Chapter 11
More Than One Input File

Objectives

Upon completion of this chapter you will be able to:
- Sort data files as necessary in order to process them using matching logic, and
- Design and code a program using matching logic to process two input files simultaneously.

Introduction

All of the programs which we have seen up to this point have had only one input file and one output file. The next two chapters will look at the logic required to process two input files. When processing two input files, we can usually think of one of the files as a master file and the other as a transaction file. The nature of the transactions may or may not necessitate some change to the corresponding record on the master file. In this chapter we will process two files, but we will not make changes to the master file: we simply produce a report making use of the data contained in both files. In the next chapter we will apply changes to the master file as a result of the transactions processed.

* *

The report which we will produce in this chapter is a list of teaching assignments for the Fall 1992 semester. The report will appear as follows:

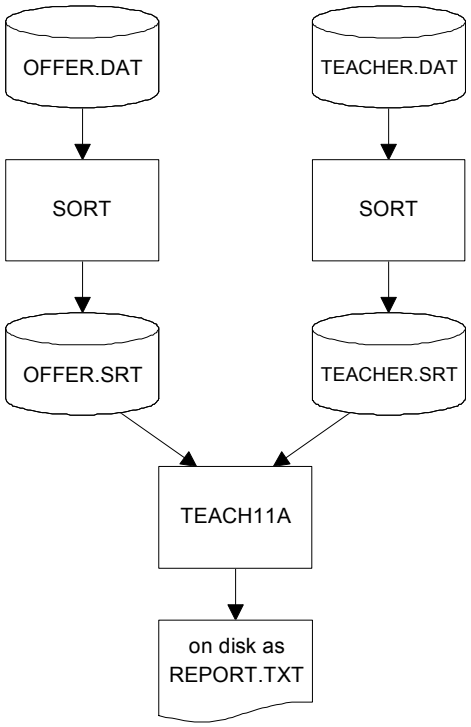
```
         1         2         3         4         5         6
12345678901234567890123456789012345678901234567890123456789012345678901234567890
          F92 Teaching Assignments      PageBZZ9

TID       Name          Course  Section  Room
---  ---------------    ------  -------  ----
XXX  XXXXXXXXXXXXXXX    XXXXX      X     XXXX
                         XXXXX      X     XXXX

XXX  XXXXXXXXXXXXXXX    XXXXX      X     XXXX
                         XXXXX      X     XXXX
```

Observe that we will list the teacher ID and name for each teacher teaching a course in the Fall 1992 semester, along with the course ID, section number and room number for each course taught. The teacher ID, course ID, section number and room number will come from the OFFER file, *but the teacher name is found on the TEACHER file only!* We can think of the TEACHER file as the master, and the OFFER file as the transactions.

What we need is some way to find the teacher name based on the teacher ID. Of course, this is a trivial task if you are working with a relational database system, but we are not. We must code this matching logic ourselves. Demonstrating that logic is the purpose of this chapter. Note: we do not introduce any new assembler instructions in this chapter.

Sorting the Input Files

We have said the teacher ID, course ID, section number and room number will come from the OFFER file, and the teacher name will come from the TEACHER file. To print the teacher name based on the teacher ID, we will match the two files based on the teacher ID. The teacher ID is referred to as the **key** field. In order to do so, both files must be in teacher ID sequence; that is, they must be sorted. DOS' SORT command is crude but will suffice. (In the mainframe world you would use a much more sophisticated sort package, such as SyncSort.) To sort these files, we type the following at the DOS prompt:




```
sort /+10 < offer.dat > offer.srt
sort /+1 < teacher.dat > teacher.srt
```

These instructions will sort the .DAT files, creating the .SRT files. We can then use DOS' TYPE command to view these files. For example:

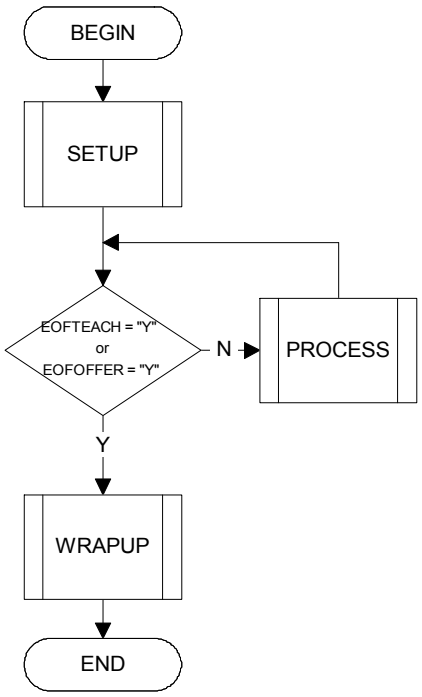
```
A:\MIN>type offer.srt
W92MA1071218A2
F92AC1011218B1
F92BU1011218B1
W92PE1511574GYM
F92PE1511574GYM
W93PE1511574GYM
F92MA1011626A2
F92MA1012626A2
W93MA1011626A2
W93MA1071626A3
W92EG1021732A1
F92EG1011732A1
F92EG1012732A1
W93EG1021854A1
```

```
A:\MIN>type teacher.srt
218HINCKLEY, G.B. MBA N5509
574SMITH, J.      MS  Y5320
626YOUNG, B.      MBA Y5664
732BENSON, E.T.   PHD N5156
854KIMBALL, S.W.  PHD Y5594
```

These .SRT files will then be included in the DDNAME parameter of the DCBs in the program:

```
TEACHERS DCB    LRECL=29,RECFM=F,MACRF=G,EODAD=ATENDTCH,
                DDNAME='TEACHER.SRT'
OFFER    DCB    LRECL=18,RECFM=F,MACRF=G,EODAD=ATENDOFF,
                DDNAME='OFFER.SRT'
```

The Mainline Structure



The mainline structure of the program is really not all that different from what we've seen before: we continue to process records until *either* file is at EOF. This type of check will require two end-of-file switches:

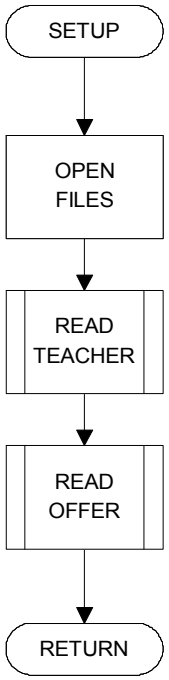
```
EOFTEACH DC    CL1'N'
EOFOFFER DC    CL1'N'
```

Note that it is possible (and in fact likely) that both files will not reach EOF at the same time. (This condition is handled in the `WRAPUP` routine.)

The mainline code is repeated here. It is not particularly "clean" code when compared to the flowchart, but this is because BAL does not have an `IF..OR` structure.

```
         BAL   R10,SETUP
MAIN     EQU   *
         CLI   EOFTEACH,C'Y'
         BE    EOJ
         CLI   EOFOFFER,C'Y'
         BE    EOJ
         BAL   R10,PROCESS
         B     MAIN
EOJ      EQU   *
         BAL   R10,WRAPUP
```

The `SETUP` Routine

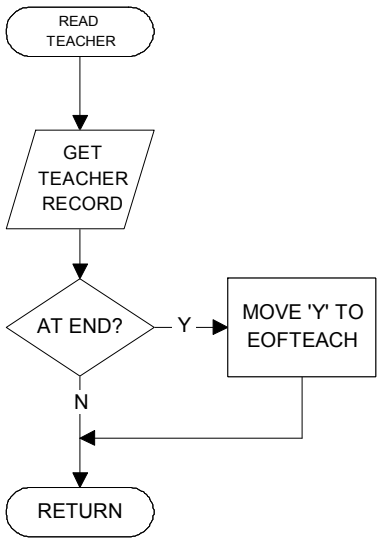


Within the `SETUP` routine, we open all files and read the first record from each input file:

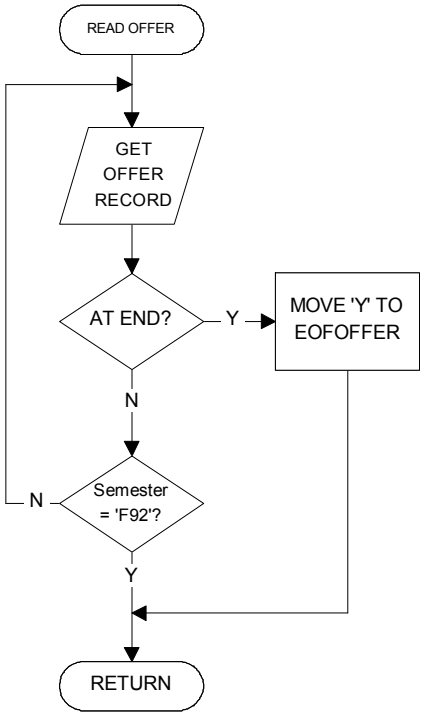
```
SETUP    EQU   *
         ST    R10,SVSETUP
         OI    TEACHERS+10,X'08'
         OI    OFFER+10,X'08'
         OI    REPORT+10,X'08'
         OPEN  TEACHERS
         OPEN  OFFER
         OPEN  REPORT
         BAL   R10,READTCH
         BAL   R10,READOFF
         L     R10,SVSETUP
         BR    R10
```

The READ Routines

In every program we have seen so far, we have had one read routine, one end-of-file switch, one input DCB, etc. In this program we will, of course, have two of each. The read routines are shown next. They are very similar with the exception that since we are producing a class list for semester F92 only, we will skip any OFFER records not from that semester:




```
READTCH  EQU   *
         ST    R10,SVREADT
         GET   TEACHERS,TREC
         B     READTX
ATENDTCH EQU   *
         MVI   EOFTEACH,C'Y'
READTX   EQU   *
         L     R10,SVREADT
         BR    R10
```

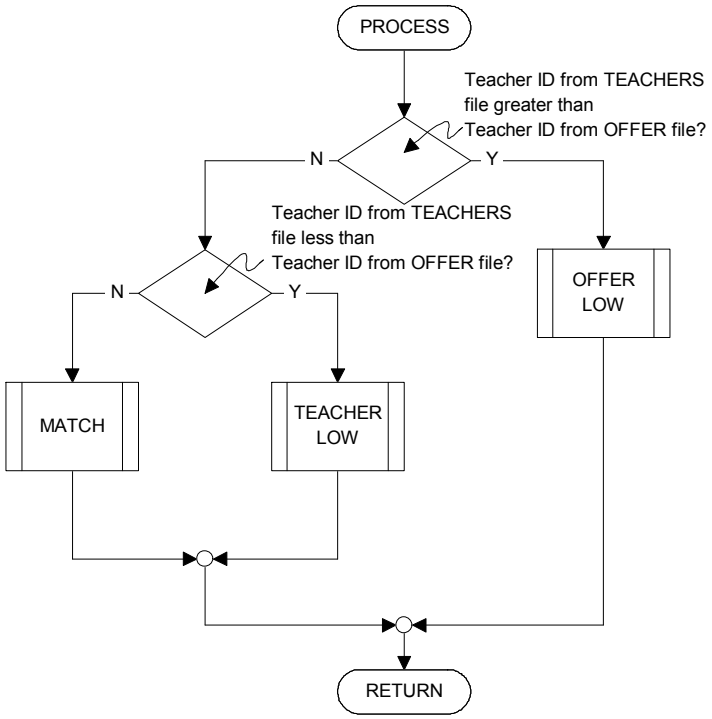



```
READOFF  EQU   *
         ST    R10,SVREADO
READOFF2 EQU   *
         GET   OFFER,OREC
         CLC   OSEM,=CL3'F92'
         BNE   READOFF2
         B     READOX
ATENDOFF EQU   *
         MVI   EOFOFFER,C'Y'
READOX   EQU   *
         L     R10,SVREADO
         BR    R10
```

The PROCESS Routine

At the time we perform the PROCESS routine, we have one record from each file. We then compare the teacher ID from the two files. There are three possible conditions as a result of this compare:

- the ID on the TEACHER file is greater than the ID on the OFFER file,
- the ID on the TEACHER file is less than the ID on the OFFER file, or
- the ID on the TEACHER file is equal to the ID on the OFFER file.



The first condition would indicate that we have an OFFER record with a teacher ID which is *not* found on the TEACHER file. Clearly, this would indicate an error condition.

