



“十二五” 高等院校精品规划教材

“SHI ER WU” GAODENG YUANXIAO JINGPIN GUIHUA JIAOCAI

Visual FoxPro

数据库程序设计

实习实训指南

主编 盖玉莲

赠送配套电子课件



北京交通大学出版社
<http://press.bjtu.edu.cn>

“十二五”高等院校精品规划教材

Visual FoxPro 数据库程序设计实习实训指南

主 编 盖玉莲
副主编 谢 勇 邓小盾
主 审 孙姜燕

北京交通大学出版社

· 北京 ·

内 容 简 介

本书与《Visual FoxPro 数据库程序设计》章节对应, 配套使用。全书在内容上加强了每个知识单元实践技能训练环节的指导, 精心设计了大量的相关案例, 并对案例进行了认真细致的分析。为了便于学生掌握, 全书的主要内容都从知识要点、典型试题分析、上机实习指导和实训案例分析四个层次进行编排与阐述。

本书在知识结构上是参考教育部考试中心制定的《全国计算机等级考试二级 Visual FoxPro 数据库程序设计考试大纲》要求编写的。全书知识要点全面, 典型试题及难点分析透彻, 重点突出, 具有较强的针对性, 以加深学生对课堂教学内容的理解, 提高学生的动手操作能力。

本书适合作为高等院校(本科)非计算机专业的教材, 也可作为报考全国计算机等级考试二级 Visual FoxPro 数据库程序设计考试考生的参考用书。

版权所有, 侵权必究。

图书在版编目(CIP)数据

Visual FoxPro 数据库程序设计实习实训指南 / 盖玉莲主编.

— 北京: 北京交通大学出版社, 2013. 2

(“十二五”高等院校精品规划教材)

ISBN 978 - 7 - 5121 - 1376 - 3

I. ① V… II. ① 盖… III. ① 关系数据库系统 - 程序设计 - 高等学校 - 教学参考资料 IV. ① TP311. 138

中国版本图书馆 CIP 数据核字 (2013) 第 027707 号

责任编辑: 薛飞丽

出版发行: 北京交通大学出版社

电话: 010 - 51686414

北京市海淀区高粱桥斜街 44 号

邮编: 100044

印 刷 者: 北京时代华都印刷有限公司

经 销: 全国新华书店

开 本: 185 × 260 印张: 10.5 字数: 252 千字

版 次: 2013 年 2 月第 1 版 2013 年 2 月第 1 次印刷

书 号: ISBN 978 - 7 - 5121 - 1376 - 3/TP · 729

印 数: 1 ~ 4 000 册 定价: 21.00 元

本书如有质量问题, 请向北京交通大学出版社质监组反映。对您的意见和批评, 我们表示欢迎和感谢。

投诉电话: 010 - 51686043, 51686008; 传真: 010 - 62225406; E-mail: press@bjtu. edu. cn。

前 言

本书与《Visual FoxPro 数据库程序设计》配套使用,是按照《全国计算机等级考试二级 Visual FoxPro 数据库程序设计考试大纲》的要求,对“Visual FoxPro 数据库程序设计”课程实习实训部分教学目标的定位。

国家教育部文科计算机基础教学指导委员会提出,新形势下应加强大学计算机公共基础中数据库技术和程序设计方面的知识的学习,使学生掌握数据库程序设计的基本知识和操作技能,培养大学生计算机应用意识。因此,将计算机等级考试“二级”要求的知识点融入到 Visual FoxPro 数据库程序设计课程中,使学生通过对该课程的学习获取计算机等级证书,是目前大学计算机基础教学的一个改革方向。

目前,Visual FoxPro 数据库程序设计课程用书以纯理论的统编教材居多,与课程用书配套并将等级考试机考内容融入实习实训的教材很少,不便于学生进行技能训练,本书正是根据这个需要编写的。

本书采用“课程”与“证书”相结合,避免采用“纯教学而教学”的教学方式。学生通过课程学习,可以具备参加计算机等级考试的能力。同时,本书在编写过程中,注意将课程要求的操作技能与“计算机等级考试所要求的上机操作技能紧密结合,学生通过实训便能掌握相关考试的上机操作技能,顺利通过“机试”部分的考试。

本书主要从三个方面展开对教材中知识点的讨论。第一,重、难点分析解答;第二,上机实习指导;第三,实训案例分析。其中,重、难点分析解答共分9章,每章都包含知识要点、典型试题分析和思考题三部分内容;上机实习指导根据不同章节分为8个部分共31个实习知识点;实训案例分析主要面对操作性较强的章节,分为4个部分共6个实训案例。大量案例的运用,更适合读者根据自己的特点进行实训,通过上机练习可获得较强的 Visual FoxPro 基本操作与实践能力。在“重难点试题分析”中主要侧重于对计算机“二级”知识点的查漏补缺,力争做到在完成课堂教学的同时达到“二级”等级考试的基本要求。

