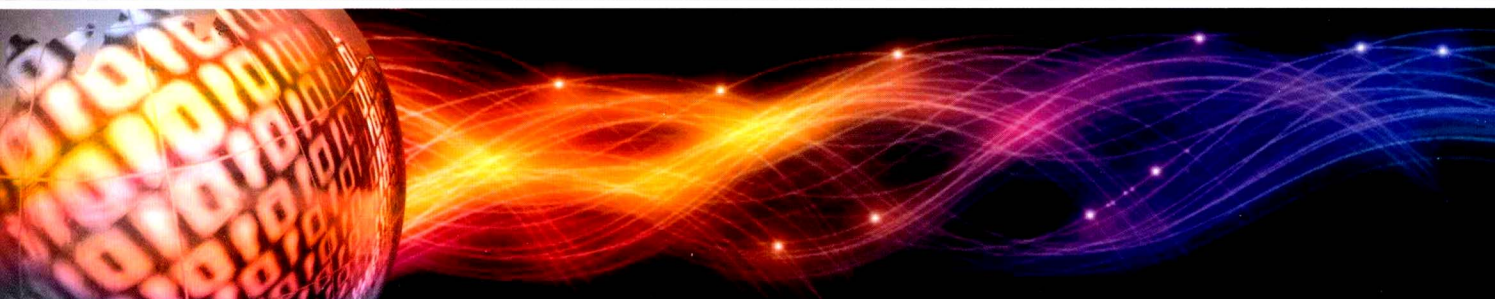


全国研究生 入学计算机统一考试考点、 题解与模拟试卷 (2009年新大纲)(第2版)

梁 旭 张振琳 黄 明 编著



→ 数据结构

←----- 计算机组成原理

←-----→ 操作系统

-----→ 计算机网络



电子工业出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

<http://www.phei.com.cn>

全国研究生入学计算机统一考试考点、 题解与模拟试卷 (2009 年新大纲) (第 2 版)

数据结构 计算机组成原理
操作系统 计算机网络

梁 旭 张振琳 黄 明 编著

电子工业出版社

Publishing House of Electronics Industry

北京 • BEIJING

内 容 简 介

本书是根据“全国研究生入学考试(计算机专业)考试大纲(2009年)”,精选各个著名高校历年考研真题和习题编写的。本书主要包括5大部分:第1部分是数据结构;第2部分是计算机组成原理;第3部分是操作系统;第4部分是计算机网络;第5部分是模拟试卷及参考答案:另外,还包括应试策略和附录。

本书紧扣考试大纲,广泛收集了近几年全国20余所重点高校考研试卷,对典型的真题进行了深入、细致的分析和解答。叙述通俗易懂,每门课程都附有习题及参考答案。书中还给出了模拟试卷,可用于考前训练。

本书适合报考计算机专业研究生的考生有针对性地进行专业课的复习,也适合希望深入学习计算机专业知识的高校学生作为学习相关课程的参考书,同时也适用于讲授该课程的教师及自修该课程的其他人员。

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。

版权所有,侵权必究。

图书在版编目(CIP)数据

全国研究生入学计算机统一考试考点、题解与模拟试卷:2009年新大纲/梁旭,张振琳,黄明编著.—2版.

北京:电子工业出版社,2009.8

ISBN 978-7-121-09246-6

I. 全… II. ①梁… ②张… ③黄… III. 电子计算机—研究生—入学考试—自学参考资料 IV. TP3

中国版本图书馆CIP数据核字(2009)第116597号

策划编辑:赵平

责任编辑:周宏敏

印刷:北京市海淀区四季青印刷厂

装订:涿州市桃园装订有限公司

出版发行:电子工业出版社

北京市海淀区万寿路173信箱 邮编 100036

开本:850×1168 1/16 印张:30.25 字数:1024千字

印次:2009年8月第1次印刷

印数:4000册 定价:48.00元

凡所购买电子工业出版社图书有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系,联系及邮购电话:(010)88254888。

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。

服务热线:(010)88258888。

修 订 说 明

《全国研究生计算机统一考试习题详解》是针对全国计算机科学与技术学科硕士研究生招生联考而出版的辅导书，在 2009 年 1 月第一次全国联考结束后，我们对 2009 考试真题的性质、内容、范围、形式、难度、能力要求、题型、分值等方面严格分析，并据此修订了本书内容，对每部分增加了考点分析，结合例题对大纲的涵盖方面，添加了一些高校最近两年的硕士生考试真题，优化了解析的内容。此外，在本书出版后，受到了广大读者的欢迎，许多读者给我们写信，提出了宝贵的意见和建议，我们也将这部分意见加入了这次修订，在此，向他们表示最真挚的感谢！

我们希望通过这次修订，使本书更符合考研要求，以帮助考生把握复习的重点和规律，为考生提供更好的帮助。此外我们欢迎广大读者继续和我们交流，作者信箱：liangxu00@263.net。

