

# 通用全屏幕编辑软件的设计与实现

伍素兰 (西安公路交通大学)

郭润熙 (西安市育才中学)

**摘要:**本文利用 Foxbase 的数组功能和 INKey()函数的功能成功地编制了一个通用全屏幕编辑的软件,它集全屏幕编辑、显示于一体,来实现对数据库中的数据进行搜索、编辑。

Foxbase 以其功能完善,易扩充以及与多种高级语言有接口而深受众多 MIS 研究和开发者们的欢迎。但在使用 Foxbase 对数据库进行操作进程中,由于每一种命令只能完成简单的操作,使得对库中数据的编辑极不方便。尽管 browse 在 Foxbase 中算是一个有效的全屏幕编辑命令,利用它可以对数据库中的记录进行全屏幕编辑。但在进入编辑显示首屏记录时,命令首先清屏。在显示完一屏记录换屏时,还是先清屏后显记录。由于反复清屏,致使无法设置用户自己所需的界面。对上下记录、左右字段的操作既不直观亦不方便。更无随意增删记录的功能。笔者利用 Foxbase 的数组功能和 INKey()函数的功能,通过实践摸索研制出了一个比较完善的“通用全屏幕编辑程序”,实现对数据库中的数据进行编辑。

## 一、程序的设计思想

程序总体结构由一个主程序 QPMBJ.PRG 和两个子程序 BIJI.PRG 及 HK.PRG 组成。下面简略介绍程序的设计思想,并给出流程图及功能介绍。

### 1. 设计思想

根据用户输入的每屏字段数,设置了 8 个数组:ARRAY(3, ZDXS), ARRAY1(PZDS), ARRAY2(JLS, PZDS)。其中 PZDS 是每屏最多字段数, ZDXS 为数据库文件总字段个数(最多为 128 个字段), JLS 是一屏的记录数(最多为 15 个记录)。将数据库文件的字段名存入数组 ARRAY(1, ZDZS), 字段长存入数组 ARRAY(1, ZDZS), 相应的小数位数存入数组 ARRAY(3, ZDZS)。调用 HK.PRG 子程序画出编辑框及给出功能提示。然后,计算出各字段的列坐标并且将其存入数组 ARRAY1, 再根据字段的列坐标把字段名显示到方框的顶行。对其长度超过 80 个字符的字段作了特殊处理。接着把记录中各字段的值存入数组 ARRAY2, 再把字段的

值进行类型转换后显示到屏幕上,最后调用 BIJI.PRG 对记录进行编辑。

在 BIJI.PRG 中,先把存入到数组 ARRAY2 的字段值送入变量 GA,再把它转换成字符类型复盖到原来显示的位置上,然后用 INKEY()函数接收键盘输入的字符,判断其是否为控制字符。如果不是控制字符,则把输入的字符经过类型转换后做为该字段(光标所在字段)的内容存入数组 ARRAY2,若是控制字符,则改变起始字段号变量 ZDH 和起始记录号变量 JLH,然后返回主程序继续显示、编辑。当一屏编辑完成后,需换屏或退出程序时,在编辑窗口的底行询问是否存入数据,若回答“Y”,则把数组 ARRAY2 的值即各记录编辑后的字段值存入文件,否则,不存入文件。

