

汉

字

FoxBASE+
高级程序设计

汉

郑阿奇 编著
东南大学出版社

字

FoxBASE+

汉字 FoxBASE+高级程序设计

郑阿奇 编著

东南大学出版社

(苏)新登字第 012 号

内 容 提 要

本书以 FoxBASE+2.10 为背景介绍其高级程序设计方法, 共分为三篇。第一篇简要介绍 FoxBASE+命令, 第二篇介绍通用高级程序设计, 第三篇介绍高级应用技术。书中列出的例子均是实用程序。

本书比较适合具有一定能力的应用软件开发人员, 也可作为大学本科和专科计算机专业的教材或教学参考书。为便于学习和推广使用, 备有教学盘和通用信息管理系统应用盘。

责任编辑 冉榴红

责任校对 张新荣

汉字 FoxBASE+高级程序设计

郑阿奇 编著

*

东南大学出版社出版发行

(南京四牌楼 2 号 邮编 210018)

江苏省新华书店经销

大丰印刷二厂印刷

*

开本: 787×1092 毫米 1/16 印张: 17.5 字数: 426 千

1993 年 4 月第 1 版 1994 年 12 月第 2 次印刷

印数: 1—5000 册

ISBN 7—81023—745—4/TP·45

定价: 12.50 元

(凡因印装质量问题, 可直接向承印厂调换)

前 言

汉字 FoxBASE+2.10 数据处理的功能很强,因而被广泛用于编写管理系统程序。一般来说对于一个特定的管理系统就须编制一套管理程序,工作量大,开发周期长。实际上,尽管各个管理系统操作的数据库项目不同,要求各异,但就基本功能来讲都是一样的,例如都必须包含数据库的建立、修改、查询、统计、报表和图形等,数据库字段的内容也是有规律的。这就是说,完全能找到一种方法,使其编制的程序对一般管理系统都适用,特定的管理系统仅需加入特定功能即可完成。

本书以 FoxBASE+2.10 为背景介绍其高级程序设计方法,书中提供的各功能单元程序可组装成一个通用信息管理系统,避免程序的重复开发,并可实现对现有管理系统的改造及 dBASE III 管理系统的直接更新。比较适合具有一定能力的应用软件开发人员,也可作为大学本科和专科计算机专业的教材或教学参考书。为便于学习和推广使用,备有教学盘和通用信息管理系统应用盘,需要者请与作者联系。

为便于教学和自学,第一篇简要介绍了 FoxBASE+2.10 的基本内容,系统设置等命令和系统函数请参阅附录。

全书共分三篇,郑进同志参加了第一篇的编写。

本书由南京大学张福炎教授主审,其他许多同志也给予了大力支持,在此一并表示感谢!

编 者
1992 年 12 月

目 录

第一篇 FoxBASE+基础

第一章 FoxBASE+特性、命令项和全屏幕编辑键	(3)
1.1 FoxBASE+特性	(3)
1.2 命令项和全屏幕编辑键	(5)
第二章 数据库建立和修改	(9)
2.1 数据库结构	(9)
2.2 数据库建立与修改	(11)
2.3 数据的转移、数据库的复制	(15)
第三章 数据库查寻、定位、检索和计算	(18)
3.1 数据库的查询	(18)
3.2 数据库的检索	(19)
3.3 多工作区数据库关联和更新	(21)
3.4 数据库的计算	(23)
第四章 内存变量和其它命令	(27)
4.1 内存变量	(27)
4.2 其它命令	(29)
第五章 系统程序设计	(33)
5.1 命令文件(程序)	(33)
5.2 程序交互性命令	(39)

第二篇 通用高级程序设计

第六章 通用高级程序设计基本技术	(52)
6.1 数据库及其结构	(52)
6.2 程序功能	(66)
第七章 主程序、菜单程序和连接程序	(73)
7.1 主程序	(73)
7.2 菜单程序	(78)

