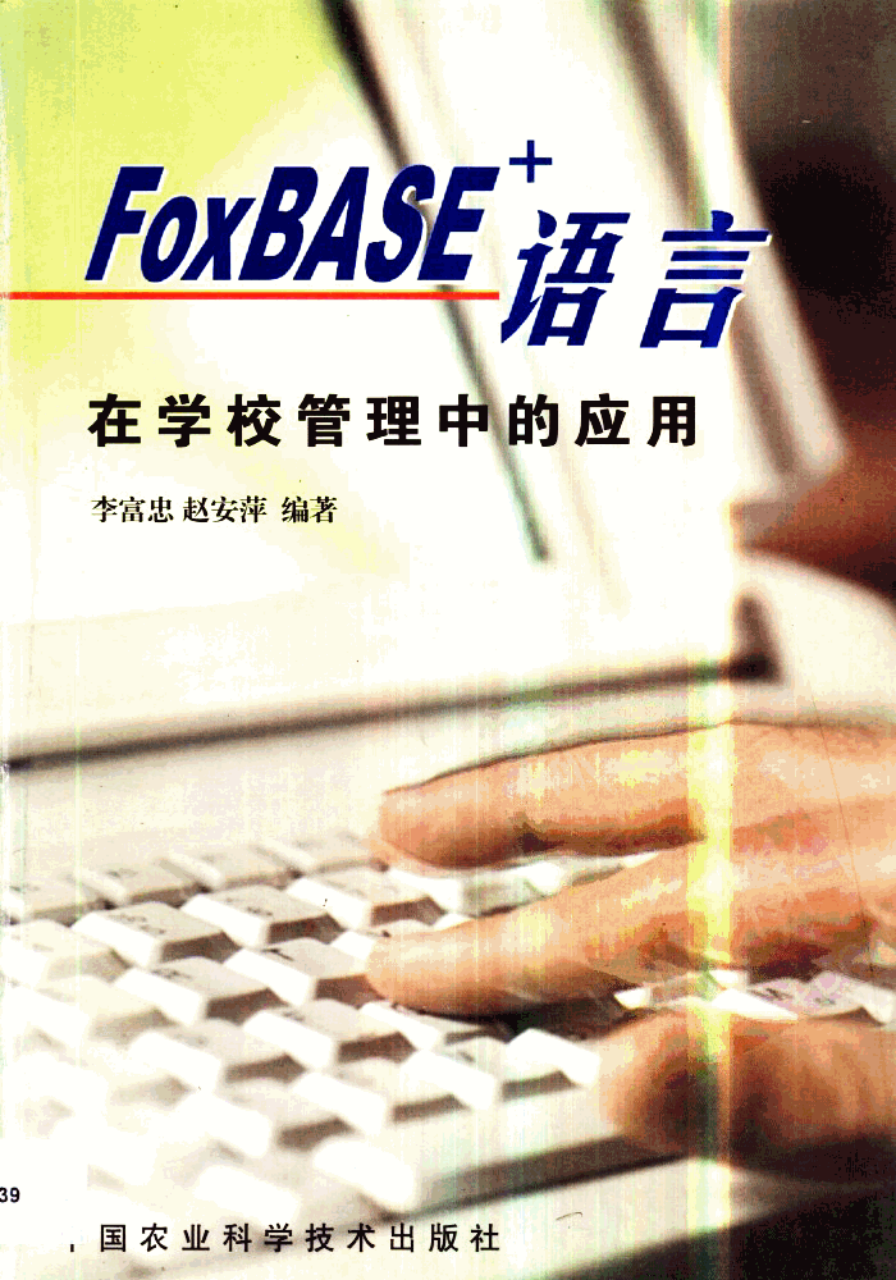




FoxBASE⁺ 语言

在学校管理中的应用

李富忠 赵安萍 编著

前 言

《FoxBASE*语言在学校管理中的应用》一书，使用通俗简炼的程序语句，对高校学生管理、图书借阅管理、医院管理、教学管理、宿舍管理等日常办公重复操作现象较多的领域，根据高校工作的特点，进行了自动化处理，既可以提高工作效率，又可以把工作人员从繁琐的重复作业中解放出来，用更多的时间去设计和实践学校发展的规划。

在程序设计过程中力求通俗化、傻瓜化，以满足不同层次工作人员的要求。

本书由山西农业大学的李富忠同志、赵安萍同志合作完成，其中李富忠同志完成了第一章、第二章、第三章、第四章、第五章、第八章、第十一章的设计与编写任务，赵安萍同志完成了第六章、第七章、第九章、第十章及附录的设计与编写任务。

