

2014

考研

计算机学科专业基础综合

辅导讲义

主编：崔巍

副主编：蒋本珊 孙卫真 白龙飞

- 考纲要求提纲挈领
- 考点精讲层次分明
- 常考点明确重难点
- 例题设置紧扣大纲



北京航空航天大学出版社
BEIHANG UNIVERSITY PRESS

013044357

TP3
914
2014

2014

考研

计算机学科专业基础综合

辅导讲义

主编：崔巍

副主编：蒋本珊 孙卫真 白龙飞

- 考纲要求提纲挈领
- 考点精讲层次分明
- 常考点明确重难点
- 例题设置紧扣大纲



北航

C1651631



北京航空航天大学出版社
BEIHANG UNIVERSITY PRESS

TP3
914

2014

内 容 简 介

本书严格依据最新考研大纲的要求编写,分为四部分:第一部分为数据结构,第二部分为计算机组成原理,第三部分为操作系统,第四部分为计算机网络。每章内容包括考纲要求、常考点、考点精讲三个部分。在考纲要求中明确本章的主要知识点,阐述清晰;在考点精讲中对相关课程考纲的各个知识点进行集中讲解和提炼,以帮助考生有针对性的复习,并选择了典型例题及部分真题进行分析,方便考生对每部分知识的考核方式有所把握,加强考生的应试能力;常考点部分更是编者在研究历年真题的命题思路和统考的命题规律的基础上,列出本章重点考核的知识点及考查形式以及重点知识点的应试技巧与方法。本书适合所有计算机专业考研学子。

图书在版编目(CIP)数据

2014 考研计算机学科专业基础综合辅导讲义 / 崔巍

主编. — 北京:北京航空航天大学出版社,2013.5

ISBN 978-7-5124-1077-0

I. ①2… II. ①崔… 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(2013)第 038851 号

版权所有,侵权必究。

2014 考研计算机学科专业基础综合辅导讲义

主 编 崔 巍

副主编 蒋本珊 孙卫真 白龙飞

策划编辑 谭 莉

责任编辑 郑 方

*

北京航空航天大学出版社出版发行

北京市海淀区学院路 37 号(邮编 100191) <http://www.buaapress.com.cn>

发行部电话:(010)82317024 传真:(010)82328026

读者信箱:bhpress@263.net 邮购电话:(010)82316936

北京时代华都印刷有限公司印装 各地书店经销

*

开本:787×1 092 1/16 印张:27 字数:691 千字

2013 年 5 月第 1 版 2013 年 5 月第 1 次印刷

ISBN 978-7-5124-1077-0 定价:49.80 元

若本书有倒页、脱页、缺页等印装质量问题,请与本社发行部联系调换。联系电话:(010)82317024

前 言

《2014 考研计算机学科专业基础综合辅导讲义》是崔巍考研计算机系列丛书之一。该系列丛书的编者全部为具有多年考研辅导和命题经验的名师,他们是长期在全国重点院校的计算机科学与技术学科从事一线本科及研究生课程教学的教授和副教授,在相关课程中均具有 15 年以上的教学经验,并先后编写过多本教材和教学参考书。崔巍考研计算机系列丛书适用于所有计算机(包括统考和自主命题考试)考研学子。

本书自 2008 年出版后受到全国各地考生的一致好评与推崇,每年都根据当年的考研情况作了细致的修订,为了帮助备考 2014 年计算机专业的考生在深入了解最新考试命题规律的基础上更好地把握计算机学科专业基础综合的复习要点,编者认真研究了计算机统考与自主命题考试的特点,结合多年来对这些课程的潜心研究编写此书,以帮助同学们迅速抓住考试重点、掌握难点。

全书分为四个部分:第一部分数据结构,第二部分计算机组成原理,第三部分操作系统,第四部分计算机网络。每章内容包括考纲要求、常考点、考点精讲三个部分。在考纲要求中明确本章的主要知识点,阐述清晰;在常考点部分中,编者在研究历年真题的命题思路和统考的命题规律的基础上,列出了本章重点考核的知识点及考查形式,更值得一提的是,还给出了每章重要知识点的应试技巧与方法。在考点精讲中对相关课程考纲的各个知识点进行集中讲解和提炼,以帮助考生有针对性地复习,并选择了典型的例题及部分真题进行分析,方便考生对每部分知识的考核方式有所把握,加强考生的应试能力。

