



---

# Printer software

*H. J. C. Otten*

Als we iets netjes uit een computer op papier willen krijgen moet er een programma komen dat rekening houdt met het aantal regels per pagina, de pagina's nummert, een titel boven een pagina zet en eventueel de regels nummert. Het hier gegeven programma doet dit allemaal en geeft bovendien de mogelijkheid een subtitel per pagina op te nemen en een pagina eerder af te sluiten.

Het programma is in de lijst te zien en is afgedrukt door het programma zelf. De mogelijkheden van het programma worden er meteen door gedemonstreerd. De regelnummers, zoals die zijn afgedrukt, zijn door het printprogramma zelf op verzoek gegenereerd, evenals de titel bovenaan de bladzijde.

VSPASCAL, de IBM-versie van Pascal, is de taal waarin het programma is geschreven. Er zijn geen uitbreidingen gebruikt; alle

constructies en datatypen behoren tot standaard Pascal.

De systeemafhankelijke zaken zitten in de terminal in- en uitvoer. De procedures „termin” en „terminal” openen de files „input” en „output” als interactieve terminalfiles. Andere Pascal-compilers zullen overeenkomstige procedures kennen.

Een speciale procedure „datetime” is gebruikt om datum en tijd van het systeem op te vragen. Als een

dergelijke procedure niet voorhanden is kan om invoer via de terminal worden gevraagd op een soortgelijke manier als de titel.

## **Programma**

Het hoofdprogramma begint met het openen van de files „interfile” en „outfile” na de files „input” en „output” als interactief te hebben geopend. De regels die naar de printer worden gestuurd komen uit „infile”, „outfile” is de file die naar

## Printer software



de printer wijst. Het operating systeem is verantwoordelijk voor het verbinden van „infile” en „outfile” met de af te drukken respectievelijk de printerfiles.

De procedure GetInfo haalt de informatie voor de titel en de datum en tijd binnen en vraagt of wel of geen regelnummers moeten worden gegenereerd.

De hoofdloop van het programma haalt achtereenvolgens een regel van de „infile”, besluit of er een kop bovenaan een pagina moet worden afgedrukt (PrintHeader) en drukt dan de regel zelf af (OutBuffer) met al of niet een regelnummer.

De laatste pagina wordt afgemaakt door EndPage.

De procedure om een regel van de „infile” te lezen, GetBuffer, test of in de regel de commando's „pa” of „st” voorkomen, beginnend op kolom 3. „pa” veroorzaakt in het hoofdprogramma een nieuwe pagina, „st” geeft de subtitel een nieuwe waarde. Achter „st” kan deze subtitel, afgesloten met een punt, worden geplaatst.

Regels met deze twee commando's worden door het programma wel afgedrukt, in het programma (regel 103) is met een commentaar aangegeven hoe dit is te onderdrukken. Voor het afdrukken van

een programma is het wel raadzaam deze opdrachten af te drukken want ze staan als commentaar in de code.

Het programma is nu losstaand maar kan eenvoudig in een ander programma worden opgenomen om als laatste actie nette resultaten uit te printen.

