

中国计算机软件专业技术资格和水平考试

1997 年度
试题分析与解答

初级程序员级 程序员级 高级程序员级

吴家勋 王 萍 主编

44

1

清华大学出版社

<http://www.tup.tsinghua.edu.cn>



TP31-66
WJX/1

中国计算机软件专业技术资格和水平考试

1997 年度试题分析与解答

(初级程序员级 程序员级 高级程序员级)

吴家勋 王萍 主编

清华大学出版社

(京)新登字 158 号

内 容 简 介

本书是 1997 年度中国计算机软件专业技术资格和水平考试(初级程序员级、程序员级、高级程序员级)试题的题解,书中不仅给出了试题的答案,还分析了解答每道题的思路。读者可以通过阅读本书来熟悉题型,学习解题的方法,提高阅读和分析程序的能力。本书是应试者备考复习必备的辅导教材,可供相关级别的应试者及相应层次的计算机技术人员阅读。

版权所有,翻印必究。

本书封面贴有清华大学出版社激光防伪标签,无标签者不得销售。

图书在版编目(CIP)数据

JSSS / 16

中国计算机软件专业技术资格和水平考试 1997 年度试题分析与解答:初级程序员级、程序员级、高级程序员级/吴家勋,王萍主编. —北京:清华大学出版社,1998
ISBN 7-302-02998-9

I. 中… II. ①吴… ②王… III. 计算机软件-水平考试-学习参考资料 N. TP31

中国版本图书馆 CIP 数据核字(98)第 14381 号

出版者:清华大学出版社(北京清华大学校内,邮编 100084)

<http://www.tup.tsinghua.edu.cn>

印刷者:北京昌平环球印刷厂

发行者:新华书店总店北京科技发行所

开 本:787×1092 1/16 印张:8.25 字数:156 千字

版 次:1998 年 7 月第 1 版 1998 年 7 月第 1 次印刷

书 号:ISBN 7-302-02998-9/TP·1591

印 数:00001~20000

定 价:11.00 元

前 言

准备参加中国计算机软件专业技术资格和水平考试的人员,除了要系统地学习掌握计算机软件、硬件知识,训练程序设计的基本技能外,还应阅读前些年统一考试的试题及解答,从中了解试题的形式、难易程度和考试内容的大致分布。这样可以帮助检查基本知识的掌握情况,有助于提高对程序的设计和阅读分析能力,以及应试能力。

近几年,我们对全国统一考试的试题编写了试题分析与解答,发给参加软件水平考试培训班的学员,作为备考复习的辅导材料,受到学员的欢迎。今年,应清华大学出版社之约,将我们编写的 1997 年初级程序员级、程序员级、高级程序员级的试题分析与解答出版,希望对备考复习人员有所帮助。由于编写时间仓促,难免存在缺点和错误,请读者批评指正。

参加本书编写的有(以姓氏笔划为序):王萍、王爱英、张立昂、李志刚、沈林兴、吴家勋、耿素云。由吴家勋、王萍主编。

编 者

1998 年 4 月

目 录

1997 年度 初级程序员级 上午试题分析与解答	1
1997 年度 初级程序员级 下午试题分析与解答	6
1997 年度 程序员级 上午试题分析与解答	14
1997 年度 程序员级 下午试题分析与解答	23
1997 年度 高级程序员级 上午试题分析与解答	35
1997 年度 高级程序员级 下午试题分析与解答	46
附录 1 1997 年度 初级程序员级 上午试卷	58
附录 2 1997 年度 初级程序员级 下午试卷	65
附录 3 1997 年度 程序员级 上午试卷	80
附录 4 1997 年度 程序员级 下午试卷	90
附录 5 1997 年度 高级程序员级 上午试卷	102
附录 6 1997 年度 高级程序员级 下午试卷	110
附录 7 CASL 汇编语言文本	123

1997 年度 初级程序员级 上午试题分析与解答

试题 1 分析

线性表是最简单、最常用的一种数据结构。它包括 n 个类型相同的数据元素,在这些数据元素之间有着一个接一个的有序关系。在计算机存储器上存放线性表可有两种方式:一种是按它们的有序关系依次连续存放,用存储的邻接来体现其有序关系;另一种方式是独立地存放各个数据元素,再通过链接来体现其有序关系。为了将存储的各个数据元素链接起来,先要知道各个数据元素存储的起始地址。在存储每个数据元素时,还要存储一个指向下一个数据元素的指针,即下一个数据元素的存放地址。线性表的链式存储还需要有一个首指针,指向第一个数据元素。最后一个数据元素的指针需要用一种特殊的符号(如 0)来表示,表明其没有后续元素了。

本题中已知线性表 $L = (23, 17, 47, 05, 31)$, 其中第一个数据元素是 23, 其后继元素是 17, 接着是 47, ...。而根据试题中的存储图示可知,每个数据元素占 4 个字节,前 2 个字节存储数据本身,后 2 个字节存储指针。存储各个数据元素的起始地址如下表:

各数据元素	05 U	17 X	23 V	31 Y	47 Z
存储始址	100	104	108	112	116

由于线性表 L 的第一个数据元素 23 的存储始址是 108, 所以该链表的首指针为 108; 数据元素 23 的后继元素为 17, 而数据元素 17 的存储始址是 104, 所以数据元素 23 的指针部分 V 为 104; 依次类推, 得到链接图示:

→ 23 V → 17 X → 47 Z → 05 U → 31 Y
108 104 116 100 112

因此, 指针 $V = 104$, $X = 116$, $Z = 100$, $U = 112$, $Y = 0$ 。

解答 A: ⑤ B: ⑦ C: ① D: ③ E: ④

试题 2 分析

按照结构程序设计方法, 在高级程序设计语言中, 程序控制一般有三种基本结构: 顺序结构、分支结构(又称选择结构)和循环结构(又称重复结构)。顺序结构中, 程序中的语句逐条顺序地执行; 分支结构中, 程序将根据某种条件选择不同的路径执行(如: IF 条件 THEN 语句 1 ELSE 语句 2); 循环结构中, 程序将根据某种要求重复若干次执行某一段程序(如: FOR $I := 1$ TO 5 DO 语句)。此外, 高级语言还提供两种子程序控制机制: 函数和过程。函数用于按预定算法求值, 函数的调用一般出现在表达式中, 如: $2 * \sin(X) + f(x) - 1$; 而过程的调用代表执行某个预定的处理, 需要用单独的语句, 如: 过程名(参数表)。

解答 A: ⑥ B: ④ C: ② D: ④ E: ②

试题 3 分析

对软盘进行格式化的 DOS 命令为: `FORMAT A: (或 B:) [/S][/4]...`。

可选参数/S 表示在格式化时将 DOS 系统文件复制到相应的软盘中,而且在该软盘上建立引导块,使其成为可引导的(可开机启动的)。可选参数/4 表示按低密度软盘进行格式化。

DOS 命令 `REN *.DOC *.TXT` 表示将当前目录中所有以 DOC 为扩展名的文件改名,使其主文件名不变,扩展名改为 .TXT。命令中的匹配符“*”表示若干个任意字符。例如,执行此命令后,A.DOC 和 BC.DOC 将分别改名为 A.TXT 和 BC.TXT。

注意:匹配符“?”只代表一个任意字符。匹配符的使用实际上是一种简单的模糊技术。

DOS 命令 TYPE 用于查看文件的内容;命令 DIR 则用于查看文件目录。命令 `DIR \*. *` 用于显示根目录下所有的文件名清单,输出的结果显示在屏幕上。DOS 命令中的“>”表示输出重定向,PRN 表示打印机设备,“>PRN”表示将前面的输出重定向到打印机上。因此,命令 `DIR \*. * >PRN` 表示将当前盘中根目录下的所有文件名清单输出到打印机上。

命令 `TYPE ABC.TXT` 的功能是在屏幕上显示文件 ABC.TXT 的内容。命令 `COPY ABC.TXT CON` 的功能是将文件 ABC.TXT 拷贝到设备文件 CON 上。CON 的含义是控制台,作输入设备时代表键盘,作输出设备时代表显示器。因此,命令 `COPY ABC.TXT CON` 就是将文件 ABC.TXT 的内容在屏幕上显示出来。

设当前目录下已有文件 ABC.TXT,则命令 `COPY ABC.TXT \` 表示将文件 ABC.TXT 拷贝到当前盘的根目录中。如果当前目录不是根目录,则该命令将使当前盘上增加一个文件。

解答 A: ① B: ④ C: ③ D: ① E: ②

试题 4 分析

FOXBASE 中,类型函数 TYPE 的格式为: `TYPE (<字符串表达式>)`,其功能为测试由<字符串表达式>表示的数据的类型。函数的参量需用单引号或双引号等括起来。例如:

`TYPE ('09/23/96')` 返回值为 N,表示表达式 09/23/96 的数据类型(/为除号)是数值型。

`TYPE ('"09/23/96"')` 返回值为 C,表示表达式“09/23/96”是字符型。

`TYPE ('DATE()')` 返回值为 D,表示表达式 DATE()属于日期型。

`TYPE ('.T.')` 返回值为 L,表示表达式 .T. 属于逻辑型。

`TYPE (09/23/96)` 返回值为 U,因为函数的格式错误。对于未定义的数据类型也会返回 U。

FIND 和 SEEK 命令都可以根据索引文件来查找,但 FIND 只能对字符型或数值型的关键字实现检索,命令参数只能是常量;而 SEEK 对字符型、数值型、日期型和逻辑型的关键字都能实现检索,命令参数可以是表达式。本题根据日期查找,在 FOXBASE 中不