参加本书编写人员全部是来自西安外事学院本科教学第一线的骨干教师,书中融入了他们在数据库程序设计课程教学中宝贵的教学经验,使本书更具可读性。

全书共分9章,第1章由盖玉莲、周蓉编写;第2、3、7章由商娟叶编写;第4、6章由谢勇编写;第5章由邓小盾编写;第8、9章由周小娟编写。全书由盖玉莲、孙姜燕负责规划总体框架、制定编写大纲和统稿,由孙姜燕主审。

在本书的编写过程中,参考了大量的相关书籍和资料,编者在此对这些书籍和资料的作者表示感谢,同时也感谢西安外事学院相关领导及人员所给予的大力支持和帮助。

计算机科学技术发展很快,需要学习的知识内容不断变化,书中难免会有需要改进和更新的地方,诚请各位读者提出宝贵意见和建议。

编 者
2013年1月

目 录

第1章 公共基础知识	1
1.1 知识要点	1
1.2 典型试题分析	7
1.3 强化练习	16
第2章 Visual FoxPro 系统概述	22
2.1 知识要点	22
2.2 典型试题分析	24
2.3 实习	29
第3章 Visual FoxPro 数据库基本操作	37
3.1 知识要点	37
3.2 典型试题分析	41
3.3 实习	49
3.4 实训	58
第4章 查询与视图	63
4.1 知识要点	63
4.2 典型试题分析	65
4.3 实习	70
第5章 关系数据库标准语言 SQL	72
5.1 知识要点	72
5.2 典型试题分析	73
5.3 实习	82
第6章 Visual FoxPro 编程基础	89
6.1 知识要点	89
6.2 典型试题分析	97
6.3 实习	101
第7章 数据表单的设计与应用	109
7.1 知识要点	109
7.2 典型试题分析	111
7.3 实习	116
7.4 实训	122
第8章 菜单的设计与应用	126
8.1 知识要点	126
8.2 典型试题分析	129
8.3 实习	132

8.4 实训	138
第9章 设计报表和标签	140
9.1 知识要点	140
9.2 典型试题分析	142
9.3 实习	145
9.4 实训	157
参考文献	159

第 1 章

公共基础知识

1.1 知 识 要 点

数据库程序设计公共基础知识主要包括数据结构与算法、程序设计基础、软件工程基础和数据库系统概述四个知识点。这四个知识点在全国计算机等级考试二级 Visual FoxPro 数据库程序设计笔试考试中都是必考内容，应试者要全面掌握这方面的内容。

1.1.1 内容提要

本章主要内容可概括以下四个部分。

1. 数据结构与算法

数据结构方面的知识点包括：数据结构的基本概念和常用术语、数据结构的图形表示和数据结构的类型。

算法方面的知识点包括算法的基本概念、基本特征、基本要素、基本设计方法和算法复杂度。

2. 程序设计基础

程序设计基础主要包括程序设计和软件开发方法两个知识点。

与程序设计相关的知识点主要包括：程序设计方法与风格，具体包括源程序文档化、数据说明的方法、语句的结构和输入输出四个知识点。

与软件开发方法相关的知识点包括：结构化程序设计原则及结构特点、面向对象程序设计的基本概念和主要优点。

3. 软件工程基础

软件工程主要研究软件开发技术和软件工程管理，可通过以下五个方面进行论述说明。

- (1) 软件工程基本概念。
- (2) 结构化分析方法。
- (3) 结构化设计方法。
- (4) 软件测试。
- (5) 程序调试。

4. 数据库系统概述

本部分内容主要介绍数据库系统基本概念、关系数据库系统和数据库设计与管理等方面的知识，具体包括：数据库、数据模型和数据库管理系统基本概念；关系数据库基本概念、关系中的数据约束和关系运算。

1.1.2 学习重点

本章的学习重点首先是充分了解数据结构与算法在程序设计中的重要地位,了解程序设计与数据管理之间的关系,了解软件开发、维护和管理等方面的原理、方法、工具与环境。

除此之外,对于二级等级考试的应试者来说,在掌握以上各基本概念的基础上,还要在全面系统阅读教材及相关资料的同时,适当分析解答一些习题来进一步掌握这些概念性的纯记忆知识。

