

全国计算机等级考试指定教材配套辅导



新大纲

National Computer
Rank Examination

全国计算机等级考试 考点解析与模拟训练 (二级 Visual Basic)

陈志泊 编著

- 突出考试重点、难点及经验分析
- 题量、题型及分值统计分析
- 250道全真考试真题解析
- 10套全真模拟试卷, 500道试题



清华大学出版社

TP3
430
:2
2007

National Computer
Rank Examination

全国计算机等级考试 考点解析与模拟训练 (二级 Visual Basic)

陈志泊 编著

清华大学出版社
北京

内 容 简 介

本书根据中华人民共和国教育部考试中心颁布的《全国计算机等级考试二级考试最新大纲》关于 Visual Basic 语言程序设计的考试要求而编写,并按照考试知识点组织教材内容,共 14 章。

全书涵盖了考试大纲所要求的全部内容和知识点,前 13 章每章都包括了考试大纲要求、考试题型及比例分配、考试重点与经验分析、真题分析等四部分。大纲要求部分简明地概括了考试大纲所要求的知识点,便于读者对考点把握;考试题型及比例分配部分对 2003 年 9 月至 2006 年 4 月共 6 次考试试题进行了统计分析,列出了各知识点所占的题量、题型及分值,便于读者抓住考试的重点;考试重点与经验分析部分分析和总结了读者在学习过程中易犯错误、较难理解的内容以及有关注意事项,突出了难点;真题分析部分收集了自 2003 年 9 月至 2006 年 4 月间的全真考试试题共 250 道,并对各题进行了详细剖析,分析了解题方法和相关考点,从而指导读者加深对知识点的理解和掌握。第 14 章包括 10 套全真模拟试卷,约 500 道试题,每套模拟试卷都根据考点要求进行命题,内容紧扣知识点,读者通过练习模拟试卷不仅可以巩固所学知识,而且对考试的规律和出题特点有准确的把握。所有的模拟试卷都附有答案,便于读者检查学习效果。书中所有的例题和模拟试卷中的程序都已在 Visual Basic 6.0 中文版环境中调试通过。

本书可以作为参加计算机等级考试人员的自学、培训或辅导教材,也可以作为大专院校 Visual Basic 语言程序设计课程的教学辅助参考书。

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 13501256678 13801310933

图书在版编目(CIP)数据

全国计算机等级考试考点解析与模拟训练(二级 Visual Basic) / 陈志泊编著. —北京:清华大学出版社, 2007.2
ISBN 978-7-302-13957-7

I. 全… II. 陈… III. BASIC 语言—程序设计—水平考试—自学参考资料 IV. TP312

中国版本图书馆 CIP 数据核字(2006)第 120359 号

责任编辑:冯志强 林都佳

责任校对:张 剑

责任印制:李红英

出版发行:清华大学出版社 地 址:北京清华大学学研大厦 A 座

<http://www.tup.com.cn> 邮 编:100084

c-service@tup.tsinghua.edu.cn

社 总 机:010-62770175 邮购热线:010-62786544

投稿咨询:010-62772015 客户服务:010-62776969

印 刷 者:北京市清华园胶印厂

装 订 者:三河市新茂装订有限公司

经 销:全国新华书店

开 本:210×285 印 张:19 字 数:607 千字

版 次:2007 年 2 月第 1 版 印 次:2007 年 2 月第 1 次印刷

印 数:1~5000

定 价:29.00 元

本书如存在文字不清、漏印、缺页、倒页、脱页等印装质量问题,请与清华大学出版社出版部联系调换。联系电话:(010)62770177 转 3103 产品编号:024357-01



前言 Preface

Basic 语言自 1964 年问世以来,深受初学者欢迎,几十年来长盛不衰。随着微软公司推出窗口式操作系统 Windows 以来,促使了 Visual Basic 语言的诞生,也极大地推动了 Visual Basic 的发展,受到了广大计算机爱好者的欢迎。

现在许多大学、中专、小学相继开设了 Visual Basic 语言课程。教育部考试中心在 2001 年的全国计算机等级考试大纲中,把 Visual Basic 作为二级(程序设计)考试的可选语种之一。为了帮助准备参加全国计算机等级考试的读者以及 Visual Basic 语言的爱好者学习和掌握 Visual Basic 语言程序设计,我们根据教育部考试中心颁布的《全国计算机等级考试二级考试最新大纲》关于 Visual Basic 语言程序设计的考试要求编写了本书。并与《Visual Basic 程序设计(二级)教程》、《Visual Basic 程序设计应试辅导(二级)》组成一套丛书,本套丛书是广大 Visual Basic 语言读者的重要教材和参考书,它们各自有所侧重,又相互补充,形成一套完整的教材体系。