在本书编写、修订和程序调试过程中，得到了山西农业大学有关部门和中国农业出版社的大力支持，在这里谨向他们表示衷心的感谢！

由于作者水平有限，书中难免有不足之处，欢迎各位同仁批评指正。

目 录

第一章 FoxBASE+ 基础知识	(1)
第一节 基础知识	(1)
第二节 常量、变量与表达式	(7)
第三节 函 数	(13)
第二章 数据库文件的建立与维护	(30)
第一节 数据库文件的建立	(30)
第二节 记录的显示和打印	(35)
第四节 库记录的追加和内容的修改	(40)
第五节 记录的删除与恢复	(47)
第六节 库结构和结构库文件	(50)
第七节 数据库文件的复制及磁盘文件的操作	(53)
第三章 数据库的排序、索引与检索	(56)
第一节 数据库记录的排序	(56)
第二节 数据库记录的索引	(58)
第三节 数据库记录的检索	(63)
第四章 数据库记录的统计	(68)
第一节 记录数的统计	(68)
第二节 数值型字段求和和求平均数	(69)
第三节 分类汇总	(71)
第五章 多工作区操作	(73)
第一节 工作区的概念	(73)
第二节 选择当前数据库工作区命令 SELECT	(74)
第三节 非当前工作字段的引用	(75)
第四节 数据库文件之间的关联命令 SET RELATION TO	(76)
第五节 数据库文件之间的修改命令 UPDATE	(78)
第六节 数据为文件之间的连接命令 JOIN	(79)
第六章 数据库辅助操作命令	(81)
第一节 SET 命令	(81)
第二节 常用辅助命令	(100)
第七章 程序结构	(107)
第一节 命令文件的建立与运行	(108)
第二节 交互式命令	(110)
第三节 顺序结构	(112)
第四节 分支结构	(113)

第五节 循环结构	(116)
第六节 菜单程序设计	(121)
第八章 内存变量和数组	(125)
第一节 内存变量	(125)
第二节 数 组	(131)
第九章 过程文件和自定义函数	(136)
第一节 子程序	(136)
第二节 过程文件	(137)
第三节 自定义函数	(140)
第十章 应用实例	(142)
第十一章 全国计算机等级考试试题题解	(208)
附录 1 FoxBASE* 2.1 命令一览表	(258)
附录 2 FoxBASE* 2.1 函数一览表	(266)
附录 3 标准 ASCII 码表	(272)

第一章 FoxBASE+ 基础知识

FoxBASE 是美国的 Fox Software 公司于 1985 年推出的微型计算机数据库管理系统。它是用 C 语言编写的，共有 1.0、2.0、2.1 三个版本。目前我国使用最多的是 FoxBASE+2.1。本章将介绍数据库及关系型数据库 FoxBASE 的基础知识。

第一节 基础知识

一、数据库

1. 数据库概念和模型

数据库 (DATA BASE) 是指按一定方式组织起来的、存储在计算机存储设备上的、独立于应用程序并能为用户共享的结构化的数据的集合。它不仅要管理数据的具体值，而且要反映数据之间的联系。

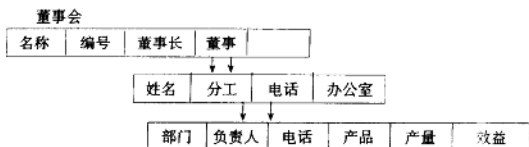
在这个定义中，我们首次使用了结构化和数据两个新概念。那么什么是数据呢？所谓数据就是存储在计算机存储设备上的计算机能够处理的各种物理符号。随着计算机的发展，多媒体技术的广泛应用，我们日常生活中所提到的数字、字符、声音、图像、图形、动画等计算机都能够处理，都是数据。这些数据在计算机中用计算机能够识别的特定的物理符号表示。人们在使用中须根据实际情况对数据进行收集、存储（记载）、分类、计算（加工）、排序、检索、制表、传播等，以提炼出自己需要的数据，实现自己决策及管理的目的。这个过程通常称为数据的处理过程，即数据处理。被提炼出的有用的数据称为信息。所以数据处理也称为信息处理。信息就是加工后的数据。即：

$$\text{信息} = \text{数据} + \text{处理}$$

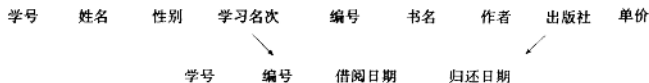
所谓结构化是指按特定的方式对信息（数据）进行组织，以反映数据本身及数据之间的联系。我们在实际应用中通常用数据模型来直观地反映数据及数据间的联系。我们经常谈到的数据库模型一般分为三大类，即层次型、网状型和关系型。

层次型就是利用树状模型来反映数据及数据之间联系的模型，通常一个企业集团的组织机构设置就是一个简单的层次状模型。如下图：

从图中可看出层次状模型只反映一层对一层或一层对多层之间的联系，即每一层的父结点（层）只能是一个，实际上直接联系只有两个点，即：本结点（层）和父结点（层），不能直接反映多对多的联系。同时，由于多个层次组成一个模型（系统），数据扩充较困难。



