用 TURBO PASCAL 实现数据输入全屏幕编辑

何亚群　赵亦朋　　（中国矿业大学）

摘要：本文介绍了一个用 *PASCAL* 语言编写的通用全屏幕编辑数据输入程序，并对该程序的特点及使用方法作了简单叙述，提供了源程序清单。

使用 C，PASCAL 等高级语言编程时，由于它们没有提供象 FOXBASE,FOXPRO 语言 PROMT，MENU TO 语句这样的输入全屏幕编辑命令，因此仅用这些高级语言的读语句输入数据时，无法做到全屏幕编辑。操作人员一旦输入一个错误数据并按了回车键后，再要修改这个数据就必须由程序控制并返回重新输入，因而给编程和输入均带来很大的不便。针对这一问题，笔者用 TURBO PASCAL 6.0 编写了一组通用过程，用它们可简便有效地实现数据输入的全屏幕编辑。通用过程具有以下特点：

1.主要通用过程包括 ADDGET，READGET，GETREAL，GETSTR，GETINT 等，其中只要调用 ADDGET 和 READGET 两个过程，即可实现数据输入的全屏幕编辑。GETREAL，GETSTR，GETINT 则分别是实型、字符串型及整型数据的输入过程。

2.输入数据时可根据用户要求，在屏幕的任何位置，对屏幕允许范围内的任意项数据进行数据的全屏幕编辑和修改。输入结束并确认无误后，按“ESC”键即可将变量读入内存。

3.ADDGET 过程带有五个参数，即 ADDGET (P_Y,P_X,X,CH,TF)；其中 P_Y,P_X 为整型变量，表示输入项的纵，横坐标。X 是不定类型变量，代表输入的数据。CH 为字符型变量，当 CH 为′S′或′s′时，表示输入数据 X 为字串型变量；CH 为′I′或′i′时，X 是整型变量；CH 为′R′或′r′时，X 是实型变量。换言之，根据不同参数，该过程可输入字串、整型、实型三种类型的变量，基本满足了一般程序设计的需要。TF 为布尔型变量，当 TF 为 TRUE 时，输入的变量自动赋零或空字串；当 TF 为 FALSE 时，变量 X 为原内存中的值，这样使得修改已输入的数据成为可能。

4.程序对功能键 UP、DOWN、LEFT、RIGHT、HOME、END、DEL、INS、PgDn、PgUp、BACKSPACE、TAB、ENTER、ESC 等键作了定义，这样不仅可以用功能键在屏幕上任意选择输入项，又可以在选定的某一项修改、删除、编辑数据。