编著者

2009 年 4 月

前 言

一年一度的研究生入学考试即将开始，这也意味着新一轮复习的开始。如今研究生入学考试已经成为继高考之后全国第二大考试，而随着 IT 产业的高速发展，计算机专业几乎成为全国最热门的专业之一。

2009 年，全国硕士研究生入学统一考试新大纲对计算机科学与技术专业的研究生入学考试专业课进行了调整，将以往专业课由各个学校独立命题，调整为全国统一命题形式。计算机科学与技术学科的初试科目调整后为 4 门，即政治理论、外语、数学和计算机学科专业基础综合，卷面满分值分别为 100 分、100 分、150 分和 150 分。计算机学科专业基础综合的考试内容包括：数据结构、计算机组成原理、操作系统和计算机网络，重点考查考生掌握相关基础知识、基本理论和分析与解决问题的能力。

本书针对全国统一命题要求，紧扣新大纲，精选各个著名高校历年考研真题，进行详尽的解析，并给出答案，使读者可以真正掌握题目的解题思路和方法。其内容覆盖了计算机学科专业基础综合所涉及的 4 大部分，既强化了重点，又进一步细化了知识点，以帮助读者了解新大纲的要点，提高复习效率。

本书具有以下两个特点：

- (1) 紧密结合新大纲；
- (2) 全面而有重点地选择真题。

本书紧扣国家教育部制定的研究生入学考试大纲，合理安排书中各个章节内容，条理清晰，详略分明，深入浅出，解疑答惑，并广泛收集近年全国 20 余所重点高校考研试卷，对典型的真题进行了深入剖析，以求达到举一反三的目的，使读者能掌握这四门专业课程的主要原理和方法。

对于专业课的复习，必须注重“全面”与“重点”相结合，因为一门课程的内容非常多，考试只是考到其中的一小部分，所谓重点就是出题概率较高，并且所占分值也较高的内容。编者通过仔细研究各个学校的历年专业课试题，发现其中一些内容重复性很强，对这些相关典型题目进行了细致、深入的分析和解答，并进行了适当的扩展，使读者能够明确整个题目的分析过程和需要注意的地方，而不仅是知道答案而已，从而达到融会贯通的境界。对于每章的例题，读者应注重其解题的思路，而不应拘泥于特定的题目，为了使读者有更多的练习机会，每门课程都配有大量的习题及参考答案。最后还附有两套模拟试卷，以配合考生的学习。

本书由梁旭、张振琳、黄明共同编著，同时要特别感谢张韶华、孟旭升、唐玲、牛文颖、刘剑等，他们做了大量的工作。由于编著者水平有限，编写时间仓促，书中错误和不妥之处在所难免，请读者和专家批评指正。

读者在使用本书的过程中如有问题，可与下列 E-mail 联系：liangxu00@263.net。

最后，预祝广大考生旗开得胜，万事如意！

编著者

2009 年 4 月

目 录

应试策略	1
第1部分 数据结构	2
考查目标和知识点解析	2
第1章 概论	9
一、单项选择题	9
二、综合应用题	11
第2章 线性表	13
一、单项选择题	13
二、综合应用题	20
第3章 栈、队列和数组	28
一、单项选择题	28
二、综合应用题	36
第4章 树与二叉树	47
一、单项选择题	47
二、综合应用题	58
第5章 图	70
一、单项选择题	70
二、综合应用题	76
第6章 查找	84
一、单项选择题	84
二、综合应用题	90
第7章 排序	96
一、单项选择题	96
二、综合应用题	102
习题	112
一、单项选择题	112
二、综合应用题	118
参考答案	120
一、单项选择题	120
二、综合应用题	120
第2部分 计算机组成原理	131
考查目标和知识点解析	131
第8章 计算机系统概述	147
一、单项选择题	147
二、综合应用题	149
第9章 数据的表示和运算	151
一、单项选择题	151

二、综合应用题	158
第 10 章 存储器层次结构	165
一、单项选择题	165
二、综合应用题	171
第 11 章 指令系统	181
一、单项选择题	181
二、综合应用题	185
第 12 章 中央处理器 (CPU)	191
一、单项选择题	191
二、综合应用题	194
第 13 章 总线	202
一、单项选择题	202
二、综合应用题	204
第 14 章 输入/输出 (I/O) 系统	206
一、单项选择题	206
二、综合应用题	213
习题	217
一、单项选择题	217
二、综合应用题	222
参考答案	225
一、单项选择题	225
二、综合应用题	225
第 3 部分 操作系统	231
考查目标和知识点解析	231
第 15 章 操作系统概述	244
一、单项选择题	244
二、综合应用题	249
第 16 章 进程管理	252
一、单项选择题	252
二、综合应用题	265
第 17 章 内存管理	294
一、单项选择题	294
二、综合应用题	302
第 18 章 文件管理	311
一、单项选择题	311
二、综合应用题	319
第 19 章 输入/输出 (I/O) 管理	328
一、单项选择题	328
二、综合应用题	331
习题	334
一、单项选择题	334
二、综合应用题	341
参考答案	350