网状模型实质上是带结点的网络来反映数据及数据之间联系的模型，它突破了层次模型父结点只有一个的限制，能够实现多对多的联系。如下图：



从图中可以看出，网状模型允许结点有多个父结点，在反映数据之间的联系时，比层次模型更灵活。当然，在网状模型中，我们也可以看出，数据的结构更加复杂。使用起来不方便，由于多个结点组成一个模型（系统），数据扩充较困难。

关系数据模型就是用二维表的结构来反映数据及数据间的联系的模型。一个二维表就是一个关系。这种模型以关系数学理论为基础，操作的对象和结果都是一个二维表（关系），简单方便，数据在表中很容易扩充。它和层次模型及网状模型的本质区别就是数据描述的一致性。如下图

山西农业大学学生基本情况表（关系）

学 号	姓 名	性 别	英 语	语 文	婚 否	奖 金	年 龄	籍 贯	专 业	简 历
97012	张建武	男	64.5	65	T	100	20	山东	林学	Memo
98013	郑小红	女	90	87	F	180	19	山西	农学	Memo
96113	张春琳	女	56	71	T	60	21	广东	食品	Memo
97114	王伟大	男	43	25	T	0	20	广东	动科	Memo
98052	李 晨	男	90.5	80	F	180	19	广西	动医	Memo
99201	王博士	女	80	85	F	180	18	山西	林学	Memo
96313	李大雨	男	71	70	T	160	21	河南	食品	Memo

从表中我们可以看出，关系模型简单，数据及数据之间的联系一目了然。一个关系就是一个二维表，表名就是关系名，在这个二维表（关系）中，表的列被称为关系的属性，能够说明这个关系中某一方面的性质和结构，例如性别这一列中，我们能够看出某人的性别及所有人男女比例等，这一列的最上边，即性别称为属性名，具体的“男”、“女”则被称为属性的值。根据某个条件（属性的值）所确定的范围为关系的域。表的行由若干个属性的具体值组成的，称为元组；在表中某一个人的全部情况，能够通过某个属性的值或若干属性的值唯一确定，这时我们把能够确定一个具体元组的属性或属性的组合称为关键字。如上图的学号即是一个关键字，它的值可以唯一确定一个具体的元组；籍贯为山东同时专业是林学也可以确定一个具体的元组，这时籍贯和专业的组合称为一个关键字。从这个关系中，我们还可以看出关系的特征：元组的次序和属性的次序都不会影响数据及数据间的联系，属性是一个变量，属性值是常量，同一个属性的值的

数据类型必须一致；不能出现完全相同的属性名和完全相同的元组，每个属性是关系的最小单元，不能再分解，即一个关系中不能包含另一个关系（表中不能再有表）。FoxBASE* 中的数据库文件就是一个关系，就是一个二维表。

现在计算机能管理的除以上三种模型外，还有一种面向对象型，它是计算机数据库管理技术和面向对象编程技术结合在一起产生的一种新型数据库模型。在 FoxBASE* 语言中暂不涉及这种模型。

2. 数据库管理系统 DBMS 和数据库系统 DBS

数据库管理系统 DBMS (Data Base Management System) 是管理数据库的各类软件的总称。其功能是根据用户的需要，对数据库进行操作和访问，以实现数据资源共享，保障数据安全，维护和管理数据库的目标。FoxBASE 就是专门定义、操作、控制和管理关系型数据的数据库管理系统。

数据库系统 DBS (Data Base System) 是指引进数据库技术后的计算机系统。其功能主要是采用相关的数据模型，实现数据资源共享，减少数据冗余；其次是统一控制数据，实现数据较高的独立性。它由计算机（含硬件系统和软件系统）、数据库集合、数据库管理系统和用户（含数据库管理员、程序员和普通用户）四部分组成。其中数据库管理系统是数据库系统的核心。

由程序员利用数据库系统资源开发面向某一领域或某一方面的应用软件系统通常被称为数据库应用系统。它也是数据库系统的一个部分。

二、关系型数据库 FoxBASE

关系型数据库 FoxBASE 实质上就是一个二维表，表的列（属性）在 FoxBASE 中被称为字段，属性名就是字段名，属性值就是字段值；表的行（元组）被称为记录；表名就是 FoxBASE 数据库文件名。其中 FoxBASE* 是 1986 年 6 月被 Fox Software 公司推出，FOXBASE* 2.1 于 1988 年 7 月被推出，其运行速度快，能够根据用户需要准确迅速地进行查找、检索、统计、增删、插入、分类、汇总、排序、打印、编程等操作和管理。

