	99
3.4 软件测试	101
重点、难点透析	101
易错题型讲解	104
同步训练	106
同步训练答案及解析	106
3.5 程序的调试	107
重点、难点透析	107
易错题型讲解	108
同步训练	110
同步训练答案及解析	111
综合测试题三	112

第4章 数据库设计基础

4.1 数据库系统的基本概念	119
重点、难点透析	119
易错题型讲解	123
同步训练	128
同步训练答案及解析	130
4.2 数据模型	133
重点、难点透析	133
易错题型讲解	138
同步训练	140
同步训练答案及解析	141
4.3 关系代数	143
重点、难点透析	143
易错题型讲解	145
同步训练	146
同步训练答案及解析	147
4.4 数据库设计与管理	148
重点、难点透析	148
易错题型讲解	151
同步训练	152
同步训练答案及解析	152
综合测试题四	153

第5章 全真模拟试题

5.1 笔试全真模拟试题(1)	169
-----------------------	-----

5.2 笔试全真模拟试题(2)	170
5.3 笔试全真模拟试题(3)	171
5.4 笔试全真模拟试题(4)	172
5.5 笔试全真模拟试题(5)	173
5.6 笔试全真模拟试题(6)	174
5.7 笔试全真模拟试题(7)	175
5.8 笔试全真模拟试题(8)	176
5.9 笔试全真模拟试题(9)	177
5.10 笔试全真模拟试题(10)	178
5.11 笔试全真模拟试题(11)	179
5.12 笔试全真模拟试题(12)	179
5.13 笔试全真模拟试题(13)	180
5.14 参考答案及解析	181

附 录

附录 A 教程配套习题精解	199
附录 B 综合测试题一~试题四答案	204
附录 C 2005 年 9 月二级公共基础知识笔试试题	210



第1章

数据结构与算法



本章知识网络图

算法	算法的基本概念	★
	算法复杂度	★★
数据结构的基本概念	数据结构及图形表示	★
	线性结构与非线性结构	★
线性表及其顺序存储结构	线性表的基本概念	★
	线性表的顺序存储结构	★★
	顺序表的插入运算及删除运算	★
栈和队列	栈及其基本运算	★★★
	队列及其基本运算	★★★
线性链表	线性链表的基本概念	★
	线性链表的基本运算	★★
	循环链表及其基本运算	★
树与二叉树	树的基本概念	★
	二叉树及其基本性质	★★★
	二叉树的存储结构及遍历	★★★
查找技术	顺序查找法	★★
	二分查找法	★★
排序技术	交换类排序	★★
	插入类排序	★★
	选择类排序	★★

1.1 算 法

重点、难点透析

1. 算法的基本特征

计算机解题的过程实际上是在实施某种算法,这种算法称为计算机算法。算法(algorithm)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,且是明确的,此顺序将在有限的次数后终止。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,其中每一条指令表示一个或多个操作。

算法是指解题方案的准确而完整的描述。一般具有以下几个基本特征:

(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)

算法的每一步骤都必需具有明确定义的解释,不允许有多义性。

(3) 有穷性(finiteness)

算法必需在有限时间内做完,即算法必需能在执行有限个步骤之后终止。算法的有穷性还应包括合理的执行时间的含义。

(4) 拥有足够的情报

要使算法有效必需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,且是明确的;此顺序将在有限的次数下终止。

2. 算法的基本要素

一个算法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。

(1) 算法中对数据的运算和操作

在一般的计算机系统中,基本的运算和操作有以下4类:

① 算术运算。主要包括加、减、乘、除等运算。

② 逻辑运算。主要包括“与”、“或”、“非”等运算。

③ 关系运算。主要包括“大于”、“小于”、“等于”、“不等于”等运算。

④ 数据传输。主要包括赋值、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。描述算法的工具通常有传统流程图、N-S结构化流程图、算法描述语言等。一个算法一般都可以用顺序、选择、循环三种基本控制结构组合而成。

3. 算法设计基本方法

计算机解题的过程实际上是在实施某种算法,这种算法称为计算机算法。工程上常用的几种算法设计方法如下:

(1) 列举法

列举法是计算机算法中的一个基础算法。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验哪些是需要的,哪些是不需要的。

列举法的特点是算法比较简单。但当列举的可能情况较多时,执行列举算法的工作量将会很大。因此,在用列举法设计算法时,使方案优化,尽量减少运算工作量,是应该给以重点注意的。

(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分析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。如果一个算法显式地调用自己则称为直接递归,如果通过另一算法调用自己则称为间接递归。