(1) 数据结构是指反映数据元素之间关系的数据元素的集合,数据的逻辑结构反映了数据元素之间逻辑关系的数据结构,它有两个要素:一是数据元素的集合;二是数据元素之间的关系。各数据元素在计算机存储空间中的位置关系与它们的逻辑关系不一定是相同的,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,或称为数据的物理结构。算法是对解题方案的准确而完整的描述,是一组严谨地定义运算顺序的规则,每一个规则都是有效而明确的,并且此顺序将在有限的次数下终止。

(2) 程序设计的方法与风格主要要求简单、清晰和可理解,这里强调“清晰第一,效率第二”。结构化程序设计的原则为自顶向下、逐步细化、模块化和限制使用转移语句。采用结构化程序设计方法,可以得到结构良好、易读、易理解、易维护的程序,程序设计语言也仅用顺序、选择和循环三种基本控制结构就可以表达出各种其他形式结构的程序设计方法。

(3) 软件工程学是用工程学和数学的原理与方法研制和维护计算机软件的有关技术及管理方法的一门工程学科。软件生命周期是指软件产品从提出→实现→使用→维护→停止使用→退役的整个过程,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。软件生命周期可以分为软件定义、软件开发及软件运行维护三个阶段。软件生命周期的主要活动阶段包括:可行性研究与计划制订、需求分析、软件设计、软件实现、软件测试、运行和维护。

1.1.3 学习中的难点分析

作为 Visual FoxPro 数据库程序设计课程,本章的学习难点体现在以下几个方面。

1. 算法与数据结构中的难点分析

算法与数据结构中的难点主要有以下几个方面。

1) 算法和算法的复杂度

算法是按解题要求从当前环境能进行的所有操作中选择合适的操作组成的一组指令序列。但需要强调的是,算法既不等于程序也不等于计算方法,因为在编写程序时要受到诸如语法规则等计算机系统运行环境的限制,还要考虑许多与方法和分析无关的问题,这就使程序的编制不会优于算法设计,但程序可以作为对算法的一种描述。一般在设计算法的初期,并不直接用计算机程序来描述算法,而是用流程图、算法语言,甚至用自然语言来描述算法。对于一个具体问题,如果能够通过一个计算机程序,并在有限存储空间内运行有限长时间而得到正确的结果,那么就可以称这个问题是算法可解的。

算法的复杂度是本章的重点也是难点。

在实际选用算法时,既要考虑算法的正确性,又要考虑执行算法所耗费的时间和存储空间,同时还要考虑算法是否易于理解、易于编码和易于调试等问题。算法的复杂度是衡量算

法优劣的标志,其又体现在时间复杂度和空间复杂度两个方面。

(1) 算法的时间复杂度是指执行算法所需要的计算工作量。例如,当在一个表中进行查询操作时,可以将两个数据之间的比较作为基本运算。

算法所执行的基本运算次数还与问题的规模有关。一般情况下,算法的工作量用算法所执行的基本次数来度量,而算法所执行的基本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。也就是说,随着问题规模 n 的增大,算法执行时间的增长率和 $f(n)$ 的增长率相同,称做算法的渐进时间复杂度,简称时间复杂度。

(2) 算法的空间复杂度一般是指执行这个算法所需要的存储空间,这个存储空间包括:算法程序所占用的空间、初始输入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。

2) 关于数据结构

数据结构是计算机的一门科学,是软件设计的基础。简单地说,数据结构是指相互有关系(关联)的数据元素的集合。

数据结构中各数据元素具有广泛的含义,一般来说,现实世界中客观存在的一切个体都可以称为数据元素。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(联系),这种关系反映了该集合中数据元素所固有的一种结构,通常把这种数据元素之间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。

3) 栈的基本概念

栈又称为堆栈,是一种仅允许在表的一端进行插入和删除运算的线性表。人们将一端称为栈顶,栈顶的第一个元素称为栈顶元素,另一端称为栈底。向一个栈插入新元素(称为进栈或入栈),是把该元素放到栈顶元素的上面,使之成为新的栈顶元素;从一个栈删除元素(称为出栈或退栈),是把栈顶元素删除掉,使其下面的相邻元素成为新的栈顶元素。

4) 队列

队列简称队,也是一种运算受限的线性表,其限制是仅允许在表的一端(称作队尾)进行插入运算,在表的另一端(称作队首)进行删除运算。向队列中插入新元素(称为进队或入队),新元素进队后就成为新的队尾元素;从队列中删除元素(称为离队或出队),元素离队后,其后件元素就成为队首元素。