1. FoxBASE 的组成及运行环境

单用户的 FoxBASE* 数据库管理系统一般由六个文件组成，它们分别是：

系统执行文件	FOXPLUS.EXE
系统覆盖文件	FOXPLUS.OVL
文字原始数据文件	FOXPLUS.RSC
过程组合文件	FOXBIND.EXE
伪编译文件	FOXPCOMP.EXE
帮助文件	FOXPHelp.HLP

其中 FOXPLUS.EXE 和 FOXPLUS.OVL 是 FoxBASE 的两个最基本、最核心的文件，缺其中之一，FoxBASE 就不能启动。有了这两个文件，就可以进行 FoxBASE 的常规操作。

多用户的 FoxBASE+ 数据库管理系统可以在网络环境下运行，它的两个核心文件是：

系统执行文件 MFOXPLUS.EXE

系统覆盖文件 MFOXPLUS.OVL

现在人们常用的计算机都可以运行 FoxBASE+。FoxBASE+ 在个人电脑上的最低硬件要求是 360 KB 的内存、一个硬盘驱动器和一个软盘驱动器。如果运行标准版的 FoxBASE+ 内存要求在 512 KB。汉化的 FoxBASE+ 要求内存还要大，一般要求内存存在 1 兆以上。单用户 FoxBASE+ 的最低软件要求是操作系统为 MS-DOS2.0 以上版本，多用户 FoxBASE+ 要求操作系统为 MS-DOS3.1 以上版本。运行汉字 FoxBASE+ 还必须有汉字操作系统支持，目前流行的 CCDOS、UCDOS、XSDOS 都支持 FoxBASE+。

汉字 FoxBASE+ 在运行时，要求根目录的 CONFIG.SYS 文件至少能够打开 40 个文件。

2. FoxBASE 的性能及语法规则

FoxBASE+ 主要性能指标如下：

每个库文件最大记录数	10 亿
每个记录的最多字符数	4000
每个记录的最多字段数	128
每个字段的最多字符数	254
每个字符型字段的宽度	< = 254
每个数值型字段的宽度	< = 19
每个日期型字段的宽度	= 8
每个逻辑型字段的宽度	= 1
每个备注型字段的宽度	= 10
每个备注文件的总内容	< = 64KB
一个库文件对应的备注文件数	= 1
可同时可打开的库文件个数	< = 10
可同时可打开的索引文件数	< = 7
同时可打开的所有文件数	< = 48
一个字符串最多字符数	= 254
一个命令行最多字符数	= 254
索引关键字最多字符数	= 100
内存变量的最多个数	= 3600
系统默认的内存变量数	= 256
数组的最多个数	= 3600
每个数组中最多数组元素个数	= 3600
数值计算中有效数字位数	< = 16

FoxBASE+ 的主要语法规则如下：

文件的命名规则遵循 DOS 文件的命名规则；

每条命令以动词开头，后面各短语的位置可以交换，各短语之间要用至少一个空格字符隔开；

命令动词和保留字、函数可以用前 4 个字母或连续不间断的 4 个以上字母表示；表示其的英文字母不分大小写；

一个命令最长 254 字符，若一行写不完或不想在一行写完时，本行行尾要用“；”，然后在下一行继续书写；如果不想人为换行，也可以一直写下去，由系统自动换行；

一行只能写一条命令；

命令行中字母和各种符号要求使用非纯中文格式，即半角字符；

每条命令结束不用标点符号，直接打回车键即可。

FoxBASE* 的命令结构如下：

命令动词 [<范围>] [<条件>] [<表达式表>] [<字段名表>] [<其它>]

根据语法规则，范围、条件、字段名表和其它，可以互换位置，但动词不能变位置。

在 FoxBASE* 中，范围有 4 种可能，分别是：

ALL 表示命令动词对库中全部记录进行操作

REST 表示命令动词对库中当前记录到最后一个记录进行操作

NEXT <n> 表示命令动词对库中从当前记录开始对 n 个记录进行操作

RECORD <n> 表示命令动词只对库中记录号为 n 的记录进行操作

如果当前记录是第一个记录，则 REST 和 ALL 操作范围一样；

如果当前记录是第一个记录且 n 的值等于全部记录数时，则 NEXT <n> 和 ALL 一样。

在 FoxBASE* 中，条件有两种可能，分别是：

FOR <条件> 符合条件的都，表示命令动词对库中符合条件的的所有记录进行操作。

WHILE <条件> 遇到第一个不符合条件的止，表示命令动词从当前记录开始对库中符合条件的记录按记录顺序进行操作，遇到第一个不符合条件的记录就停止操作。

在 FoxBASE* 中，表达式表用来指明命令动词操作的相关内容或有关公式，当表达式多于一个时，表达式之间要用“，”隔开。

在 FoxBASE* 中，字段名表用来指明命令动词操作的字段，要用 FILEDS 引导，当字段多于一个时，字段名之间要用“，”隔开。

在 FoxBASE* 中，“其它”包括打印 to print，不显示记录号 off，生成新数据文件 to <库名>，使用关键字 on <字段名或字段表达式> 等。