一、单项选择题	350
二、综合应用题	350
第 4 部分 计算机网络	365
考查目标和知识点解析	365
第 20 章 计算机网络体系结构	381
一、单项选择题	381
二、综合应用题	386
第 21 章 物理层	388
一、单项选择题	388
二、综合应用题	391
第 22 章 数据链路层	396
一、单项选择题	396
二、综合应用题	407
第 23 章 网络层	415
一、单项选择题	415
二、综合应用题	419
第 24 章 传输层	426
一、单项选择题	426
二、综合应用题	427
第 25 章 应用层	429
一、单项选择题	429
二、综合应用题	433
习题	437
一、单项选择题	437
二、综合应用题	440
参考答案	442
一、单项选择题	442
二、综合应用题	442
第 5 部分 模拟试卷及参考答案	445
全真模拟试卷(一)	445
一、单项选择题(本大题共 40 小题, 每小题 2 分, 共 80 分)	445
二、综合应用题(本大题共 11 小题, 共 70 分)	447
全真模拟试卷(一) 分析解答	449
一、单项选择题	449
二、综合应用题	449
全真模拟试卷(二)	454
一、单项选择题(本大题共 40 小题, 每小题 2 分, 共 80 分)	454
二、综合应用题(本大题共 11 小题, 共 70 分)	456
全真模拟试卷(二) 分析解答	459
一、单项选择题	459
二、综合应用题	459
附录 A 2009 年计算机专业硕士入学考试大纲	462
参考文献	472

应试策略

数据结构、计算机组成原理、操作系统和计算机网络这四门课都是计算机专业的基础课程，每门课程所涉及的内容十分广泛，具有知识点多、难度和跨度大等特点。而硕士考试中的计算机学科专业基础综合要同时涵盖这四门课程，其难度可想而知。

作为即将参加考研的考生，如何在有限的时间内取得更好的成绩，是每个考生都非常关心的问题。为此，我们从考生的实际出发，结合多年的教学经验，总结了一些迎考备战中应注意的问题，希望能在考生的复习过程中起到一定的指导作用。

（一）深入理解考试大纲

全国计算机硕士统一考试计算机学科专业基础综合考试大纲对考试的基本要求、考试内容都做了具体规定（详见附录部分）。

在计算机考研专业基础课统考科目中，一共考查数据结构、计算机组成原理、操作系统和计算机网络 4 门课程，满分为 150 分。其中数据结构 45 分、计算机组成原理 45 分、操作系统 35 分、计算机网络 25 分。统考只有两种题型：单项选择题和综合应用题，选择题占了 80 分，共 40 道题，从这点上可以很明显地看出来，考查的知识点将会相对地比较全面。此外，综合应用题占了 70 分。大纲的考查范围说得比较笼统，考查的知识点方面也只是把有关教材的章节名列举出来，并没有强调每个知识点应该掌握到什么程度。因此考生在复习的过程中要全面复习，注重基础，重点是基本概念的理解。

（二）应试策略

1. 选择合适的教材。近年来，有关这 4 门课程的书籍越来越丰富。国内许多高校都有各自不同的自编教材，同时，一些国外的经典教材也陆续被引进。这些教材，在讲授同一门课程时，侧重点往往有所不同，覆盖的知识面及知识点的取舍不尽相同，对一些概念的叙述和解释往往也存在着一定的差异。这些问题考生在复习时都应该加以注意。

2. 重视基础题的练习。这 4 门课程都有许多基本概念，这些题目看似简单，但很容易出错，而且这也反映出考生对基础概念的理解是否透彻，尤其在一些难度较大的题目中，这些基础概念也是解题的基础。

第1部分 数据结构

考查目标和知识点解析

【考查目标】

1. 理解数据结构的基本概念,掌握数据的逻辑结构、存储结构及其差异,以及各种基本操作的实现。
2. 在掌握基本的数据处理原理和方法的基础上,能够对算法进行设计与分析。
3. 能够选择合适的数据结构和方法进行问题求解。

【知识点解析】

1. 线性表

线性表是一种最简单的数据结构,但线性表在线性结构的学习乃至整个数据结构学科的学习中都具有非常重要的作用。本书系统地引入了链式存储的概念,链式存储概念是整个数据结构学科的重中之重,无论哪一章都涉及了这个概念,所以需要透彻理解。

这部分要识记线性表相关的基本概念和线性表的结构特点,掌握线性表的顺序存储方式和链式存储方式的实现方法及相应的运算,掌握链表与顺序表的相似及不同之处,掌握几种常用的链表的特点和运算。在理解线性表的顺序存储及链式存储的情况下,了解其不同的优缺点的比较及各自适用的场合。另外,还要掌握线性表的基本应用。

(1) 线性表的定义和基本操作

线性表是若干个数据元素的有限序列,线性表的特点是具有唯一的首结点和唯一的尾结点,除了首结点外的每个结点有唯一的前驱,除了尾结点外每个结点有唯一的后继。与线性表相关的基本操作包括:初始化线性表、销毁线性表、判表空、置表空、取表长、定位、取第 i 个元素、取前驱、取后继、插入元素、删除元素等。