能直接表示日期型常量,通常由字符串转换函数 CTOD(...)这种表达式来表示。

逻辑变量“婚否”有两种值: .T. 表示已婚; .F. 表示未婚。即逻辑变量“婚否”表示的是已婚情况。表示未婚情况的逻辑表达式应为: .NOT. 婚否。

解答 A: ① B: ② C: ⑥ D: ④ E: ②

试题 5 分析

文字处理系统 WPS 通常编辑 WPS 格式的文件,这种文件常以 WPS 为扩展名。WPS 也能编辑文本文件。一般以 TXT, BAT, BAS 为扩展名的文件都是文本文件,经 WPS 编辑后也能转换成文本文件保存。WPS 作编辑处理时常把原文件保存在以 BAK 为扩展名的文件中,称为备份文件。若要作废上次的编辑处理,可以将该备份文件换名后再由 WPS 进行编辑。WPS 不能直接对以 BAK 为扩展名的文件进行编辑。

WPS 具有较强的字符串查找替换功能,可以在全文中查找替换,也可以在定义的块内进行查找替换,可以忽略(不区分)字母的大小写进行查找,也可以从当前位置往前反向查找给定的字符串,但是,查找和替换的字符串不能超过 80 个字节长度。

Microsoft 公司的 Word 也是文字处理软件,可以读入 WPS 格式的文件进行处理。在安装 Word 6.0/7.0 后可以通过插入一数据库,将 FoxBASE 数据库文件插入正在编辑的文件中进行表格套用。

文字处理时,文字的大小取决于字号的规定。图书中的汉字一般为 5 号字,因此,文字处理软件中规定汉字的字号缺省定义(隐含)为 5 号字,4 号字则更大些。

软件产品中有些文件很大,常用压缩软件 ARJ. EXE 等进行压缩,以便于保存。压缩后的文件常用特定的扩展名来标记(如. ARJ),在安装软件时再将这些文件解压缩展开。

解答 A: ② B: ③ C: ① D: ① E: ③

试题 6 分析

无符号整数(即二进制整数)1111 1111 0000 0000
=1111 1111 1111 1111-1111 1111,其十进制值为
 $(2^{16}-1)-(2^8-1)=2^{16}-2^8$ 。

带符号的反码 1111 1111 0000 0000,其原码为 1000 0000 1111 1111,其十进制值为 $-(2^8-1)$ 。该数为负数,其补码等于反码的末位加 1,为 1111 1111 0000 0001。

已知字母 K 的十六进制奇校验码为 CB,其二进制形式为 1100 1011,最高位 1 是奇校验位,后面七位是 K 的 ASCII 码: 100 1011,所以大写字母 E 的 ASCII 码为 100 0101 (大写字母的 ASCII 码是依次顺序排列的)。它的数位上已有奇数个 1,奇校验位应为 0,将 0 放在最高位上,得奇校验码 0100 0101,其十六进制形式为 45。

解答 A: ② B: ③ C: ⑤ D: ③ E: ②

试题 7 分析

本题可以直接通过逻辑运算求出各个逻辑表达式的值,同时检查供选答案中的等式是否成立。也可根据逻辑代数的运算规则简化逻辑表达式;还可以通过检验真值表是否

相同判断逻辑表达式是否等价；或通过画文氏图(类似于集合运算图)来简化逻辑表达式。

本题中, $A = x + y = 1001 + 0011 = 1011$ 。

$B = \overline{(0110 + 0011)}(1001 + 1100) = \overline{0111} \cdot \overline{1101} = \overline{0101} = 1010$ 。注意供选答案中的等式⑩不成立。

$C = xyz + 0 + xy\bar{z} = xy = 1001 \cdot 0011 = 0001$ 。选⑥

$D = x\bar{y}(z + \bar{z}) + \bar{x} = x\bar{y} + \bar{x} = \bar{y} + \bar{x} = 0110 + 1100 = 1110$ 。选②

$E = (1001 \oplus 1100) \oplus (1001 + 1100) = 0101 \oplus 1101 = 1000$,而在供选答案⑩中, $1001 \cdot 1100 = 1000$,等式成立。

解答 A: ③ B: ⑨ C: ⑥ D: ② E: ⑩

试题 8 分析

Cache 是高速缓冲存储器,是为解决 CPU 速度很快但主存的存取速度较慢影响存取效率的问题而设置的。Cache 比主存存取速度快得多,但由于价格高,其容量比主存小得多。主存中常用的数据放入 Cache 中,CPU 处理数据时先到 Cache 中去取,取不到时再到主存中去取,而且把所用的数据调入 Cache 便于以后使用。Cache 和主存一样,都是随机存取存储器,即可以按地址存取所需的数据;而磁带则是顺序存取存储器,不先取出其前面的数据就难于取出所需的数据。

在各种辅存中,硬盘一般安装在机箱内,不便于脱卸和携带。软盘、光盘和磁带则便于携带,适用于在不同计算机之间的脱机交换数据。

Cache 一般采用静态随机读写半导体存储器件 SRAM,其特点是读写速度快(小于 20ns),在不停电情况下能长时间保存数据;主存目前常用动态半导体存储器件 DRAM,存取时间约为 70ns,一般在 2ms 时间内要对所有的存储单元重写一遍,否则信息会自动消失。