例 DISPLAY ALL 英语 + 语文 FOR 性别 = [男] FILEDS 姓名，性别，奖学金

DISPLAY 可以书写为 DISP，是一个显示库文件记录的命令。

ALL 指要显示库内全部记录。

FOR 短语 指的是要显示全部男同学。

FILEDS 短语 指的是要显示的字段及其值。
英语 + 语文 是一个表达式, 指明要显示两门课程之和

3. FoxBASE 的数据类型

在 FoxBASE* 中, 常用的数据类型有字符型数据、数值型数据、日期型数据、逻辑型数据、备注型数据和屏幕型数据六种。

字符型数据也称 C 型 (Character) 数据, 其最大宽度为 254 个字符。计算机能处理的各种符号, 都可以当作字符。汉字相当于两个西文字符。

数值型数据也称 N 型 (Numeric) 数据, 由 0~9 共 10 个阿拉伯数字和小数点及正负号共计 13 个符号根据需要组成, 在实际操作中可以直接进行计算。包括小数点、正负号和数字在内的最大宽度为 19, 小数部分的最大宽度为 15。

逻辑型数据也称 L 型 (Logical) 数据, 是专门用于“真”与“假”逻辑判断的数据, 其值只有两个, 一个是“假”值 (False 或 No), 在操作中, 用 .N. 或 .F. 表示, 一个是“真”值 (True 或 Yes), 用 .T. 或 .Y. 表示, 在实际应用中字母不分大小写。逻辑数据的宽度为 1。特别注意的是在实际应用中逻辑值一边一定要有一个 “.” 否则, 计算机不当作逻辑值处理, 在数据库内进行逻辑数据录入时, 直接输入 T 或 F 即可, “.” 由系统自动加上。在库外使用时, 则必须输入 “.”。

日期型数据也称 D 型 (Date) 数据, 其长度固定为 8 位, FoxBASE 默认格式为 “mm/dd/yy”, 即 “月/日/年”, 在数据库内进行数据录入时, 直接使用这种格式, 而在库外操作时, 须用日期型变量或函数。

备注型数据也称 M 型 (Memo) 数据, 有的书上称为记忆型或明细型, 它的最大长度可达 64KB, 在数据库中定义这种数据类型时, 长度固定为 10, 一般用于存放个人爱好、简历、档案材料等, 它的内容独立于标准的数据库文件外, 在对数据库进行操作时, 备注型字段只存放一个指针, 以便使其与记录相对应。

屏幕型数据也称 S 型 (Screen) 数据, 只有内存变量有屏幕型, 用于保存一个屏幕, 包括屏幕格式与内容。一般, 一个屏幕内容大约占到 4 KB 的内存空间。

4. FoxBASE 的安装、启动及退出

FoxBASE 的安装不需要专门的安装程序, 一般它的全部文件 (至少要有两个核心文件) 拷贝到 FOX 子目录下或直接拷贝到汉字操作系统子目录中即可。

FoxBASE 的启动也非常简单, 一般先引导 MS-DOS 操作系统, 然后引导汉字操作系统, 最后启动 FoxBASE, 出现 “.” 提示。假设一台计算机的操作系统为 MS-DOS, 汉字系统为 UC DOS, 全部文件在 UC DOS 子目录中, FoxBASE 的全部文件在 FOX 子目录中, UC OS 和 FOX 都是根目录的子目录, 则启动过程如下:

①开机, MS-DOS 引导成功, 屏幕出现 C: \ >

②打开 UC DOS 子目录, 即在 C: \ > 后光标处输入 CD \ UC DOS 按回车键, 屏幕出现 C: \ UC DOS >

③启动汉字 UC DOS, 即在 C: \ UC DOS > 后的光标处输入 UP 按回车键, 屏幕又出

现 C: \ UC DOS >

④装入五笔字型输入法,即在 C: \ UC DOS > 后的光标处输入 WB 按回车键,屏幕仍然出现 C: \ UC DOS >

⑤进行 FOX 子目录,即在 C: \ UC DOS > 后的光标处输入 CD \ FOX 按回车,屏幕出现 C: \ FOX >

⑥启动 FOX,即在 C: \ FOX > 后的光标处输入 FOXPLUS 按回车键,屏幕出现“.”,说明 FoxBASE 启动成功,可以进行命令操作。“.”是 FoxBASE 的人机交互式工作的提示符。