本书具有以下特点:

1. 编者全部为具有多年考研辅导和命题经验的名师。
2. 明确重点考核内容。“常考点”部分简单明了地列出计算机学科专业基础综合考试中各章的考核重点、考查形式以及应试技巧与方法。
3. 复习要点层次分明。“考点精讲”部分均逐层展开,脉络清楚,利于考生建立知识框架。
4. 例题设置紧扣大纲。为使考生充分掌握相关知识要点及考试出题规律而设的例题均围绕大纲要求编制,并给出了配套习题参考。

另外,编者作为计算机专业课的授课教师,在此也为准备参加 2014 年研究生入学考试计算机专业统考的同学给出一些复习建议。

专业课的复习可分为以下三个阶段:

第一阶段:基础复习阶段(开始复习—2013 年 6 月)。这一阶段需要对“数据结构”、“计算机组成原理”、“操作系统”、“计算机网络”的教材仔细阅读一遍,了解四门课程的内容,理解每一个知识点,弄清每门课程的内在逻辑结构、重点章节等。这一阶段的复习要注意全面性。

第二阶段:强化提高阶段(2013 年 7 月—2013 年 11 月上旬)。这一阶段使用优秀的考研参考书进行深入复习,加强知识点的前后联系,建立整体框架结构。分清、整理、掌握重点和难点,完成参考书配有的习题,加深解题思路,提升解题速度。并且针对历年真题,梳理真题答案,弄清每一道题属于教材中的哪一章、哪个知识点。通过做真题要了解考试形式、考试重点、



题型设置和难易程度等内容,揣摩命题思路。这一阶段的复习要注意系统性。

第三阶段:冲刺阶段(2013年11月中下旬—考前)。这一阶段总结所有重点知识点,包括重点概念、理论和模型等,查漏补缺。复习笔记和历年真题,分析真题的出题思路,预测本年度可能考查的内容和出题思路。多做模拟试卷,进一步归类整理总结。最后全面回顾知识点、易考题目及答案,准备应考。这一阶段的复习要注意目的性。

本书的编者为了更好地帮助考试复习,针对计算机专业课考试共编写了以下五本辅导教材,分别为:

《2014 考研计算机学科专业基础综合辅导讲义》;

《2014 考研计算机学科专业基础综合考试大纲同步练习》;

《2014 考研计算机学科专业基础综合考点速记手册》;

《2014 考研计算机学科专业基础综合历年真题名师详解及 100 知识点聚焦》;

《2014 考研计算机学科专业基础综合全真模拟试卷及精析》。

其中《辅导讲义》、《同步练习》、《考点速记手册》这三本教材适用于考生在复习的各个阶段(基础阶段、强化阶段、冲刺阶段)中使用,《历年真题名师详解及 100 知识点聚焦》、《全真模拟试卷及精析》这两本适用于考生在复习的强化及冲刺阶段中使用。

本书数据结构部分由崔巍老师编写,计算机组成原理部分由蒋本珊老师编写,操作系统部分由孙卫真老师编写,计算机网络部分由白龙飞老师编写。全书由崔巍老师统稿。

在本书的编写过程中,参考了一些相关的书籍和资料,在此向这些书的作者表示深深的谢意。在编写、修改和出版本书的过程中,我们本着对考生高度负责的态度,精益求精。但由于编者水平有限,时间也比较仓促,尽管经过反复校对与修改,书中难免还存在错漏和不妥之处,敬请广大读者和专家批评指正,以便再版完善。

衷心地希望本书能帮助考生在考试中取得理想的成绩!圆梦 2014!