解答 A: ① B: ④ C: ④ D: ⑤ E: ④

试题 9 分析

ISO 提出的 OSI 参考模型共有七层,从低层到高层依次为:1. 物理层,2. 数据链路层,3. 网络层,4. 传输层,5. 会话层,6. 表达层,7. 应用层。通信子网一般在低三层范围内。中继器用于放大传输介质上的传输信号,补偿信号衰减,位于 OSI 的最低层(第一层)。网桥用于局域网之间存储和转发信息,提供数据链路层上的协议转换。路由器用于将局域网与广域网的连接,提供网络层上的协议转换。

解答 A: ⑦ B: ③ C: ③ D: ② E: ①

试题 10 分析

参考译文:计算机将信息存储在软盘、硬盘或随机存取存储器上。在你使用软盘或硬盘之前,应先做格式化。软盘设计得使你需要时很容易把它插入计算机,用完后又很容易取出。一旦将信息保存在软盘上,这些信息就一直会保留在那里,直到被改写或删除,关机

也不会影响这些数据。

主要术语: information 信息	floppy disk 软盘	hard disk 硬盘
design 设计	remove 取出	save 保存
overwrite 改写	erase 删除	turn off 关
remain 保留	store 存储	affect 影响
insert 插入	so that 以至于	once 一旦

解答 A: ⑤ B: ③ C: ③ D: ① E: ①

试题 11 分析

参考译文: 以下是 DOS 命令 XCOPY 的简述:

功能: 拷贝文件(除隐文件和系统文件外)和目录树。

语法: XCOPY 源 [目的地] [/D: date] [/P] [/S] [/E] [/V] [/W]

参数: 源 指定需要拷贝的文件

目的地 指定新文件名或位置

开关: /D: date 拷贝指定日期当天及之后生成的文件

/P 在建立每个目标文件前都给出提示

/S 拷贝非空的目录和子目录

/E 拷贝每个子目录(即使是空的)

/V 对每个新文件都要进行校验

/W 在拷贝之前给出按键提示

主要术语: command 命令	hidden file 隐文件	system file 系统文件
directory tree 目录树	specify 指定	location 位置
prompt 提示	create 创建	subdirectory 子目录
empty 空	even if 即使	verify 校验

解答 A: ④ B: ① C: ③ D: ⑥ E: ③

试题 12 分析

连结 BD, 交 AC 于 O。矩形 ABCD 的两条对角线将矩形 ABCD 分成四个面积相等的三角形, 而且 $AO=BO=CO=DO$ 。等腰三角形 ADO 中已知一个角为 60° , 因此它就是等边三角形。其中的高 DE 又平分三角形 ADO。所以, 三角形 AED 与矩形 ABCD 的面积之比为 1:8。

圆心为 $(-1, 1)$, 圆周点为 $(2, 2)$, 其间的距离即为半径。半径的平方等于 $(2+1)^2 + (2-1)^2 = 9+1=10$, 因此, 该圆的方程为 $(X+1)^2 + (Y-1)^2 = 10$ 。

从 $g(2)=0$ 可得 $4a+2b=0$, 即 $b=-2a$ 。因 $g(x)=x$, 即 $x(ax+b-1)=0$ 有两个相等的实根, 即有重根 $x=0$, 所以 $b-1=0$ 。因此, $b=1, a=-1/2$ 。

D 的四个供选答案中只有 $1+i$ 的平方为 $2i$ 。

首先要注意到 $g(x)$ 的定义域 $[-1, +1]$ 关于原点对称, 这是奇(偶)函数的必要条件。由于 $g(-x)=g(x)$, 所以 $g(x)$ 是偶函数。奇函数的特点是 $g(-x)=-g(x)$ 。

解答 A: ② B: ② C: ③ D: ④ E: ③

1997 年度 初级程序员级 下午试题分析与解答

试题一 分析

本题给出四段 BASIC 程序,要求考生阅读程序并填写程序运行结果。

程序 1.1 主要使用了条件分支和逻辑表达式。初始 $A=1, B=-1$; 因为 B 不等于 $ABS(B)$, 第一个条件语句的条件不满足。程序执行第二条分支语句时 A 值为 2, B 值为 -1 , A 不等于 B , 表达式 $NOT ((A+B)<(A-B))$ 返回逻辑假值, 使得第二个分支语句的条件也不满足; 此时 A 值为 3, B 值为 -1 , 第三个分支语句中的条件 $(A=B+4 \text{ AND } NOT (A<>3))$ 返回真值, 程序无条件转去执行标号 P 处给出的语句, 打印出 $A+ABS(B)$ 的值, 最后程序执行结果为 4。

