

全国硕士研究生招生考试
计算机科学与技术学科联考

计算机学科专业基础
综合科目 (408)

综合教程及历年真题详解

最 新 版

开点工作室 编著



清华大学出版社

全国硕士研究生招生考试计算机科学与技术学科联考

计算机学科专业基础 综合科目(408)

综合教程及历年真题详解

最 新 版

开点工作室 编著

清华大学出版社
北京

内 容 简 介

本书作为全国硕士研究生招生考试中计算机科学与技术专业的计算机专业基础综合科目(408)的复习教材,主要包括考试大纲的内容解析以及历年真题详解。全书分上篇和下篇两个部分,上篇给出了数据结构、计算机组成原理、操作系统和计算机网络4门课程的考研大纲中涉及的主要内容概述;下篇主要对数据结构、计算机组成原理、操作系统和计算机网络等4门课程历年真题中具有代表性的题目进行详细解析。

本书重点突出,通俗易懂,实例丰富。本书可以作为高等学校在校学生的计算机专业考研教材,也可以作为计算机专业本科生、研究生或计算机技术人员的参考书。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

全国硕士研究生招生考试计算机科学与技术学科联考 计算机学科专业基础综合科目(408)综合教程及历年真题详解:最新版/开点工作室编著. —北京:清华大学出版社,2015

ISBN 978-7-302-40720-1

I. ①全… II. ①开… III. ①电子计算机—研究生—入学考试—自学参考资料 IV. ①TP3

中国版本图书馆CIP数据核字(2015)第152422号

责任编辑:张瑞庆 李 晔

封面设计:傅瑞学

责任校对:梁 毅

责任印制:刘海龙

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦A座 邮 编:100084

社总机:010-62770175 邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质量反馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

课件下载: <http://www.tup.com.cn>, 010-62795954

印 刷 者:清华大学印刷厂

装 订 者:三河市新茂装订有限公司

经 销:全国新华书店

开 本:185mm×260mm 印 张:20.5 字 数:509千字

版 次:2015年7月第1版 印 次:2015年7月第1次印刷

印 数:1~2000

定 价:39.50元

产品编号:065520-01

前言

FOR E W O R D

全国硕士研究生招生考试(简称研考)计算机科学与技术专业计算机专业基础综合科目(科目代码为408,简称408)从2009年开始实施全国统一命题考试,到2015年已经实施统考七个年头了。计算机研考专业课统考在刚开始实施之时,曾在广大计算机专业招生单位和学生中引起了广泛的关注和强烈的反响,赞成和反对的声音同时存在。很多计算机专业的学生抱怨计算机统考内容广,难度大,分数线还要和其他非统考的专业课统一划线;很多学校和老师反映由于统考的难度大,挡住了相当一批计算机专业的学生上线,只能调剂其他非计算机专业上线的考生。另一方面,赞成的观点认为计算机统考能够最大程度地保证招生录取的公平性,非名校的学生有了更多进入名校计算机专业读研的机会,而且统考试题的质量普遍较高,基础好、能力强的学生能真正通过成绩展示出水平。不同观点的讨论和碰撞引起了主管部门的注意,于是在2013年,教育部对研考计算机专业课政策进行了调整,在保留计算机专业课统考科目(408)的基础上,允许各招生单位自主选择参加统考,还是自主命题。这样的政策对于不同的招生单位,可以说是各取所需。到目前为止,仍然选择统考(408)作为计算机专业研究生招生初试科目的招生单位包括清华大学、北京大学、复旦大学、上海交通大学、中国科学技术大学、浙江大学、华中科技大学、中国科学院系统、中国航天集团等100多家。