本书共包括 14 章,分别是:计算机公共基础知识,Visual Basic 程序开发环境,Visual Basic 对象及其操作,数据类型、运算符和表达式,数据输入输出,常用标准控件,控制结构,数组,过程,菜单和对话框,多重窗体与环境应用,键盘与鼠标事件过程,数据文件,全真模拟试卷。全书涵盖了考试大纲所要求的全部内容和知识点,前 13 章每章都包括了考试大纲要求、考试题型及比例分配、考试重点与经验分析、真题分析等四部分。全书以 Visual Basic 的考点和知识点为线索,科学合理地组织有关章节内容,分析和总结了读者在学习过程中易犯错误、较难理解的内容以及有关注意事项,突出了难点和重点,便于读者全面、系统地学习和掌握 Visual Basic 程序设计语言及其二级考试考点。同时,收集了自 2003 年 9 月至 2006 年 4 月的全真考试试题共 250 道,对考试真题进行了统计分析,列出了各知识点所占的题量、题型及分值,并对各题进行了详细剖析,分析了解题方法和相关考点,从而指导读者加深对知识点的理解和掌握。

针对相关的知识点,本书编写了 10 套全真模拟试卷并附有参考答案,约 500 道试题,每套模拟试卷都根据考点要求进行命题,内容紧扣知识点,读者通过练习模拟试卷不仅可以巩固所学知识,而且对考试的规律和出题特点有准确的把握,便于读者检查学习效果。

本书重要特点是:内容全面,涵盖笔试全部内容;讲解深入浅出,文字通俗易懂,对知识点、考点讲解透彻,循序渐进,使读者能很容易地掌握和理解相关知识点;以考试真题为例题,更具有针对性、实用性;提供全真模拟试卷训练,提高读者实战能力。

全书全部由陈志泊编写,由谭浩强教授修改定稿。另外,黄金燕、张惠萍、冯振兴、高映雪、罗云深参加了全书的文字校对工作并提出了很多宝贵的修改意见,在此表示衷心的感谢。

本书如有不当之处,敬请专家和广大读者批评指正。

编 著
2006 年 9 月



目录 Contents

第1章 计算机公共基础知识	1
1.1 大纲要求	2
1.2 考试题型及比例分配	2
1.3 考试重点与经验分析	2
1.3.1 基本数据结构与算法	2
1.3.2 程序设计基础	6
1.3.3 软件工程基础	7
1.3.4 结构化分析方法	8
1.3.5 结构化设计方法	9
1.3.6 软件测试	10
1.3.7 程序的调试	11
1.3.8 数据库系统的基本概念	11
1.3.9 数据模型	13
1.3.10 关系代数	14
1.3.11 数据库设计	15
1.4 真题分析	15
1.4.1 数据结构与算法	15
1.4.2 程序设计基础	18
1.4.3 软件工程基础	18
1.4.4 数据库设计基础	21
第2章 Visual Basic 程序开发环境	24
2.1 大纲要求	25
2.2 考试题型及比例分配	25
2.3 考试重点与经验分析	25
2.4 真题分析	26
第3章 Visual Basic 的对象及其操作	28
3.1 大纲要求	29
3.2 考试题型及比例分配	29
3.3 考试重点与经验分析	29
3.3.1 面向对象程序设计的有关概念	29
3.3.2 窗体及其常用属性与事件	31
3.3.3 几点注意事项	31
3.3.4 事件驱动的编程机制与运行方式	33
3.4 真题分析	33