Het programma kan ook als voorbeeld dienen om het algoritme in een andere taal of assembler te implementeren.

```
File : Print.pascal Date : 02/18/82 Time : 11:04:20 Page : 1
-----
1:  program print ;
2:
3:  (* maakt van de file infile een file die naar een printer kan worden
4:  gestuurd met een kop boven de bladzijde, datum en tijd, pagina
5:  nummering etc. *)
6:  Het aantal lijnen per bladzijde is maxline, het aantal blanke regels
7:  onderaan is bottom, het maximum aantal bytes is maxbuf.
8:  De kop , inhoud en aantal regels , wordt in PrintHeader bepaald.
9:  Lijnen uit de infile kunnen van een regelnummer worden voorzien,
10: het aantal bytes op een regel maxbuf wordt daarmee 5 minder.
11: In de tekst kunnen commando's voor een nieuwe pagina en een subtitel
12: worden opgenomen.
13: Een nieuwe pagina wordt geforceerd door vanaf kolom 3 in de tekst op
14: te nemen „pa”.
15: Een nieuwe subtitel wordt gegenereerd door vanaf kolom 3 in de tekst
16: op te nemen „st” gevolgd door de nieuwe subtitel van max 20 kar
17: afgesloten met een punt. *)
18:
19: const
20:     maxbuf = 132 ;
21:     bottom = 6 ;
22:     maxline = 66 ;
23:
24: var
25:     linecnt,pagecnt,
26:     bufptr,numrec,j ;
27:     linemax : integer ;
28:
29:     yesno : char ;
30:
31:     numwant,newpage,
32:     subtitle : boolean ;
33:
34:     buffer : array ( 1 .. maxbuf ) of char ;
35:     title,bufsubtitel : packed array ( 1 .. 20 ) of char ;
36:     date,time : packed array ( 1 .. 8 ) of char ;
37:
38:     infile, outfile : text ;
39:
40: (*..st Get procedures..*)
```

```
File : Print.pascal Date : 02/18/82 Time : 11:04:20 Page : 2
-----
41: (*..pa*)
42:
43: PROCEDURE GetInfo ;
44: (* lees info voor print layout *)
45:
46: begin
47:
48:     datetime(date,time) ; (* haal datum en tijd van systeem *)
49:     writeln('Tijel :');
50:     readln(title) ; (* haal titel van terminal invoer *)
51:     writeln('Regelnummers (y/n) :');
52:     readln(yesno) ; (* vraag wel of geen regelnummers *)
53:
54:     numwant := ( ( yesno = 'y' ) or ( yesno = 'Y' ) ) ;
55:     if numwant then linemax := maxbuf - 5 else linemax := maxbuf ;
56:
57: end ; (* getinfo *)
58:
59:
60: PROCEDURE GetBuffer( var bufptr : integer ) ;
61: (* haal record van infile op *)
62:
63: var i : integer ;
64:
65: begin
66:
67:     bufptr := 0 ;
68:     while (not eof(infile)) and ( bufptr < linemax ) do
69:     begin
70:         bufptr := bufptr + 1 ;
71:         read(infile,buffer ( , bufptr )) ;
72:     end ;
73:
74:     numrec := numrec + 1 ;
75:     newpage := (buffer(.3) = ',') and (buffer(.4) = 'p') and
76:               ( buffer(.5) = 'a' ) ;
77:
78:     subtitle := (buffer(.3) = ',') and (buffer(.4) = 's') and
79:               ( buffer(.5) = 't' ) ;
80:     if subtitle then
81:     begin
82:         for i := 1 to 20 do bufsubtitel(i) := ' ' ;
83:         i := 7 ;
84:         repeat
85:             bufsubtitel( (i - 6) ) := buffer(i) ;
86:             i := i + 1 ;
87:         until ( ( i = 20 ) or ( buffer(i) = ',' ) ) ;
88:         end ;
89:         readln(infile) ;
90:
```

```
File : Print.pascal Date : 02/18/82 Time : 11:04:20 Page : 3
-----
91: end ; (* GetBuffer *)
92:
93: (*..st Print Procedures..*)
94:
95: (*..pa*)
96:
97: PROCEDURE OutBuffer( bufptr : integer ) ;
98:
99: var i : integer ;
100:
101: begin
102:     if numwant then
103:         write(outfile,numrec:4,' ',' ');
104:     (* if (not newpage) and (not subtitle) then *)
105:         for i := 1 to bufptr do
106:             write(outfile,buffer(i)) ;
107:             writeln(outfile,' ');
108:         end ; (* OutBuffer *)
109:
110: PROCEDURE PrintHeader( pagecnt : integer; var linecnt : integer ) ;
111:
112: (* print de kop boven de bladzijde *)
113:
114: var i : integer ;
115:
116: begin
117:     write(outfile,'File : ');
118:     for i := 1 to 20 do
119:         write(outfile,title(i)) ;
120:     writeln(outfile,' Date : ',date,' Time : ',time);
121:     writeln(outfile,' Page ',pagecnt:4) ;
122:     write(outfile,' ');
123:     for i := 1 to 20 do
124:         write(outfile,bufsubtitel(i)) ;
125:     writeln(outfile,' ');
126:     write(outfile,'-----');
127:     writeln(outfile,'-----');
128:     writeln(outfile,' ');
129:
130:     linecnt := 3 (* linecnt een minder dan lijnen in header *)
131:
132: end ; (* PrintHeader *)
133:
134: PROCEDURE EndPage(var linecnt : integer ) ;
135:
136: (* maak laatste pagina vol met linefeeds *)
137:     printer met forefeed kunnen hier beter forafeed van maken *)
138:
139: begin
140:
141:     while linecnt < (maxline - 1) do
142:     begin
143:         linecnt := linecnt + 1 ;
144:         writeln(outfile,' ');
145:     end
146:     end ; (* EndPage *)
147:
148: (*..st Main Program..*)
```

```
File : Print.pascal Date : 02/18/82 Time : 11:04:20 Page : 4
-----
149: (*..pa..*)
150:
151: begin (* main *)
152:
153:     termin(input) ; (* interaktief terminal i/o *)
154:     termin(output) ;
155:     reset(infile) ; (* lees infile van begin *)
156:     rewrite(outfile) ; (* outfile is nieuwe file *)
157:
158:     GetInfo ;
159:
160:     pagecnt := 1, numrec := 0 ;
161:     for i := 1 to 20 do bufsubtitel(i) := ' ' ;
162:     PrintHeader(pagecnt,linecnt) ;
163:     while not eof(infile) do
164:     begin
165:         GetBuffer(bufptr) ;
166:         linecnt := linecnt + 1 ;
167:         if ((linecnt = (maxline - bottom - 1)) or newpage) then
168:         begin
169:             pagecnt := pagecnt + 1 ;
170:             for i := 1 to (maxline - linecnt) do (* forfeed *)
171:                 writeln(outfile,' ');
172:             PrintHeader(pagecnt,linecnt) ;
173:             end ;
174:             OutBuffer(bufptr) ;
175:             end ;
176:         EndPage(linecnt) ; (* Laatste pagina *)
177:     end ;
178: end.
```