计算机专业基础综合科目(408)的考试内容由四部分组成:数据结构、计算机组成原理、操作系统和计算机网络。在庞大复杂的计算机科学与技术专业体系中,这四门课程可以说是最基础、最核心的部分。数据结构是研究数据在计算机中的表达、存储和处理的方法和过程的系统理论,也是设计和实现编译程序、操作系统、数据库系统及其他系统软件和大型应用程序的重要基础,培养的是学生抽象能力、算法能力和编程能力。计算机组成原理研究计算机的硬件组成和基本工作原理,其课程目标是要让学生了解数据的表示与运算、指令集体系结构、指令的执行过程及中央处理器组织、存储器层次结构以及输入输出组织等,并使学生将计算机硬件组成、指令系统和上层的操作系统以及高级语言程序建立关联,最终构建完整的计算机系统概念。操作系统是现代计算机系统的核心软件,是其他一切软件运行的基础,是应用软件开发的基础平台。操作系统在用户和硬件之间架起了一个桥梁,通过这个桥梁,用户可以方便地使用计算机,硬件可以更高效地发挥其作用。学习和了解操作系统的工作机理和工作方式,进而设计和实现操作系统,是计算机专业学生必备的知识和技能。计算机网络是目前以及未来计算机最主要的应用领域,几乎所有的新技术、新应用都离不开计算机网络技术的支持。

这四门课程是各高校计算机专业的核心课程,它们既自成体系,互相之间又有很强的联系。计算机专业的学生不管是读研、读博,继续从事计算机科学与技术的研究工作,还是进入就业市场,获得各大IT企业的工作机会,都需要认真学好这四门课程,真正掌握其中的基本概念、基本理论和基本技能。计算机专业基础综合科目(408)把这四门课程作为考试内容,其目的是全面考查学生计算机专业的基础知识和综合素养,评价和评测学生是否具有进一步从事计算机相关领域的开发、设计、工程以及理论研究的能力。

不可否认,408科目的难度是很大的,以至于不少计算机专业考研学生形容408为“变态难”。分析其原因,我们认为主要有三点:第一,内容多、范围广。因为408科目规定了考试范围是四门课程,考研学生要复习的内容很多,而这些课程本身的知识容量就很大,是计算机专业中学生难学、老师难教的课程,合并在一起作为考试内容就更增加了难度。第二,试题灵活、多变,重在考查应用能力。从几年来的统考试题来看,408科目的试题形式新颖灵活,基本上都不是那种靠记忆背诵就能一眼看出答案的试题,需要学生对于相关知识理解透彻,主要考查学生的计算机知识应用能力,体现了计算思维和系统思维的思想。这种突出能力考查的命题方式对一部分学生来说增加了考试难度。第三,客观地讲,目前的高校计算机专业教育教学水平参差不齐,有些学校缺少高水平的师资力量,另外部分学校使用的教材陈旧,缺乏配套的实验课程,学生的基本知识和基本技能掌握不扎实。

开点工作室是由一群重点高校计算机专业中青年骨干教师组成的教育教学研究兴趣小组,这些老师活跃在计算机教学工作的第一线,虽然来自不同的学校,但有着共同的特点:热爱计算机教学,喜欢钻研专业知识,具有丰富的教学、科研以及著作编写经验,并且希望能真正培养出知识和能力扎实过硬,能够在科研和就业领域都有所作为的计算机专业学生。我们自己招的研究生中就是考过408的,有自己教的本科生去考408的,还有的老师参加过408的阅卷工作。通过近几年来的相关工作和研究,我们越来越感觉到408科目是对计算机专业本科学生进行能力检测和选拔的很好标尺,同时也能较系统地引导学生准确掌握计算机专业最基础最核心的知识。如果学生真正能把408科目试题中所蕴涵的知识和能力掌握好,对于学生在专业上的进一步发展是非常有益的。而现实情况是,相当数量的学生在复习和准备408科目的时候,花费了大量的精力,看了大量的教材,做了海量的练习,却没有抓住要领,对于必须掌握的知识的认识仍然是模糊的,更没有建立起计算机系统整体概念,缺乏将不同知识点进行关联来解决问题的思路。因此,我们花费了大量的时间和精力,对408科目的历年真题进行了深入的分析研究,将其中典型试题的知识点、能力点、解题思维过程进行了细致的解析,力图把这些“变态难”的试题像剥洋葱一样层层展开,让学生能够清晰地了解试题所传递的信息,掌握同类试题的解题方法,在学习中提高能力;同时,根据408科目考试大纲,将各门课程的考核知识点展开并进行系统论述,有针对性地将这些知识讲清楚,而省略了408科目不要求的内容,相当于是四门课程的一个精简版集合教材。这两部分内容相结合,构成了这样一本融四门专业课知识讲解和历年考试真题分析于一体的指导用书。本书具有以下几个特色和亮点:

第一,考试大纲仅仅是列出了考试的知识点,并没有阐述对这些知识点应该掌握到什么程度,而本书能够帮助考生深刻理解大纲,把握大纲,做到有的放矢,重点突出。考生经常发现,各种教材都看过了,但还是不会做题,不知道从何下手。其中的主要原因是并没有真正具备综合运用教材中知识点的能力,这正是本书要解决的问题。本书提升的是考生分析问

题和解决问题的能力,而不仅仅是会做题。

第二,本书与目前通用的课程教材有所区别。教材侧重于讲授知识点,而疏于介绍解题思路,更不会花笔墨在多个知识的综合应用上。教材的内容平面化、全面化、系统化,而本书对其进行提炼精选,角度独特。

第三,本书与市面上其他参考资料也有所区别。本书以大纲为准则,以真题为素材,以考生为对象,以提升为目标,针对试题的考点、容易出错的地方、非答案选项的错误原因均给出详细的讲解,可起到举一反三的作用,达到事半功倍的效果,也让考生了解题目的形式和难度。另外,典型题所涉及的知识点都是大纲中要求重点掌握的,也是经常出现在考题中的知识点,可以帮助考生避免题海战术。

我们希望通过这本凝结了集体智慧和心血的教材,不仅仅能够帮助参加 408 科目的考生更轻松、更有效地复习;而且,更多不参加 408 科目考试的学生也可以使用该教材进行计算机专业知识的自学、复习、练习以及就业笔试、面试的准备,以提升自己的专业水平和技能。“拨开学习中的荆棘迷雾,指点探索中的灵感思路”是本书的主旨,也是我们“开点工作室”名字的由来。未来我们会继续致力于策划编写更多高质量的计算机专业领域各类教材和工具书,为广大热爱计算机、渴望学习计算机的学生提供更多的帮助。

本书分上、下两篇。上篇是考研计算机专业基础综合考试大纲解析(注:本书针对的是 2015 年考试大纲,待 2016 年考试大纲公布后进行相应调整),包括数据结构、计算机组成原理、操作系统和计算机网络四个部分;下篇是 2009 年至 2015 年历年典型真题详解,包括了数据结构、计算机组成原理、操作系统和计算机网络四部分的单项选择题 107 道和综合应用题 22 道。在本书的编写过程中,我们深深感受到计算机专业真的是博大精深,越深入研究就会越感觉到自己知识不够。由于水平有限,加之时间仓促,书中肯定会有这样或那样的错误,敬请读者提出宝贵意见,以便帮助我们修改完善。

开点工作室

2015 年 5 月于北京

目 录

上 篇

计算机专业基础综合大纲解析

第 1 章 数据结构	3
1.1 线性表	4
1.1.1 线性表的定义和基本操作	4
1.1.2 线性表的实现	5
1.2 栈、队列和数组	10
1.2.1 栈和队列的基本概念	10
1.2.2 栈和队列的顺序存储结构	11
1.2.3 栈和队列的链式存储结构	13
1.2.4 栈和队列的应用	13
1.2.5 特殊矩阵的压缩存储	15
1.3 树与二叉树	17
1.3.1 树的基本概念	17
1.3.2 二叉树	18
1.3.3 树、森林	22
1.3.4 树与二叉树的应用	25
1.4 图	30
1.4.1 图的基本概念	30
1.4.2 图的存储及基本操作	31
1.4.3 图的遍历	33
1.4.4 图的基本应用	34
1.5 查找	39
1.5.1 查找的基本概念	39
1.5.2 顺序查找法	40
1.5.3 折半查找法	40
1.5.4 分块查找法	41
1.5.5 B 树及其基本操作、B ⁺ 树的基本概念	42
1.5.6 散列(Hash)表	45
1.5.7 字符串模式匹配	48
1.5.8 查找算法的分析及应用	49