程序 1.2 分两部分, 第一部分是主程序, 第二部分是用户自定义函数。函数是位于 $FUNCTION$ 和 $END FUNCTION$ 语句之间的代码段, 函数遵循和子程序相同的规则。用户自定义函数在使用前必须定义, 除此之外与 BASIC 系统提供的函数等同。主程序中首先声明了函数 P(X,Y)$, 定义三个全局变量 $A, B, C$$ 并通过键盘或赋值语句分别对 $A, B, C$$ 赋值, 最后输出函数 P(A,B)$ 的值。下面我们分别看两组不同数据输入后产生的结果如何。首先对 A, B 输入 16 和 -3 , $C$$ 是常数, 执行 $PRINT P$(A,B)$ 语句, 此时 A, B 的值分别传递给 X, Y 。因为 $X>Y$, 所以将字符串运算结果值“ $RESULT1=3.5$ ”赋给 $Z$$, 并将 $Z$$ 作为返回值赋给函数 $P$$, 第一次输出为“ $RESULT1=3.5$ ”, 同理, 第二组输入 23.3 和 39.5, 因为 $X<Y$, 将字符串运算结果值“ $RESULT2=6200$ ”赋给 $Z$$, 输出结果为“ $RESULT2=6200$ ”。

程序 1.3 首先定义 16 位字符串 $C$$, 从键盘接受两个整型数 S, D ; 要求整数 S 大于等于 2 且小于等于 16; D 大于等于 0 且小于等于 32767。如果 S, D 输入合法, 进入 $DO WHILE$ 循环体。在循环体中 D 对 S 求模, 或者说数 D 除以 S 的商放回 D 中, 余数放在变量 B 中, 按余数值加 1 的位置从字符串 $C$$ 中取一字符赋给数组 X()$ 相应的单元中, 商为 0 时循环结束。运行程序对变量 S, D 分别输入 16 和 175, 满足程序给定的条件, 进入 $WHILE$ 循环体执行第一次循环, 将 D 值赋给 A , A 值为 175。将 $INT(A/S)$ 赋给 D , D 值为 10, $B=A-D*S=175-10*16=15$, X(16)=MID$(C$, B+1, 1)=MID$(C$, 16, 1)$, X(16) 值为“F”, I 值减 1 变为 15, 第一次循环结束, 返回循环头, 此时 $D=10$, 不等于 0。继续执行下一轮循环, $A=D=10$, $D=INT(10/15)$, D 值为 0, $B=A-D*S$, B 值为 10, X(15)=MID$(C$, 11, 1)$, X(15) 值为 A, I 递减 1 变为 14, 返回循环头判别条件。因为 $D=0$, 循环结束, 程序转向 $LOOP$ 语句的下一条语句执行, 将 X(15)、X$(16) 内容输出为“AF”。$$$

程序 1.4 是一段排序程序, 采用了直接插入排序算法。主要算法含在二重循环体中。假定第一个数有序, 外层循环执行五次, 处理第 2 到第 6 个数, 每次处理将待排序的一个数按其大小插在前面已排序的有序序列中, 直到全部无序数据插入完毕为止。分析程序,

串日期转换成日期型存入 P,再分别将‘东方之珠:香港’、‘庆祝回归’存入 Q、R 中,用取子字符串函数,从 Q 中取出“香港”,从 R 中取出“回归”,再与“日”相连后存入 M,最后用打印语句输出的结果为“07/01/97 是庆祝香港回归日”。

程序 3.2 中有两段程序 P1. PRG 和 P2. PRG, P1 是主程序, P2 是外部子程序。主程序 P1 带参数调用外部子程序 P2, 子程序中当 U 值递增到 21 时返回主程序, 主程序的打印语句输出结果是 94.00。

程序 3.3 主要考察考生对变量应用范围的理解。数组也是内存变量,在程序中引用的符号名有.dbf 库文件名、库文件的字段名和内存变量名等等。内存变量可定义为全局(public)变量或局部变量(private),若没有显式说明,变量为全局可用。全局变量在各程序中可以共享,每个程序均可修改它所含的内容,它总是保存最近一次修改的值,局部变量只在所定义的程序段中起作用,出了这段程序,局部变量也就不存在了。局部变量可以与全局变量重名。主程序 G1. PRG 中,定义了全局变量一维数组 X、简单变量 A、B、C,并分别对其赋值后调用子程序 G2. PRG。在 G2 中定义了局部变量 B,并赋值为 476.22。要注意此时的 B 与 G1 中定义的全局变量 B 不同;向数组 X 赋值‘AA’相当于对 X 中的每个单元赋值‘AA’。因此,程序执行后? A+C 语句输出“保卫祖国”;? X(1)+X(2), B+100.1 语句输出‘AAAA 246.87’。

程序 3.4 使用了用户自定义函数。主程序 P. PRG 分别接收三个值送入 A、B、C 中,然后调用外部函数 CAL. PRG,将返回结果计算处理后打印输出。运行程序分别输入 3, 4, 5 三个数,函数返回 $\text{SQRT}(6 * (6-3) * (6-4) * (6-5)) = 6$,主程序变量 G 中值为 600,输出语句中的数转换串函数 STR(G, 5, 1) 将 600 转换成字符串‘600.0’,输出结果为‘程序输出: 600.0’。

