



汉字 FoxBASE+ 2.10

程序设计 100例

李春葆 编著
熊可宜 审校

海洋出版社

北京希望电脑公司计算机技术丛书

汉字 FoxBASE+ 2.10 程序设计 100 例

李春葆 编著
熊可宜 审校

海洋出版社

1992 年·北京

内 容 简 介

本书主要讨论 FoxBASE+ 2.10 系统的编程技术, 共分十九个专题开发了 105 个程序, 每个专题中, 先简要介绍有关基本概念, 然后列出相关例子, 并描述了各个例子程序的功能及主要采用的技术, 可供 FoxBASE 编程技术人员参考借鉴使用。

需要本书的用户, 请直接与北京 8721 信箱联系, 电话: 2562329, 邮政编码: 100080。

(京)新登字 087 号

责任编辑: 刘莉霞

FoxBASE+ 2.10 程序设计 100 例

李春葆 编著

熊可宜 审校

海洋出版社出版 (北京市复兴门外大街 1 号)

海洋出版社发行 双青印刷厂印刷

开本: 787×1092 1/16 印张: 10.475 数: 243 千字

1993 年 9 月第二版 1993 年 9 月第二次印刷

印数: 4000 - 10000 册

ISBN7-5027-2958-5 / TP·148 定价: 9.00 元

目 录

前 言	1
一、例中使用的数据文件	2
二、宏代换	5
三、键码操作	8
四、动态画面设计	14
五、颜色、日期和时间	21
六、保持系统安全性	23
七、菜单设计	28
八、加密和解密	37
九、使用数据库文件	43
十、使用结构伸展文件	47
十一、数据修改和删除	52
十二、数据输入	55
十三、数据查询和数据显示	64
十四、数据统计	98
十五、数据库操作	101
十六、绘制统计图	110
十七、打印	121
十八、源程序结构检测与分析	129
十九、其它	157
参考文献	163

前 言

FoxBASE 是目前国内使用最普及的关系数据库管理系统之一,它以编程灵活方便、程序运行速度快、支撑硬件档次较低等特点深受广大的计算机信息系统开发人员的青睐。新推出的汉字 FoxBASE+2.10 系统除保留原版的一些特点外,还提供了若干开发工具,如图形生成器 FoxGraph、屏幕格式生成器 FoxVIEW,应用程序生成器 FoxCODE 以及程序文档生成器 FoxDOC 等,从而构成了一个强有力的集成化开发系统,减少用户开发应用的时间。

尽管如此,基础编程仍是十分重要和不可缺少的,因为这些工具并不能完全替代编程,也不能完成所有的编程工作,一些复杂的功能仍要靠精心编制的程序来实现。而且,具有良好的编程技术可以充分利用 FoxBASE+2.10 的各种功能,提高应用程序的运行效率和速度。

本书主要讨论 FoxBASE+2.10 系统的编程技术,其实这些技巧对于其它较低的 FoxBASE 版本如 FoxBASE+2.0 也同样实用,我们在 FoxBASE+2.10 系统上开发了 105 个小程序,并分成十几个专题加以讨论,每个专题中,先简要介绍有关基本概念,然后列出相关的例子,并描述了各个例子程序的功能及主要采用的技术。值得说明的是,专题的划分并非十分严格,例如,宏代换专题部分仅包含了 4 个例子,实际上后面的大部分程序都使用了宏代换函数。

本书中所有程序均在 AST386 微机上和汉字 FoxBASE+2.10 系统环境下调试通过的,因此这里没有讨论有关长城系列机的特殊功能,如使用 GRD.SYS 图形系统绘制统计图等。为了简便,本书中的 FoxBASE 指的是 FoxBASE+2.10 系统。

书中大部分程序是作者编制的,另一部分参考了有关计算机书刊和杂志,但作者对这部分程序也作了一些改进,在此对这些作者们表示感谢。

由于作者水平有限,本书只能起到抛砖引玉的作用,书中难免存在缺点和错误之处,欢迎广大读者不吝指教。

本书的出版得到北京希望电脑公司秦人华经理的大力支持,在此表示衷心感谢。

编者

1992·北京

一、例中使用的数据文件

本书的例子程序中使用了十几个数据库文件，为了方便阅读这些程序，本部分列出所有使用的数据文件的结构。

STUD.DBF 文件结构:

Field	Field Name	Type	Width	Dec
1	XH	Character	4	
2	XM	Character	6	
3	XB	Character	1	
4	NL	Numeric	2	
5	BH	Character	5	
** Total **				

STUDENT.DBF文件结构:

Field	Field Name	Type	Width	Dec
1	XH	Character	5	
2	XM	Character	6	
3	XB	Character	2	
4	ZSRQ	Date	8	
5	ZZMM	Character	6	
6	BH	Character	6	
** Total **			34	

BHK.DBF文件结构:

Field	Field Name	Type	Width	Dec
1	姓名	Chaaracter	8	
2	学号	Character	5	
** Total **				

DZK.DBF文件结构:

Field	Field Name	Type	Width	Dec
1	学号	Chaaracter	5	
2	性别	Character	2	
3	出生日期	Date	8	
4	入学日期	Date	8	
5	系编号	Character	5	
** Total **				

XHK.DBF文件结构:

Field	Field Name	Type	Width	Dec
1	系编号	Character	5	

2	系名	Character	18
** Total **			

CJK.DBF文件结构:

Field	Field Name	Type	Width	Dec
1	班号	Character	6	
2	学号	Character	4	
3	姓名	Character	6	
4	英语	Character	3	
5	哲学	Character	3	
6	数字电路	Character	3	
7	逻辑设计	Character	3	
8	原理	Character	3	
9	PASCAL	Character	3	
10	数据结构	Character	3	
11	汇编语言	Character	3	
12	编译原理	Character	3	
13	操作系统	Character	3	
14	平均分	Character	4	
15	高数	Character	54	

**** Total ****

KMK.DBF文件结构:

Field	Field Name	Type	Width	Dec
1	JLH	Character	4	
2	KMMC	Character	16	
3	KMDM	Character	8	
4	KMLX	Character	2	
5	KMJB	Character	2	
6	OPER	Character	6	
7	RQ	Date	8	

**** Total ****

CCC.DBF文件结构:

Field	Field Name	Type	Width	Dec
1	YJH	Character	78	

**** Total ****

PWD.DBF文件结构:

Field	Field Name	Type	Width	Dec
1	YJH	Character	4	
2	UN	Character	8	
3	PWD	Character	8	
** Total **				

TMP.DBF文件结构:

Field	Field Name	Type	Width	Dec
1	FILENAME	Character	16	
2	RECNUM	Numeric	11	
3	UTIME	Character	12	
4	FSIZE	Numeric	12	
** Total **			52	

JLK.DBF文件结构:

Field	Field Name	Type	Width	Dec
1	LAN	Character	78	
** Total **			52	

LANK.DBF文件结构:

Field	Field Name	Type	Width	Dec
1	BH	Numeric	4	
2	LAN	Character	78	
3	LX	Numeric	1	
4	PD	Character	1	
** Total **			85	

NAME.DBF文件结构:

Field	Field Name	Type	Width	Dec
1	LAN	Character	20	
** Total **			21	

KJGK.DBF文件结构:

Field	Field Name	Type	Width	Dec
1	KM	Character	8	
2	FN	Character	10	
3	FT	Character	1	
4	FL	Numeric	3	
5	FD	Numeric	3	
** Total **			26	

KJG.DBF文件结构:

Field	Field Name	Type	Width	Dec
1	FIELD_NAME	Character	10	
2	FIELD_TYPE	Character	1	
3	FIELD_LEN	Numeric	3	
4	FIELD_DEC	Numeric	3	
** Total **			18	

SZK.DBF文件结构:

Field	Field Name	Type	Width	Dec
1	KM	Character	8	
** Total **			9	

SZK.DBF文件结构:

Field	Field Name	Type	Width	Dec
1	KM	Character	8	
2	BJ	Numeric	3	
** Total **			12	