(2) 线性表的实现

① 顺序存储结构

线性表的顺序存储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,用这种方法存储的线性表称为顺序表。

② 链式存储结构

链式存储的线性表称为链表,链表是用一组任意的存储单元存储线性表的数据元素(存储单元可以是连续的,也可以是不连续的)。在实现时,构成结点的结构体有两个域,一个数据域存放数据,一个指针域存放后继结点的地址。

循环链表是一种特殊的链表,它的特点是表中最后一个结点的指针域指向头结点,整个链表形成环状结构。

双向链表中有两个指针域,一个指针域指向后继,另一个指针域指向前驱。

当双向链表中的最后一个结点的后继指向头结点,而头结点的前驱指向最后一个结点时,就构成了双向循环链表。

③ 线性表的应用

线性表应用于多项式的表示及加法和乘法运算。

2. 栈、队列和数组

栈、队列和数组都属于线性结构的拓展，栈和队列是特殊的线性表，是操作受限的线性表，数组是数据元素为非原子类型的线性表。这部分要识记栈、队列的定义及相关数据结构的概念，理解栈和队列的特点与运算的实现，掌握栈和递归的关系，掌握循环队列的各种判断和运算，了解栈和队列的应用。

对于数组这部分，主要掌握数组的存储结构，例如按行优先、按列优先等，掌握多维数组中某个元素的位置求解。了解特殊矩阵的定义，掌握特殊矩阵的压缩存储。

(1) 栈和队列的基本概念

栈是插入、删除操作只能在表的一端进行的线性表。栈中允许插入、删除的一端称为栈顶，另一端称为栈底。栈的特点是后进先出 (Last In First Out, LIFO)。

队列是允许在一端进行插入而在另一端进行删除的线性表。允许插入的一端称为队尾，允许删除的一端称为队头。队列的特点是先进先出 (First In First Out, FIFO)。

(2) 栈和队列的顺序存储结构

顺序存储的栈称为顺序栈，在实现时也是借助数组。但是插入和删除元素必须在约定好的栈顶位置。

顺序存储的队列称为顺序队列，在实现时同样借助数组。分别设立队头指针和队尾指针，以方便进行插入和删除，插入必须在队尾进行，删除必须在队头位置。为了解决假溢出，通常采用循环队列，循环队列中对于元素的插入和删除及其队空、队满的判定是重点。

(3) 栈和队列的链式存储结构

栈的链式存储也叫链栈，我们把插入和删除操作均在链表表头进行的链表称为链栈。链栈的优点是不受连续存储空间大小的限制。

队列的链式存储实现称为链队，它实际上就是一个同时带有头指针和尾指针的单链表。头指针指向队头结点，尾指针指向队尾结点。

(4) 栈和队列的应用

由于栈结构的特殊性，栈的应用非常广泛，常见的应用有表达式求值、数制转换、括号匹配、迷宫问题及递归问题。

队列的应用主要是排队问题。

(5) 特殊矩阵的压缩存储

特殊矩阵的压缩存储包括数组和特殊矩阵的存储两部分知识。

数组的顺序表示中有两种顺序存储方法，即行主序和列主序。数组元素的位置是其下标的线性函数，需要掌握行主序和列主序两种方法所决定的数组元素位置的计算公式。

简单地说，特殊矩阵是值或零元素在矩阵中的分布有一定规律的矩阵。常见的特殊矩阵有对称矩阵、下(上)三角矩阵、对角矩阵。稀疏矩阵中非 0 元素很少且分布没有规律。

为了节约空间，常常对稀疏矩阵和特殊矩阵进行压缩存储。稀疏矩阵常用的存储方法有三元组表示和十字链表表示。

矩阵压缩存储的关键是矩阵元素位置的计算方法。

3. 树与二叉树

树和二叉树是考试的重点和难点章节，二叉树和树是两种不同的概念。在这个部分，要掌握树的定义、二叉树的定义及主要特征 (特殊的二叉树、二叉树的性质)。掌握二叉树的性质和不同存储方式及各自的优缺点和适用场合。熟练掌握二叉树的遍历方法。

在特殊的二叉树中，完全二叉树的概念是必须要熟练掌握的。另外，要掌握线索二叉树的基本概念和构造，二叉排序树、平衡二叉树的基本概念和应用。

要掌握树的存储结构和遍历、森林的遍历、树和二叉树的转换、森林和二叉树的转换等知识。最后介绍

树的应用, 要掌握哈夫曼树的构造和等价类问题。

(1) 树的概念

树是 0 个或多个结点的有限集合, 在一棵非空树中, 有一个树根和若干棵根的子树。树的定义是递归的, 即在树的定义中又用到了树的概念。

树中有许多基本术语, 如结点、结点的度、树的度、叶子、分支结点、孩子、双亲、子孙、祖先、兄弟、结点的层次、树的深度、森林等。

树中也有一些基本性质需要掌握。

(2) 二叉树

① 二叉树的定义及其主要特征

二叉树是结点的有限集合, 这个集合或者为空, 或者是由一个根结点及两棵不相交的分别称为这个根的左子树和右子树的二叉树组成。二叉树中每个结点至多有两棵子树, 且有左、右之分, 次序不能颠倒。二叉树是一种重要的树形结构, 但二叉树不是特殊的树。树和二叉树是两种不同的数据结构。