1.6	排序	49
1.6.1	排序的基本概念	49
1.6.2	插入排序	49
1.6.3	起泡排序	50
1.6.4	简单选择排序	51
1.6.5	希尔排序	51
1.6.6	快速排序	52
1.6.7	堆排序	52
1.6.8	二路归并排序	53
1.6.9	基数排序	53
1.6.10	外部排序	53
1.6.11	各种排序算法的比较	54

第2章 计算机组成原理 55

2.1	计算机系统概述	55
2.1.1	计算机发展历程	55
2.1.2	计算机系统层次结构	55
2.1.3	计算机性能指标	57
2.2	数据的表示和运算	58
2.2.1	数制和编码	58
2.2.2	定点数的表示和运算	59
2.2.3	浮点数的表示和运算	62
2.2.4	算术逻辑单元 ALU	63
2.3	存储器层次结构	64
2.3.1	存储器的分类	64
2.3.2	存储器的层次化结构	64
2.3.3	半导体随机存取存储器	65
2.3.4	主存储器和 CPU 的连接	66
2.3.5	双口 RAM 和多模块存储器	66
2.3.6	高速缓冲存储器	67
2.3.7	虚拟存储器	70
2.4	指令系统	71
2.4.1	指令格式	71
2.4.2	指令的寻址方式	72
2.4.3	CISC 和 RISC 的基本概念	73
2.5	中央处理器(CPU)	74

2.5.1	CPU 的功能和基本结构	74
2.5.2	指令执行过程	75
2.5.3	数据通路的功能和基本结构	76
2.5.4	控制器的功能和工作原理	78
2.5.5	指令流水线	79
2.6	总线	80
2.6.1	总线概述	80
2.6.2	总线仲裁	82
2.6.3	总线操作和定时	82
2.7	输入输出(I/O)系统	83
2.7.1	I/O 系统基本概念	84
2.7.2	外部设备	84
2.7.3	I/O 接口(I/O 控制器)	86
2.7.4	I/O 方式	87
第 3 章 操作系统		89
3.1	操作系统概述	89
3.1.1	操作系统的概念、特征和操作系统的服务	89
3.1.2	操作系统的发展与分类	91
3.1.3	操作系统的运行环境	93
3.1.4	操作系统体系结构	96
3.2	进程管理	97
3.2.1	进程与线程	97
3.2.2	CPU 调度	102
3.2.3	同步与互斥	105
3.2.4	死锁	111
3.3	内存管理	112
3.3.1	内存管理基础	112
3.3.2	虚拟内存管理	119
3.4	文件管理	124
3.4.1	文件系统基础	124
3.4.2	文件系统实现	126
3.4.3	磁盘组织与管理	128
3.5	输入输出(I/O)管理	130
3.5.1	I/O 管理概述	130
3.5.2	I/O 核心子系统	132