5) 二叉树及其遍历

二叉树及其遍历相关的概念和操作是本章中非常重要的知识点之一。二叉树是一种非线性结构,它有两个特点:① 非空二叉树只有一个根结点;② 每个结点至多只有两棵子树,并且有左右之分,分别称为该结点的左子树与右子树。

另外,二叉树中的每一个结点的子树被明显地分为左子树与右子树。一个结点可以只有左子树而没有右子树,也可以只有右子树而没有左子树。当一个结点既没有左子树也没有右子树时,则该结点称为叶子结点。

在了解二叉树遍历前,一定要深刻理解和掌握二叉树的四个基本性质。

(1) 在二叉树的第 k 层上,最多有 2^{k-1} ($k \geq 1$) 个结点。

(2) 深度为 m 的二叉树最多有 $2^m - 1$ 个结点。

(3) 在任意一棵二叉树中,度为 0 的结点(叶子结点)总比度为 2 的结点多一个。例如,

某二叉树中度为 2 的结点有 18 个, 则该二叉树中有 19 个叶子结点。

(4) 具有 n 个结点的二叉树, 其深度至少为 $\lfloor \log_2 n \rfloor$, 其中 $\lfloor \log_2 n \rfloor$ 表示 $\log_2 n$ 的整数部分。

遍历是对树的一种最基本的运算, 所谓遍历二叉树, 就是按一定的规则和顺序走遍二叉树的所有结点, 使每个结点都被访问一次, 而且只被访问一次。由于二叉树是非线性结构, 所以树的遍历实质上是将二叉树的各个结点转换成为一个线性序列表示过程。

对一棵二叉树的遍历有前序遍历、后序遍历和中序遍历三种情况, 这三种遍历二叉树的过程都是一个递归的过程。由此可见, 掌握遍历的先左后右原则是关键。

6) 查找与排序

查找一般有两种方法: 顺序查找和二分查找。

(1) 顺序查找又称为顺序搜索, 一般是指在线性表中查找指定的元素, 对于大的线性表来说, 顺序查找的效率是很低的。其基本查找过程如下:

① 从表中最后一个记录开始, 逐个进行记录的关键字和给定值的比较;

② 若某个记录的关键字和给定值比较相等, 则找到所查记录, 查找成功; 若直至第一个记录, 其关键字和给定值比较都不相等, 则表明表中没有所要查找的记录, 查找不成功。

(2) 二分查找是针对有序表进行查找的简单、有效而又较常用的方法, 其基本查找过程如下:

① 首先选取表中间位置的记录, 将其与给定关键字 k 进行比较, 若相等, 则查找成功;

② 若 k 值比该关键字值大, 则要查找的元素一定在表的后半部分 (或称右子表), 继续对右子表进行二分查找;

③ 若 k 值比该关键字值小, 则要查找的元素一定在表的前半部分 (或称左子表), 继续对左子表进行二分查找。

在二分查找中, 每进行一次比较, 要么找到要查找的元素, 要么将查找范围缩小一半, 如此递推, 直到查找成功或把要查找的范围缩小为空, 查找失败。这里要注意一个重要概念: 当有序线性表为顺序存储时才能使用二分查找, 并且二分查找的效率要比顺序查找高得多。

2. 程序设计基础中的难点分析

程序设计基础中的难点主要体现在以下几个方面。

1) 程序设计方法与风格

程序设计主要经历了结构化程序设计和面向对象的程序设计阶段, 要形成良好的程序设计风格, 需要注意和考虑以下几个因素。

(1) 源程序文档化, 包括以下几点。

① 符号名的命名。符号名也称为标识符, 包括模块名、变量名、常量名、标号名、子程序名、数据区名以及缓冲区名等。

② 程序的注释。程序中的注释是程序员与程序读者之间沟通的重要手段, 注释不是可有可无的。注释一般分为序言性注释和功能性注释两种。

③ 程序的视觉组织。为使程序层次更加清晰, 结构一目了然, 可以在程序中使用空格、空行、缩进等技巧。

(2) 数据说明的风格。在设计阶段已经确定了数据结构的组织及其复杂性。在编写程序时, 需要注意数据说明的风格。数据说明的风格一般应注意以下几点。

① 数据说明的次序要规范化。可使数据属性容易查找,也利于测试、排错和维护。

② 说明语句中变量安排有序化。当多个变量名在一个说明语句中时,应该对这些变量按字母顺序排列。

③ 使用注释来说明复杂数据的结构。如果设计一个复杂的数据结构,应当使用注释来说明在程序实现时这个数据结构的固有特点。

(3) 语言的结构。程序编写应简单易懂,语句构造应简单直接。

(4) 输入和输出。输入、输出信息是与用户的使用直接相关的。输入、输出的方式和格式应尽量做到方便用户的使用。一定要避免因设计不当给用户带来麻烦。系统能否被用户所接受,很大程度上取决于输入、输出的风格。