第 4 章 数据类型、运算符及表达式	38
4.1 大纲要求	39
4.2 考试题型及比例分配	39
4.3 考试重点与经验分析	39
4.3.1 变量及其定义	39
4.3.2 用户自定义数据类型及使用	41
4.3.3 枚举类型及其使用	42
4.3.4 用户自定义符号常量	42
4.3.5 表达式与运算符的优先级	43
4.3.6 注意事项	43
4.4 真题分析	44
4.4.1 常量、变量与数据类型	44
4.4.2 表达式及其运算	45
第 5 章 数据输入输出	50
5.1 大纲要求	51
5.2 考试题型及比例分配	51
5.3 考试重点与经验分析	51
5.3.1 Print 方法与 Format 函数	51
5.3.2 InputBox 函数及其使用	52
5.3.3 MsgBox 函数与 MsgBox 语句	53
5.3.4 Move 方法及其使用	54
5.4 真题分析	54
5.4.1 Print 方法与 Format 函数	54
5.4.2 InputBox 函数及其使用	55
5.4.3 MsgBox 函数与 MsgBox 语句	56
5.4.4 Move 方法及其使用	57
第 6 章 常用标准控件	59
6.1 大纲要求	60
6.2 考试题型及比例分配	60
6.3 考试重点与经验分析	60
6.3.1 文本类控件	60
6.3.2 按钮类控件	61
6.3.3 图形类控件	61
6.3.4 选择类控件	62
6.3.5 滚动条控件	64
6.3.6 计时器控件	64
6.3.7 框架控件	65
6.3.8 控件的默认属性	65
6.3.9 对象的输入焦点与 Tab 键序	65
6.3.10 事件驱动应用程序的工作方式	66

6.4	真题分析	66
6.4.1	文本类控件与按钮类控件	66
6.4.2	图形类控件	70
6.4.3	选择类控件	71
6.4.4	滚动条控件	73
6.4.5	计时器控件	75
6.4.6	对象的输入焦点与 Tab 键序	77
第 7 章	控制结构	78
7.1	大纲要求	79
7.2	考试题型及比例分配	79
7.3	考试重点与经验分析	79
7.3.1	选择结构	80
7.3.2	循环结构	82
7.4	真题分析	87
7.4.1	选择结构	87
7.4.2	循环结构	92
第 8 章	数组	109
8.1	大纲要求	110
8.2	考试题型及比例分配	110
8.3	考试重点与经验分析	110
8.3.1	数组的概念及其定义方法	110
8.3.2	数组的基本操作	112
8.3.3	控件数组	113
8.3.4	几点注意事项	114
8.4	真题分析	115
8.4.1	数组的概念及其定义方法	115
8.4.2	数组的基本操作	116
8.4.3	控件数组	128
第 9 章	过程	134
9.1	大纲要求	135
9.2	考试题型及比例分配	135
9.3	考试重点与经验分析	135
9.3.1	Sub 过程的定义与调用	136
9.3.2	Function 过程的定义与调用	137
9.3.3	过程的参数及其传递	137
9.3.4	Shell 函数	140
9.3.5	变量与过程的作用范围	140
9.4	真题分析	141
9.4.1	Sub 过程的定义、调用与参数传递	141



9.4.2	Function 过程的定义、调用与参数传递	147
9.4.3	Shell 函数与变量作用范围	155
第 10 章	菜单和对话框	156
10.1	大纲要求	157
10.2	考试题型及比例分配	157
10.3	考试重点与经验分析	157
10.3.1	菜单设计及菜单属性控制	157
10.3.2	通用对话框控件与通用对话框	159
10.4	真题分析	164
10.4.1	菜单设计及菜单属性控制	164
10.4.2	通用对话框控件与通用对话框	166
第 11 章	多重窗体与环境应用	169
11.1	大纲要求	170
11.2	考试题型及比例分配	170
11.3	考试重点与经验分析	170
11.4	真题分析	172
第 12 章	键盘与鼠标事件过程	175
12.1	大纲要求	176
12.2	考试题型及比例分配	176
12.3	考试重点与经验分析	176
12.4	真题分析	181
第 13 章	数据文件	188
13.1	大纲要求	189
13.2	考试题型及比例分配	189
13.3	考试重点与经验分析	189
13.4	真题分析	199
第 14 章	全真模拟试卷	205
14.1	全真模拟试卷 (一)	206
14.2	全真模拟试卷 (二)	214
14.3	全真模拟试卷 (三)	221
14.4	全真模拟试卷 (四)	230
14.5	全真模拟试卷 (五)	238
14.6	全真模拟试卷 (六)	246
14.7	全真模拟试卷 (七)	254
14.8	全真模拟试卷 (八)	261
14.9	全真模拟试卷 (九)	269
14.10	全真模拟试卷 (十)	277

附录 全真模拟试卷参考答案	286
全真模拟试卷（一）参考答案	287
全真模拟试卷（二）参考答案	287
全真模拟试卷（三）参考答案	288
全真模拟试卷（四）参考答案	288
全真模拟试卷（五）参考答案	289
全真模拟试卷（六）参考答案	289
全真模拟试卷（七）参考答案	290
全真模拟试卷（八）参考答案	290
全真模拟试卷（九）参考答案	291
全真模拟试卷（十）参考答案	292
参考文献	293