- 解答** (1) 07/01/97 是庆祝香港回归日 (2) 94.00
(3) 保卫祖国 (4) AAAA 246.87
(5) 程序输出: 600.0

试题四 分析

该程序采用冒泡方法对一组数降序排序。

首先定义三个一维数组 N(10)、M(10)、L(10),分别存放编号、得分、名次。从程序结构看, S1 标号前的语句是数组准备部分, 初始的 FOR...NEXT 语句将 10 对数从 DATA 语句依次读入数组。从标号 S1 到标号 S2 是冒泡排序算法。从打印语句开始到程序结束语句是输出结果部分。

标号 S1 到标号 S2 的算法: 由题意及程序内容可知, 空(1)是 FOR 循环控制语句, 循环控制变量是 J, 循环的终值是 10, 空(1)可填“J=2 TO I”或“J=2 TO 10”。序列中的数两两比较, 如果有序, 进入下一轮循环, 否则交换位置, 置交换状态标志后进入下一轮循环。因此空(2)应填“S2”。空(3)是判别“冒泡”是否结束的语句, 若满足条件 FLAG=0 或 I<2 则表明排序结束, 因此空(3)可填“I>1 AND FLAG=1”。具有同一得分的运动员其名次相同, 在打印输出时要做相应的处理, 空(4)可填“L(I)=L(I-1)”; 空(5)可填“L(I)=I”。

- 解答** (1) J=2 TO I (2) S2
 (3) I>1 AND FLAG=1 (4) L(I)=L(I-1)
 (5) L(I)=I

试题五 分析

对已知公式和算法的求解一般使用自定义函数及函数调用的方法。根据用户自定义函数先定义后使用的原则,空(1)应填函数说明语句;根据程序打印语句所引用的表达式‘a-b’,可以推测:空(2)和空(3)所在的行应是对应于 Machin 公式的函数调用赋值语句“a=16.0 * arctan(1/5.0)”和“b=4.0 * arctan(1/239.0)”。为满足算法要求,函数及变量均采用双精度实数。函数中用 while 循环实现满足条件的级数计算,空(4)所在的赋值语句赋值号右边是由三元运算符“?:”构成的条件表达式,r 是所求级数,f 是级数中的通项,当级数中通项系数为 1/3,1/7,...,1/(4n-1)时,符号为负,否则符号为正,因此空(4)应填“r+f:r-f”;空(5)应填“r”。

- 解答** (1) double arctan() (2) 16.0 * arctan(1/5.0)
 (3) 4.0 * arctan(1/239.0) (4) r+f:r-f
 (5) r

试题六 分析

仔细阅读程序说明和程序中相关语句,可以看出:这段程序主体由一个 WHILE 循环体构成,程序按题目要求对库文件 GZ.DBF 从头至尾扫描一遍并做相应处理。题目要求功能(1):显示 1970 年 10 月 1 日后出生并具有工程师职称的正式工的姓名、年龄及工资,因此我们可以推测空(1)应填“ZC='工程师'.AND. CSRQ>=CTOD('10/01/70').AND. ZSGF”,空(2)填 YEAR(DATE())-YEAR(CSRQ)。根据题目要求功能(2):空(3)应填“.AND. .NOT. ZSGF”;空(4)应填“REPLACE GZ WITH GZ * 1.3”。空(5)实现程序功能(3),应填“AVERAGE GZ TO PG FOR ZSGF”。

- 解答** (1) ZC='工程师'.AND. CSRQ>=CTOD('10/01/70').AND. ZSGF
 (2) YEAR(DATE())-YEAR(CSRQ)
 (3) .AND. .NOT. ZSGF
 (4) REPLACE GZ WITH GZ * 1.3
 (5) AVERAGE GZ TO PG FOR ZSGF

试题七 分析

程序使用自定义函数 TSIN (X) 计算试题中 SIN X 的近似值。这段程序分两部分,从 REM 语句到 END 语句是主程序部分;其余部分是用户自定义函数 TSIN (X)。主程序中变量 R 和 S 用来接收用户从键盘输入的两个数值,程序依据输入值按给定的公式调用函数计算并输出结果。由题意,当 $R^2 \leq S^2$ 时计算 $K = \sqrt{\text{SIN}^2(R) + \text{SIN}^2(S)}$; 当 $R^2 > S^2$ 时计算 $K = \frac{1}{2} \text{SIN}(R * S)$ 。由此我们可以知道空(1)应填写语句 $K = \text{SQR}(\text{TSIN}(R)^2 +$

TSIN(S)^2)。在自定义函数 TSIN(X) 中, 变量 P 存放值为 0.000001, 作为循环结束的控制条件用来控制计算的精度。G 存放展开式的和, T 用来存放展开式中的某一项。DO UNTIL 后的空(2)应填入展开式累加计算结束的控制条件, 由题意知道当 T 中所含值小于 0.000001 时控制累加结束, 因此空(2)应填入 $ABS(T) < P$ 。产生展开式中某项以及展开式求和的过程是在循环体内实现的。观察 DO...LOOP 之间循环体的三条语句, 可知第一条赋值语句产生展开式的和、第二条赋值语句为控制变量增值以便产生展开式的第 N 项; 第三条语句是循环体的关键部分, 它产生展开式中的第 N 项。LOOP 语句与 END FUNCTION 语句之间的空(5)是在循环结束之后、程序控制从函数返回主程序之前执行的语句, 它必定要完成将计算结果赋值给函数的功能, 空(5)填 'TSIN=G'。