满二叉树、完全二叉树是两种特殊的二叉树。二叉树有许多性质需要读者掌握。

② 二叉树的顺序存储结构和链式存储结构

顺序存储二叉树就是用一组地址连续的存储单元依次自上而下、自左至右存储完全二叉树上的结点元素。对于一般二叉树, 则应将其每个结点与完全二叉树上的结点对照, 存储在一维数组的相应分量中。

二叉树的链式存储结构是指用一个链表来存储一棵二叉树, 二叉树中每个结点用链表的一个链结点存储。因链表的结点类型不同, 存储二叉树的链表有很多种, 常用的有二叉链表和三元链表。

③ 二叉树的遍历

二叉树的遍历是指按照一定的次序访问树中的所有结点, 并且每个结点仅被访问一次的过程。根据访问结点的顺序分为先序遍历、中序遍历和后序遍历。

④ 线索二叉树的基本概念和构造

n 个结点的二叉链表中有 $n+1$ 个空链域, 用这些链域存放结点在遍历中的前驱和后继信息构成了线索链表。其中, 指向结点前驱和后继的指针叫线索。加上线索的二叉树叫线索二叉树。对二叉树以某种次序遍历使其成为线索二叉树的过程叫线索化。

⑤ 二叉排序树

二叉排序树或者是一棵空树, 或者是具有下列性质的二叉树:

- 若左子树不空, 则左子树上所有结点的值均小于根结点的值; 若右子树不空, 则右子树上所有结点的值均大于根结点的值。
- 左、右子树也都是二叉排序树。

⑥ 平衡二叉树

平衡二叉树或者是一棵空树, 或者是具有下列性质的二叉排序树: 它的左子树和右子树都是平衡二叉树, 且左子树和右子树高度之差的绝对值不超过 1。

在平衡二叉树上插入或删除结点后, 可能使树失去平衡, 因此, 需要对失去平衡的树进行平衡化调整。设 a 结点为失去平衡的最小子树根结点, 对该子树进行平衡化调整归纳起来有以下 4 种情况: 左单旋转、右单旋转、先左后右双向旋转、先右后左双向旋转。

(3) 树和森林

① 树的存储结构

在计算机中, 树的存储有多种方式, 既可以采用顺序存储结构, 也可以采用链式存储结构, 但无论采用何种存储方式, 都要求存储结构不但能存储各结点本身的数据信息, 还要能唯一地反映树中各结点之间的逻辑关系。基本的树的存储方式包括: 双亲表示法、孩子表示法、双亲孩子表示法、孩子兄弟表示法。

② 森林与二叉树的转换

从树的孩子兄弟表示法可以看到: 如果设定一定规则, 就可以用二叉树结构表示树和森林, 这样, 对树

的操作就可以借助二叉树存储,利用二叉树上的操作来实现。

将一棵树转换为二叉树的方法如下:

- 树中所有相邻兄弟之间加一条连线。
- 对树中的每个结点,只保留它与第一个孩子结点之间的连线,删去它与其他孩子结点之间的连线。
- 以树的根结点为轴心,将整棵树顺时针转动一定的角度,使之结构层次分明。

由森林的概念可知,森林是若干棵树的集合,只要将森林中各棵树的根视为兄弟,每棵树又可以用二叉树表示,这样,森林也同样可以用二叉树表示。

森林转换为二叉树的方法如下:

- 将森林中的每棵树转换成相应的二叉树。
- 第一棵二叉树不动,从第二棵二叉树开始,依次把后一棵二叉树的根结点作为前一棵二叉树根结点的右孩子,当所有二叉树连起来后,所得到的二叉树就是由森林转换得到的二叉树。

③ 树和森林的遍历

树的遍历通常有两种方式:先根遍历和后根遍历。树的先根遍历与其转换的相应二叉树的先序遍历的结果序列相同;树的后根遍历与其转换的相应二叉树的中序遍历的结果序列相同。树的遍历算法是可以采用相应二叉树的遍历算法来实现的。

森林的遍历有前序遍历和中序遍历两种方式。森林的前序遍历和中序遍历与所转换的二叉树的先序遍历和中序遍历的结果序列相同。

(4) 树的应用

树的应用如下:

- 等价类问题。
- 哈夫曼(Huffman)树和哈夫曼编码。

哈夫曼树是指对于一组带有确定权值的叶结点,构造的具有最小带权路径长度的二叉树。哈夫曼树的构造和编码需要读者掌握。

4. 图

在数据结构中,图的结构是最复杂的,这里的概念也是最多的。读者需要掌握图的基本概念(有向图、无向图、连通、路径、子图、出度、入度、生成树、最短路径、关键路径等),掌握相关概念的计算方法。