2) 结构化程序设计的主要原则

结构化程序设计的主要原则是自顶向下、逐步求精、模块化、限制使用 goto 语句。

结构化程序设计的基本结构和特点:采用结构化程序设计方法编写程序,可使程序结构良好、易读、易理解、易维护。

3) 面向对象程序设计的基本概念

(1) 对象。面向对象的程序设计方法中,设计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可执行的一组操作组成。

(2) 类和实例。类是具有共同属性、共同方法的对象的集合。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,而一个对象是其对应类的一个实例。

(3) 消息。消息是一个实例与另一个实例之间传递的信息,它请求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。消息的使用类似于函数的调用。

(4) 继承。继承是类之间的一种关系,一个类的上层可以有父类,下层可以有子类,一个类可以直接继承其父类的描述(数据和操作)或特征,一个子类可以自动地共享基本类中定义的数据和方法。

(5) 多态性。对象根据所接受的信息而做出具体动作,同样的消息被不同的对象接受时导致完全不同的行动,该现象称为多态性。

面向对象方法有以下五个优点:与人类习惯的思维方式一致;稳定性好;可重用性好;可维护性好;易于开发大型软件产品。

3. 软件工程基础中的难点分析

软件工程基础中的难点主要体现在以下几个方面。

1) 软件危机与软件工程

所谓软件危机是指在计算机软件开发和维护过程中遇到的一系列严重问题,为了消除软件危机,软件工程应运而生。软件工程的主要思想是强调在软件开发过程中需要应用工程化原则,是应用于计算机软件的定义、开发和维护的一整套方法、工具、文档、实践标准和工序。软件工程的目标是在给定成本、进度的前提下,开发出具有有效性、可靠性、可维护性、可适应性、可移植性和可互操作性,并且满足用户需求的产品。

2) 软件生命周期

软件生命周期通常是指将软件产品从提出、实现、使用维护到停止使用的过程。这个过程一般包括可行性研究、需求分析、软件设计、软件实现、软件测试和交付使用后的运行维

护等活动。

- (1) 可行性研究。
- (2) 需求分析。
- (3) 软件设计。
- (4) 软件实现。
- (5) 软件测试。
- (6) 运行和维护。

3) 数据字典

数据字典是关于数据的信息的集合,同时作为数据流程图的补充工具,对数据流程图中的各个元素做完整的定义与说明。目前,数据字典已成为结构化分析方法的一个重要工具,尽管各工具字典的形式不同,但都包含以下信息。

- (1) 名称:数据或数据项、数据存储或外部实体的主要名称。
- (2) 别名:第一项的其他名字。
- (3) 何处使用、如何使用:使用数据或控制项的加工列表以及如何使用。
- (4) 内容描述:表示内容的符号。

数据字典和数据流图密切配合,能够清楚地表达数据处理的要求。

4) 软件测试的目的与方法

软件测试是为了发现错误而执行程序的过程,其主要过程涵盖了整个软件生命周期的全过程。如果从是否需要执行被测软件的角度划分,软件测试方法和技术可以分为静态测试和动态测试两种;若按照功能划分,可以分为黑盒测试和白盒测试。软件测试的实施过程一般按单元测试、集成测试、确认测试和系统测试四个步骤进行,并通过这些步骤的实施来验证软件是否合格和能否交付用户使用。

5) 程序调试

程序进行了成功的测试之后进入调试阶段,程序调试是为了诊断和改正程序中潜在的错误,主要在开发阶段进行。其活动过程主要由两部分组成:一是根据错误的迹象来确定程序中错误的确切性质、原因和位置;二是对程序进行修改进而排除错误。从是否跟踪和执行程序的角度出发,可以将程序调试的方法分为静态调试和动态调试。静态调试主要是指通过人的思维来分析源程序代码并排除错误,这是主要的调试手段,而动态调试是静态调试的辅助操作。

4. 数据库设计基础中的难点分析

数据库设计基础中的难点主要体现在以下几个方面。

1) 数据库管理系统

通俗地说,数据库管理系统是帮助用户创建和管理数据库的应用程序的集合。因此,数据库管理系统也就是一个可以帮助完成定义、构造和操纵数据库等处理目的的通用软件系统。其主要功能有以下几个。