第 4 章	计算机网络	136
4.1	计算机网络体系结构	136
4.1.1	计算机网络综述	136
4.1.2	计算机网络体系结构与参考模型	136
4.2	物理层	138
4.2.1	通信基础	138
4.2.2	传输介质	140
4.2.3	物理层设备	141
4.3	数据链路层	142
4.3.1	数据链路层的功能	142
4.3.2	组帧	142
4.3.3	差错控制	142
4.3.4	流量控制与可靠传输机制	143
4.3.5	介质访问控制	144
4.3.6	局域网	148
4.3.7	广域网	150
4.3.8	数据链路层设备	151
4.4	网络层	152
4.4.1	网络层功能	152
4.4.2	路由算法	153
4.4.3	IPv4	154
4.4.4	IPv6	159
4.4.5	路由协议	160
4.4.6	IP 组播	162
4.4.7	移动 IP	162
4.4.8	网络层设备	163
4.5	传输层	164
4.5.1	传输层提供的服务	164
4.5.2	UDP 协议	165
4.5.3	TCP 协议	165
4.6	应用层	169
4.6.1	网络应用模型	169
4.6.2	DNS 系统	169
4.6.3	FTP	172
4.6.4	电子邮件	172
4.6.5	WWW	174

下 篇

历年典型真题详解

第5章 数据结构	179
第6章 计算机组成原理	217
第7章 操作系统	258
第8章 计算机网络	281
参考文献	314

上

篇

计算机专业基础
综合大纲解析

PART

【术语解释】

数据是客观事物的符号表示,是对现实世界的事物采用计算机能够识别、存储和处理的进行描述的符号的集合。

数据元素是数据的基本单位。一个数据元素由若干个数据项组成。数据项又分为简单数据项及复合数据项两种。简单数据项不能再分割,复合数据项由若干个数据项组成。

类型是一组值的集合。

抽象数据类型(ADT)用一种类型和定义在该类型上的一组操作来定义。

数据结构是抽象数据类型的实现,可以使用二元组表示:

$$\text{数据结构} = (D, R)$$

其中, D 是数据对象,即数据元素的有限集; R 是该数据对象中所有数据元素之间关系的有限集。

数据元素及数据元素之间的逻辑关系称为数据的逻辑结构;数据元素及数据元素之间的关系在计算机中的存储表示,称为数据的存储结构或物理结构。

数据的逻辑结构分为 4 类:集合、线性结构、树结构和图结构。

线性结构中,每个元素最多只有一个前驱和一个后继。线性表是典型的线性结构。

树结构和图结构都属于非线性结构,其中每个元素都可能多个前驱和多个后继。

如果结构中每个元素最多只有一个前驱,而可能多个后继,且有一个元素没有前驱,则为树结构。

如果结构中每个元素都可能多个前驱及多个后继,则为图结构。

数据的基本存储结构有 4 种:顺序存储、链式存储、索引存储及散列存储。

顺序存储方式通常采用数组实现,也称为静态存储。

链式存储方式通常采用指针实现,也称为动态存储。

问题是一个需要完成的任务,即一组输入就有一组相应的输出。

算法是指解决问题的一种方法或者一个过程。一个问题可以用多种算法来解决,一种给定的算法解决一个特定的问题。算法应具备有输入、有输出、确定性、有穷性及可执行性。

一个计算机程序是使用某种程序设计语言对一种算法的具体实现。

算法的复杂度度量分为时间复杂度和空间复杂度度量。当问题的规模从 1 增加到 n 时,若解决这个问题的算法在执行时所耗费的时间由 1 增加到 $T(n)$,则称算法的时间复杂度为 $T(n)$;同时,若解决这个问题的算法在执行时所需附加的存储空间由 1

增加到 $S(n)$, 则称此算法的空间复杂度为 $S(n)$ 。

渐近算法分析是衡量算法效率的一种手段, 它估算出当问题规模变大时, 一种算法及实现它的程序的效率和开销。

如果存在常数 $c > 0$ 与 n_0 , 当 $n > n_0$ 时有 $T(n) \leq cf(n)$, 则称(复杂度)函数 $T(n)$ 是 $O(f(n))$ 的, 即 $T(n) = O(f(n))$ 。

如果存在常数 $c > 0$ 与 n_0 , 当 $n > n_0$ 时有 $T(n) \geq cf(n)$, 则称(复杂度)函数 $T(n)$ 是 $\Omega(f(n))$ 的, 即 $T(n) = \Omega(f(n))$ 。