按照 DOS 操作系统编辑批处理文件的办法,将以上六步编为一个 C: \ FOX.BAT 文件,再次引导 MS-DOS 成功后,在 C: \ > 后的光标处直接按 FOX 回车即可启动 FoxBASE。

FoxBASE 的退出:在“.”后光标处输入 QUIT 按回车,屏幕出现 C: \ FOX >,表明已成功退出 FoxBASE,这时计算机处于汉字操作系统 UC DOS (假定计算机的汉字操作系统是 UC DOS),如果这时不需要在汉字系统下工作,再次输入 QUIT,即退出汉字操作系统。

第二节 常量、变量与表达式

FoxBASE 在进行数据输入和处理时,从形式上把数据划分为四大类:常量、变量、函数和表达式,这就人们常说的 FoxBASE 数据元素。

一、常量

常量是在操作或程序运行中保持不变的量。一般在输入数据时给出。在 FoxBASE 中,常量有四种数据类型:字符型(C型)、数值型(N型)、日期型(D型)和逻辑型(L型)。

1. 字符型常量

字符型常量是由定界符定起的字符串,FoxBASE 规定,字符型常量有三种定界符,分别是单引号、双引号和中括号,这三种定界符必须是半角字符,且要成对出现,在实际使用中,它们没有区别。定界符中字符串的长度就是字符型常量的长度,也就是说,字符串的长度不包括定界符。如:[中华人民共和国]就是一个字符型常量,它的长度为 14。“12345”也是一个字符型常量,它的长度为 5。“[39.4]”也是一个字符型常量,它的长度为 6。特别需要指出的是当一种定界符作为字符串的一部分对称出现在字符串两边时,则字符串的定界符要用另一种定界符,否则,字符串不合法。如:[[456]]是不合法的。定界符之间没有任何字符,表明字符串长度为 0,在字符串内空格也是一个字符。

2. 数值型常量

数值型常量就是平时我们谈到的整数和小数，它有正数和负数之分。其最大宽度为 19 位，最大精度为 16 位有效数字，表示超过 16 位有效数字的数时要用科学记数法，如 1.23×10^8 ，可以表示为 1.23E+8； 1.23×10^{-8} ，可以表示为 1.23E-8。

3. 日期型常量

日期型常量在 FoxBASE 中宽度为 8，系统默认格式为美国格式，采用“月/日/年”的格式表示，月、日、年分别用两位数表示。在实际操作中，一般使用转换函数，如表示 1999 年 12 月 6 日则用 CTOD (“12/06/99”)。如果不用转换函数，则系统会将 12/06/99 当作数值型表达式而计算出结果。即先计算 12/6，再计算 2/99。

4. 逻辑型常量

在 FoxBASE 中，逻辑型常量只有两个值，即 .T. 和 .F.，系统在操作时，遇到需作出逻辑真假的判断时就会得出两个逻辑常量之一。

二、变 量

变量是在数据处理过程或程序运行过程中值可以变化的量。假定一个程序有以下三步：

S=0

X=1

S=S+X

它的含义是将数值型常量 0 赋值给变量 S，将数值型常量 1 赋值给变量 X，最后再将 S 与 X 的和赋值给 S，S 在整个运行过程中发生变化，这就是变量。变量名可以由汉字、字母、数字和下划线组成，必须以汉字或字母开头，长度不超过 10 个字符。在 FoxBASE 中变量分为字段变量和内存变量两种类型。

1. 字段变量

字段变量就是数据库的字段（属性），变量名就是字段名，这就是说有了数据库并打开库文件以后，才有字段变量。从二维表中我们可以得知，字段的值是不一样的，所以字段是变量。除非修改库文件结构，否则不能变更字段变量的数据类型；对字段变量重新赋值需要通过修改数据库记录的命令来完成。从库文件中，我们可以得知，字段变量是多值变量，数据库中记录的值得都是变量的值。在 FoxBASE 中字段变量分为五种类型，即字符型字段变量（C）、数值型字段变量（N）、日期型字段变量（D）、逻辑型字段变量（L）和备注型字段变量（M）。

2. 内存变量

内存变量独立于数据库而存在，和数据库的存在及打开与否无关。它是用来存储常

量运行或运算的中间结果和最后结果的，当然也可以用来控制程序的运行过程。内存变量可以重新赋值，一个变量只有一个值，是单值变量。变量的数据类型及其值都可以重新赋值。从适用范围上内存变量可分为全局变量与局部变量。内存变量也是数据的一种，它可从形式上分为字符型内存变量（C）、数值型内存变量（N）、日期型内存变量（D）、逻辑型内存变量（L）和屏幕型内存变量（S）。在使用过程中，内存变量一般需要赋初值，也可以重新赋值。给内存变量赋值的命令很多。在此我们主要讲下 STORE 命令，它的格式是：