7.3 系统功能连接程序	(82)
7.4 系统维护	(84)
第八章 数据库记录卡片格式编辑与显示	(88)
8.1 数据库记录定位及功能连接	(88)
8.2 数据库记录卡片格式显示	(91)
8.3 数据库记录卡片格式编辑	(98)
第九章 规范信息的输入及条件表达式的自动生成	(108)
9.1 规范信息输入	(108)
9.2 条件表达式自动生成	(116)
第十章 数据库查询和字段选择	(125)
10.1 数据库查询	(125)
10.2 字段选择	(136)
第十一章 随机统计汇总报表的自动生成	(139)
11.1 基本技术	(139)
11.2 报表程序	(143)
第十二章 分析图形的显示、打印和绘制	(167)
12.1 分析数据库	(167)
12.2 分析图形的显示和打印	(171)
12.3 分析图形的绘制	(186)

第三篇 高级应用技术

第十三章 远程通信、时钟、计算器	(200)
13.1 远程通信	(200)
13.2 远程通信处理程序	(201)
13.3 通信程序举例	(207)
13.4 时钟、计算器	(212)
第十四章 数据库、记忆库和内存变量文件的内部结构及其高级语言的直接存取 ..	(215)
14.1 数据库文件(.DBF)的结构	(215)
14.2 备注型文件(.DBT)文件结构	(216)
14.3 内存变量(.MEM)文件结构	(216)
14.4 直接操作数据库文件	(217)
第十五章 虚拟画图设备、汇编语言画图程序和打印字库的通用方法	(222)
15.1 虚拟画图设备	(222)
15.2 汇编语言画图程序	(223)
15.3 打印字库的通用方法	(226)
第十六章 FoxBASE+与汇编语言多参数传递和图象的存取	(229)

16.1	FoxBASE+与汇编语言多参数传递	(229)
16.2	FoxBASE+图象的存取	(231)
第十七章 多用户 FoxBASE+		(236)
17.1	概述	(236)
17.2	冲突及其处理	(237)
17.3	死锁的产生及其处理	(244)
附录一	命令文件的编译和组合	(247)
附录二	系统配置文件(CONFIG.FX)	(248)
附录三	FoxBASE+命令清单	(251)
附录四	系统函数	(259)
附录五	FoxBASE+错误信息表	(265)

第一篇

FoxBASE+基础

第一章 FoxBASE+特性、命令项和全屏幕编辑键

1.1 FoxBASE+特性

1.1.1 技术指标

1. 数据库文件

记录个数: 10 亿.

总字节数: 20 亿.

每个记录字节数: 在.DBF 文件中为 4000 字节.

2. 字段

记录允许的字段个数: 128.

字段数据类型: 5.

字段宽度:

字符型字段 (C) ——最大 254 个字节.

数值型字段 (N) ——最大 19 个字节.

逻辑型字段 (L) ——1 个字节.

日期型字段 (D) ——8 个字节.

记忆型字段 (M) ——4096 字节.

3. 数值精度

有效位数: 最多 15 位, 其中小数部分最多 9 位.

4. 内存变量

最多可同时使用的内存变量为 3600 个.

允许使用最大 3600 个数组, 每个数组最大为 3600 个元素. 一个数组系统仅作一个内存变量个数计算.

当然, 上面这些数据都受到计算机可使用的内存空间的限制.

5. 文件操作

可同时打开 48 个不同类型的文件.

可同时打开 10 个数据库文件.

数据库文件中使用了记忆型字段, 每个数据库文件须打开一个备注型文件.

每个数据库文件可同时打开 21 个索引文件.

每个数据库文件可打开 1 个格式文件.

6. 系统本身要求的最小内存

FoxBASE+标准版本 360K.

FoxBASE+/386 保护模式版本 2M.

1.1.2 文件类型

系统规定的隐含扩展名与相应文件类型的对应表如下:

文件类型	隐含扩展名
数据库文件	.DBF
记忆文件	.DBT
索引文件	.IDX
存贮变量文件	.MEM
命令文件	.PRG (编译后.FOX)
格式文件	.FMT (编译后.FMX)
标签文件	.LBL
报表格式文件	.FRM
文本文件	.TXT
现场文件	.VUE