二、宏代换

宏代换就是使用宏代换函数“&”进行变量代换。&是FoxBASE中最实用，也是使用最为广泛的函数之一，它用于代换一个字符型内存变量的内容，因此，&必须放在字符型变量之前，且该字符型变量已赋值，其运算结果为&函数之后字符型变量的值。

&<内存变量>组合形式与跟在它后面的其它字符之间一般要隔一个或多个空格，如果代换后的值需要与后面的其它字符连结在一起，则必须在&<内存变量>之后与其它字符之间插入一个“.”号，用于终止一个宏代换的展开。

&允许多级嵌套使用，最多可达256层，嵌套的方法是使&函数代换出来的字符串中包括&函数本身，在执行时，FoxBASE以外向里逐层进行宏代换，直至将跟在最内层的&后的变

量内容代换出来。同时在一个语句中&嵌套的次数不得超过 256 次。

注意：用于宏代换的字符型内存变量必须是标量变量，而不能是数组或数组中的单元。

例 1. 用宏代换实现字符型数据到数值型数据的转换。

本例中 STUD 的 nl 字段为数值型字段，而内存变量 aa, bb 均为字符型变量，使用&函数使得&aa、&bb 都成为数值数据，于是可以与 nl 进行比较。

*文件名:exam1.prg

set talk off

set colo to 7+/1

clea

use stud

aa=space(2)

bb=space(2)

@10,30 say "输入年龄下限:" get aa

@12,30 say "输入年龄上限:" get bb

read

set filter to nl>=&aa .and. nl<=&bb

clea

list

close all

return

例 2. 用宏代换函数代换 FoxBASE 命令。

*文件名:exam2.prg

set talk off

set colo to 7+/1

clea

aa="list"

bb="list for xb='m'"

cc="list for xb='f'"

use stud

do while .t.

@22,16 say "F1:全体同学 F2:全体男同学 F3:全体女同学 ESC:退出"

@20,34 say "请选择:"

j=inkey(0)

clea

do case

case j=28

&aa

case j=-1

```

        &bb
    case j=-2
        &cc
    case j=27
        exit
    endcase
enddo
close all
return

```

本例中，aa, bb, cc 均包含一条可执行的 Foxbase 命令，使用&后，&aa, &bb, &cc 成为可执行的命令。

例 3. 用宏代换函数代换数据库名。

*文件名:exam3.prg

```

set talk off
set colo to 7+/1
clea
fn=space(10)
@10,18 say "输入数据库名(不带扩展名):" get fn
read
fn=trim(fn)
if .not. file("&fn..dbf")
    use ybk
    copy stru to &fn
    use &fn
else
    use &fn
    zap
endif
appe
close all
return

```

本例中先读取任一不带扩展名.DBF 的数据库名，若当前目标下存在该文件，则删除其全部记录，否则用复制 ybk 库结构方法生成该文件。由于数据库文件名从用户获取，这里必须使用&fn 指定相应的库文件。

例 4. 用宏代换函数重新定义功能键。

*文件名:exam4.prg

```

set talk off

```

```

set function 2 to "use &km;list;"
set function 3 to "delete all;"
set function 4 to "recall all;"
set function 5 to "zap;"
return

```

本例中把 F2 设置成 “use &km; list; ”，一旦运行本程序后，只要给内存变量 km 赋值，那么按功能键 F2 便打开相应的库并列出其所有记录。

三、键码操作

FoxBASE 提供了三种键码，即 ASCII 码、inkey 码和 readkey 码，ASCII 码是任一可打印字符对应的机器内部编码，可以使用 asc() 函数得到任何可打印字符的 ASCII 码，反过来，可以使用函数 chr() 得到任一 ASCII 码(1—255)对应的字符。

inkey 码是使用 inkey() 函数时，任一按键返回的结果，对于可打印字符，其 inkey 码为相应字符的 ASCII 码；对于不可打印字符，如光标键和控制键等，其对应的 inkey 键码为一些特殊的值，如 Home 键对应的 inkey 码为 1。

readkey 码是用户退出全屏幕编辑状态时按键所对应的 readkey() 函数的返回值，全屏幕编辑命令有 Append, Browse, Change, Create, Edit, Inter, Modify 和 Read 等，readkey 码又分为退出码和修改码，退出码是指执行全屏幕编辑命令过程中没有修改过数据，修改码是指退出全屏幕编辑状态之前已修改过数据，两者相差 256，例如，Home 键对应的退出码为修改码 258。