第1章

计算机公共基础知识

1.1 大纲要求

1. 掌握算法的基本概念。
2. 掌握基本数据结构及其操作。
3. 掌握基本排序和查找算法。
4. 掌握逐步求精的结构化程序设计方法。
5. 掌握软件工程的基本方法, 具有初步应用相关技术进行软件开发的能力。
6. 掌握数据的基本知识, 了解关系数据库的设计。

1.2 考试题型及比例分配

比例: 30%。一般 10 个选择题和 5 个填空题, 具体如表 1.1 所示。

表 1.1 考试题型及比例分配

年 份	题 型	分 数	年 份	题 型	分 数
2005 年 4 月	选择 (1~10) 填空 (1~5)	30	2006 年 4 月	选择 (1~10) 填空 (1~5)	30
2005 年 9 月	选择 (1~10) 填空 (1~5)	30			

※ 括号内为考试题号。

1.3 考试重点与经验分析

1.3.1 基本数据结构与算法

1. 算法的基本概念及特征

算法的概念是考试的重点, 是指解题方案的准确而完整的描述, 它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。

算法具有可行性、确定性、有穷性、拥有足够的情报等特征。其中, 确定性和有穷性是考试的重点。

算法的确定性, 是指算法中的每一步骤都必须有明确定义, 不允许有模棱两可的解释, 也不允许有多义性。

算法的有穷性, 是指算法必须能在有限的时间内做完, 即算法必须能在执行有限个步骤之后终止。

2. 算法复杂度的概念和意义

一个算法质量的好坏可从算法的时间复杂度和空间复杂度两个方面来衡量。算法的复杂度也是每次考试的重点, 要注意明确有关概念。

算法的时间复杂度是指算法所需要的计算工作量; 算法的空间复杂度是指执行这个算法所需要的内存空间。

3. 数据结构的定义

数据结构主要研究和讨论以下三个方面的问题:

- ① 数据集合中各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。

要注意数据的逻辑结构与存储结构的区别与联系。

4. 线性结构与非线性结构

根据数据结构中各元素之间前后件关系的复杂程度,一般数据结构分为两大类型:线性结构与非线性结构。要注意这两种结构的特征、它们之间的区别以及常见的有关结构。

(1) 线性结构(或称线性表)有以下主要特征:

- ① 有且只有一个根结点,它无前件。
- ② 有且只有一个终结点,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称为线性表的长度,当结点个数为0时,该线性表为空表。

常见的线性结构有:线性表、栈、队列等。

(2) 如果一个数据结构不是线性结构,则称之为非线性结构,常见的非线性结构有:树、二叉树、图等。

5. 线性表的顺序存储结构(顺序表)及其插入与删除运算

线性表既可以采用顺序存储结构,又可以采用链式存储结构进行存储。要注意掌握二者在存储数据方面的方式与特点。

(1) 线性表的顺序存储结构的特点

- 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。

由此可见,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。

(2) 线性表在顺序存储结构下的插入与删除运算

线性表在顺序存储结构下,若在第 i ($1 \leq i \leq n$, n 为线性表中元素的个数)个位置上插入一个新元素,则首先从最后一个(即第 n 个)元素开始,直到第 i 个元素之间共有 $n-i+1$ 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 个位置。插入结束后,线性表的长度增1。

显然,在最好的情况下,插入位置在线性表的末尾进行,即在第 n 个元素之后插入运算,此时,不需要移动表中的元素。而在最坏的情况下,插入位置在第1个元素上,此时需要移动表中所有的元素。在平均情况下,要在线性表中插入一个新元素,需要移动表中一半的元素。

同理,线性表在顺序存储结构下的删除运算,也需要移动表中的元素,只不过是向前移动,在最好的情况下,删除运算在线性表的末尾进行,即删除第 n 个元素,此时,不需要移动表中的元素。而在最坏的情况下,删除位置在第1个元素上,此时需要移动表中所有的元素。在平均情况下,要在线性表中删除一个元素,需要移动表中一半的元素。

线性表的顺序存储结构的特点,以及在顺序存储结构下插入与删除运算的效率是考试的重点。

6. 栈与队列

要深刻领会二者的概念,以及对二者进行插入、删除运算的特点,这是考试的重点。

栈实际上也是线性表,只不过是一种特殊的线性表。在这种特殊的线性表中,其插入与删除运算都只在线性表的一端进行。即在这种线性表的结构中,一端是封闭的,不允许进行插入与删除元素;另一端是开口的,允许插入与删除元素。允许插入与删除运算的一端称为栈顶,而不允许插入与删除运算的一端称为栈底。栈顶元素总是最后被插入的元素,从而也是最先能被删除的元素;栈底元素总是最先被插入的元素,从而也是最后