(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,对于每一步的试探,若试探成功,就得到问题的解,若试探失败,就逐步回退,换别的路线逐步试探。

4. 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。

(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。

算法的工作量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输入时,可以用以下两种方法来分析算法的工作量。

① 平均性态 (Average Behavior)

所谓平均性态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。

设 x 是有可能输入中的某个特定输入, $p(x)$ 是 x 出现的概率(即输入为 x 的概率), $t(x)$ 是算法在输入为 x 时所执行的基本运算次数,则算法的平均性态定义为

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

其中 D_n 表示当规模为 n 时,算法执行的所有可能输入的集合。

② 最坏情况和复杂情况 (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

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

(2) 算法的空间复杂度

算法的空间复杂度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。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。

易错题型讲解

一、选择题

例题 1 算法的时间复杂度取决于_____。

- A) 问题的规模
C) 问题的难度

- B) 待处理的数据的初态
D) A) 和 B)

【答案】D)

【解题要点】算法的时间复杂度不仅与问题的规模有关,在同一个问题规模下,而且与输入数据有关。即与输入数据所有的可能取值范围、输入各种数据或数据集的概率有关。

【错解分析】算法的时间复杂度与问题的难度无关,而是与问题的规模和数据初态有关,所以 C) 选项错误, A) 选项和 B) 选项正确。

例题 2 用顺序搜索法在长度为 n 的一维数组中查找值为 x 的元素,在平均情况下需要做的比较次数为

$$A(n) = \sum_{i=1}^{n+1} P_i t_i = \sum_{i=1}^n (q/n) i + (1-q)n = (n+1)q/2 + (1-q)n$$

已知需要查找的 x 一定在数组中,此时 $A(n) =$ _____。

A) $(n+1)/2$

B) $n/2$

C) $3n/4$

D) n

【答案】A)

【解题要点】如果查找的 x 一定在数组中,此时 $q=1$,则 $A(n) = (n+1)/2$ 。也就是说,在这种情况下,用顺序搜索法在长度为 n 的一维数组中查找值为 x 的元素,在平均的情况下需要检查数组中一半的元素。

【错解分析】如果已知需要查找的 x 有一半机会在数组中,此时 $q=1/2$ 。则 $A(n) = [(n+1)/4] + n/2 \approx 3n/4$ 。 x 不在数组中时, $A(n) = n$ 。

二、填空题

例题 1 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的算法设计方法是_____。

【答案】归纳法

【解题要点】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。

例题 2 如果算法 P 调用另一个算法 Q ,而算法 Q 又调用算法 P ,则称为_____。

【答案】间接递归调用

【解题要点】如果一个算法显式地调用自己则称为直接递归,如果通过另一算法调用自己为间接递归调用。

同步训练

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

2. 利用分治法解决实际问题的复杂度和规模的算法,工程上常用的分治法是_____。

- A) 归纳法 B) 递推法 C) 回溯法 D) 减半递推技术

二、填空题

1. 算法的复杂度主要包括时间复杂度和_____。

2. 在 $N \times N$ 矩阵相乘的算法中, 整个算法的执行时间与该基本操作 (乘法) 重复执行的次数 n^3 成正比, 时间复杂度为_____。

同步训练答案及解析

一、选择题

1. 【答案】C)

【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。算法的工作量用算法所执行的基本运算次数来度量。

2. 【答案】D)

【解析】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。

二、填空题

1. 【答案】空间复杂度

【解析】算法的复杂度主要包括时间复杂度和空间复杂度。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; 算法的空间复杂度, 一般是指执行这个算法所需要的内存空间。

2. 【答案】 $O(n^3)$

【解析】所谓算法时间复杂度, 是指执行算法所需要的计算工作量。算法工作量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, 即时间复杂度 $f(n) = O(n^3)$ 。

1.2 数据结构的基本概念

重点、难点透析

1. 数据结构的概念

数据结构作为计算机的一门学科, 主要研究和讨论以下三个方面的问题:

- (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计算。

讨论以上问题的主要目的是为了提高数据处理的效率。主要包括两个方面:

- (1) 提高数据处理的速度;
- (2) 尽量节省在数据处理过程中所占用的计算机存储空间。

在具有相同特征的数据元素集合中, 各个数据元素之间存在着某种关系 (即联系), 这种关系反映了该集合中的数据元素所固有的一种结构。在数据处理领域中, 通常把这种数据元素之间这种固有的关系简单地用前后件关系 (或直接前驱与直接后继关系) 来描述。