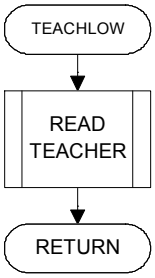
The second condition would indicate that we have a TEACHER record with a teacher ID which is *not* found on the OFFER file. This would indicate that this particular teacher is not teaching any courses during the F92 semester. (We will simply skip these teachers.)

The third condition would indicate that we have an OFFER record with a teacher ID which *is* found on the TEACHER file. This indicates that this teacher and course should be listed on the report.

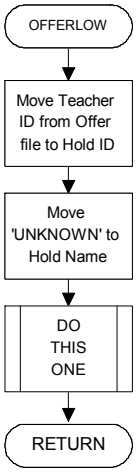
Each of these conditions will be handled in a separate routine.

```
PROCESS  EQU   *
         ST    R10,SVPROC
         CLC   TTID,OTID
         BH    PROC2
         BL    PROC3
         BAL   R10,MATCH
         B     PROCESSX
PROC2    EQU   *
         BAL   R10,OFFERLOW
         B     PROCESSX
PROC3    EQU   *
         BAL   R10,TEACHLOW
PROCESSX EQU   *
         L     R10,SVPROC
         BR    R10
```

The "Teacher File Low" condition indicates we have a teacher who is *not* teaching any classes this semester. No processing is required; just get the next TEACHER record.

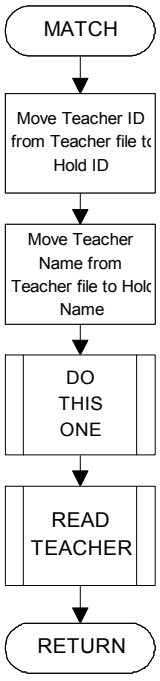



```
TEACHLOW EQU   *
         ST    R10,SVTCHLOW
         BAL   R10,READTCH
         L     R10,SVTCHLOW
         BR    R10
```



The "Offer File Low" condition indicates we have a course assigned to a teacher *not* on the TEACHER file. We will go ahead and show this course anyway, and show the teacher name as "Unknown".

```
OFFERLOW EQU   *
         ST    R10,SVOFFLOW
         MVC   HOLDTID,OTID
         MVC   HOLDNAME,=CL15'Unknown'
         BAL   R10,DOTHIS1
         L     R10,SVOFFLOW
         BR    R10
```

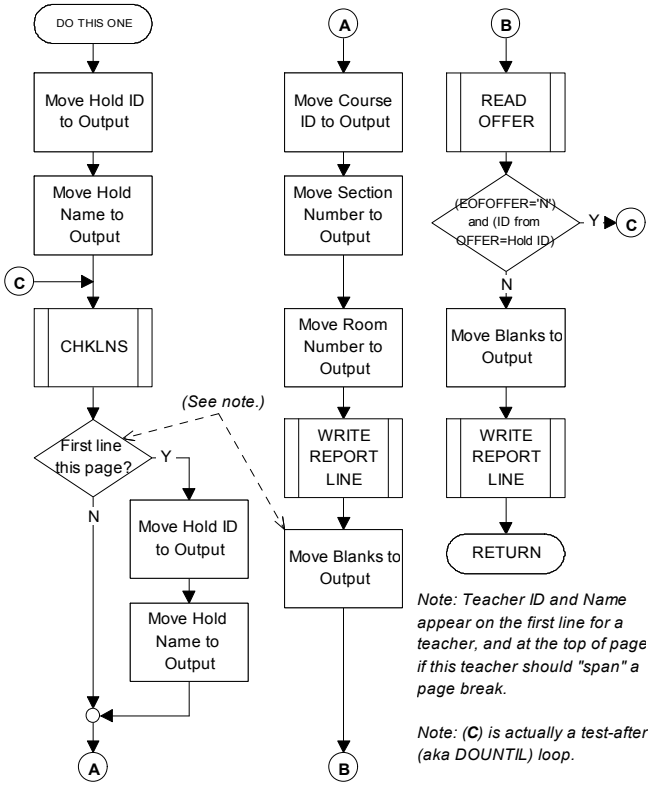


The "Match" condition indicates that we have a course assigned to a teacher who *is* on the TEACHER file. Process all course records for this teacher.

```
MATCH    EQU   *
         ST    R10,SVMATCH
         MVC   HOLDTID,TTID
         MVC   HOLDNAME,TTNAME
         BAL   R10,DOTHIS1
         BAL   R10,READTCH
         L     R10,SVMATCH
         BR    R10
```

