Chapter 8
Page Break Logic

Objectives

Upon completion of this chapter you will be able to:

- Describe and code page break logic, including...
- Produce a form feed to force the printer to advance to the top of a page.

Introduction

In chapter two we saw how to produce headings for our reports. We said at that time that we would put off our discussion of page counts because such counts required the use of arithmetic, which we had not yet covered. Up to now, all of our reports have been limited to a single page of output, but having discussed packed decimal arithmetic, we are now ready to produce "true" report headings. In this chapter we discuss page break logic, including how to produce headings with page counts and how to force the printer to advance to the top of a new page.

* * * * * * * * * * * * * * * * * * * *

In this chapter we will modify `TEACH2C.MLC` from chapter two. That program produced a list of all teachers at the Small Town Community College. This list included report and column headings, but did not include page break logic. We will modify this program to include headings *with page numbers*. In order to demonstrate page break logic while still working with a dataset of manageable size, we will assume that a maximum of three detail lines will be printed per page. Our new output will appear as follows:

```
            LIST OF TEACHERS       Page 001

ID#        Name         Degr  Ten  Phone
---   ---------------   ----  ---  -----
732   BENSON, E.T.      PHD    N   5156
218   HINCKLEY, G.B.    MBA    N   5509
854   KIMBALL, S.W.     PHD    Y   5594

            LIST OF TEACHERS       Page 002

ID#        Name         Degr  Ten  Phone
---   ---------------   ----  ---  -----
626   YOUNG, B.         MBA    Y   5664
574   SMITH, J.         MS     Y   5320
```

There are five steps to producing page breaks: these steps are shown in the flow chart on the next page. These steps require that we add several fields to our program:

```
PGS      DC    PL2'0'              Nbr of pages printed.
LNS      DC    PL2'3'              Lines printed on this page.
MAXLNS   DC    PL2'3'              Max nbr lines per page.
*                                  My line counts exclude hdgs.
```

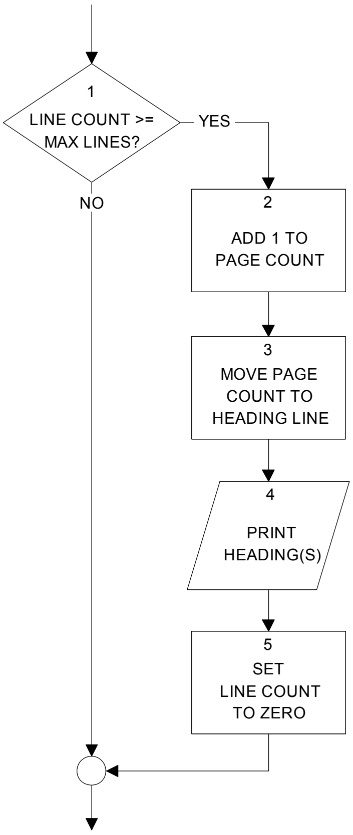
Step 1: Before printing a line, check the line count to see if it is greater than or equal to the maximum number of lines allowed per page. If so, then...

Step 2: Add one to the page count,

Step 3: Move the page count to one of the headings,

Step 4: Print the headings, and

Step 5: Reinitialize the line count to zero.



Note that the page count (PGS) is initialized to zero. The maximum number of lines per page (MAXLNS) is initialized to the desired maximum. Since we are dealing with small datasets, and for demonstration purposes, we will use a maximum of three. In the real world you would probably use 50-55. Finally, the line count (LNS) is initialized to the maximum value as well. The reason for doing so may not be readily apparent: we do this so that the first time we check the line count, the program will think the current page is full, and will therefore force a page break.

My preference is to count detail lines only: headings are not included in LNS and MAXLNS. I say, "Headings are free." You (or your company) may prefer to do otherwise.

The CHKLNS Routine

In our current program I will put the line count check in the PROCESS section, after FORMAT and before WRITE. I will use a separate routine in order to minimize the number of branches (and labels) within PROCESS:

```
PROCESS  EQU   *
         ST    R10,SVPROC
         BAL   R10,FORMAT
         BAL   R10,CHKLNS
         BAL   R10,WRITE
         BAL   R10,READ
         L     R10,SVPROC
         BR    R10
```

where CHKLNS is defined as...

```
****************************************************************
*        CHKLNS - Check lines printed. Full page?              *
****************************************************************
CHKLNS   EQU   *
         ST    R10,SVCHKLNS
         CP    LNS,MAXLNS
         BL    CHKLNSX
         BAL   R10,HDGS
CHKLNSX  EQU   *
         L     R10,SVCHKLNS
         BR    R10
```

Adding CHKLNS necessitates defining a fullword to save the return address:

```
SVCHKLNS DC    F'0'               CHKLNS
```

Including the Page Number in the Headings Definition

Next, we change the definition of the first heading line, HD1, to include the page number:

```
HD1      DS    0CL62
         DC    CL40'            LIST OF TEACHERS       Page '
HDPGS    DC    CL3' '
         DC    CL17' '
         DC    XL2'0D25'
```

There are no changes necessary for the other heading lines.