5.编写 ADDGET 和 READGET 过程时，使用了动态数据的技术。

通过双向链表的建立，实现了数据输入的全屏幕编辑，同时也使得变量内存单元的分配更为合理。

6.所有通用过程皆放在一单元(UNIT TESTUNIT)中，这样处理使主程序(TEST.PAS)的编写更加简捷、使用更加灵活方便。

7.当用户用 PASCAL 语言编写程序时可直接使用这些过程。用户也可根据具体要求，将主程序编译成 EXE 文件后嵌入用其它语言编写的程序中使用。

本过程设计中所用到的语句较庞杂，源程序也较为冗长，故很难作详细解释。现将在 IBM 286 或 286 以上机型和各类兼容机上运行通过的源程序附后，供读者参考。运行时可采用 SPDOS、STNCS(网络汉字)、CXDOS 等各类常用汉字操作系统，程序均已通过。

```
UNIT TESTUNIT;(*过程单元*)
{——} INTERFACE {——}
USES Crt;
CONST
str w=8;int w=8;dec w=3;
TYPE
 AddGetPtr=^GetType;
 GetType=Record
 last:AddGetPtr; next:AddGetPtr;
 Getx,Gety:Integer; valueType:Char;
```

```
  value:Pointer;        End;
  Get ST=String[20];
 VAR
  Get x,Get y                 :Byte;
  Finish,GetFinish          :Boolean;
  Top,Middle,BotTom         :AddGetPtr;
  Get LOC,STR LEN,code,ii:Integer;
  Get String                  :Get ST;
 PROCEDURE READGET;
 PROCEDURE AddGet(x,y:Integer;
         Var x ;ch:Char;tf:Boolean);
 {——} IMPLEMENTATION {——}
 function getkey:integer;
  var
   ch:Char; km:Integer;
 Begin
  ch:=ReadKey; km:=ord(ch);
  If km=0 Then Begin
  ch:=ReadKey;km:=256+ord(ch);
  End; GetKey:=km; End;{GetKey}
 Procedure move right(Var Get LOC:Integer);
 Begin
  If Get LOC<STR LEN Then Begin
    Get LOC:=Get LOC+1;
    If Get String[Get LOC]='.' Then
    Get LOC:=Get LOC+1; End; End;
 FUNCTION NOBLANK(S:STRING):STRING;
 BEGIN
  while pos(' ',s)>0 do begin
   delete(s,pos(' ',s),1);end;
   noblank:=s; END;{NOBLANK}
 Function sTon(st:String):Real;
 Var
    xrr:Real;
 Begin st:=noblank(st);
  val(st,xrr,code);sTon:=xrr; End;{ston}
 Procedure recover(Var Get String:Get ST);
 Var
    i w,d w:Integer;
 Begin ii:=1;
  While Get String[ii]=' ' Do ii:=ii+1;
  If (Get LOC<ii) and (ii<=STR LEN-d w) Then
  Begin While ii<=STR LEN-d w Do Begin
    If Get String[ii]<>'.' Then
```

```
    Get String[ii]:=' '; ii:=ii+1;
    End;End;End;{recover}
 procedure colour(a,b:integer);
 begin
  TextColor(a); TextBackground(b); end;
 Procedure GetReal(Var RB:Real;i w,d w:
                 Integer;Var Inkey:Integer);
 Begin
  colour(14,5);
  GET X:=wherex;GET Y:=wherey;
  str(RB:i w:d w,Get String);
  STR LEN:=Length(Get String);
  Write(Get String);
  Get LOC:=1; Finish:=False;
   Repeat
    GotoXY(Get x-1+Get LOC,Get y);
    Inkey:=GetKey;
    Case Inkey Of
     Ord('+'),Ord('-'):Begin (*定义+,-键*)
      Get LOC:=1; recover(Get String);
      Get String[1]:=Chr(Inkey);
      GotoXY(Get x,Get y);
      Write(Get String); Get LOC:=2; End;
     48..57,32 : Begin        (*数字*)
      recover(Get String);
      Get String[Get LOC]:=Chr(Inkey);
      GotoXY(Get x,Get y);
      Write(Get String);
      move right(Get LOC); End;
     46 : If d w>0 Then Begin(*小数点*)
      RB:=sTon(Get String);
      str(RB:i w:d w,Get String);
      GotoXY(Get x,Get y);
      Write(Get String);
      If d w>0 Then
      Get LOC:=i w-d w+1; End;
     331: If Get LOC>1 Then Begin (*LEFT 键*)
      Get LOC:=Get LOC-1;
      If Get String[Get LOC]='.' Then
      Get LOC:=Get LOC-1; End;
     333: move right(Get LOC);  (*RIGHT 键*)
     327:If Get LOC>1 Then Begin (*HOME 键*)
      If d w=0 Then Get LOC:=1 Else Begin
        ii:=1; While Get String[ii]=' ' Do
        ii:=ii+1; If ii=i w-d w Then ii:=ii-1;
      If Get  LOC>ii Then Get  LOC:=ii
      Else Get  LOC:=1; End; End;
  335: Get  LOC:=STR  LEN;        (* END 键 *)
  8 :If Get  LOC>1 Then Begin(* BACKSPACE 键 *)
   If d  w=0 Then Begin Get  LOC:=Get  LOC-1;
   Delete(Get  String,Get  LOC,1);
   Insert(' ',Get  String,i  w);
   End Else Begin
    If (Get  LOC<=i  w-d  w) Then
    Begin Get  LOC:=Get  LOC-1;
     Delete(Get  String,Get  LOC,1);
     Insert(' ',Get  String,i  w-d  w-1);
    End Else If Get  LOC=(i  w-d  w+1) Then
    Begin Get  LOC:=Get  LOC-2;
     Get  String[Get  LOC]:=' '; End Else
     If Get  LOC>(i  w-d  w+1) Then Begin
     Get  LOC:=Get  LOC-1;
     Delete(Get  String,Get  LOC,1);
     Insert(' ',Get  String,i  w);End;End;
     GotoXY(Get  x,Get  y);