才能被删除的元素。即栈是按照“先进后出”(First In Last Out, FILO)或“后进先出”(Last In First Out, LIFO)的原则组织数据的,因此,栈也被称为“先进后出”表或“后进先出”表。由此可以看出,栈具有记忆作用。对栈常可以进行进栈、出栈、读取栈顶元素的运算。

队列是指允许在一端进行插入运算、而在另一端进行删除运算的线性表。允许插入运算的一端称为队尾,通常用一个称为队尾指针的指针指向队尾元素,即队尾指针总是指向最后被插入的元素。允许删除运算的一端称为队头,通常也用一个队头指针指向队头的元素。显然,在队列这种数据结构中,最先插入的元素将最先能够被删除,反之,最后插入的元素将最后才能被删除。因此,队列又称为“先进先出”(First In First Out, FIFO)或“后进后出”(Last In Last Out, LILO)的线性表。对队列可以进行入队、退队运算。

7. 循环队列

重点注意循环队列的概念、存储方式。

循环队列是队列顺序存储结构的一种,它将 m 个物理上连续的存储单元,在逻辑上形成一个环状,供队列循环使用。

具体来说,在循环队列中,用队尾指针 $rear$ 指向队列中的队尾元素,用队头指针 $front$ 指向队头元素的前一个位置,因此,从队头指针 $front$ 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。

8. 线性表的链式存储结构(线性链表)

(1) 线性表的链式存储结构及其有关运算

在线性表的链式存储结构中,一个元素用一个结点来存储,每个结点含有两个域,一个数据域用于存放数据元素值,一个指针域,用于存放指针,该指针用于指向该结点的前一个或后一个结点(即前件或后件)。

在链式存储结构中,存储数据结构的存储空间可以不连续,各数据结点的存储顺序(即存储空间位置)与数据元素之间的逻辑关系可以不一致,而数据元素之间的逻辑关系是由指针域来确定的。要特别注意,线性表的链式存储结构与顺序存储结构方式的不同。

线性表的链式存储结构又称为线性链表。

对线性链表的运算主要包括:查找指定元素、插入、删除运算等。不像顺序存储结构那样,对线性链表的插入与删除运算不需要移动数据元素,而只需改变有关结点的指针即可。

(2) 循环链表

在对线性链表进行运算的过程中,虽然其插入与删除运算比较方便,但还存在一个问题,即对于空表和对第一个结点的处理必须单独考虑,使空表与非空表的运算不统一。为了克服线性链表的这个缺点,可以采用另一种链接方式,即循环链表的结构,使整个链成为一个环状结构。在此,需要注意线性链表与循环链表在存储方式上的不同。

循环链表的结构与线性链表相比,具有以下两个特点:

① 在循环链表中增加了一个表头结点,其数据域为任意或者根据需要来设置,指针域指向线性表的第一个元素的结点。循环链表的头指针指向表头结点。

② 循环链表中最后一个结点的指针域不是空,而是指向表头结点。即在循环链表中,所有结点的指针构成了一个环状链。

9. 树与二叉树

树是一种非线性结构,在这种结构中,所有数据元素之间的关系具有明显的层次特性。而二叉树也是一种非线性结构,它与树结构相似,并且树结构的所有术语都可以用到二叉树这种数据结构上。

二叉树具有以下两个特点:

① 非空二叉树只有一个根结点。

② 每一个结点最多有两棵子树,且分别称为该结点的左子树与右子树。

因此,二叉树中每一个结点的度最大为 2,即所有子树(左子树或右子树)也均为二叉树。

对于二叉树,其概念与性质是考试的重点。要特别注意二叉树的有关性质。

10. 满二叉树与完全二叉树

满二叉树与完全二叉树是两种特殊形态的二叉树,对这两种二叉树的概念上的理解是考试的重点。

(1) 满二叉树

满二叉树是指这样的一种二叉树:除最后一层外,每一层上的所有结点都有两个子结点,也就是说,在满二叉树中,每一层上的结点数都达到最大值,即在满二叉树的第 k 层有 2^{k-1} 个结点,且深度为 m 的满二叉树有 $2^m - 1$ 个结点。