The HDGS Routine

Finally, steps 2 through 5 are done within the HDGS routine:

```
HDGS     EQU   *
         ST    R10,SVHDGS
         AP    PGS,=P'1'          Add 1 to page count
         UNPK  HDPGS,PGS          Move page count to headings
         MVZ   HDPGS+L'HDPGS-1(1),=X'F0'    Remove sign
         PUT   REPORT,FORMFEED    PC/370 ONLY
         PUT   REPORT,HD1
         PUT   REPORT,HD2
         PUT   REPORT,HD3
         PUT   REPORT,HD4
         ZAP   LNS,=P'0'          Reset line count to zero
         L     R10,SVHDGS
         BR    R10
```

The FORMFEED Line

Note the addition of a new PUT statement: PUT REPORT,FORMFEED. We need some way to indicate that the printer should advance to the top of the next page. An easy way to do this is to define a new heading line which is:

- the same length as the other heading lines (62), and
- consists of all blanks,
- except the first character, which is a form feed character.

This is done as follows:

```
FORMFEED DS    0CL62               PC/370 only
         DC    X'0C'               EBCDIC formfeed
         DC    CL59' '
         DC    X'0D25'             EBCDIC CR/LF
```

A X'0C' is an EBCDIC formfeed. (Coincidentally, it is also an ASCII formfeed.) Most PC printers will advance to top of page when they receive this character. Note this is *not* how one would force a skip to top of page in the mainframe world: that is discussed in the appendix.

Of course, the line count must be incremented after each write:

```
WRITE    EQU   *
         ST    R10,SVWRITE
         PUT   REPORT,OREC         Write report line
         AP    LNS,=P'1'
         L     R10,SVWRITE
         BR    R10
```

Sample Program Solution

The complete program, TEACH8A.MLC, follows.

```
         PRINT NOGEN
****************************************************************
*        FILENAME:  TEACH8A.MLC                                *
*        AUTHOR  :  Bill Qualls                                *
*        SYSTEM  :  PC/370 R4.2                                *
*        REMARKS :  This is a revision of TEACH2C.MLC.         *
*                   Produce list of teachers, with headings.   *
*                   Introduce page break logic.                *
****************************************************************
         START 0
         REGS
BEGIN    BEGIN
```

(continued)

```
         WTO   'TEACH8A ... Begin execution'
         BAL   R10,SETUP
MAIN     EQU   *
         CLI   EOFSW,C'Y'
         BE    EOJ
         BAL   R10,PROCESS
         B     MAIN
EOJ      EQU   *
         BAL   R10,WRAPUP
         WTO   'TEACH8A ... Normal end of program'
         RETURN
****************************************************************
*        SETUP - Those things which happen one time only,      *
*                before any records are processed.             *
****************************************************************
SETUP    EQU   *
         ST    R10,SVSETUP
         OI    TEACHERS+10,X'08'  PC/370 ONLY - Convert all
*                                 input from ASCII to EBCDIC
         OI    REPORT+10,X'08'    PC/370 ONLY - Convert all
*                                 output from EBCDIC to ASCII
         OPEN  TEACHERS
         OPEN  REPORT
         BAL   R10,READ           Note: Removed BAL R10,HDGS
         L     R10,SVSETUP
         BR    R10
****************************************************************
*        HDGS - Print headings.                                *
****************************************************************
HDGS     EQU   *
         ST    R10,SVHDGS
         AP    PGS,=P'1'          Add 1 to page count
         UNPK  HDPGS,PGS          Move page count to headings
         MVZ   HDPGS+L'HDPGS-1(1),=X'F0'    Remove sign
         PUT   REPORT,FORMFEED    PC/370 ONLY
         PUT   REPORT,HD1
         PUT   REPORT,HD2
         PUT   REPORT,HD3
         PUT   REPORT,HD4
         ZAP   LNS,=P'0'          Reset line count to zero
         L     R10,SVHDGS
         BR    R10
****************************************************************
*        PROCESS - Those things which happen once per record.  *
****************************************************************
PROCESS  EQU   *
         ST    R10,SVPROC
         BAL   R10,FORMAT
         BAL   R10,CHKLNS
         BAL   R10,WRITE
         BAL   R10,READ
         L     R10,SVPROC
         BR    R10
****************************************************************
*        READ - Read a record.                                 *
****************************************************************
READ     EQU   *
         ST    R10,SVREAD
         GET   TEACHERS,IREC      Read a single teacher record
         B     READX
```