1.1.3 运行环境

1. 硬件环境

IBMPC/XT, 286, 386, 486 等微机及其各种兼容机。

RAM 足够大, 在装入汉字操作系统 (包括汉字字库) 后有不小于运行系统需要的最小内存空间。

单色或彩色显示器一台及其接口卡。

并行打印机一台及其接口卡。

一台到两台软盘驱动器及一台硬盘驱动器与相应的软盘硬盘接口卡。

键盘一个。

2. 软件配置

2.0 以上版本的汉字操作系统及其汉字打印字库和相应的汉字驱动程序。用于操作系统启动的盘上包括系统配置文件 CONFIG.SYS。

CONFIG.SYS 文件包含有 PC 机用于操作方式的信息。每次开机时, 它就检查这个文件是否存在, PC 机通常允许一次打开最多 8 个文件, 用户可以使用其中的 5 个。数据库操作中打开的文件大于 5 个就需进行设定。同样也应根据用户使用的实际情况设定磁盘缓冲区大小, 否则会出现打开的文件数太多和内存空间不够等错误。

用文本编辑软件修改操作系统盘上的 CONFIG.SYS 文件, 使其包含下列二行内容:

FILES = 20

BUFFERS = 24

根据实际情况, 还须用文本编辑软件建立一个 CONFIG.FX 文件, 修改系统的默认参数, 以满足运行用户程序的需要。

需要注意, 为了留出足够的内存空间给 FoxBASE+使用, 若主机内存不大于 640K, 宜将字库驻留在硬盘上; 对于大于 640K 的机器, 宜将字库放在扩展内存 (虚拟盘) 上, 否则 FoxBASE+无法进入或者 MODIFY COMMAND, RUN 等命令无法运行。

FoxBASE+2.10 基本系统须有以下文件:

FOX.BAT

INT10.EXE

MFOXPLUS.EXE 系统程序及常驻内存可执行程序文件

MFOXPLUS.OVL 系统可覆盖程序

MFOXHELP.HLP 帮助文件

FOXPCOMP.EXE 命令文件编译程序

FOXBIND.EXE 命令文件组合过程程序

EUROPEAN.MEM 欧洲文字排序基准

1.2 命令项和全屏幕编辑键

1.2.1 命令项符号

[...] 表示方括号中的项是可选的, 而不是必须的, 任选项[]中的内容作为一个整体不能分开。

<...> 表示其中的内容是必须的, 不是可选的。

/ 表示其前后的内容任选一个。

,... 表示同类项目的重复出现。

<Cexp> 字符型表达式。

<Nexp> 数字型表达式。

<Dexp> 日期型表达式。

<Lexp> 逻辑型表达式。其值为.T.为真, .F.为假。

<exp> 表达式。表示可以是上述表达式中的任意一种。

[OFF]表示不显示记录号。

[TO PRINT]表示在打印机上打印出来, 不选它可在执行命令前按^P, 也能联机打印。

<范围> 包括:

RECORD <Nexp> 第<Nexp>号记录。

NEXT <Nexp> 当前记录开始的<Nexp>个记录。

REST 当前记录开始至文件末尾的所有记录。

ALL 所有的记录。

不选<范围>隐含, 大多数系统默认 ALL, 个别命令为当前记录。

WHILE <条件> 表示从<范围>内的第一个记录开始, 只要条件成立, 就对该记录进行操作并使记录指针指向下一个记录, 一旦遇到不符合<条件>的记录就结束该命令。

FOR <条件> 表示从<范围>内的第一个记录开始, 如果条件成立, 就对该记录进行操作, 否则不对该记录操作; 然后使记录指针指向下一个记录, 重复上述操作, 直到范围内的记录搜索完毕。

<条件>用逻辑表达式表达的操作条件,它有真和假两个值,真表示条件成立,假表示条件不成立。

[FIELDS] <字段名表>表示字段名序列,字段名之间用“,”分隔。<字段名表>一般均可包含多个工作区的字段。使用非当前工作区中的字段变量,须在字段名变量前冠以前缀“<工作区名>/<别名>->”。

<表达式表>为表达式序列,表达式之间用“,”分隔。

<表达式>由运算符和数据项经适当组合而成的式子,通过运算得到表达式的值。

<数据项>最基本的元素是常量、变量(包括字段变量和内存变量)和函数。这些元素本身就是一个简单的表达式,由它们通过运算符适当组合起来的式子就是一般的表达式。

<运算符>对数据进行运算操作的符号。FoxBASE+共有四种类型的运算符。

(1) 算术运算符

+ - 单目运算符正负

* * 或 ^ 幂

* / 乘,除

+ - 加,减

算术运算符对数字型数据进行算术运算操作,产生算术运算结果,优先顺序从上到下。

(2) 关系运算符

< 小于 > 大于

= 等于 < > 或 # 不等于

<= 小于等于 >= 大于等于

= = 字符串精确比较

关系运算符用于对两个相同类型的表达式进行比较,运算结果只能是逻辑真(T.)或假(F.)。

(3) 逻辑运算符

.NOT.或! 非

.AND. 与

.OR. 或

优先次序: .NOT. → .AND. → .OR.