2. 数据的逻辑结构

一个数据结构应包含以下两方面的信息:

- (1) 表示数据的信息;
- (2) 表示各数据元素之间的前后件关系。

在以上所述的数据结构中, 其中数据元素之间的前后件关系是指它们的逻辑关系, 而与它们在计算机中的存储位置无关。因此上面所述的数据结构实际上是数据的逻辑结构。

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。

由前面的叙述可以知道, 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 D 中各数据元素之间的前后件关系,通常记为 R 。即一个数据结构可以表示成

$$B = (D, R)$$

其中 B 表示数据结构。为了反映 D 中各数据元素之间的前后件关系,一般用二元组来表示。

3. 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(也称数据的物理结构)。

由于数据元素在计算机存储空间中的位置关系可能与逻辑关系不同,因此,为了表示存放在计算机存储空间中的各数据元素之间的逻辑关系(即前后件关系),在数据存储结构中,不仅要存放各数据元素的信息,还需要存放各数据元素之间的前后件关系的信息。

一种数据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、索引等存储结构。

4. 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;没有后件结点称为终端结点(也称为叶子结点)。数据结构中除了根结点与终端结点外的其他结点一般称为内部结点。

通常,一个数据结构中的元素结点可能是在动态地变化的。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)。插入与删除是对数据结构的两种基本运算。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。在对数据结构的处理过程中,不仅数据结构中的结点(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动态地变化。

5. 线性结构与非线性结构

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。如果一个非空的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。线性结构又称线性表。在一个线性结构中插入或删除任何一个结点后还应是线性结构。如果一个数据结构不是线性结构,则称之为非线性结构。线性结构与非线性结构都可以是空的数据结构。

易错题型讲解

一、选择题

例1 在数据结构中,从逻辑上可以把数据结构分成_____。

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

【答案】B)

【解题要点】逻辑结构反映数据元素之间的逻辑关系,线性结构表示数据元素之间为一对一的关系,非线性结构表示数据元素之间为一对多或者多对一的关系,所以答案为 B)。

【错解分析】数据的逻辑结构就是指数据元素之间的相互关系,A)选项、C)选项和 D)选项均不能表示数据元素之间的关系。只有 B)选项可以。线性结构和非线性结构是根据数据元素之间关系的不同特性加以分类的,即从逻辑上加以分类。

【解析】数据结构作为计算机的一门学科,主要研究和讨论以下三个方面的问题: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。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。对各种数据结构进行的运算。

2.【答案】线性结构

【解析】数据的逻辑结构主要是指数据元素之间的相互关系,它主要包括线性结构和非线性结构。

1.3 线性表及其顺序存储结构

重点、难点透析

1. 线性表的基本概念

线性表由一组数据元素构成,数据元素可以是简单项。在复杂的线性表中,一个数据元素还可以由若干个数据项组成。在这种复杂的线性表中,由若干数据项组成的数据元素称为记录(record),而由多个记录构成的线性表又称为文件(file)。

综上所述,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。

显然,线性表是一种线性结构。数据元素在线性表中的位置只取决于它们自己的序号,即数据元素之间的相对位置是线性的。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。

2. 线性表的顺序存储结构

在计算机中存放线性表,一种最简单的方法是顺序存储,也称为顺序分配。

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。

在线性表的顺序存储结构中,其前后件两个元素在存储空间中是紧邻的,且前件元素一定存储在后件元素的前面。

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 $ADR(a_1)$, 每一个数据元素占 k 个字节,则第 $i+1$ 个元素的存储位置 $ADR(a_{i+1})$ 和第 i 个数据元素的存储位置 $ADR(a_i)$ 之间满足下列关系:

$$ADR(a_{i+1}) = ADR(a_i) + k$$

线性表中第 i 个元素 a_i 在计算机存储空间中的存储地址为

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

即在顺序存储结构中,线性表中每一个数据元素在计算机存储空间中的存储地址由该元素在线性表中的位置序号惟一决定。线性表的顺序存储结构如图 1-1 所示。

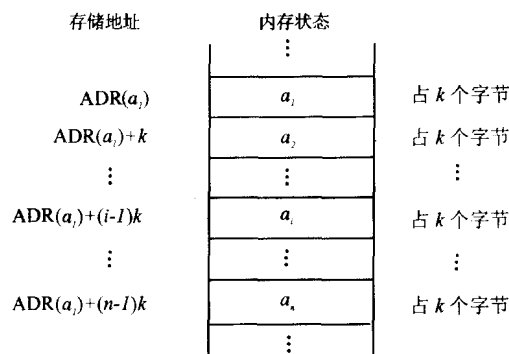


图 1-1 线性表顺序存储结构

3. 线性表顺序存储结构的运算

在线性表的顺序存储结构下,可以对线性表进行各种处理。主要的运算有以下几种:

- (1) 在线性表的指定位置处加入一个新的元素(即线性表的插入);
- (2) 在线性表中删除指定的元素(即线性表的删除);
- (3) 在线性表中查找某个(或某些)特定的元素(即线性表的查找);
- (4) 对线性表中的元素进行排序(即线性表的排序);
- (5) 按要求将一个线性表分解成多个线性表(即线性表的分解);
- (6) 按要求将多个线性表合并成一个线性表(即线性表的合并);
- (7) 复制一个线性表(即线性表的复制);
- (8) 逆转一个线性表(即线性表的逆转)等。

4. 顺序表的插入运算

一般来说,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 ,插入后得到长度为 $n+1$ 的线性表为

$$(a_1', a_2', \dots, a_j', a_{j+1}', \dots, a_n', a_{n+1}')$$

则插入前后的两线性表中的元素满足如下关系:

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

在一般情况下,在要第 i 个($1 \leq i \leq n+1$)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 个元素之间共 $n-i+1$ 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。

5. 顺序表的删除运算

一般来说,设长度为 n 的线性表为

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

现删除线性表的第 i 个元素 a_i ,删除后得到长度为 $n-1$ 的线性表为

$$(a_1', a_2', \dots, a_j', \dots, a_{n-1}')$$

则删除前后的两线性表中的元素满足如下关系:

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

在一般情况下,要删除第 i 个($1 \leq i \leq n+1$)个元素时,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,删除结束后,线性表的长度就减小了 1。

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是合适的,但对于元素经常变动的大线性表就不太合适。

易错题讲解

一、选择题

例题1 下面关于线性表的叙述中,错误的是_____。

- A) 线性表采用顺序存储,必须占用一片连续的存储单元
- B) 线性表采用顺序存储,便于进行插入和删除操作
- C) 线性表采用链接存储,不必占用一片连续的存储单元
- D) 线性表采用链接存储,便于进行插入和删除操作

【答案】B)

【解题要点】采用顺序存储的线性表,当对其进行插入和删除操作时需要移动大量的元素,所以 B) 选项说法是错误的,答案选 B)。

【错解分析】顺序存储结构必须占用一片连续的存储单元,A) 选项说法是正确的;当线性表采用链接存储时,其占用的存储空间是不连续的,C) 选项说法也是正确的;正是由于链接存储不是占用一片连续的存储空间,所以便于进行插入和删除操作。

例题2 对线性表,在下列_____情况下应当采用链表表示。

- A) 经常需要随机地存取元素
- B) 经常需要进行插入和删除操作
- C) 表中元素需要占据一片连续的存储空间
- D) 表中元素的个数不变

【答案】B)

【解题要点】由于链表不是顺序存储的,即当进行插入和删除操作时不需要进行移动大量的元素,所以当一些需要经常进行插入和删除操作的情况,应当采用链表的方式存储,答案应是 B)。

【错解分析】经常需要随机地存取元素的情况适合采用顺序存储的数组来表示,因为可以通过数组下标来随机访问,所以 A) 选项错误;表中元素需要占据一片连续的存储空间正是顺序存储的特点,所以 C) 选项也不正确;数组在一开始就分配固定个数的空间,不能动态分配,而链表的元素是可以动态改变的,所以 D) 选项也不正确。

二、填空题

例题1 一个数据元素可以由若干_____组成。

【答案】数据项

【解题要点】每个数据元素可以简单到是一个字母或是一个数据,也可能是比较复杂的由多个数据项组成的。由若干数据项组成的数据元素称为记录。

例题2 对于长度为 n 的顺序存储的线性表,当随机插入和删除一个元素时,需平均移动元素的个数为_____。

【答案】 $n/2$

【解题要点】删除一个元素,平均移动的元素的个数为 $(n-1+n-2+\cdots+0)/n=(n-1)/2$;

插入一个元素,平均移动元素个数为 $(n+n-1+n-2+\cdots+1)/n=(n+1)/2$,所以总体移动元素个数为 $n/2$ 。

同步训练

一、选择题

1. 下述哪一条是顺序存储方式的优点_____。

- A) 插入运算方便