We now process all course offerings for a single teacher ID. In the OFFERLOW and MATCH routines we moved the teacher ID and teacher name (or "Unknown") to a hold area. These hold area fields serve two purposes. First, the hold ID is used to check for a change in teacher ID. Second, these are the fields which will be moved to the teacher ID and name on the report. We will show the teacher ID and name on the first line for this teacher only, unless the course listing for this teacher should happen to "span" a page break, in which case the teacher ID and name will be shown on the first line of the next page as well.

```
DOTHIS1  EQU   *
         ST    R10,SVTHIS1
         MVC   RREC,BLANKS
         MVC   RTID,HOLDTID
         MVC   RTNAME,HOLDNAME
DOTHIS2  EQU   *
         BAL   R10,CHKLNS
         CP    LNS,=P'0'
         BNE   DOTHIS3
         MVC   RTID,HOLDTID
         MVC   RTNAME,HOLDNAME
DOTHIS3  EQU   *
         MVC   RCID,OCID
         MVC   RSECT,OSECT
         MVC   RROOM,OROOM
         BAL   R10,WRITE
         MVC   RREC,BLANKS
         BAL   R10,READOFF
         CLI   EOFOFFER,C'Y'
         BE    DOTHIS4
         CLC   HOLDTID,OTID
         BE    DOTHIS2
DOTHIS4  EQU   *
         MVC   RREC,BLANKS
         BAL   R10,WRITE
         L     R10,SVTHIS1
         BR    R10
```

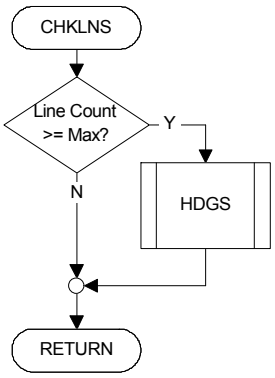


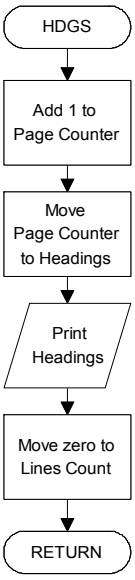
A check is made for a full page prior to each record being printed (see CHKLNS). In the process of formatting a report line, we check to see if the line count (LNS) is zero. If so, then this is the first line on a page: show the teacher ID and name again. After each line is formatted and printed, the next course offerings record is read. The process is repeated until the course offerings file is at EOF or the teacher ID has changed (compared to the hold ID).

The CHKLNS, HDGS, and WRITE Routines

The CHKLNS, HDGS, and WRITE routines are the same as we have seen before:

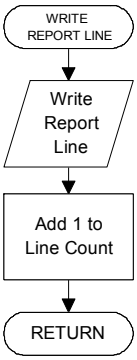
```
CHKLNS   EQU   *
         ST    R10,SVCHKLNS
         CP    LNS,MAXLNS
         BL    CHKLNSX
         BAL   R10,HDGS
CHKLNSX  EQU   *
         L     R10,SVCHKLNS
         BR    R10
```






```
HDGS     EQU   *
         ST    R10,SVHDGS
         AP    PGS,=P'1'
         MVC   HDPGS,=X'40202120'
         ED    HDPGS,PGS
         PUT   REPORT,FORMFEED
         PUT   REPORT,HD1
         PUT   REPORT,HD2
         PUT   REPORT,HD3
         PUT   REPORT,HD4
         ZAP   LNS,=P'0'
         L     R10,SVHDGS
         BR    R10
```

```
WRITE    EQU   *
         ST    R10,SVWRITE
         PUT   REPORT,RREC
         AP    LNS,=P'1'
         L     R10,SVWRITE
         BR    R10
```