解答 (1) $K = \text{SQR}(\text{TSIN}(R)^2 + \text{TSIN}(S)^2)$

或 $K = \text{SQR}(\text{TSIN}(R) * \text{TSIN}(R) + \text{TSIN}(S) * \text{TSIN}(S))$

(2) $ABS(T) < P$

(3) $G + T$

(4) $-T * X * X / (2 * N - 1) / (2 * N - 2)$

(5) $\text{TSIN} = G$

试题八 分析

这段 C 程序用来找出整数 11~999 之间的数 m , 它满足 m, m^2, m^3 均为回文数, 所谓回文数就是指其各位数字左右对称的整数、或者说该整数的逆序数与自身相同。判别某数是否为回文数的算法: 对给定整数, 产生它的逆序数, 如果与自身相同, 这个整数就是回文数。我们首先分析程序, 在主程序中整数 m 遍历 11 到 999, 每次均调用函数 `symm` 判别 m 是否回文数并将回文数打印出来。由题意, 空(1)应为 "`symm(m * m) && symm(m * m * m)`"。函数 `symm()` 用来判别 m 与其逆序数是否相同并返回比较结果。在函数中, 首先将遍历的某个整数值 n 赋给 i , 函数局部变量 m 初值为零, m 存放生成的逆序数。空(2)填写 " $i = n$ ", 空(3)填写 " $m = 0$ ", 进入循环, 当 $i < 1$ 时循环结束并返回 n 与 m 比较的结果。循环体中的两条语句用来产生该整数的逆序数; 空(4)可填写 " $i = i / 10$ "、空(5)填写 " $m = n$ "。

解答 (1) `symm(m * m) && symm(m * m * m)` (2) $i = n$

(3) $m = 0$; (2)和(3)可以交换

(4) $i /= 10$, 或 $i = i / 10$,

(5) $m = n$,

试题九 分析

这段 FoxBASE 程序主要考察考生“宏代换、别名及多分区操作”的几个基本概念。程序在两个分区中打开了两个 dbf 文件。本题的重点在于首先要能正确判别两个 dbf 文件各自使用了哪个分区, 这个问题解决了, 其他问题便迎刃而解。我们首先阅读整个程序, 发现在第 2 分区中打开的库文件别名为 F2, 第 1 分区打开的库文件别名为 F1, 又由 ACCEPT 和 INPUT 语句知道, 变量 F1、F2 分别存放着人员库和成绩库的库文件名, 变量 ZH 中存放着起始的证书号。DO 循环体完成本段程序的主要处理功能。我们知道, 对于理

论和操作成绩之和大于等于 160 分的合格考生,应将成绩库的成绩填入到人员库相应的字段中、并赋给一个证书号。在循环体中替换字段时,人员库所在的分区应是主分区 F1,因此空(1)和空(2)应分别填写“&F2”和“&F1”,空(3)应填写判别是否替换的条件“F2->LL+F2->CZ>=160”,空(4)填入“WITH F2->LL,CZ WITH F2->CZ”;空(5)填写“STR(ZH,6)”。此处应注意引用非主分区上的字段名时,应加上分区别名前缀,以示区别。

- 解答**
- (1) &F2
 - (2) &F1
 - (3) F2->LL+F2->CZ>=160 (其中 F2 可用 B 代替)
 - (4) WITH F2->LL,CZ WITH F2->CZ (其中 F2 可用 B 代替)
 - (5) STR(ZH,6)

试题十 分析

这是一段依据一定的算法对压缩数据实现复原的程序。数组 PACKED 存放已压缩的数据、数组 UNPACK 存放复原后的原始数据,我们不妨假设由压缩规则(1)和(2)而压缩存放的一组数据为“数据包”。压缩数据的总个数由键盘输入、存放在变量 N1 中并由此定义数组 PACKED(N1)。程序执行后,首先通过 FOR...NEXT 循环从键盘接收压缩数据并存入数组 PACKED 中。变量 I 作为 PACKED 数组的下标变量,它总是指示下一个将被处理的“数据包”在数组中的起始位置。变量 J 总指向下一个将被复原的“数据包”在复原数组中的起始位置减 1。程序中 DO...LOOP 循环体实现了复原功能,在进入 DO 循环之前变量 I 值为 1, J 值为 0;意味着初始要复原的“数据包”是压缩数组 PACKED 中的第一个单元,要在 UNPACK 数组的第一个单元开始复原。由压缩规则(3),我们知道数据 0 是压缩数据的结束标志,所以 DO 循环条件是 PACKED(I)<>0。在循环体内首先将“数据包”的首字符赋值给变量 L,然后用 IF 条件语句判别“数据包”的类型,如果 L>0 则表明此“数据包”是 L 个相同的数据,因此,可以将 PACKED(I+1)的内容分别赋给 UNPACK(J+1),...,UNPACK(J+L)的 L 个单元中,空(1)可以填入“PACKED(I+1)”,因为复原了 L 个数据,要修改变量 J 的值,因此需要执行 J=J+L 操作,由于相同数据构成的压缩“数据包”长度均为 2,所以指示下一个要被处理的压缩“数据包”起始位置的变量 I 也要做相应的修改,即执行 I=I+2 操作。反之,如果 L<0,则表明要处理的压缩“数据包”长度是 -L+1,有 -L 个相邻不同的数据。类似 L>0 的处理方法,FOR 循环条件可以是“FOR K=1 TO -L”(L<0),也可以用“FOR K=1 TO ABS(L)”。空(3)与空(1)有所不同。空(1)仅取 PACKED(I+1)单元上的一个数分别赋值给 UNPACK 的相应单元中,空(3)则不然,这里需要将 PACKED(I+1),...,PACKED(I-L)这 -L(L<0)个单元分别复原到 UNPACK(J+1),...,UNPACK(J-L)(L<0)中去,同样在这里也需要修改变量 I 和 J 的值。空(3)可填写“PACKED(I+K)”;空(4)处为“J-L”或“J+ABS(L)”;空(5)处语句为“I=I-L+1”或“I=I+ABS(L)+1”。