图的存储主要有邻接矩阵法和邻接表法,要掌握有向图和无向图的这两种存储方法,要掌握图的连通和存储方法之间的关系以及在图的两种存储方法中识别图的各种基本信息。

图的遍历方法有深度优先搜索和广度优先搜索,要熟练掌握图的两种遍历算法。

图的基本应用很多,也比较复杂,要掌握这些应用的复杂度分析。掌握最小生成树的概念及构造,掌握拓扑排序、关键路径的定义与计算方法以及最短路径的计算。

(1) 图的概念

图状结构是一种比树形结构更复杂的非线性结构。在图状结构中,任意两个结点都可能相关,即结点之间的邻接关系可以是任意的。图由非空的顶点集合和一个描述顶点之间关系——边(或者弧)的集合组成。图中有许多术语需要掌握,例如无向图、有向图、顶点、边、弧、完全图、顶点的度、入度、出度、边的权、网图、路径、路径长度、回路、简单路径、简单回路、子图、连通图、连通分量、强连通图、生成树等。

(2) 图的存储及基本操作

从图的定义可知,一个图的信息包括两部分,即图中顶点的信息以及描述顶点之间的关系——边或者弧的信息。因此无论采用什么方法建立图的存储结构,都要完整、准确地反映这两方面的信息。

① 邻接矩阵法

所谓邻接矩阵的存储结构,就是用一维数组存储图中顶点的信息,用矩阵表示图中各顶点之间的邻接关系。

② 邻接表法

邻接表法是图的一种顺序存储与链式存储结合的存储方法。邻接表表示法类似于树的孩子链表表示法。

(3) 图的遍历

图的遍历是指从图中任一顶点出发,对图中的所有顶点访问一次且只访问一次。图的遍历操作和树的遍历操作功能相似。

① 深度优先搜索

深度优先搜索遍历类似于树的先根遍历,是树的先根遍历的推广。假设初始状态是图中所有顶点未曾被访问,则深度优先搜索可从图中某个顶点 v 出发,访问此顶点,然后依次从 v 的未被访问的邻接点出发深度优先遍历图,直至图中所有和 v 有路径相通的顶点都被访问到。若此时图中尚有顶点未被访问,则另选图中一个未曾被访问的顶点作为起始点,重复上述过程,直至图中所有顶点都被访问到为止。

② 广度优先搜索

广度优先搜索遍历类似于树的按层次遍历的过程。假设从图中某顶点 v 出发,在访问了 v 之后依次访问 v 的各个未曾访问过的邻接点,然后分别从这些邻接点出发依次访问它们的邻接点,并使“先被访问的顶点的邻接点”先于“后被访问的顶点的邻接点”被访问,直至图中所有已被访问的顶点的邻接点都被访问到。若此时图中尚有顶点未被访问,则另选图中一个未曾被访问的顶点作为起始点,重复上述过程,直至图中所有顶点都被访问到为止。

(4) 图的基本应用及其复杂度分析

① 最小(代价)生成树

如果无向连通图是一个网,那么它的所有生成树中必有一棵边的权值总和最小的生成树,我们称这棵生成树为最小(代价)生成树,简称为最小生成树。最小生成树的概念可以应用到许多实际问题中。常用的构造最小生成树的方法是 Prim 算法和 Kruskal 算法,其时间复杂度分别是 $O(n^2)$ 和 $O(e \log e)$, n 为结点数, e 为边的条数。

② 最短路径

最短路径问题是图的又一个比较典型的应用问题。在网图中,求点 A 到点 B 的所有路径中边的权值之和最短的那一条路径。这条路径就是两点之间的最短路径,并称路径上的第一个顶点为源点,最后一个顶点为终点。求源点到其他所有顶点的最短路径的时间复杂度是 $O(n^2)$ 。

③ 拓扑排序

拓扑排序的方法有多种,按照算法原理不同,可以有以下两类主要方法:无前驱顶点优先算法和无后继顶点优先算法。

④ 关键路径

当一个工程可以划分为若干个子活动时,其中的任何子活动可能又是以另外一些子活动为先决条件的。关键路径问题的目标是计算完成整个工程的最短路径,确定关键路径,以找出哪些活动是影响工程进度的关键。

5. 查找

查找是重点和难点,概念较多,相互联系较为紧密。查找的基本方法主要有顺序查找法、折半查找法、B-树、散列(Hash)表及其查找。需要识记基本概念,掌握平均查找长度的计算,掌握线性表上的查找,熟练掌握树表上的查找,熟练掌握哈希表的基本思想和方法。

另外,要掌握各种查找算法的分析及应用,掌握各种查找在查找成功、查找失败情况下的最好、平均、最坏的平均查找次数的计算方法。

(1) 查找的基本概念

所谓查找,就是按给定的某个值 kx ,在查找表中查找关键码为给定值 kx 的数据元素(记录)。

(2) 顺序查找法

顺序查找又称线性查找,是最基本的查找方法之一。其查找方法为:从表的一端开始,向另一端逐个按给定值 kx 与关键码进行比较,若找到,查找成功,并给出数据元素在表中的位置;若整个表检测完时仍未找到与 kx 相同的键码,则查找失败,给出失败信息。