(2) 完全二叉树

完全二叉树是这样的二叉树,除最后一层外,每一层上的结点数均达到最大值。

11. 二叉树的遍历

二叉树的遍历是指不重复地访问二叉树中的所有结点。二叉树的遍历可以分为三种:前序遍历、中序遍历、后序遍历。这三种遍历方式是每次考试的重点,要求对于某一棵二叉树应能写出对应的遍历序列。

12. 顺序查找及其特点

需要重点掌握顺序查找的概念及查找的效率。

顺序查找又称顺序搜索。它从线性表的第一个元素开始,依次将线性表中的元素与被查元素进行比较,若相等则表示找到(即查找成功);若线性表中所有的元素都与被查元素进行了比较但都不相等,则表示线性表中没有要找的元素(即查找失败)。

很明显,在进行顺序查找过程中,在最好的情况下,如果线性表中的第一个元素就是被查元素,则只需做一次比较就查找成功,查找效率最高;但如果被查的元素是线性表中的最后一个元素,或者被查元素根本不在线性表中,则为了查找这个元素需要与线性表中所有的元素进行比较,这是顺序查找的最坏情况。在平均情况下,利用顺序查找法在线性表中查找一个元素,大约与线性表中一半的元素进行比较。

13. 二分法查找及其特点

重点掌握二分法查找的适用对象、查找的方法与查找效率。

二分法查找只适用于顺序存储的有序表。此处的有序表是指线性表中的元素按值非递减排列(即从小到大,但允许相信元素值相等)。

设有序线性表的长度为 n ,被查元素为 x ,则二分查找的方法如下所述。

将 x 与线性表的中间项进行比较:

若中间项的值等于 x ,则说明查到,查找结束;

若 x 小于中间项的值,则在线性表的前半部分(即中间项以前的部分)以相同的方法进行查找;

若 x 大于中间项的值,则在线性表的后半部分(即中间项以后的部分)以相同的方法进行查找。

这个过程一直进行到查找成功或子表长度为 0(说明线性没有这个元素)为止。

显然,当有序线性表为顺序存储时才能采用二分查找,并且,二分查找的效率要比顺序查找高得多。对于长度为 n 的有序线性表,在最坏情况下,二分查找只需要比较 $\log_2 n$ 次。

14. 排序技术

排序是指将一个无序序列整理成按值非递减顺序排列的有序序列。常见的排序方法主要有交换类排序、插入类排序和选择类排序。各种排序方法的特点以及在最坏情况下的排序效率是考试的重点。

(1) 交换类排序法

交换类排序法是指借助数据元素之间的互相交换进行排序的一种方法。冒泡排序法与快速排序法都属于交换类排序方法。

冒泡排序法是一种最简单的交换类排序方法,它是通过相邻数据元素的交换逐步将线性表变成有序。假设线性表的长度为 n ,则在最坏情况下,冒泡排序需要经过 $n/2$ 遍的从前往后的扫描和 $n/2$ 遍的从后往前的扫描,需要的比较次数为 $n(n-1)/2$ 。但这个工作量不是必需的,一般情况下要小于这个工作量。

快速排序法也是一种交换类的排序方法,但由于它比冒泡排序法的速度快,因此称之为快速排序法。其关键是对线性表进行分割,以及对各分割出的子表再进行分割。

(2) 插入类排序法

插入类排序法主要有简单插入排序法和希尔排序法。

简单插入排序法,是指将无序序列中的各元素依次插入到已经有序的线性表中。在这种排序方法中,每一次比较后最多移掉一个逆序,因此,这种排序方法的效率与冒泡排序法相同。在最坏情况下,简单插入排序需要 $n(n-1)/2$ 次比较。

希尔排序法对简单插入排序做了较大的改进。它是将整个无序序列分割成若干小的子序列分别进行插入排序。希尔排序的效率与所选取的增量序列有关。在最坏情况下,希尔排序所需要的比较次数为 $O(n^{1.5})$ 。

(3) 选择类排序

选择类排序主要有简单选择类排序法和堆排序法。

简单选择排序法的基本思想是:扫描整个线性表,从中选出最小的元素,将它交换到表的最前面(这是它应有的位置);然后对剩下的子表采用同样的方法,直到子表空为止。对于长度为 n 的线性表,在最坏情况下需要比较 $n(n-1)/2$ 次。