大 O 表示法和 Ω 表示法使我们能够描述某一算法的上限(如果能找到某一类输入下开销最大的函数)和下限(如果能找到某一类输入下开销最小的函数)。如果一种算法既是 $O(f(n))$, 又是 $\Omega(f(n))$, 则称它是 $\Theta(f(n))$ 的。

对于某些算法, 即使问题规模相同, 如果输入的数据不同, 其时间代价也不同, 又分为最佳情况、平均情况及最差情况。

1.1 线性表

线性表是最简单的一种线性结构, 有很广泛的应用, 同时也是其他非线性结构的基础。线性表中各元素的类型是一致的。

由任意元素组成的表称为广义表, 广义表是对线性表的扩展。线性表是大纲中数据结构部分的重要内容, 对广义表不做要求。

要深入理解线性表的概念, 了解线性表基本操作中各参数的含义, 正确给出操作的结果。能正确实现基于线性表应用的程序。

1.1.1 线性表的定义和基本操作

1. 线性表的定义

线性表是最常用且最基本的一种数据结构, 是由称为元素的数据项组成的一种有限且有序的序列。表中元素可以是任意类型。由 $n(n \geq 0)$ 个元素组成的线性表记为

$$(a_0, a_1, \dots, a_{n-1})$$

其中, a_0 称为表头, a_{n-1} 称为表尾。元素的个数 n 称为表长。 $n=0$ 时称为空表, 记为 $()$ 。

线性表各元素的位置关系是确定的。线性表中常使用整数表示各元素的位置, 表头 a_0 的位置为 0, a_1 的位置为 1, 一般地, a_i 的位置为 i 。

元素 $a_{i-1} (1 \leq i \leq n-1)$ 称为 a_i 的直接前驱(简称为前驱), a_i 称为 a_{i-1} 的直接后继(简称为后继)。除 a_0 外, 每个元素有且仅有一个前驱; 除 a_{n-1} 外, 每个元素有且仅有一个后继。

线性表中各元素可以是可比类型的, 也可以是不可比类型的。如果是可比类型的, 其“大小”可以任意, 既可以有序也可以无序。“大小”有序的线性表称为有序表。

2. 线性表的基本操作

线性表的基本操作包括:

- (1) 创建一个空表 Create(): 返回一个空的线性表 L 。
- (2) 在表的指定位置插入一个元素 Insert(L, i, e): 在表 L 的位置 i 插入元素 e 。
- (3) 删除表中指定位置的元素 Delete(L, i): 删除表 L 中位置 i 的元素。
- (4) 访问表中指定位置的元素 GetElem(L, i): 返回表 L 中位置 i 的元素。

例如, 已知表 $L=(20, 14, 12, 15, -3, 7)$, 操作 Insert($L, 3, 11$) 要在表 L 的位置 3 插入元素 11, 操作后得到的结果为: $(20, 14, 12, 11, 15, -3, 7)$ 。插入、删除及访问操作中都有一个表示位置的参数, 这个值必须在合理的范围内。对于表长为 n 的线性表, 插入的合理位置为 $0 \leq i \leq n$, 删除及访问的合理位置为 $0 \leq i \leq n-1$ 。

除此之外, 还有以下一些操作:

- (1) 判定表空 IsEmph(L): 如果表为空返回真, 否则返回假。
- (2) 判定表满 IsFull(L): 如果表已满返回真, 否则返回假。
- (3) 求表长 Length(L): 返回表中元素的个数。
- (4) 求表中指定元素的直接前驱 Prev(L, e): 返回表 L 中元素 e 的直接前驱。
- (5) 求表中指定元素的直接后继 Next(L, e): 返回表 L 中元素 e 的直接后继。