顺序查找的基本工作就是键码的比较,因此,查找长度的量级就是查找算法的时间复杂度,其为 $O(n)$ 。

(3) 折半查找法

折半查找法只能在有序表上进行,有序表是将表中数据元素按键码升序或降序排列。折半查找的思想为:在有序表中,取中间元素作为比较对象,若给定值与中间元素的键码相等,则查找成功;若给定值小于中间元素的键码,则在中间元素的左半区继续查找;若给定值大于中间元素的键码,则在中间元素的右半区继续查找。不断重复上述查找过程,直到查找成功,或所查找的区域无数据元素,查找失败。

(4) B-树

B-树是一种平衡的多路查找树,它在文件系统中很有用。B-树的查找类似二叉排序树的查找,所不同的是 B-树每个结点上是多键码的有序表,在到达某个结点时,先在有序表中查找,若找到,则查找成功;否则,到按照对应的指针信息指向的子树中查找,当到达叶子结点时,说明树中没有对应的键码,查找失败。即在 B-树上的查找过程是一个顺指针查找结点和在结点中查找键码交叉进行的过程。

(5) 散列(Hash)表及其查找

散列方法就是选取某个函数,依该函数按键码计算元素的存储位置,并按此存放。查找时,由同一个函数对给定值 kx 计算地址,将 kx 与地址单元中元素键码进行对比,确定查找是否成功。散列方法中使用的转换函数称为散列函数,按这个思想构造的表称为散列表。因而经过散列函数变换后,可能将不同的键码映射到同一个散列地址上,这种现象称为冲突,冲突不可能避免,只能尽可能减少。所以,散列方法需要解决以下两个问题:

- 构造好的散列函数。
- 制定解决冲突的方案。

(6) 查找算法的分析及应用

读者应综合上面各种查找方法,比较各个查找方法的特点和适用情况。

6. 内部排序

内部排序也是重点和难点,且概念更多,联系更为紧密,概念之间容易混淆。在这一部分中,主要应掌握直接插入排序、折半插入排序、冒泡排序、简单选择排序、希尔排序、快速排序、堆排序、二路归并排序、基数排序的基本概念和方法。掌握这些排序方法的思想 and 流程,以及它们之间的区别。

在这部分,一个很重要的考查点就是各种内部排序算法的比较,考查各种排序算法及其思想以及优缺点和性能指标(时间复杂度,空间复杂度),进而了解各种排序方法适用的场合。

最后,就是要掌握内部排序算法的基本应用以及算法的实现。

(1) 排序的基本概念

排序是计算机程序设计中的一种重要操作,其功能是对一个数据元素集合或序列重新排列成一个按数据元素某个项值有序的序列。排序方法分为内部排序和外部排序,按照稳定性,分为稳定排序和不稳定排序。

(2) 插入排序

① 直接插入排序

直接插入排序方法是:对 n 个记录的表,可从第二个记录开始直到第 n 个记录,逐个向有序表中进行插入操作,从而得到 n 个记录按键码排序的表。

② 折半插入排序

由于插入排序的基本操作是在一个有序表中进行查找和插入,因此可以利用这个特性,使这个查找操作利用折半查找来实现,由此进行的插入排序称为折半插入排序。

(3) 气泡排序 (bubble sort)

气泡排序在每一趟排序中, 两两比较待排序记录, 若两个记录的次序相反即进行交换, 直到没有反序的记录为止。对 n 个记录的表, 第一趟排序得到一个关键码最大的记录, 第二趟排序对 $n-1$ 个记录的表再得到一个关键码最大的记录, 这样每趟排序选出的元素所在的位置也是其最终位置。

(4) 简单选择排序

简单选择排序第一趟从 n 个记录中找出关键码最小的记录与第一个记录交换; 第二趟从第二个记录开始的 $n-1$ 个记录中再选出关键码最小的记录与第二个记录交换; 如此, 第 i 趟则从第 i 个记录开始的 $n-i+1$ 个记录中选出关键码最小的记录与第 i 个记录交换, 直到整个序列按关键码排序为止。

(5) 希尔排序 (shell sort)

希尔排序又称缩小增量排序。由于直接插入排序算法简单, 在 n 值较小时, 效率比较高; 在 n 值很大时, 若序列按关键码基本有序, 效率依然较高, 其时间效率可提高到 $O(n)$ 。希尔排序即是从这两点出发给出的插入排序的改进方法。

(6) 快速排序

快速排序是通过比较关键码、交换记录, 以某个记录为界 (该记录称为支点), 将待排序列分成两部分。其中, 一部分所有记录的关键码大于等于支点记录的关键码, 另一部分所有记录的关键码小于支点记录的关键码。我们将待排序列按关键码以支点记录分成两部分的过程称为一次划分。对各部分不断划分, 直到整个序列按关键码排序为止。