(4) 字符串运算

+ 把两个字符串连接起来

- 两个字符串连接,把前串尾部空格移到连接后的字符串后面

除日期型外表达式中运算符两边参与运算的数据的类型必须完全一致,否则会给出错误信息。在日期型表达式中,除了运算符两边是日期型外还允许日期型常数、变量与数值型常数或变量运算,这是允许运算符两端的数据类型不一致的唯一特例。

同一表达式中使用几种不同类型的运算符时,这些不同类型的运算符的优先级从高到低是:算术运算符和字符串运算符 → 比较运算符 → 逻辑运算符。同一表达式中使用优先级相同的运算符时,则按从左到右的次序进行运算。括号()可以改变运算的优先级,运

算时总是先算括号里面的。注意，括号必须成对出现。

例如，下面就是一个一般的逻辑表达式：

(职称="工程师".OR.工龄>20).AND.DTOC(出生时间)>"60".OR.! 工资>N+82.AND.婚否

1.2.2 全屏幕编辑控制键及其主要功能

键	等价键	功 能
↑	^E / ^K	光标上移一行
↓	^J / ^X	光标下移一行
←	^S	光标左移一个字符位置
→	^D / ^L	光标右移一个字符位置
回车	^M	光标下移一个项目或下移一行
Ins	^V	打开或关闭插入方式
Del	^G	删除光标处的一个字符
Backspace	^H	删除光标左边的一个字符
^Y	^T	删除光标到项目尾部的内容，编辑程序时删除光标所在的程序行
^U		在定义和修改库结构时删除一个字段，在操作数据库时加上或去掉删除记录标记
^N		在定义和修改库结构时插入一个字段，在编辑程序时插入一行
Esc	^Q	操作数据库时存盘退出(但当前记录修改无效)，编辑命令程序和修改库结构时放弃本次所作的所有修改
^End	^W	保存本次所作的所有修改内容后退出
Home	^A	光标左移一个字
End	^F	光标右移一个字
PgUp	^R	上移一个记录或上移一屏
PgDn	^C	下移一个记录或下移一屏
^→	^B	在建立修改库结构时右移显示；在 BROWSE 命令时整屏右移一个字；在编辑程序时移至行尾
^←	^Z	在建立修改库结构时左移显示；在 BROWSE 命令时整屏左移一个字；在编辑程序时移至行首
^PgDn	^-	进入记忆型字段编辑
^PgUp	^^	退出并存贮记忆型字段
^O		多用户系统中在 EDIT 和 CHANGE 中加锁一个记录
^P		使打印机联机或脱机
F1		开关光标编辑键提示
^Home	F10 / ^]	BROWSE、MODIFY STRUCTURE 等命令开关子功能菜单
^KT		文本编辑时移至文件头
^KE		文本编辑时移至文件尾
^KF		文本编辑时将光标定位到文件中指定字符串的第一个位置
^KL		文本编辑时继续定位指定字符串的下一个位置
^KR		文本编辑时在光标处插入读入的指定文件内容
^KW		文本编辑时将当前文件写至指定文件中
^KB		文本编辑时重新格式一个段落

注：^为 Ctrl 键

1.2.3 命令规则

(1) 命令行由命令动词和短语组成, 命令动词指明了本命令的功能, 它是命令中必不可少的主要部分。命令行中的短语用于扩展命令的基本功能。例如 FOR 引导条件, NEXT 引导范围, 短语部分是可选的, 不是必需的。系统本身所用到的英文单词称为保留字, 一般不要把它定义为用户的文件名、字段名等, 以免引起混乱。可选项的书写先后顺序任意, 其功能不变。系统对英文字母的大小写等同处理。

【例】 LIST NEXT 100 FOR 职称="助工"