1.1.2 线性表的实现

线性表可以采用不同的存储结构保存, 主要存储结构有两种: 数组及链表。

采用数组保存的线性表称为顺序表, 这种存储结构称为顺序存储结构或静态存储结构。采用指针方式保存的线性表称为链表, 相应的存储结构称为链式存储结构或动态存储结构。结合两种存储结构的特点, 将线性表以动态存储的策略保存在数组中, 得到静态链表。

1. 顺序存储

顺序存储方式下, 线性表采用一维数组保存, 采用数组保存的线性表称为顺序表。线性表中各元素依次保存在数组各存储单元中, 表中逻辑上相邻的两个元素, 其实际的存储位置也相邻。

若线性表每个元素占用 l 个存储单元, 线性表中元素 a_i 的存储地址表示为 $LOC(a_i)$, 则有下列关系:

$$LOC(a_i) = LOC(a_0) + i * l$$

只要确定了顺序表存储的起始位置(数组首址), 根据上式可以立即得到线性表中任一元素的存储位置。由于数组能够按照下标直接访问元素, 所以顺序表可实现随机存取。

根据已知条件计算顺序表中某个元素的实际存储地址, 是考试中常见的题型。

设顺序表元素类型表示为 ElemType, 使用 C 语言定义的顺序表如下:

```
typedef struct{
    ElemType * listArray;    //指向 ElemType 类型数组的指针
    int length;             //数组长度
    int listSize;           //数组最大容量
}SeqList;
```

数组一经分配, 其大小不可改变, 数组空间大小由 listSize 表示, 顺序表中当前元素个数

保存在 length 中。

实现顺序表插入 $\text{Insert}(L, i, e)$ 的代码描述为:

```
Status Insert(SeqList &L, int i, ElemType e)
{
    int k=0;
    if(i<0||i>L.length) return ERROR;
    for(k=L.length; k>i; k--)
        L.listArray[k]=L.listArray[k-1];
    L.listArray[k]=e;
    L.length++;
    return OK;
}
```

实现顺序表删除 $\text{Delete}(L, i, e)$ 的代码描述为:

```
Status Delete(SeqList &L, int i, ElemType *e)
{
    int k=0;
    if(i<0||i>=L.length) return ERROR;
    *e=L.listArray[i];
    for(k=i; k<L.length-1; k++)
        L.listArray[k]=L.listArray[k+1];
    L.length--;
    return OK;
}
```

顺序表保存在数组中,数组的特点要求数据必须保存在一段连续的地址空间内。在顺序表中插入、删除元素时不可避免地要进行或多或少的元素移动。在顺序表中插入一个新元素、删除指定位置元素时,最优时间复杂度均为 $O(1)$,最差时间复杂度和平均时间复杂度均为 $O(n)$ 。所列的其他基本操作的时间复杂度均为 $O(1)$ 。由此可知,插入、删除操作的开销较大。

2. 链式存储

线性表中的各元素也可以不必保存在一段连续的地址空间中,而是放在任意的存储单元内。为了能方便地找到这些存储单元,采用指针结构将这些存储单元串在一起,形成链表。这种存储方式称为链式存储,采用链式存储结构的线性表称为链表。这样的表示法也称为动态表示法。

链表由一系列的结点构成,每个结点包含数据域和指针域,其中数据域中保存线性表的一个数据元素的信息,指针域中保存指向该数据元素后继结点的指针信息,表尾结点的指针域中保存 NULL,表示表的结束。指针也称为链,链表的名称由此得来。

每个结点中除数据域外只包含一个指针域,这样的链表称为单链表。单链表中各元素的位置仍然用整数表示,通常使用一个指针(称为表头指针)指向表头元素结点,使用另外一个指针(称为当前工作指针)来指示操作位置,如图 1-1 所示为一个单链表,指针 head 指向表头元素,指针 p 指向操作位置。表尾结点的指针域保存 NULL,使用 \wedge 表示。

为了操作上的方便,常常在单链表的表头添加一个空结点,称为表头结点,得到带头表的单链表,如图 1-2 所示。