### 2. 编辑功能介绍

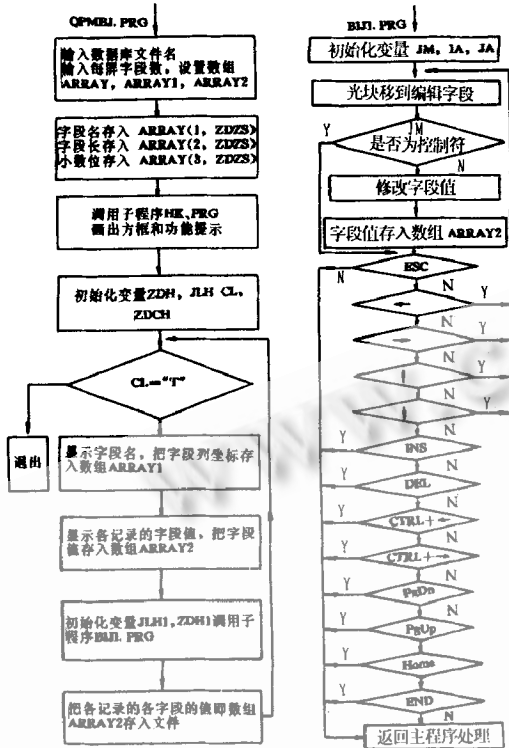
当选定要编辑的一屏后,光块停留在本屏第一个记录的第一个字段处。

←	左移光块,选取编辑字段
→	右移光块,选取编辑字段
↑	上移光块,选定记录
↓	下移光块,选定记录
CTRL←	整屏左移一个字段
CTRL→	整屏右移一个字段
PGUP	上翻一页(屏)
PGDN	下翻一页(屏)
HOME	移到起始页(屏)
END	移到最后一页(屏)
INS	在光标所在记录后打入一个空记录
DEL	当选定所删除记录时,按此键直接删除。在删除之前设置了询问功能,对删除的记录进行确认。
ESC	退出编辑,返回 Foxbase“.”状态

以上编辑功能用户可按自己的需要随意选用,对字

段的选取并不局限于一条记录而是可将光块随意移在任一记录的任一个字段上。使得对数据库中的数据操作极为方便灵活。

### 3. 主程序流程图



### 4. 使用注意事项

在使用该程序时,要注意不能用程序中的变量做字段名,以免引起混乱。另外,当数据库文件的字段太多时,程序中的内存变量可能超过 256,这时应在 CONPIG·SYS 文件中,用 MVCOUNT 重新设置内存变量个数。

## 二、功能扩充

利用数组与 INkey()函数的配合来完成对数据库中数据的编辑,仅仅是我们在这方面的起步,我们将继续探索,以求扩展出下列功能:

1. 可设置修改或不可修改的字段
2. 修改库结构
3. 根据索引条件显示记录
4. 其他提高扩展功能

附:(主程序)QPMBJ.PRG

```

SET TALK OFF
SET STATUS OFF
SET SCOREBOARD OFF
    
```

```

SET CENTURY OFF
CLEAR
FNAME = SPACE(12)
@1,16 SAY "数据库文件名:" GET FNAME
READ
PZDS = SPACE(4)
@1,44 SAY "每屏字段数:" GET PZDS
READ
PZDS = VAL(PZDS)
IF PZDS <= 0
    @1,0
    @1,0 SAY "字段数错"
    RETURN
ENDIF
DIMENSION ARRAY(3,128),ARRAY1(PZDS),ARRAY2(15,
PZDS)
* 输入数据库文件名,每屏字段数,设置数组 ARRAY, AR-
RAY1, ARRAY2
USE &FNAME
JLZS = RECCOUNT()
IF FILE("F-EXTE.DBF")
    ERASE F-EXTE.DBF
ENDIF
COPY TO F-EXTE STRUCTURE EXTENDED
USE F-EXTE
DELETE ALL FOR FIELD-TYPE = "M"
PACK
IA = 1
DO WHILE .NOT. EOF()
    ZDM = TRIM(FIELD-NAME)
    IF FIELD-TYPE = "L".AND. LEN(ZDM) < 3
        ARRAY(1, IA) = IIF(LEN(ZDM) = 1, ZDM + SPACE
(2), ZDM + SPACE(1))
    ELSE
        ARRAY(1, IA) = ZDM
    ENDIF
    ARRAY(2, IA) = FIELD-LEN
    ARRAY(3, IA) = FIELD-DEC
    SKIP
    IA = IA + 1
ENDDO
ZDZS = RECCOUNT()
* 字段名存入 ARRAY(1, ZDZS), 字段长存入 ARRAY(2,
ZDZS), 小数位数存入 ARRAY(3, ZDZS)
USE
ERASE F-EXTE.DBF
IF FILE("C:HK.PRG")
    DO C:HK.PRG
ELSE
IF FILE("A:HK.PRG")
    DO A:HK.PRG
ELSE
    DO B:HK.PRG
ENDIF
ENDIF
USE &FNAME
SET COLOR TO W + /R
ZDH = 1
JLH = 1
    
```

```

CL="F"
ZDCH=1
DO WHILE .T.
  IF CL="T"
    EXIT
  ENDIF
  @3,2 SAY SPACE(76)
  @3,3 SAY "记录号"
  NA=10
  IA=0
  DO WHILE IA< IIF(PZDS< ZDZS, PZDS, ZDZS) .AND.
ZDH+ IA< = ZDZS
  LAR1 = LEN(ARRAY(1, ZDH + IA))
  LN = IIF(LAR1 > ARRAY(2, ZDH + IA), LAR1, AR-
RAY(2, ZDH + IA))
  NA = NA + LN + 1
  IF NA > 76
    IF IA = 0
      @3,10 SAY ARRAY(1, ZDH)
      IA = 1
    ENDIF
    EXIT
  ELSE
    ARRAY1(IA + 1) = NA - LN
    @3, ARRAY1(IA + 1) SAY ARRAY(1, ZDH +
IA)

    N = LN - LAR1
    IF N > 0
      @3, COL() SAY REPLICATE("-", N)
    ENDIF
    IA = IA + 1
  ENDIF
ENDDO
ZDS = IA
IF JLZS = 0
  APPEND BLANK
  JLZS = 1
ENDIF
* 显示字段名,把各字段的列坐标存入 ARRAY1
GO JLH
IA = 0
DO WHILE .NOT.EOF() .AND. IA < 15
  @IA + 4, 2 SAY SPACE(76)
  @IA + 4, 3 SAY STR(JLH + IA, 4, 0)
  JA = 0
  DO WHILE JA < ZDS .AND. JLZS > = 1
    W = ARRAY(1, ZDH + JA)
    STORE &W TO ARRAY2(IA + 1, JA + 1)
    WA = ARRAY2(IA + 1, JA + 1)
    IF TYPE("WA") = "L"
      @IA + 4, ARRAY1(JA + 1) SAY WA
    ELSE
      IF TYPE("WA") = "N"
        WA1 = STR(WA, ARRAY(2, ZDH + JA),
ARRAY(3, ZDH + JA))
      ELSE
        WA1 = IIF(TYPE("WA") = "D", DTOC
(WA), WA)

```

```

ENDIF
  @IA + 4, ARRAY1(JA + 1) SAY SUBSTR(WA1,
(ZDCH - 1) * 66 + 1, 66)
ENDIF
  JA = JA + 1
ENDDO
  IA = IA + 1
SKIP
ENDDO
  JLS = IA
  DO WHILE IA < = 15
    @IA + 4, 2 SAY SPACE(76)
    IA = IA + 1
  ENDDO
  * 显示字段值,把字段值存入数组 ARRAY2
  CL = "F"
  CSJI = "F"
  JLH1 = JLH
  ZDH1 = ZDH
  IF FILE("C:BIJI.PRG")
    DO C:BIJI.PRG
  ELSE
    IF FILE("A:BIJI.PRG")
      DO A:BIJI.PRG
    ELSE
      DO B:BIJI.PRG
    ENDIF
  ENDIF
  CLEAR GETS
  IF CSJI = "T"
    ZM = ""
    @19, 3 SAY SPACE(70)
    @19, 3 SAY "是否存入数据(Y/N)?" GET ZM PIC-
TURE "Y"
    READ
    IF ZM = "Y"
      IA = 1
      DO WHILE IA < = JLS
        GO JLH1 + IA - 1
        JA = 1
        DO WHILE JA < = ZDS
          WA = ARRAY(1, ZDH1 + JA - 1)
          WA1 = ARRAY2(IA, JA)
          REPLACE &WA WITH WA1
          JA = JA + 1
        ENDDO
        IA = IA + 1
      ENDDO
    ENDIF
  ENDIF
  * 把修改后的字段值即数组 ARRAY2 存入文件
ENDDO
USE
SET COLOR TO W/N
RETURN

```

(编者注:两个子程序因篇幅关系略去,有兴趣读者请与作者联系。)