FoxBASE 的键码在程序设计中十分有用，特殊是在分支条件判定和菜单设计更是如此。

例 5. 显示任一按键对应的 inkey 码。

*文件名:exam5.prg

```

set talk off
set colo to 7+/1
clea
do while .t.
    @10,25 say "按任意键(按 ESC 键退出):"
    j=inkey(0)
    @10,47 say str(j,3)
    if j=27
        exit
    endif
enddo
return

```

例 6. 显示所有的 ASCII 码字符。

*文件名:exam6.prg

```

set talk off

```

```

set colo to 7+/1
clea
@1,30 say "显示所有 ASC 码字符"
row=3
col=2
j=1
do while j<256
    @row,col say str(j,3)+" "+chr(j)
    row=row+1
    if row>22
        row=3
        col=col+6
    endif
    j=j+1
enddo
return

```

本例在一屏中显示所有 ASCII 码及对应的字符。

例 7. 显示任一可打印字符对应的 ASCII 码。

*文件名:exam7.prg

```

set talk off
set colo to 7+/1
clea
achar=space(1)
do while .t.
    @10,26 say "键入任一可打印字符(按 z 退出):" get achar
    read
    @12,29 say "该字符对应的 ASCII 码:"+str(asc(achar),3)
    if asc(achar)=122
        exit
    endif
enddo
return

```

本例的执行不同于例 5, 当按任一功能键时, 不会触发 get...read 语句, 只有可打印字符才可正常执行。

例 8. 显示汉字库中所有字符及汉字的 ASCII 码。

*文件名:exam8.prg

```

set talk off

```

```

set escape off
set colo to 7+/1
clea
i=161
row=1
col=3
np=1
k=1
do while i<255
j=161
do while j<255
    if k=1
        @0,20 say "显示汉字库中所有字符汉字的 ASCII 码"
        @0,65 say "第 "+ltrim(str(np,3))+ " 屏"
        k=0
    endif
    if col>65
        @24,25 say "按 ESC 退出,其它键继续显示下一屏"
        m=inkey(0)
        if m=27
            return
        endif
        col=3
        k=1
        np=np+1
    else
        if row>23
            col=col+15
            row=1
        else
            @row,col say str(i,3)+"," +str(j,3)+space(2)+chr(i)+chr(j)
            row=row+1
        endif
    endif
    j=j+1
enddo
i=i+1
enddo
close all
return

```

在中文操作系统下,汉字库(如 ucdos 下的 cclib.dat 文件)中的汉字和制表符等均由两个字节组成,这两个字符的最高位均为 1,它们对应的 ASCII 码均大于 160,且值为 161-255,由此采用循环方法,以 chr(i)+chr(j)获取字符中所有字符。该程序采用一屏一屏地显示方法。按 Esc 键退出,每个汉字的显示形式为:

<第一字节的 ASCII 码>, <第二个字节的 ASCII 码><汉字>例如第一屏的显示结果如下:

显示汉字库所有字符汉字的 ASCII 码				第一屏			
161,161	161,185	161,209	161,233	162,163			
161,162	161,186	161,210	161,234	162,164			
161,163	161,187	161,211	161,235	162,165			
161,164	161,188	161,212	161,236	162,166			
161,165	161,189	161,213	161,237	162,167			
161,165	161,190	161,214	161,238	162,168			
161,167	161,191	161,215	161,238	162,169			
161,168	161,192	161,216	161,240	162,170			
161,169	161,193	161,217	161,241	162,171			
161,170	161,194	161,218	161,242	162,172			
161,171	161,195	161,219	161,243	162,173			
161,172	161,196	161,220	161,244	162,174			
161,173	161,197	161,221	161,245	162,175			
161,174	161,198	161,222	161,246	162,176			
161,175	161,198	161,223	161,247	162,177	1.		
161,176	161,200	161,224	161,248	162,178	2.		
161,177	161,201	161,225	161,249	162,179	3.		
161,178	161,202	161,226	161,250	162,180	4.		
161,179	161,203	161,227	161,251	162,181	5.		
161,180	161,204	161,228	161,252	162,182	6.		
161,181	161,205	161,229	161,253	162,183	7.		
161,182	161,206	161,230	161,254	162,184	8.		
161,183	161,207	161,231	161,255	162,185	9.		

按 ESC 退出,其它键继续显示下一屏

例 9. 显示汉字库中所有字符及汉字的区位码。

汉字库中任一字符或汉字的区位码由两位的区号和两位的位号组成,区号为该汉字的第一字节的 ASCII 码-160,不是两位者,在前面补一“0”;位号该汉字的第二字节的 ASCII 码-160,不足两位者在前补一“0”,根据这一关系得到汉字库中所有汉字或字符的区位码,该程序结构和例 8 的程序结构相似,例如第一屏的显示结果如下:

0102	,	0126	¡	0150	}	0174	£	0204	
0103	.	0127	ª	0151	¸	0175	‰	0205	
0104	•	0128	«	0152	■	0176	§	0206	
0105	-	0129	»	0153	≡	0177	№	0207	
0106	√	0130	[	0154	≈	0178	☆	0208	
0107	..	0131	]	0155	≈	0179	★	0209	
0108	"	0132	±	0156	∝	0180	○	0210	
0109	÷	0133	×	0157	≠	0181	●	0211	
0110	—	0134	÷	0158	◀	0182	◎	0212	
0111	~	0135	:	0159	▶	0183	◇	0213	
0112		0136	∧	0160	≡	0184	◆	0214	
0113	...	0137	∨	0161	≡	0185	□	0215	
0114	'	0138	Σ	0162	∞	0186	■	0216	
0115	,	0139	Π	0163	∴	0187	△	0217	1.
0116	"	0140	∪	0164	∴	0188	▲	0218	2.
0117	"	0141	∩	0165	♂	0189	※	0219	3.
0118	(	0142	€	0166	♀	0190	→	0220	4.
0119	)	0143	∴	0167	•	0191	←	0221	5.
0120	<	0144	✓	0168	'	0192	↑	0222	6.
0121	>	0145	⊥	0169	'	0193	↓	0223	7.
0122	◀	0146	//	0170	°C	0194	=	0224	8.
0123	▶	0147	∠	0171	\$	0201	⑩	0225	9.

按 ESC 退出，其它键继续显示下一屏

*文件名:exam9.prg

set talk off

set esca off

set colo to 7+/1

clea

i=161

row=1

col=3

np=1

k=1

do while i<255

j=161

do while j<255

if k=1

@0,20 say "显示汉字库所有字符汉字的区位码"

@0,65 say "第 "+ltrim(str(np,3))+ " 屏"

k=0

```

endif
if col>65
    @24,25 say "按 ESC 退出,其它键继续显示下一屏"
    m=inkey(0)
    if m=27
        return
    endif
    col=3
    k=1
    np=np+1
else
    if row>23
        col=col+15
        row=1
    else
        z=str(i-160,2)
        z=ltrim(z)
        if len(z)=1
            z='0'+z
        endif
        w=str(j-160,2)
        w=ltrim(w)
        if len(w)=1
            w='0'+w
        endif
        @row,col say z+w+space(2)+chr(i)+chr(j)
        row=row+1
    endif
endif
j=j+1
enddo
i=i+1
enddo
close all
return

```

例 10. 分离任一字符串中的汉字和其它字符。

汉字是由两个字节组成的（包括汉字库中所有字符），其它字符如“a”，“，”等由单个字节组成，且其 ASCII 码小于 126。本例便是根据这一原理进行汉字分离的，例如输入