编者

2013 年 5 月

目 录

第 1 部分 数据结构	1
第 1 章 绪 论	2
考纲要求	2
常考点	2
考点精讲	2
1.1 基本概念	2
1.2 算法和算法的衡量	3
第 2 章 线性表	5
考纲要求	5
常考点	5
考点精讲	6
2.1 线性表的定义	6
2.2 线性表的实现	6
第 3 章 栈、队列和数组	23
考纲要求	23
常考点	23
考点精讲	24
3.1 栈	24
3.2 队 列	31
3.3 特殊矩阵的压缩存储	35
第 4 章 树与二叉树	40
考纲要求	40
常考点	40
考点精讲	41
4.1 树的概念	41
4.2 二叉树	42
4.3 树和森林	58
4.4 树的应用	63
第 5 章 图	71
考纲要求	71
常考点	71
考点精讲	72
5.1 图的概念	72
5.2 图的存储及基本操作	73
5.3 图的遍历	76



5.4 图的基本应用	80
第6章 查找	92
考纲要求	92
常考点	92
考点精讲	93
6.1 查找的基本概念	93
6.2 顺序查找法	94
6.3 折半查找法	94
6.4 B树及其基本操作、B+树的基本概念	97
6.5 散列表	101
第7章 排序	107
考纲要求	107
常考点	107
考点精讲	108
7.1 排序的基本概念	108
7.2 插入排序	108
7.3 起泡排序	110
7.4 简单选择排序	112
7.5 希尔排序	113
7.6 快速排序	114
7.7 堆排序	117
7.8 二路归并排序	121
7.9 基数排序	122
7.10 外部排序	124
7.11 各种内部排序算法的比较	126
第2部分 计算机组成原理	129
第1章 计算机系统概述	130
考纲要求	130
常考点	130
考点精讲	130
1.1 计算机发展历程	130
1.2 计算机系统层次结构	131
1.3 计算机性能指标	134
第2章 数据的表示和运算	137
考纲要求	137
常考点	137
考点精讲	138
2.1 数制与编码	138
2.2 定点数的表示和运算	147



2.3 浮点数的表示和运算	154
2.4 算术逻辑单元 ALU	160
第3章 存储器层次结构	164
考纲要求	164
常考点	164
考点精讲	165
3.1 存储器的分类	165
3.2 存储器的层次化结构	167
3.3 半导体随机存取存储器	168
3.4 主存储器与 CPU 的连接	171
3.5 双口 RAM 和多模块存储器	178
3.6 高速缓冲存储器 (Cache)	179
3.7 虚拟存储器	182
第4章 指令系统	187
考纲要求	187
常考点	187
考点精讲	188
4.1 指令格式	188
4.2 指令的寻址方式	190
4.3 CISC 和 RISC 的基本概念	196
第5章 中央处理器 (CPU)	199
考纲要求	199
常考点	199
考点精讲	200
5.1 CPU 的功能和基本结构	200
5.2 指令执行过程	201
5.3 数据通路的功能和基本结构	203
5.4 控制器的功能和工作原理	206
5.5 指令流水线	213
5.6 多核处理器的基本概念	217
第6章 总线	218
考纲要求	218
常考点	218
考点精讲	219
6.1 总线概述	219
6.2 总线仲裁	223
6.3 总线操作和定时	225
6.4 总线标准	226
第7章 输入/输出 (I/O) 系统	228



考纲要求	228
常考点	228
考点精讲	229
7.1 I/O 系统基本概念	229
7.2 外部设备	229
7.3 I/O 接口(I/O 控制器)	234
7.4 I/O 方式	236
第 3 部分 操作系统	255
第 1 章 操作系统概述	256
考纲要求	256
常考点	256
考点精讲	257
1.1 操作系统的概念、特征、功能和提供的服务	257
1.2 操作系统的发展与分类	260
1.3 操作系统的运行环境	262
1.4 操作系统体系结构	265
第 2 章 进程管理	267
考纲要求	267
常考点	268
考点精讲	268
2.1 进程与线程	268
2.2 处理机调度	277
2.3 同步与互斥	281
2.4 死 锁	296
第 3 章 存储管理	301
考纲要求	301
常考点	301
考点精讲	302
3.1 内存管理基础	302
3.2 虚拟存储管理	312
第 4 章 文件管理	321
考纲要求	321
常考点	321
考点精讲	322
4.1 文件系统基础	322
4.2 文件系统实现	331
4.3 磁盘组织与管理	333
第 5 章 输入输出(I/O)管理	337
考纲要求	337