(continued)

```
ATEND    EQU   *
         MVI   EOFSW,C'Y'
READX    EQU   *
         L     R10,SVREAD
         BR    R10
****************************************************************
*        FORMAT - Format a single detail line.                 *
****************************************************************
FORMAT   EQU   *
         ST    R10,SVFORM
         MVC   OTID,ITID          Move teacher ID Nbr to output
         MVC   OTNAME,ITNAME      Move teacher Name to output
         MVC   OTDEG,ITDEG        Move highest degree to output
         MVC   OTTEN,ITTEN        Move tenure to output
         MVC   OTPHONE,ITPHONE    Move phone nbr to output
         MVC   OCRLF,WCRLF        PC/370 ONLY - end line w/ CR/LF
         L     R10,SVFORM
         BR    R10
****************************************************************
*        CHKLNS - Check lines printed. Full page?              *
****************************************************************
CHKLNS   EQU   *
         ST    R10,SVCHKLNS
         CP    LNS,MAXLNS
         BL    CHKLNSX
         BAL   R10,HDGS
CHKLNSX  EQU   *
         L     R10,SVCHKLNS
         BR    R10
****************************************************************
*        WRITE - Write a single detail line.                   *
****************************************************************
WRITE    EQU   *
         ST    R10,SVWRITE
         PUT   REPORT,OREC        Write report line
         AP    LNS,=P'1'
         L     R10,SVWRITE
         BR    R10
****************************************************************
*        WRAPUP - Those things which happen one time only,     *
*                 after all records have been processed.       *
****************************************************************
WRAPUP   EQU   *
         ST    R10,SVWRAP
         CLOSE TEACHERS
         CLOSE REPORT
         WTO   'TEACH8A ... Teacher list on REPORT.TXT'
         L     R10,SVWRAP
         BR    R10
****************************************************************
*        Literals, if any, will go here                        *
****************************************************************
         LTORG
****************************************************************
*        File definitions                                      *
****************************************************************
TEACHERS DCB   LRECL=29,RECFM=F,MACRF=G,EODAD=ATEND,
               DDNAME='TEACHER.DAT'
REPORT   DCB   LRECL=62,RECFM=F,MACRF=P,
               DDNAME='REPORT.TXT'
```

(continued)

```
*****************************************************************
*        RETURN ADDRESSES                                       *
*****************************************************************
SVSETUP  DC    F'0'              SETUP
SVHDGS   DC    F'0'              HDGS
SVPROC   DC    F'0'              PROCESS
SVREAD   DC    F'0'              READ
SVFORM   DC    F'0'              FORMAT
SVWRITE  DC    F'0'              WRITE
SVWRAP   DC    F'0'              WRAPUP
SVCHKLNS DC    F'0'              CHKLNS
*****************************************************************
*        Miscellaneous field definitions                        *
*****************************************************************
WCRLF    DC    X'0D25'           PC/370 ONLY - EBCDIC CR/LF
EOFSW    DC    CL1'N'            End of file? (Y/N)
PGS      DC    PL2'0'            Nbr of pages printed.
LNS      DC    PL2'3'            Lines printed on this page.
MAXLNS   DC    PL2'3'            Max nbr lines per page.
*                                My line counts exclude hdgs.
*****************************************************************
*        Input record definition                                *
*****************************************************************
IREC     DS    0CL29             Teacher record
ITID     DS    CL3               Teacher ID nbr
ITNAME   DS    CL15              Teacher name
ITDEG    DS    CL4               Highest degree
ITTEN    DS    CL1               Tenured?
ITPHONE  DS    CL4               Phone nbr
ITCRLF   DS    CL2               PC/370 only - CR/LF
*****************************************************************
*        Output (line) definition                               *
*****************************************************************
OREC     DS    0CL62
OTID     DS    CL3               Teacher ID nbr
         DC    CL3' '
OTNAME   DS    CL15              Teacher name
         DC    CL3' '
OTDEG    DS    CL4               Highest degree
         DC    CL3' '
OTTEN    DS    CL1               Tenured?
         DC    CL3' '
OTPHONE  DS    CL4               Phone nbr
         DC    CL21' '
OCRLF    DS    CL2               PC/370 only - CR/LF
*
*        Headings definitions
*
FORMFEED DS    0CL62             PC/370 only
         DC    X'0C'             EBCDIC formfeed
         DC    CL59' '
         DC    X'0D25'           EBCDIC CR/LF
HD1      DS    0CL62
         DC    CL40'            LIST OF TEACHERS       Page '
HDPGS    DC    CL3' '
         DC    CL17' '
         DC    XL2'0D25'
HD2      DS    0CL62
         DC    CL60' '
         DC    XL2'0D25'
```

(continued)

```
HD3      DS    0CL62
         DC    CL40'ID#        Name         Degr  Ten  Phone'
         DC    CL20' '
         DC    XL2'0D25'
HD4      DS    0CL62
         DC    CL40'---   ---------------   ----  ---  -----'
         DC    CL20' '
         DC    XL2'0D25'
         END   BEGIN
```

The execution and output are shown as well. `WTO` messages are used, as before, to indicate the progress of the program and to remind the user where to find the output.