- 解答**
- (1) PACKED(I+1)
 - (2) 1 TO -L 或 1 TO ABS(L)
 - (3) PACKED(I+K)
 - (4) J-L 或 J+ABS(L)

(5) $I-L+1$ 或 $I+ABS(L)+1$

试题十一 分析

这段程序要求实现的功能是将给定字符串的前导和后置空白字符删除、并将字符串中间所包含的连续多个空白字符删减成为一个空白字符。本程序主要考察考生指针及字符串处理和 for 循环语句的基本概念。程序包括主程序和函数 sdel() 两部分。主要处理功能由 sdel 函数实现,主程序中首先定义要处理的字符串数组,在打印语句中调用函数 sdel 并将字符串处理的返回结果打印输出。在函数中,首先定义字符串指针变量 p 和 q 并将主程序传来的字符串数组地址赋给他们。空(1)处 for 循环语句用来删除字符串的前导空白字符。for 循环语句的格式为 for([初值表达式];[条件表达式];[循环表达式]),执行步骤为:第一步计算初值表达式,在执行循环体之前,他仅被计值一次。第二步条件表达式被计值,如果为真则执行循环体一次,否则退出 for 循环,若省去条件表达式则条件永远为真进入死循环。第三步循环表达式计值进入下一轮循环。我们来看空(1)所在的 for 语句。没有循环体,初值表达式省略了,只有指针增值运算,根据题意空(1)填入条件表达式 *s== ' ' 后就可完成删去字符串的前导空白字符功能。接着观察下一个 for 循环语句。这条语句中只有条件表达式,其他两部分都省略了,也就是说当字符串指针为空时循环停止。阅读循环体内的语句并由题意可知,初次进入循环,将第一个非空字符赋给新串指针 q 所指内容后指针增值,如果当前字符非空白符,原串指针指向下一字符地址后返回循环头;否则进入 while 循环,原串指针下移直至遇到非空字符为止。因此空(2)填 s++,空(3)填 *s== ' '。空(4)设定字符串结束符,可填语句 *(q-1)='\0';空(5)是函数返回语句,我们知道指针 p 保存着新串的起始地址,此处可填 p。

解答 (1) *s== ' ' (2) s++ (3) *s== ' '
(4) *(q-1)='\0' (5) p

试题十二 分析

首先根据题意阅读程序,这段程序分两部分;第一部分从开始到 ENDDO 语句,实现本程序功能(1);合法性检验;从 SELECT 6 到结束是第二部分程序,实现程序功能(2)、(3)。我们先看第一部分程序中的三个填空,根据程序功能(1)要求,在输入密码时内容不能在屏幕上显示,如果输错密码允许重新输入,若密码输入超过 3 次将强制退出程序。在 FoxBASE 中,系统设置命令“SET CONSOLE ON/OFF”是用来控制输入/输出内容是否在屏幕上显示的命令。因此可在空(1)填入“SET CONSOLE OFF”,同时还应在空(2)填入“SET CONSOLE ON”,如果不加这条命令,此条语句之后的某些显示内容我们也就看不到了。密码输入合法校验及输入次数是离开 DO 循环的控制条件。如果是合法用户(输入“SKY7381”值正确无误时)则退出循环转入第二部分程序执行,否则如果次数<3 仍允许重新输入,如果密码输入次数为 3 则强制退出程序;空(3)可填入“EXIT”。在 FoxBASE 的多分区操作中,各分区均有缺省别名;如分区 1,2,3,4,5,6,...,10 分别对应别名 A,B,C,D,E,F,...,J。在分区上打开的库名同时也是分区的缺省别名。例如在程序中,在分区 6 中打开调价库文件 DJD,分区 6 的别名可以是 F;也可以是 DJD;同样,分区