常考点·····	337
考点精讲·····	338
5.1 I/O 管理概述·····	338
5.2 I/O 核心子系统·····	343
第 4 部分 计算机网络·····	349
第 1 章 计算机网络体系结构·····	350
考纲要求·····	350
常考点·····	350
考点精讲·····	351
1.1 计算机网络概述·····	351
1.2 计算机网络体系结构与参考模型·····	352
第 2 章 物理层·····	357
考纲要求·····	357
常考点·····	357
考点精讲·····	358
2.1 通信基础·····	358
2.2 传输介质·····	362
2.3 物理层设备·····	363
第 3 章 数据链路层·····	364
考纲要求·····	364
常考点·····	365
考点精讲·····	365
3.1 数据链路层的功能·····	365
3.2 组 帧·····	365
3.3 差错控制·····	366
3.4 流量控制与可靠传输机制·····	367
3.5 介质访问控制·····	370
3.6 局域网·····	374
3.7 广域网·····	377
3.8 数据链路层设备·····	378
第 4 章 网络层·····	380
考纲要求·····	380
常考点·····	381
考点精讲·····	381
4.1 网络层的功能·····	381
4.2 路由算法·····	383
4.3 IPv4·····	386
4.4 IPv6·····	394
4.5 路由协议·····	395



4.6	IP 组播	397
4.7	移动 IP	398
4.8	网络层设备	399
第 5 章	传输层	401
	考纲要求	401
	常考点	401
	考点精讲	402
5.1	传输层提供的服务	402
5.2	UDP 协议	403
5.3	TCP 协议	404
第 6 章	应用层	411
	考纲要求	411
	常考点	411
	考点精讲	412
6.1	网络应用模型	412
6.2	DNS 系统	412
6.3	FTP	415
6.4	电子邮件	416
6.5	WWW	418

数据结构

[illegible]



第 1 章 绪 论

考纲要求

本章在考试大纲中并未明确指出,但在本章也常出现一道选择题,主要考查算法的时间复杂度的分析。

常考点

2009—2013 年全国考研试题本章出题情况

年 份	单项选择题/分值	综合应用题/分值	总分值	考核知识点
2009	0 题	0 题	0 分	
2010	0 题	0 题	0 分	
2011	1 题×2 分	0 题	2 分	算法的时间复杂度分析
2012	1 题×2 分	0 题	2 分	算法的时间复杂度分析
2013	0 题	0 题	0 分	

本章常考点:算法的性能分析(主要包括时间复杂度和空间复杂度的分析)是数据结构课程教学的基本要求,常以选择题的形式单独出现,另外在算法设计题中也会涉及到。

应试技巧与方法:对于算法的时间复杂度分析的考核,需要考生注意两点:第一,需要理解并记住教材中经典算法的时间复杂度;第二,会分析简单算法的时间复杂度。

考点精讲

1.1 基本概念

★☆☆☆☆

1. 数据结构

数据结构是指相互之间存在着一种或多种关系的数据元素的集合。

数据结构是一个二元组: $\text{Data_Structure} = (D, R)$, 其中, D 是数据元素的有限集, R 是 D 上关系的有限集。

2. 逻辑结构

逻辑结构是指数据之间的相互关系。通常分为四类结构:

- (1) 集合:结构中的数据元素除了同属于一种类型外,别无其他关系。
- (2) 线性结构:结构中的数据元素之间存在一对一的关系。
- (3) 树型结构:结构中的数据元素之间存在一对多的关系。
- (4) 图状结构:结构中的数据元素之间存在多对多的关系。



3. 存储结构

存储结构是指数据结构在计算机中的表示,又称为数据的物理结构。通常由四种基本的存储方法实现:

(1) 顺序存储结构。数据元素顺序存放,每个存储结点只含一个元素。存储位置反映数据元素间的逻辑关系。存储密度大。有些操作(如插入、删除)效率较差。

(2) 链式存储结构。每个存储结点除包含数据元素信息外还包含一组(至少一个)指针。指针反映数据元素间的逻辑关系。这种方式不要求存储空间连续,便于动态操作(如插入、删除等),但存储空间开销大(用于指针),另外不能折半查找等。