- (1) 数据模式定义。
- (2) 数据存取的物理构建。
- (3) 数据操纵。
- (4) 数据的完整性、安全性定义与检查。

(5) 数据库的并发控制与故障恢复。

(6) 数据的服务。

2) 数据库系统的发展

数据库系统的发展可以分为以下三个阶段。

(1) 人工管理阶段。人工管理阶段主要用于科学计算，硬件没有磁盘，数据被直接存取，软件没有操作系统。

(2) 文件系统阶段。文件系统阶段具有简单的数据共享和数据管理能力，但无法提供统一的、完整的管理和数据共享能力。

(3) 数据库系统阶段。数据库系统阶段具有数据的集成性、数据的高共享性与低冗余性、数据独立性三个基本特点。

3) 数据模型的类型

数据是现实世界符号的抽象，而数据模型是数据特征的抽象。按不同的应用层次，可以将数据模型分成三种类型。

(1) 概念数据模型。概念数据模型是一种面向客观世界和面向用户的模型，它与具体的数据库管理系统无关，与具体的计算机平台无关。概念模型着重于对客观世界复杂事物的结构描述和它们之间的内在联系的刻画。

(2) 逻辑数据模型。逻辑数据模型是一种面向数据库管理系统的模型，只有在转换成逻辑数据模型后才能在数据库中得以表示。

(3) 物理数据模型。物理数据模型是一种面向计算机物理表示的模型，此模型给出了数据模型在计算机上物理结构的表示。

4) 关系数据模型

在关系数据模型中，把数据看成一个二维表格，每个二维表称为一个关系。表中的每一列称为一个属性，相当于记录中的一个数据项。对属性的命名称为属性名，表中的一行称为一个元组，相当于记录值。

关系模型的数据操纵是建立在关系上的数据操纵，一般有查询、增加、删除和修改四种操作。

1.2 典型试题分析

1.2.1 选择题

1. 下列叙述中正确的是 ()。

- A. 算法的时间复杂度是指执行算法所需要的计算工作量
- B. 算法的效率只与问题的规模有关，而与数据的存储结构无关
- C. 数据的逻辑结构与存储结构是一一对应的
- D. 算法的时间复杂度与空间复杂度一定相关

试题分析：算法的时间复杂度是指算法在计算机内执行时所需时间的度量，与时间复杂度类似，空间复杂度是指算法在计算机内执行时所需存储空间的度量。因此，选项 A 的说法是正确的。

数据结构包括数据的逻辑结构和存储结构两个方面，数据的结构直接影响算法的选择和

效率。因此,数据的逻辑结构和存储结构都影响算法的效率,故选项 B 是错误的。

数据之间的相互关系称为逻辑结构,通常有集合、线性结构、树形结构和网状结构四类基本逻辑结构。存储结构在计算机中有两种,即顺序存储结构和链式存储结构。由此可见,逻辑结构和存储结构不是一一对应的,选项 C 的说法是错误的。

有时人们为了提高算法的时间复杂度而以牺牲空间复杂度为代价,但是这两者之间并没有必然的联系,因此选项 D 的说法是错误的。

试题参考答案: A

2. 算法的空间复杂度是指 ()。

A. 算法程序的长度

B. 算法程序中的指令个数

C. 算法程序所占的存储空间

D. 算法执行过程中所需要的存储空间

试题分析: 算法的空间复杂度一般是指这个算法执行时所需要的内存空间,其中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,其中额外空间还包括算法程序执行过程的工作单位以及某种数据结构所需要的附加存储空间。

试题参考答案: D

3. 下列对于线性链表的描述中正确的是 ()。

A. 存储空间不一定是连续的,而且前件与元素一起存储在后件元素的前面

B. 存储空间必须连续,而且前件元素一定存储在后件元素的前面

C. 存储空间必须连续,而且各元素的存储顺序是任意的

D. 存储空间不一定是连续的,而且各元素的存储顺序是任意的

试题分析: 在链式存储结构中,存储数据的存储空间可以不连续,各数据元素的存储顺序与数据元素之间的逻辑关系可以不一致,数据元素之间的逻辑关系是由指针域来确定的。

试题参考答案: D

4. 下列叙述中正确的是 ()。

A. 线性链表是线性表的链式存储结构

B. 栈与队列是非线性结构

C. 只有根结点的二叉树是线性结构

D. 双向链表是非线性结构

试题分析: 线性表的链式存储结构称为线性链表,为了适应线性表的链式存储结构,计算机存储空间被划分成一个个小块,每一个小块占若干字节,通常称这些小块为存储结点。

试题参考答案: A

5. 按照“后进先出”原则组织数据的数据结构是 ()。

A. 队列

B. 栈

C. 双向链表

D. 二叉树

试题分析: 栈是限定在一端进行插入与删除的线性表,栈顶元素总是最后被插入的元素,同时也是最先能够被删除的元素。栈底元素总是最先被插入的元素,同时也是最后才能被删除的元素。也就是说,栈是按照“先进后出”或“后进先出”的原则组织数据的。通俗地说,对栈可以进行插入和删除操作,但必须牢记插入和删除数据都只在栈顶进行。