堆排序法也属于选择类排序法。具有 n 个元素的序列 $(h_1, h_2, \dots, h_n)$, 当且仅当满足条件:

$$\begin{cases} h_i \geq h_{2i} \\ h_i \geq h_{2i+1} \end{cases} \quad \text{或} \quad \begin{cases} h_i \leq h_{2i} \\ h_i \leq h_{2i+1} \end{cases}$$

($i=1, 2, \dots, n/2$) 时称之为堆。可见,堆顶元素(即第一个元素)必为最大项。

堆排序的方法对于规模较小的线性表并不适合,但对于较大规模的线性表来说是很有效的。在最坏情况下,堆排序需要比较的次数为 $O(n \log_2 n)$ 。

1.3.2 程序设计基础

1. 程序设计方法与风格

除了好的程序设计方法和技术外,程序设计风格也是很重要的内容。程序设计风格是指编写程序时所表现出的特点、习惯和逻辑思路。要形成良好的程序设计风格,主要应注重和考虑下述一些因素。

(1) 源程序文档化

符号名的命名:符号名的命名应具有一定的实际含义,以便于对程序功能的理解。

程序注释:注释一般分为序言性注释和功能性注释。序言性注释通常位于每个程序的开头部分,它给出程序的整体说明;而功能性注释的位置一般嵌在源程序体中,主要描述其后的语句或程序做什么。

视觉组织:可以在程序中利用空格、空行、缩进等技巧使程序层次清晰。

(2) 数据说明的方法

主要包括数据说明的次序规范化、说明语句中变量安排有序化和使用注释来说明复杂数据结构等。

(3) 语句的结构

语句构造应该简单直接,不应该为提高效率而把语句复杂化。

(4) 输入和输出

输入和输出方式和格式应尽可能方便用户的使用。

2. 结构化程序设计

结构化程序设计的主要原则、设计要素应是重点掌握的内容。

由于软件危机的出现,人们开始研究程序设计方法,其中最受关注的是结构化程序设计方法。结构化程序设计方法的主要原则可以概括为自顶向下、逐步求精、模块化、限制使用 goto 语句。

在结构化程序设计的具体实施中,要注意把握如下要素。

(1) 使用程序设计语言中的顺序、选择、循环等有限的控制结构表示程序的控制逻辑。

- (2) 使用的控制结构只准许有一个入口和一个出口。
- (3) 程序语句组成容易识别的块，每块只有一个入口和一个出口。
- (4) 复杂结构应该用嵌套的基本控制结构进行组合嵌套来实现。
- (5) 语言中所没有的控制结构，应该采用前后一致的方法来模拟。
- (6) 严格控制 goto 语句的使用。

3. 面向对象的程序设计及其重要概念

面向对象的程序设计方法及其有关概念是重点掌握的内容，也是考试的重点。

面向对象的程序方法主张从客观世界固有的事物出发来构造系统，提倡用人类在现实生活中常用的思维方法来认识、理解和描述客观事物，强调最终建立的系统能够映射问题域。它具有与人类习惯的思维方法一致、稳定性好、可重用性好、易于开发大型软件产品、可维护性好等优点。

在面向对象的程序方法中，有如下重要的概念。

(1) 对象

用来表示客观世界中的任何实体，即应用领域中有意义的、与所要解决的问题有关系的任何事物都可以作为对象。它既可以是具体的物理实体的抽象，也可以是人为的概念，或者是任何有明确边界和意义的东西。面向对象的程序设计方法中涉及的对象是系统中用来描述客观事物的一个实体，是构成系统的一个基本单位，它由一组表示其静态特征的属性和它可执行的一组操作组成。

属性即对象所包含的信息，它在设计对象时确定，一般只能通过执行对象的操作来改变。

操作描述了对象执行的功能，若通过消息传递，还可以为其他对象使用。

(2) 类和实例

将属性、操作相似的对象归为类，即是具有共同属性、共同方法的对象的集合。因此，类是对象的抽象，它描述了属于该对象类型的所有对象的性质，而一个对象则是其对应类的一个实例。

(3) 消息

面向对象的世界是通过对象与对象间彼此的相互合作来推动的，对象间的这种相互合作需要一个机制协助进行，这样的机制称为“消息”。消息是一个实例与另一个实例之间传递的信息，它请示对象执行某一处理或回答某一要求的信息，它统一了数据流和控制流。一个消息由接收消息的对象的名称、消息标识符（即消息名）、零个或多个参数组成。