STORE 值 TO 内存变量表

其中值可以是 C、N、D、L 型的常量、变量、函数、表达式，这个命令一次可以将一个值赋予一个或多个内存变量，当同时给多个变量赋值时，变量间要用“，”隔开。

例：STORE 2+3 TO X, Y, Z

执行该命令后，变量 X、Y、Z 的值都是数值型表达式 2+3 的值，它相当于以下三步过程：

X=2+3

Y=2+3

Z=2+3

这时，如果想知道它们的具体值，就使用输出命令？

？X 回车后，屏幕显示

5

？X, Y, Z 回车后，屏幕显示

5

5

5

例：STORE 123 TO X

执行该条命令后，变量 X 的值就是数值型的常量 123。该条命令相当于

X=123

所以为一个变量赋一个值时，也可以用“=”赋值。其中“=”就是一个赋值命令。

例：Y=[张三]

这时变量 Y 就是一个字符型变量，它的值是常量 [张三]。

应该注意的是，下面两条命令不一样，有着本质的区别：

X=Y 是将 Y 的值赋给变量 X

Y=X 是将 X 的值赋给变量 Y

值得注意的是：当字段变量与内存变量同名并且数据库处于打开状态时，计算机优先处理字段变量。这时如果想使用内存变量，就需要特别指明。指明的方法是：M ~> 内存变量。

三、表达式

表达式是由常量、变量、函数通过相关运算符连起来的，它的数据类型由该表达式的值决定。表达式的值有四种类型，所以表达式也有四种数据类型，分别是字符表达式、数值表达式、日期表达式和逻辑表达式。备注型的字段变量和屏幕型的内存变量都不能在表达式中使用。

1. 字符表达式

字符表达式是由字符串运算符把字符型的常量（字符串）、变量和函数连接起来的表达式。我们通常用的字符串运算符有两个，一个是直接连接字符串运算符“+”，它的运算结果是“+”两端的字符串按原来的顺序直接相连而生成新的字符串；另一个是在连接过程中将前串尾部空格移到后串尾部的运算符“-”，它的运算结果是“-”前端的字符串尾部的空格移到“-”后端的字符串尾部而生成新字符串。如果运算符前端字符串尾部没有空格时，这两个运算符的运算结果是一致的。不管是“+”，还是“-”，两个字符串连接产生的新字符串的长度等于原两字符串的长度和。

例：

```
.?"中国"+"共产党"  
      中国共产党  
.?"山西"-"农业大学"  
      山西农业大学      * * 空格在屏幕上不显示  
.?"山西"+"农业大学"  
      山西      农业大学  
.?"123"-"456"- "789"  
      123456789      * * 空格在屏幕上不显示
```

2. 数值表达式

数值表达式是由算术运算符把数值型常量、变量和函数连接在一起的表达式，又称为算术表达式。我们通常用的算术运算符有五个，按运算顺序分别是括号“()”、正负号“+”和“-”、乘方“^”或“**”、乘除“*”和“/”、加减“+”和“-”。

例：求十六分之一与八分之一的差和八的积的平方

```
.? ((1/16-1/8)*8)^2  
-0.25
```

例：显示库文件内英语和语文的总成绩

```
.LIST 英语+语文  
      129.5  
      177  
      127  
      68
```

170.5

165

141

3. 日期表达式

日期表达式是由算术运算符“+”和“-”连接有关日期型数据及相关数值型数据而生成的表达式。这两个连接符的格式分别是：

日期数据 - 日期数据 表示两个日期相差的天数，结果是数值型数据

日期数据 + 整数 表示一个日期的整数天后的日期，结果是日期型数据

日期数据 - 整数 表示一个日期的整数天前的日期，结果是日期型数据

两个日期型数据比较大小时，越晚日期越大，也就是说先按年比较，当年一样时按月比较，年月都一样时最后按日比较。特别要注意的是当日期型数据与年龄挂钩时，年龄越大，日期型数据越小；反之，日期型数据越大，年龄就越小。

4. 逻辑表达式

逻辑表达式是指用关系运算符把两个数据类型一样的表达式连接起来而生成的表达式以及用逻辑运算符把多个逻辑表达式组合起来的表达式。习惯上，我们把用关系运算符连接起来而生成的表达式称为简单逻辑表达式，这种运算称为关系运算；而把用逻辑运算符连接生成的表达式称为复合逻辑表达式，这种运算称为逻辑运算。不管是关系运算还是逻辑运算，表达式结果一定是逻辑常量“.T."或".F."。在实际操作中，关系运算符要求两边参加比较的表达式数据类型要一致，在比较运算时，数值型数据按大小比较；普通字符型数据按 ASCII 码顺序比较，汉字按机内码比较，常用的一级汉字按字典的汉语拼音顺序排列；日期型数据按年、月、日的顺序比较，值越大，日期越晚。