(3) 索引存储结构。除数据元素存储在一组地址连续的内存空间外,还需建立一个索引表,索引表中索引指示存储结点的存储位置(下标)或存储区间端点(下标)。

(4) 散列存储结构。通过散列函数和解决冲突的方法,将关键字散列在连续的有限的地址空间内,并将散列函数的值解释成关键字所在元素的存储地址。其特点是存取速度快,只能按关键字随机存取,不能顺序存取,也不能折半存取。

1.2 算法和算法的衡量

★★★★★

1. 算法

算法是对特定问题求解步骤的一种描述,是指令的有限序列。其中每一条指令表示一个或多个操作。

算法具有下列特性:(1)有穷性;(2)确定性;(3)可行性;(4)输入;(5)输出。

算法和程序十分相似,但又有区别。程序不一定具有有穷性,程序中的指令必须是机器可执行的,而算法中的指令则无此限制。算法代表了对问题的解,而程序则是算法在计算机上的特定的实现。一个算法若用程序设计语言来描述,则它就是一个程序。

2. 算法的时间复杂度

算法的时间复杂度:以基本运算的原操作重复执行的次数作为算法的时间度量。

一般情况下,算法中基本操作重复执行的次数 $T(n)$ 是问题规模 n (输入量的多少,称之为问题规模)的某个函数 $f(n)$,记作: $T(n) = O(f(n))$

也可表示 $T(n) = m(f(n))$,其中 m 为常量。记号“ O ”读作“大 O ”,它表示随问题规模 n 的增大,算法执行时间 $T(n)$ 的增长率和 $f(n)$ 的增长率相同。

注意:有的情况下,算法中基本操作重复执行的次数还随问题的输入数据集不同而不同。

常见的渐进时间复杂度有: $O(1) < O(\log_2 n) < O(n) < O(n \log_2 n) < O(n^2) < O(n^3) < O(2^n) < O(n!) < O(n^n)$ 。

【例 1.1】 (2012 年联考真题第 1 题)求整数 $n(n \geq 0)$ 阶乘的算法如下,其时间复杂度是 ()。

```
int fact( int n ){
    if ( n <= 1 ) return 1;
    return n * fact( n-1 );
}
```

A. $O(\log_2 n)$ B. $O(n)$ C. $O(n \log_2 n)$ D. $O(n^2)$ **【分析与解答】** 设 $\text{fact}(n)$ 的运行时间函数是 $T(n)$ 。

```
int fact( int n ){
```

```
    if ( n <= 1 ) return 1;           // 语句①
```

```
    return n * fact( n-1 );          // 语句②
```

```
}
```