(4) 继承

继承是面向对象的方法的一个主要特征。继承是使用已有的类定义作为基础建立新类的定义技术。已有的类可当做基类来引用，则新类相应地可当做派生类来引用。一个类的上层可以有父类，下层可以有子类。一个类直接继承其父类的描述（数据和操作）或特性，子类自动地共享基类中定义的数据和方法。

(5) 多态性

对象根据所接受的消息而做出动作，同样的消息被不同的对象接受时可导致完全不同的行动，该现象称为多态性。

1.3.3 软件工程基础

1. 软件定义与软件特点

软件的定义是重点掌握的内容。

计算机软件是计算机系统中与硬件相互依存的另一部分，是包括程序、数据及相关文档的完整集合。它具有以下特点：

- (1) 软件是一种逻辑实体，而不是物理实体，具有抽象性。
- (2) 软件的生产与硬件不同，它没有明显的制作过程。
- (3) 软件在运行、使用期间不存在磨损、老化问题。



- (4) 软件的开发、运行对计算机系统具有依赖性, 受计算机系统的限制, 这导致了软件移植的问题。
- (5) 软件复杂性高, 成本昂贵。
- (6) 软件开发涉及诸多的社会因素。

2. 软件危机与软件工程

重点掌握软件危机与软件工程的定义, 软件工程产生的背景、研究目标与研究内容。

软件工程概念的出现源自软件危机。

软件危机是泛指在计算机软件的开发和维护过程中所遇到的一系列严重问题。主要表现在: 软件需求的增长得不到满足、软件开发成本和进度无法控制、软件质量难以保证、软件不可维护或维护程度非常低、软件的成本不断提高、软件开发生产率的提高赶不上硬件的发展和应用需求的增长等。

为了消除软件危机, 逐步形成了软件工程的定义, 软件工程是试图用工程、科学和数学的原理与方法研制、维护计算机软件的有关技术及管理方法。其主要思想是强调在软件开发过程中需要应用工程化原则。

软件工程具有方法、工具和过程三个要素。方法是完成软件工程项目的技术手段; 工具支持软件的开发、管理、文档生成; 过程支持软件开发的各个环节的控制、管理。

软件工程的定义是, 在给定成本、进度的前提下, 开发出具有有效性、可靠性、可理解性、可维护性、可重用性、可适应性、可移植性、可追踪性和可互操作性且满足用户需求的产品。

基于软件工程的定义, 软件工程研究的内容主要包括: 软件开发技术和软件工程管理。

为了达到软件工程定义, 在软件开发过程中, 必须遵循以下基本原则: 抽象、信息隐蔽、模块化、局部化、确定性、一致性、完备性和可验证性。

3. 软件生命周期

通常, 将软件产品从提出、实现、使用维护到停止使用退役的过程称为软件生命周期。一般包括可行性研究与需求分析、设计、实现、测试、交付使用以及维护等活动。

4. 软件开发工具与软件开发环境

软件开发工具和环境的使用进一步提高了软件的开发效率、维护效率和软件质量。软件开发工具的完善和发展促进了软件开发方法的进步和完善, 促进了软件开发的高速度和高质量。

软件开发环境或称软件工程环境是全面支持软件开发全过程的软件工具集合。这些软件工具按照一定的方法或模式组合起来, 支持软件生命周期内的各个阶段和各项任务的完成。

1.3.4 结构化分析方法

1. 需求分析

需求分析的概念、主要工作以及常用的需求分析方法是考试的重点。

软件需求是指用户对目标软件系统在功能、行为、性能、设计约束等方面的期望。需求分析的任务是发现需求、求精、建模和定义需求的过程。

需求分析阶段的工作主要包括需求获取、需求分析、编写需求规格说明书和需求评审四个方面。

常见的需求分析方法有结构化分析方法和面向对象的分析方法 (OOA—Object Oriented method)。

其中, 在结构化分析方法中, 主要包括: 面向数据流的结构化分析方法 (Structured Analysis, SA)、面向数据结构的 Jackson 方法 (Jackson System Development method, JSD)、面向数据结构的结构化数据系统开发方法 (Data Structured, DSSD)

2. 结构化分析方法

结构化分析方法是结构化程序设计理论在软件需求分析阶段的运用。结构化分析的常用工具主要有以下四种, 是需要重点掌握的内容。

(1) 数据流图 (Data Flow Diagram, DFD): 数据流图是描述数据处理过程的工具, 是需求理解的逻辑模型的图形表示, 它从数据传递和加工的角度, 来刻画数据流从输入到输出的移动变换过程。数据流图中的主要图