常用的关系运算符有九个，分别是：

= 等于号，用来比较等号两边的表达式的值是否相等，如果等，结果为 .T.，如果不等，结果为 .F.。比较字符串时，若两串一样或后串是前串的前缀时（计算机默认），结果为真；否则，结果为假；当系统处于 SET EXACT ON 状态时属于字符串精确比较。

< > 不等号，要求与等号一样，结果与等号相反。

= = 串等号，即字符串精确比较号，当两个字符串完全一样时，结果为真；否则为假。

串不等号，结果与 = = 相反。

> 或 < 或 > = 或 < = 分别是我们常说的大于号、小于号、大于等于号、小于等于号，要求两边数据类型一致，数值型数据按大小比较，字符型数据如果是汉字（主要是一级汉字）按字典的汉语拼音顺序比较大小，其它符号字符按 ASCII 码值比较大小，常用的 ASCII 值有：空格为 32，0~9 分别为 48~57，A~Z 分别为 65~90，a~z 分别为 97~122。

\$ 串包含号，比较前字符串是否是后字符串的子串。如果是结果为 ".T."，否

则, 结果为 ".F."。

例: .?"张三"="张"

T (在上机操作时看不见两边的点)

.?"张" \$ "张三"

T

.?"张三">"张"

T

.?"张三" = "张"

N

.SET EXACT ON

.?"张三" = "张"

N

.?"张三" \$ "张"

N

.?"张三">= "张"

T

.?"张三"<= "张"

T

常用的逻辑运算符有四个, 按运算先后顺序分别是:

() 括号运算, 在运算过程中, 其内的表达式优先进行运算

.NOT. 或! 逻辑非, 运算结果是其后的逻辑值的相反值

.AND. 逻辑与, 其两边表达式的值都是真时, 运算结果为真, 否则, 运算结果为假。

.OR. 逻辑或, 其两边的表达式的值有一个为真, 运算结果就为真。只有当两边的表达式的值都为假时, 结果才为假。

需要指出的是: 当算术运算、字符串运算、关系运算、逻辑运算符同时存在于一个表达式中时, 应按照算术运算、字符串运算、关系运算、逻辑运算的顺序进行运算。

例: 求下面表达式的值

$84 < 126 - (-100)$.OR. "AB" - "CD" < "XYZ" .AND. .NOT. "86" = "8"

根据运算顺序, 逐步运算如下:

① $84 < 126$.OR. "AB" - "CD" < "XYZ" .AND. .NOT. "86" = "8"

② $84 < 126$.OR. "ABCD" < "XYZ" .AND. .NOT. "86" = "8"

③ .T. .OR. .T. .AND. .NOT. .T.

④ .T. .OR. .T. .AND. .F.

⑤ .T. .OR. .F.

⑥ .T.

所以该表达式的值为 .T.

通过以上学习, 要注意以下两个 $y = x$ 的区别。已知 $X = 8$, 请问 .Y = X 代表什么? 又知

X=8, Y=8, 请问 .? Y=X 代表什么?

换句话说以上两个 Y=X 有什么区别?

第一问题主要代表为变量 Y 赋值, 在这个题中 X 是变量, 它的值由常量 8 赋给, Y 也是变量, 它的值由 X 赋给, “Y=X” 式中 “=” 是赋值命令。

第二个问题主要代表两个表达式比较大, 在这个题中 X 是变量, 它的值由常量 8 赋给, Y 也是变量, 它的值由常量 8 赋给, “Y=X” 式中 “=” 是关系运算符, 结果是逻辑型的常量 T。

第三节 函 数

FoxBASE 提供了近百种函数, 大大地提高了系统的运算能力, 根据其功能, 这些函数可分为六大类, 分别是数学运算函数、字符操作函数、时间日期函数、数据类型转换函数、测试标识函数、输入函数。除宏替函数 & 外, 每个函数名称后都要跟一个括号。除系统提供的函数外, 我们在实际使用中还可以编写子程序作自定义函数。我们必须知道, 函数是一种数据形式, 不是动词命令, 它不能在圆点下直接使用。许多函数根据需要带有参数。下面将对各种函数作一详细地讲解。

一、数学运算函数

1. 求绝对值函数

函数格式: ABS (<数值型表达式>)

功能: 求数值型表达式值的绝对值。

例: .? ABS (-4)

4

例: .? ABS (30-60)

30

2. 取整函数

函数格式: INT (<数值型表达式>)

功能: 截去数值型表达式的小数部分, 函数值为表达式值的整数部分。

例: .? INT (-13.6)

13

注意: 取整时不四舍五入

例: .? INT (8.4×6)

50

例: 已知 X=8, Y=9, 问 X, Y 哪个是奇数

.? INT (X/2) <> X/2, INT (Y/2) <> Y/2

.F.

.T.