用 Turbo Pascal 开发节约型文本打印程序

汪 政　（泰安专用汽车制造厂）

一、前言

随着用户对计算机专用纸张消费量日趋增加的同时，由于纸张利用率低所产生的消费现象亦日益严重。当然，原因是多方面的，但主要原因是：社会上流行的文字处理软件存在功能缺陷。其主要表现：1.没有自动测试文件行最大长度的功能。2.不具备双文件同时打印功能。3.不具备单一文件作双排打印的功能。由上述功能缺陷直接产生出：不能帮助用户选择最佳型号的打印纸；不能充分而又合理地利用纸张，提高纸张的利用率，为此，笔者用 Turbo Pascal V4.0 开发了节约型文本打印程序，经个人及同事历经一年的使用，收到了很好的效果。但由于受篇幅的限制，笔者对程序做了精简，并取其核心部分奉献给广大的用户。笔者相信随着用户对本程序的使用（加之对其适当改进、扩充、完善），一定会收到很好的经济效果。

二、使用方法

几个名词解释：

单文件打印——单一文本文件的顺序打印。

双文件打印——两个单一文本文件的并排同时打印。

单文件版式打印——把单一文本文件的内容并排同时打印，象排版一样，取出纸张长度数目的文件行排在纸的左面，又取出纸张长度数目的文件行排在纸的右面，以此类推，直到打印完整个文件。

本程序设计了两个窗口，一个是主窗口，用于用户参数的输入以及程序必要的提示信息的输出。另一个是提示窗口，位于屏幕的最后一行，用于显示功能菜单及执行某一功能时显示正在执行的功能文本。

运行本程序，在提示窗口显示主功能菜单'ESC - Abort F1—单文件打印，F2—双文件打印，F3——单文件版式打印'，此时用户就可以使用 F1 或 F3 功能进行文件打印了，或使用 ESC 键终止程序执行，返回 DOS。值得一提的是：用户使用 F2 或 F3 功能时，程序若测得两文件中的最长行之和或版式打印时的单文件中的最长行的倍数大于 132 列时，则程序总在提示窗口显示'文件行太长，请使用〈F1——单文件打印〉程序打印〈Press any key＞'的提示信息。此时，用户按任意键，待程序返回主菜单后，只能使用 F1 功能了。

三、使用时的注意事项

1.本程序支持文件路径名的输入，路径名由驱动器符、目录名、文件名（包括扩展名）三者组成。若只输入文件名，省略前二者，则程序只在当前驱动器下的当前目录中查找用户文件。

2.必须在中文操作系统中使用本程序。

3.安装打印纸时第一页的顶端应预留 3 至 4 行的空间。

四、结束语

笔者推出本程序，旨在提高专用纸张的利用率把纸张浪费现象控制在最小范围，同时也是为搞文字处理软件的同仁，提供了一份具有使用价值的、可靠的第一手资料。但受笔者学识水平的限制，文中不当之处难免，望同行不吝赐教，给予批评指正。