试题参考答案: B

6. 下列对队列的叙述正确的是 ()。

A. 队列属于非线性表

B. 队列按“先进后出”的顺序组织数据

C. 队列在队尾删除数据

D. 队列按“先进先出”的顺序组织数据

试题分析: 队列是一种线性表, 按“先进先出”的顺序组织数据, 它在队尾添加数据, 在队头删除数据。

试题参考答案: D

7. 在深度为 7 的满二叉树中, 叶子节点的个数为 ()。

A. 32

B. 31

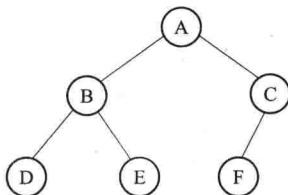
C. 64

D. 63

试题分析: 满二叉树是指除最后一层外, 每一层上的所有结点都有两个子结点的二叉树。满二叉树在其第 i 层上有 2^{i-1} 个结点, 即每一层上的结点数都是最大结点数。对于深度为 7 的满二叉树, 叶子结点所在的是第 7 层, 一共有 $2^{7-1}=64$ 个叶子节点。

试题参考答案: C

8. 对如下二叉树



进行后序遍历的结果为 ()。

A. ABCDEF

B. DBEAF C

C. ABDECF

D. DEBFCA

试题分析: 后序遍历首先遍历左子树, 然后遍历右子树, 最后访问根结点。

试题参考答案: D

9. 在长度为 64 的有序线性表中进行顺序查找, 最坏情况下需要比较 () 次。

A. 63

B. 64

C. 6

D. 7

试题分析: 长度为 64 的有序线性表中的 64 个数据元素是按照从大到小或从小到大的顺序排列的。在这样的线性表中进行顺序查找, 最坏的情况就是查找的数据元素不在线性表中或位于线性表的最后。按照线性表的顺序查找算法, 首先用被查找的数据与线性表的第一个数据元素进行比较, 若相等, 则查找成功, 否则继续与线性表的第二个数据元素进行比较, 直到在线性表中查找到该数据或查找到线性表的最后一个元素算法才结束。

试题参考答案: B

10. 在最坏情况下, 下列排序方法中时间复杂度最小的是 ()。

A. 冒泡排序

B. 快速排序

C. 插入排序

D. 堆排序

试题分析: 在最坏情况下, 冒泡排序、快速排序和插入排序需要的比较次数均为 $n(n-1)/2$, 堆排序需要比较的次数为 $n \log_2 n$, 由此可见, 堆排序的时间复杂度最小。

试题参考答案: D

11. 对建立良好的程序设计风格, 下列叙述中正确的是 ()。

A. 符号名的命名只需要符合语法

B. 要充分考虑程序的执行效率

C. 程序应该简单、清晰、可读性好

D. 对程序的注释可有可无

试题分析: 选项 A 是错误的。因为符号名的命名不仅要符合语法, 而且符号名的命名也应具有一定的实际意义, 才能加强对程序功能的理解。由于程序设计风格强调的是“清晰第

一，效率第二”，因此，选项 B 的说法是错误的。在程序中加入注释可以帮助读者理解程序，所以注释是提高程序可读性的重要手段，不能错误地认为可有可无。因此，选项 D 的说法是错误的。

试题参考答案：C

12. 下列描述中，符合结构化程序设计风格的是（ ）。

- A. 使用顺序、选择和循环三种基本控制结构表示程序的控制逻辑
- B. 模块只有一个入口，可以有多个出口
- C. 注意提高程序的执行效率
- D. 不使用 goto 语句

试题分析：应该选择只有一个入口和一个出口的模块，选项 B 是错误的。程序设计首先要考虑的是程序的正确性，然后才要求提高效率，选项 C 是错误的。要严格控制使用 goto 语句而不是绝对不使用，选项 D 是错误的。

试题参考答案：A

13. （ ）是面向对象程序设计语言不同于其他语言的主要特点。

- A. 继承性
- B. 消息传递
- C. 多态性
- D. 静态联编

试题分析：继承是一个子类直接使用另一个父类的所有属性和方法（权限为私有的属性和方法除外）。它可以减少相似类的重复说明，从而体现出一般性与特殊性的原则，这使面向对象程序设计语言有了良好的重用性，也是其不同于其他语言的最主要特点。