(7) 堆排序

堆排序的排序思想是对 n 个元素的序列进行堆排序, 先将其建成堆, 以根结点与第 n 个结点交换; 调整前 $n-1$ 个结点成为堆, 再以根结点与第 $n-1$ 个结点交换; 重复上述操作, 直到整个序列有序为止。

(8) 二路归并排序 (merge sort)

二路归并排序的基本操作是将两个有序表合并为一个有序表。

(9) 基数排序

基数排序是一种借助于多关键码排序的思想, 是将单关键码按基数分成“多关键码”进行排序的方法。基数排序不是基于关键字比较的排序方法, 它适合于 n 值很大且关键字较小的序列。

(10) 各种内部排序算法的比较

这是排序的重点, 要求读者在对上述各种排序方法掌握的基础上, 分析比较。在所有的排序方法中, 没有哪一种绝对是绝对最优的, 需要在实际应用中根据不同情况选用, 见下表。

排序方法	时间复杂度			空间复杂度	稳定性
	平均情况	最坏情况	最好情况		
插入排序	$O(n^2)$	$O(n^2)$	$O(n)$	$O(1)$	稳定
希尔排序	$O(n\log_2 n)$	$O(n\log_2 n)$		$O(1)$	不稳定
气泡排序	$O(n^2)$	$O(n^2)$	$O(n)$	$O(1)$	稳定
快速排序	$O(n\log_2 n)$	$O(n^2)$	$O(n\log_2 n)$	$O(n\log_2 n)$	不稳定
直接选择排序	$O(n^2)$	$O(n^2)$	$O(n^2)$	$O(1)$	不稳定
堆排序	$O(n\log_2 n)$	$O(n\log_2 n)$	$O(n\log_2 n)$	$O(1)$	不稳定
二路归并排序	$O(n\log_2 n)$	$O(n\log_2 n)$	$O(n\log_2 n)$	$O(n)$	稳定
基数排序	$O(d(n+r))$	$O(d(n+r))$	$O(d(n+r))$	$O(n+r)$	稳定

第1章 概 论

一、单项选择题

【例 1-1】对数据结构，下列结论中不正确的是（ ）。

- A. 相同的逻辑结构，对应的存储结构也必相同
- B. 数据结构由逻辑结构、存储结构和基本操作三个方面组成
- C. 数据存储结构就是数据逻辑结构的机内的实现
- D. 对数据基本操作的实现与存储结构有关

解析：数据结构由数据的逻辑结构、数据的存储结构和基本操作三个方面组成，所以选项 B 的结论是正确的。

数据的逻辑结构可以看做是从具体问题抽象出来的数学模型，它与数据的存储无关。我们研究数据结构的目的是为了在计算机中实现对它的操作，为此还需要研究如何在计算机中表示一个数据结构。数据结构在计算机中的标识（又称映像）称为数据的存储结构，它所研究的是数据结构在计算机中的实现方法，包括数据结构中元素的表示及元素间关系的表示，所以选项 C 的结论是正确的。

同一种逻辑结构可以采用不同的存储方法，从而得到不同的存储结构，所以选项 A 的结论是不正确的。

数据的运算是定义在数据的逻辑结构上的，常用的运算包括：检索、插入、删除、排序等，这些操作的实现要依赖于数据的存储结构，所以选项 D 的结论是正确的。故答案是 A。

答案：A

【例 1-2】在数据结构中，从逻辑上可以将之分为（ ）。

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

解析：数据的逻辑结构是从逻辑关系上描述数据，它与数据的存储无关，是独立于计算机的。数据的逻辑结构有两大类：线性结构和非线性结构。线性结构的逻辑特征是：有且仅有一个开始结点和一个终端结点，并且所有的结点最多只有一个直接前驱和直接后继。非线性结构的逻辑特征是一个结点可能有多个直接前驱和直接后继。所以选项 D 是正确的。

答案：D

【例 1-3】下面说法错误的是（ ）。

- (1) 算法原地工作的含义是指不需要任何额外的辅助空间
- (2) 在相同的规模 n 下，复杂度 $O(n)$ 的算法在时间上总是优于复杂度 $O(2n)$ 的算法
- (3) 所谓时间复杂度是指最坏情况下，估算算法执行时间的一个上界
- (4) 同一个算法，实现语言的级别越高，执行效率就越低

- A. (1)
- B. (1), (2)
- C. (1), (4)
- D. (3)

解析：下面对各个选项分别分析：算法的空间复杂度是算法所需存储空间的量度。其中所考虑的算法所需存储空间是除输入数据和算法本身之外的辅助空间，若所需辅助空间相对于输入量来说是常数，则称此算法是原地工作的。故说法 (1) 是不正确的。

对于算法时间复杂度的计算，通常是从算法中选取一种对于所研究的问题来说是基本操作的原操作，以该基本操作在算法中重复执行的次数作为算法时间复杂度的依据。这种衡量效率的办法所得出的不是时间量，而是一种增长趋势的量度。所以在相同的规模 n 下，复杂度 $O(n)$ 的算法在时间上总是优于复杂度 $O(2n)$ 的算法，故说法 (2) 是正确的。