LIST FOR 职称="助工" NEXT 100

(2) 命令中系统用的保留字都可以缩写, 写前面四个以上字母系统就能区分。例如动 APPEND 可写成 APPE, DISPLAY STRUCTURE 可写成 DISP STRU, 函数 SUBSTR 可写成 SUBS 等, 但是字母的拼法和顺序必须正确。

(3) 命令行中的单、双引号的作用相同, 加了引号的内容认定为字符串。单、双引号必须成对出现。字符串中包含引号, 则字符串两边用[]括起。

(4) 命令行中的<文件名>应包括驱动器号, 若为当前盘可以省略。当前盘也可用设置命令 (SET DEFAULT TO <驱动的名>) 改变。扩展名一般不要书写, 少数命令除外。

(5) 一个命令行可以由几个物理行组成, 系统在打满一行后自动换行。在命令程序中行与行之间也可以用回车结束, 但须在非最后一行后加";"。一个命令行字符总数不得超过 254 个字符。

(6) 一个逻辑行只能写一个命令 (&&语句可写在一个语句的后面)。

第二章 数据库建立和修改

2.1 数据库结构

2.1.1 数据库结构的建立与修改

1. 建立一个新的数据库结构

格式: CREATE [<文件名>]

数据库结构建立后, 产生<文件名>数据库文件(扩展名为.DBF), 若数据库结构中
包含记忆型字段, 则同时产生<文件名>记忆型文件(扩展名为.DBT)。此后操作该数据
库前必须打开这个数据库。

格式: USE [<文件名>]

该数据库如有记忆型字段则同时打开相应的记忆文件。无任选项则关闭当前打开的数
据库文件。

2. 数据库结构修改

格式: MODIFY STRUCTURE

修改库结构时, 如果原数据库中已有记录, 则数据结构修改后原记录对应的数据自动
加入, 但对于同一字段字段名和字段长度不能同时进行修改, 而应该分两步(打入两次命
令)进行。

3. 显示数据库的结构

格式 1: LIST STRUCTURE [TO PRINT]

格式 2: DISPLAY STRUCTURE [TO PRINT]

这两条命令的区别是, 后者每显示 20 行暂停, 用户打任意键继续显示。选[TO
PRINT]就可以把这些信息打印出来。

【例】建立人员基本情况数据库结构, 文件名为 RS1。

. CREATE RS1

. (建立数据库结构)

. USE RS1

. MODIFY STRUCTURE

. (修改操作)

. LIST STRUCTURE

数据库结构——数据库 : RS1.DBF

数据库中的数据记录个数 : 0

数据库的最后更新日期 : 01/01/92

字段	字段名	类型	宽度	小数
1	编号	字符 / 正文	4	
2	姓名	字符 / 正文	8	
3	职称	字符 / 正文	6	
4	工龄	数字型	2	0
5	工资	数字型	6	2
6	婚否	逻辑型	1	
7	出生时间	日期型	8	
8	备注	记忆型	10	
* * 总计 * *			46	

(这里, 总计的宽度比累计的宽度大 1, 这一个字节是系统作记录删除标志用.)

. USE

2.1.2 数据库结构描述数据库

1. 由数据库结构生成结构描述数据库

格式: COPY STRUCTURE EXTENDED TO <结构描述文件名>

结构描述数据库由 FIELD_NAME、FIELD_TYPE、FIELD_LEN 和 FIELD_DEC 字段组成。用以存放原数据库的字段名、字段类型、字段宽度和字段小数位, 原数据库结构中的一个字段在结构描述库中对应一个记录。

【例】

. USE RS1

. COPY STRUCTURE EXTENDED TO RS_JG

. USE RS_JG

. LIST

Record#	FIELD_NAME	FIELD_TYPE	FIELD_LEN	FIELD_DEC
1	编号	C	4	0
2	姓名	C	8	0
3	职称	C	6	0
4	工龄	N	2	0
5	工资	N	6	2
6	婚否	L	1	0
7	出生时间	D	8	0
8	备注	M	10	0

注: LIST 是显示数据库记录内容的命令。

2. 由结构描述数据库建立数据库结构

格式: CREATE <文件名> FROM <结构描述文件名>

【例】

. USE RS_JG

. CREATE RS2 FROM RS_JG