试题参考答案：A

14. 下面不属于软件工程三要素的是（ ）。

- A. 工具
- B. 过程
- C. 方法
- D. 环境

试题分析：软件工程包括的三个要素：方法、工具和过程。方法是完成软件工程项目的技术手段；工具是指支持软件的开发、管理和文档生成的软件；过程是支持软件开发各个环节的控制和管理。环境不属于软件工程的三要素之一。

试题参考答案：D

15. 下列叙述中正确的是（ ）。

- A. 软件工程只是解决软件项目的管理问题
- B. 软件工程主要解决软件产品的管理生产率问题
- C. 软件工程只是解决软件开发中的技术问题
- D. 软件工程的主要思想是强调在软件开发过程中需要应用工程化原则

试题分析：软件工程学研究软件开发和维护的普遍原理与技术的一门工程学科，主要研究对象包括：软件开发与维护技术、方法、工具和管理等。软件工程是指采用工程的概念、原理、技术和方法来指导软件的开发和维护。

试题参考答案：D

16. 在软件生命周期中，能准确地确认软件系统必须做什么和必须具备哪些功能的阶段是（ ）。

- A. 概要设计
- B. 详细设计
- C. 可行性分析
- D. 需求分析

试题分析：选项 A 和选项 B，概要设计和详细设计阶段属于软件设计阶段。在此阶段，系统设计人员和程序设计人员在反复理解软件需求的基础上，给出软件的结构、模块的划分、

功能的分配以及处理流程。选项 C, 可行性分析阶段是确定待开发系统的开发目标个数的要求, 给出它的功能、性能、可靠性以及接口等方面的方案, 制订完成开发任务的实施计划。选项 D, 需求分析阶段是待开发软件提出的需求进行分析并给出详细定义、软件规格说明书和初步的用户手册。所以, 能准确地确定软件系统必须做什么和必须具备哪些功能的阶段应该是需求分析阶段。

试题参考答案: D

17. 在结构化方法中, 用数据流程图 (DFD) 作为描述工具, 是软件开发的 () 阶段。

- A. 可行性分析 B. 需求分析 C. 详细设计 D. 程序编码

试题分析: 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, 结构化分析是使用数据流程图 (DFD)、数据字典 (DD)、结构化语句、判定表和判定树等工具, 来建立结构化规格说明的目标文档。所以, 数据流程图是在需求分析阶段使用的。

试题参考答案: B

18. 为了使模块尽可能独立, 要求 ()。

- A. 模块的内聚程度要尽量高, 且各模块间的耦合程度要尽量弱
B. 模块的内聚程度要尽量高, 且各模块间的耦合程度要尽量强
C. 模块的内聚程度要尽量低, 且各模块间的耦合程度要尽量弱
D. 模块的内聚程度要尽量低, 且各模块间的耦合程度要尽量强

试题分析: 系统设计的质量主要反映在模块的独立性上。评价模块独立性的主要标准有两个内容: 一是模块之间的耦合, 它表明模块之间相互独立的程度; 二是内聚, 它表明模块内部之间的关系是否紧密。一般情况下, 要求模块之间的耦合程度尽可能弱, 或称尽可能独立, 而要求模块的内聚程度尽量高。

试题参考答案: A

19. 下列对于软件测试的描述中正确的是 ()。

- A. 软件测试的目的是证明程序是否正确
B. 软件测试的目的是使程序符合结构化原则
C. 软件测试的目的是使程序运行结果正确
D. 软件测试的目的是尽可能多地发现程序中的错误

试题分析: 软件测试的目的是在精心控制的环境下执行程序, 以发现程序中的错误并给出程序可靠性的鉴定。测试不是为了证明程序是否正确, 而是在设想程序存在错误的前提下进行的, 目的是为了充分暴露程序中的错误和缺陷。

试题参考答案: D

20. 下列描述中正确的是 ()。

- A. 测试工作必须由编程人员自己完成
B. 测试用例和调试用例必须一致
C. 一个程序经过调试改正错误后, 一般不再进行测试
D. 以上三种说法都不对

试题分析: 测试不是为了证明程序是否正确, 而是在设想程序存在错误的前提下进行的, 目的是为了充分暴露程序中的错误和缺陷, 一般应当避免由编程者测试自己的程序, 选项 A 是错误的。测试是为了发现程序中的错误, 推断错误的原因, 进而改正错误, 所以不能证明