程序：

```
    program text-print-file(input,output);
uses graph, crt;
const fname='请输入文件名：';
nfile='文件没找到，请再次输入！';
paper80='请安装 80 列打印纸，完毕打任意键开始打印';
paper132='请安装 132 列打印纸，完毕打任意键开始打印';
printing='...正在打印';
promptstr='ESC - - abort F1 - 单文件打印 F2 - 双文件打印 F3 - 单文件版式打印';
megF1='              F1 - 单文件打印                    ';
megF2='              F2 - 双文件打印                    ';
megF3='              F3 - 单文件版式打印                ';
tooline='文件行太长，请使用 < F1 - 单文件打印 >程序打印 <press and key>';
Var graphdriver,graphmode:integer;
    file1,file2,file3:text;
procedure borderwindow;
  begin window(1,1,80,25);textbackground(black);
  window(1,1,80,24);textbackground(black);clrscr;
  end;{end of borderwindow}
  procedure promptwindow;
  begin window(1,1,80,25);textbackground(black);
  window(1,25,80,25);textbackground(lightblue);
  clrscr;
end;{end of promptwindow}
procedure file-name(var f:text;var name:string);
label again;
var str:string; exist:boolean;
begin  repeat  exist:=false;
  again:  write(fname);
    readln(name);
    if name='' then goto again;
    {$i-} assign(f,name); reset(f); {$i+}
    if ioresult<>0 then
    begin writeln(nfile);exist:=false; end
    else exist:=true;
        until exist; end;{end of enter-file-name}
function maxlength-line(var f:text):integer;
var max,max1:integer;ch:char;
begin max:=0;max1:=0;
  while not eof(f) do
    begin
      while not eoln(f) do
      begin read(f,ch); max:=max+1; end; readln(f);
      if max1<max then begin max1:=max;max:=0;end;
    max:=0;
  end;
maxlength-line:=max1;close(f); end;{end function}
procedure one-file;
var ch:char;len:integer;name:string;
begin
  borderwindow; file-name(file1,name);
      len:=maxlength-line(file1);
      if len<=80 then writeln(paper80)
      else writeln(paper132);
      repeat until keypressed; writeln(printing);
```

```
        promptwindow; write(megf1 + ' ':60);
       assign(file1,name); reset(file1);
       assign(file3,'lpt1'); rewrite(file3);
       while not eof(file1) do
        begin
          while not eoln(file1) do
          begin read(file1,ch); write(file3,ch); end;
          readln(file1); writeln(file3);
        end;   close(file1);close(file3); clrscr;
       write(promptstr); end;{end of one-file}
    procedure edit-print;
var pag1,pag2:array[1..60,1..64] of char;
linelengths:integer; pags:integer; name:string;
procedure read-pag(var f:text;mlin,mcol:integer);
var col,lin,count:integer;
begin fillchar(pag1,sizeof(pag1),' '); count:=0;
repeat count:=count+1;
fillchar(pag2,sizeof(pag2),' ');
for lin:=1 to mlin do
begin
     for col:=1 to mcol do
  if not eof(f) then
    if not eoln(f) then read(f,pag2[lin,col])
    else pag2[lin,col]:=' '
  else pag2[lin,col]:=' ';
  if not eof(f) then readln(f);
  end;
if count=1 then pag1:=pag2;
until count=2; end;{end of read-pag}
procedure prt-pag(var f:text;mlin,mcol:integer);
var col,lin:integer;
begin
  0for lin:=1 to mlin do
  begin
   for col:=1 to mcol do
   write(f,pag1[lin,col]); write(f,' ':2);
  for col:=1 to mcol do
  write(f,pag2[lin,col]); writeln(f);
end; end;{end of prt-pag}
begin BORDERWINDOW;
  file-name(file1,name);
  linelengths:=maxlength-line(file1);
  assign(file1,name); reset(file1);
  assign(file3,'lpt1'); rewrite(file3);
  if linelengths<=39 then
   begin writeln(paper80);
   repeat until keypressed;
   writeln(printing); promptwindow;
  write(megf3 + ' ':60);
  while not eof(file1) do
   begin
  read-pag(file1,56,39);
  prt-pag(file3,56,39); writeln(file3);
  end;
  clrscr; write(promptstr);
  end
    else
  if linelengths<65 then
  begin writeln(paper132);
  repeat until keypressed;writeln(printing);
  promptwindow; write(megf3 + ' ':60);
  while not eof(file1) do
   begin
     read-pag(file1,56,64);prt-pag(file3,56,64);
   writeln(file3);
  end;
    clrscr;write(promptstr);
    end
    else
    begin write(tooline);
    repeat until keypressed;
    promptwindow; write(promptstr);
     end;
close(file 1); close(file3); end;{end edit-print}
procedure two-file;
label 10,30;
var i,len,len1,len2:integer;
     ch:char; name1,name2:string;
begin borderwindow;
file-name(file1,name1);file-name(file2,name2);
len1:=maxlength-line(file1);
len2:=maxlength-line(file2);len:=len1+len2+2;
if len<=80 then begin writeln(paper80);end
else if len<=132 then begin writeln(paper132);end
else begin writeln(tooline);goto 30;end;
repeat until keypressed;
assign(file1,name1); reset(file1);
assign(file2,name2); reset(file2);
writeln(printing); promptwindow;
write(megf2 + ' ':60);
assign(file3,'lpt1'); rewrite(file3);
repeat
     for i:=1 to len1 do
  if not eof(file1) then
  if not eoln(file1) then
    begin read(file1,ch); write(file3,ch); end
    else write(file3,' ')
  else write(file3,' ');
  if not eof(file1) then readln(file1);write(file3,' ':3);
  for i:=1 to len2 do
   begin
    if not eof(file2) then
    if not eoln(file2) then
    begin read(file2,ch);write(file3,ch); end
    else goto 10
  else goto 10;
10:end;
  if not eof(file2) then readln(file2);writeln(file3);
until eof(file1) and eof(file2);
30:clrscr;write(promptstr); close(file1);
close(file2);close(file3); end;{end two-file}
procedure Run;
label 10;
const F1=#59; F2=#60; F3=#61; esc=#27;
var ch:char;
begin
repeat clrscr; borderwindow; promptwindow;
write(promptstr);
10: ch:=readkey;
case ch of
F1 :begin one-file; end;
F2 :begin two-file; end;
F3 :begin edit-print; end;
Esc:begin
      window(1,1,80,25);textbackground(black);
      clrscr; halt(0);
    end;
else goto 10;
end;{end case}
until ch=Esc; end;{end of run}
begin {main program} graphdriver:=detect;
initgraph(graphdriver,graphmode,'');
directvideo:=false; run;
end.{end of mainprogram}
```