DOS' `TYPE` command is used to view the output. Note that a formfeed character, when shown with the `TYPE` command, appears as a "female" symbol. The `COPY` command can be used to print the output. (`PRN` is a DOS device name meaning printer.)

```
A:\MIN>teach8a
TEACH8A ... Begin execution
TEACH8A ... Teacher list on REPORT.TXT
TEACH8A ... Normal end of program

A:\MIN>type report.txt
o
           LIST OF TEACHERS        Page 001

ID#        Name         Degr  Ten  Phone
---   ---------------   ----  ---  -----
732   BENSON, E.T.      PHD    N   5156
218   HINCKLEY, G.B.    MBA    N   5509
854   KIMBALL, S.W.     PHD    Y   5594
o
           LIST OF TEACHERS        Page 002

ID#        Name         Degr  Ten  Phone
---   ---------------   ----  ---  -----
626   YOUNG, B.         MBA    Y   5664
574   SMITH, J.         MS     Y   5320

A:\MIN>copy report.txt prn
        1 file(s) copied

A:\MIN>
```

Saving Paper

To save paper when testing, consider the following change:

```
FORMFEED DS    0CL62              PC/370 only
*        DC    X'0C'              EBCDIC formfeed
*        DC    CL59' '
         DC    60C'_'             For testing...
         DC    X'0D25'            EBCDIC CR/LF
```

Exercises

1. True or false.

T F a. The page count field (PGS) should be initialized to 1.
T F b. The page count field (PGS) is incremented prior to moving it to the corresponding field in the headings.
T F c. The page count field in the headings should be defined as a pack field.
T F d. The form feed symbol is ASCII 12.
T F e. The form feed symbol will appear as the female sign when viewed on the screen.
T F f. EBCDIC 12 is recognized as a form feed on most mainframe printers.
T F g. In the HDGS routine shown in this chapter, the line count (LNS) was set to zero because "headings are free".
T F h. DOS' COPY command can be used to send the report to the printer.
T F i. The FORMFEED line must be the same length as any other heading line.
T F j. The CHKLNS routine is invoked for the first time in the SETUP routine.
T F k. The line count (LNS) should be initialized to zero.
T F l. CHKLNS will invoke HDGS if LNS is less than MAXLNS.
T F m. LNS is incremented after PUT in the WRITE routine.

2. Show the changes necessary to the sample program, TEACH8A.MLC, if the report was is expected to be over 999 pages long.

3. Show the changes necessary to the sample program, TEACH8A.MLC, if, unlike the author, you prefer headings be included in the line count.

4. Many of the original PC printers did not recognize ASCII 12 as a form feed character. Most of these printers did not even have a form feed button. To advance to the top of page on these printers, it is necessary to write a sufficient number of blank lines, such as 66-LNS.

   Modify the HDGS routine to use this method instead of FORMFEED as shown. To save paper, you can assume each page is 22 lines long, so you can simulate three pages of output (and therefore three sets of headings) on a single page.

5. Exercise 6 of chapter 2 asked you to produce a formatted list of the records in the student file. Modify that program so the first heading line will include a page number. Print a maximum of five detail lines per page. In addition, print all detail lines double spaced.

Execises

6. Exercise 7 of chapter 2 asked you to produce a formatted list of the records in the grade file. Modify that program so the first heading line will include a page number. Print a maximum of five detail lines per page. In addition, print all detail lines double spaced.

7. Exercise 8 of chapter 2 asked you to produce a formatted list of the records in the course file. Modify that program so the first heading line will include a page number. Print a maximum of five detail lines per page. In addition, print all detail lines double spaced.

8. Exercise 16 of chapter 3 asked you to produce a formatted list of all grades for English classes for the 1992-93 school year (semesters `F92` and `W93` only.) Modify that program so the first heading line will include a page number. Print a maximum of four detail lines per page. In addition, print all detail lines double spaced.

9. It is not uncommon to include footers, as well as headers, on a report. Modify one of the above exercises to print a footer, with the appropriate page number, at the bottom of each page. This footer should appear in addition to the header.