     Write(Get  String);End;
   339:Begin                  (* DEL 键 *)
     Delete(Get  String,Get  LOC,1);
     If (d  w=0) Then
     Insert(' ',Get  String,i  w) Else
     If Get  LOC<(i  w-d  w) Then
     Insert(' ',Get  String,i  w-d  w-1)
     Else Insert(' ',Get  String,i  w);
     GotoXY(Get  x,Get  y);
     Write(Get  String); End;
   13,328,336,329,337,27,9: Finish:=True;
  End;Until Finish;(* ENTER,UP,DOWN,ESC 等 *)
  RB:=sTon(Get  String); colour(15,1);
  GotoXY(Get  x,Get  y); Write(RB:i  w:d  w);
  colour(15,1); End;{GetReal}
Procedure Getstr(Var sv:string;s  w:
                Integer;Var Inkey:Integer);
Begin
 colour(14,5);
 GET  X:=wherex;GET  Y:=wherey;
 get  string:=sv;
 STR  LEN:=Length(Get  String);
 Write(Get  String); Get  LOC:=1;
 Finish:=False;
 Repeat
 GotoXY(Get  x-1+Get  LOC,Get  y);
 Inkey:=GetKey;
 Case Inkey Of
  47..57,65..90,95,97..122,32:Begin
   recover(Get  String);
   Get  String[Get  LOC]:=Chr(Inkey);
   GotoXY(Get  x,Get  y);
   Write(Get  String);
   move  right(Get  LOC); End;
  331: If Get  LOC>1 Then
   Get  LOC:=Get  LOC-1;
  333: move  right(Get  LOC);
  327:If Get  LOC>1 Then
   Get  LOC:=1;
  335: Get  LOC:=STR  LEN;
  8 : begin If Get  LOC>1 Then
   begin Get  LOC:=Get  LOC-1;
   delete(get  string,Get  LOC,1);
   insert(' ',get  string,s  w);
   end; gotoxy(get  x,get  y);
   write(get  string); End;
  339:Begin
   Delete(Get  String,Get  LOC,1);
   insert(' ',get  string,s  w);
   gotoxy(get  x,get  y);
   write(get  string); End;
 13,328,336,329,337,27,9: Finish:=True;
 End; Until Finish;
 sv:=get  string; colour(15,1);
 GotoXY(Get  x,Get  y); Write(sv:s  w);
 colour(15,1); End;{Getstr}
Procedure getint(Var i,Inkey:Integer);
Var
 r:Real;
Begin
 R:=I;GetReal(R,4,0,Inkey);I:=TRUNC(R);
End;
Procedure AddGet(x,y:Integer;
                Var x ;ch:Char;tf:Boolean);
Begin
 ch:=upCase(ch);
 If GetFinish Then Begin
 GetFinish:=false;
 New(Top); Middle:=Top;
```

```
    BotTom:=Top; Middle^.last:=NIL;
    Middle^.next:=NIL;
    End Else Begin
   New(Middle^.next);
   Middle:=Middle^.next;
   Middle^.last:=BotTom;
   Middle^.next:=NIL;
   BotTom:=Middle; End;
   Middle^.valueType:=ch;
   With Middle^ Do Begin
   Getx:=x; Gety:=y;
   If tf Then Begin
   Case ch Of
 ′R′:Real(x  ):=0.0;
 ′I′:Integer(x  ):=0;
 ′S′:String(x  ):=′′;
 End; value:=    x  ; End Else
 value:=    x  ; End; End;
    Procedure releaseGet;
    Begin
   Repeat
 Middle:=Top^.next; DisPose(Top);
 Top:=Middle; Until Top=NIL; End;
    Procedure ReadGet;
    Var
   Inkey:Integer;
   maxy:Byte;
    Begin
 If Top=NIL Then exit;
 GetFinish:=false; Middle:=Top;
    Repeat
  With Middle^ Do Begin
  GotoXY(Getx,Gety); colour(15,1);
  Case valueType Of
 ′I′:Write(Integer(value^):4);
 ′R′:Write(Real(value^):int  w:dec  w);
 ′S′:Write(String(value^):str  w);End;End;
  Middle:=Middle^.next;
  Until Middle=NIL; WriteLn;
  maxy:=Hi(WindMax)-Hi(WindMin);
  GotoXY(1,maxy+1);
  Write(′移动亮条,输入数据,按 Esc 键结束′);
  Middle:=Top;
   Repeat
  With Middle^ Do Begin
  GotoXY(Getx,Gety);
   Case valueType Of
   ′R′:GetReal(Real(value ^),int  w,dec  w,Inkey);
   ′I′:Getint(Integer(value^),Inkey);
   ′S′:getstr(string(value^),str  w,inkey);
   End;End;
   Case Inkey Of
    328:If Middle=Top Then (* UP 键 *)
     Middle:=BotTom Else
     Middle:=Middle^.last; (* DOWN 键 *)
    336:If Middle=BotTom Then
     Middle:=Top Else Middle:=Middle^.next;
    13 :If Middle^.next=NIL Then (* ENTER 键 *)
     Middle:=Top Else Middle:=Middle^.next;
    329:Middle:=Top; (* PgUp 键 *)
    337:Middle:=BotTom; (* PgDn 键 *)
    27 :GetFinish:=True; End; (* ESC 键 *)
   Until GetFinish; releaseGet;End;END.
```

```
PROGRAM TEST;{主程序}
USES CRT,TESTUNIT;
VAR
  RRR:REAL;
  III:INTEGER;
  SSS:STRING;
BEGIN
  CLRSCR;
  ADDGET(3,5,RRR,′R′,TRUE);
  ADDGET(3,6,III,′I′,TRUE);
  ADDGET(3,7,SSS,′S′,TRUE);
  READGET;
END.
```

·投稿须知·

1.内容开门见山,文笔简练通顺;

2.图形正规;

3.程序一律上机通过并打印清楚;

4.如有录好的软盘,请随稿附寄.