该函数中语句①的运行时间是 $O(1)$ 。语句②的运行时间是 $T(n-1) + O(1)$, 其中 $O(1)$ 为乘法运算的时间。

因此, 当 $n \leq 1$ 时, $T(n) = O(1)$; 当 $n > 1$ 时, $T(n) = T(n-1) + O(1)$ 。

则, $T(n) = O(1) + T(n-1)$

$$= 2 * O(1) + T(n-2)$$

$$= \dots$$

$$= (n-1) * O(1) + T(1)$$

$$= n * O(1)$$

$$= O(n)$$

即 $\text{fact}(n)$ 的时间复杂度为 $O(n)$, 正确答案为 B。

【配套习题】 《考试大纲同步练习》数据结构——第 1 章——第 5—9 题

3. 算法的空间复杂度

算法的空间复杂度: 是对一个算法在运行过程中临时占用的存储空间大小的量度。只需要分析除输入和程序之外的辅助变量所占额外空间。

原地工作: 若所需额外空间相对于输入数据量来说是常数, 则称此算法为原地工作, 空间复杂度为 $O(1)$ 。



第2章 线性表

考纲要求

(一) 线性表的定义和基本操作

(二) 线性表的实现

1. 顺序存储结构

(1) 线性表的顺序存储结构,靠元素存储的先后位置反映出数据元素的逻辑关系。

(2) 用一维数组表示,给定下标,可以存取相应元素,属于随机存取结构。

(3) 线性表的顺序存储结构实现插入、删除、定位等运算的算法。

2. 链式存储结构

(1) 线性表的链式存储结构,靠指针来反映数据元素的逻辑关系。

(2) 链表的存取需要从头指针开始,顺链而行,不属于随机存取结构,而是顺序存取结构。

(3) 几种常用链表的特点和相关算法设计,即单链表、单循环链表、双向链表、双向循环链表的生成、检索、插入、删除、遍历、逆置、分解、归并等操作。

(4) 从时间复杂度和空间复杂度的角度综合比较线性表在顺序和链式两种存储结构下的特点及其各自适用的场合。

3. 线性表的应用

运用顺序表和链表的特点解决复杂的应用问题。

常考点

2009—2013 年全国考研试题本章出题情况

年份	单项选择题/分值	综合应用题/分值	总分值	考核知识点
2009	0 题	1 题×15 分	15 分	链表的应用
2010	0 题	1 题×13 分	13 分	顺序表的应用
2011	0 题	1 题×15 分	15 分	顺序表的应用
2012	0 题	1 题×13 分	13 分	链表的应用
2013	1 题×2 分	1 题×13 分	15 分	顺序表、链表的应用

本章常考点:顺序表、链表应用的综合应用题,主要是以算法设计题的形式出现。

应试方法与技巧:算法设计题是很多考生的软肋,对这个题目如何备考?编者建议:第一,需要熟练掌握顺序表、链表的基本操作的实现;第二,考生应该清楚“复杂的操作都是由一种或几种基本操作的组合来实现的”,因此对于这些复杂的操作首先要分析它是由哪些基本操作实现的;第三,考试时,如果不具备拿全部分值的能力,就要考虑如何尽可能拿到较多的部分分值,也就是说,题目中“时间和空间上尽可能高效的算法”这一点要求,考生若不能写出最优的



算法或是花费时间过长而影响后面题目的作答时间,那么次优的或者较笨的算法能写出来也会拿到部分分值;第四,当然需要考生平时也多积累算法设计的经验。本章对于整个数据结构课程的学习都非常重要,是进行算法设计的基础。

考点精讲

2.1 线性表的定义

★★★★☆

线性表是一种线性结构,在一个线性表中数据元素的类型是相同的,或者说线性表是由同一类型的数据元素构成的线性结构,定义如下:

线性表是具有相同数据类型的 $n(n \geq 0)$ 个数据元素的有限序列,通常记为:

$(a_1, a_2, \dots, a_{i-1}, a_i, a_{i+1}, \dots, a_n)$

其中 n 为表长, $n=0$ 时称为空表。

需要说明的是: a_i 为序号为 i 的数据元素 ($i=1, 2, \dots, n$), 通常将它的数据类型抽象为 ElemType , ElemType 根据具体问题而定。

2.2 线性表的实现

2.2.1 线性表的顺序存储结构

★★★★★

1. 顺序表

线性表的顺序存储是指在内存中用地址连续的一块存储空间顺序存放线性表的各元素,用这种存储形式存储的线性表称其为顺序表。因为内存中的地址空间是线性的,因此,用物理上的相邻实现数据元素之间的逻辑相邻关系是既简单又自然的。

设 a_1 的存储地址为 $\text{Loc}(a_1)$, 每个数据元素占 d 个存储地址, 则第 i 个数据元素的地址为: $\text{Loc}(a_i) = \text{Loc}(a_1) + (i-1) * d \quad 1 \leq i \leq n$

这就是说只要知道顺序表首地址和每个数据元素所占地址单元的个数就可求出第 i 个数据元素的地址来,这也是顺序表具有按数据元素的序号随机存取的特点。

在程序设计语言中,一维数组在内存中占用的存储空间就是一组连续的存储区域,因此,用一维数组来表示顺序表的数据存储区域,如下描述。

线性表的静态分配顺序存储结构:

```
#define LISTSIZE 100    // 存储空间的最大分配量
typedef struct {
    ElemType elem[LISTSIZE];
    int length;          // 当前长度
} SqList;
```

在线性表的静态分配顺序存储结构中,线性表的最多数据元素个数为 LISTSIZE , 元素数量不能随意增加,这是以数组方式描述线性表的缺点。为了实现线性表最大存储数据元素数