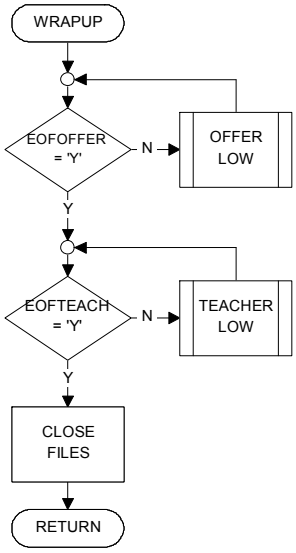


The WRAPUP Routine

Recall from our discussion of the mainline logic that we will continue to process records until either file is at EOF, and that it is likely that both files will not reach EOF at the same time. We must, therefore, keep in mind that when we reach WRAPUP, it is likely that one of the files is *not* at end of file. We will execute one of the following loops:

- If the OFFER file is not at EOF, process all remaining course offerings records as unmatched; that is, OFFERLOW.
- Otherwise, if the TEACHER file is not at EOF, process all remaining teacher records as unmatched; that is, TEACHLOW.

```
WRAPUP   EQU   *
         ST    R10,SVWRAP
WRAPUP2  EQU   *
         CLI   EOFOFFER,C'Y'
         BE    WRAPUP3
         BAL   R10,OFFERLOW
         B     WRAPUP2
WRAPUP3  EQU   *
         CLI   EOFTEACH,C'Y'
         BE    WRAPUP4
         BAL   R10,TEACHLOW
         B     WRAPUP3
WRAPUP4  EQU   *
         CLOSE TEACHERS
         CLOSE OFFER
         CLOSE REPORT
         WTO   'TEACH11A ... Teacher
               list on REPORT.TXT'
         L     R10,SVWRAP
         BR    R10
```



Program Output

```
                F92 Teaching Assignments      Page   1

      TID       Name          Course  Section  Room
      ---  ---------------    ------  -------  ----
      218  HINCKLEY, G.B.     AC101       1     B1
                              BU101       1     B1

      574  SMITH, J.          PE151       1     GYM

      626  YOUNG, B.          MA101       1     A2

                F92 Teaching Assignments      Page   2

      TID       Name          Course  Section  Room
      ---  ---------------    ------  -------  ----
      626  YOUNG, B.          MA101       2     A2

      732  BENSON, E.T.       EG101       1     A1
                              EG101       2     A1
```

Program Solution

The complete annotated program, `TEACH11A.MLC`, follows. In this program, `MAXLNS` was set to 6 so that at least one teacher (Young) would "span" a page break. This was done in order to demonstrate that a teacher's name would be shown at the top of the page if that teacher's courses spanned a page break.

```
         PRINT NOGEN
*****************************************************************
*        FILENAME:  TEACH11A.MLC                                *
*        AUTHOR  :  Bill Qualls                                 *
*        SYSTEM  :  PC/370 R4.2                                 *
*        REMARKS :  Produce list of F92 teaching assignments.   *
*                   This program illustrates matching logic.    *
*****************************************************************
         START 0
         REGS
BEGIN    BEGIN
         WTO   'TEACH11A ... Begin execution'
         BAL   R10,SETUP
MAIN     EQU   *
         CLI   EOFTEACH,C'Y'
         BE    EOJ
         CLI   EOFOFFER,C'Y'
         BE    EOJ
         BAL   R10,PROCESS
         B     MAIN
EOJ      EQU   *
         BAL   R10,WRAPUP
         WTO   'TEACH11A ... 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    OFFER+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  OFFER
         OPEN  REPORT
         BAL   R10,READTCH
         BAL   R10,READOFF
         L     R10,SVSETUP
         BR    R10
```

(continued)

```
*****************************************************************
*        HDGS - Print headings.                                 *
*****************************************************************
HDGS     EQU   *
         ST    R10,SVHDGS
         AP    PGS,=P'1'           Add 1 to page count
         MVC   HDPGS,=X'40202120' Edit pattern for page count
         ED    HDPGS,PGS           Move page count to heading
         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
         CLC   TTID,OTID           Attempt match on teacher ID
         BH    PROC2               Offerings low
         BL    PROC3               Teacher low
         BAL   R10,MATCH           Otherwise a match was found
         B     PROCESSX
PROC2    EQU   *                   No teacher for this offering
         BAL   R10,OFFERLOW
         B     PROCESSX
PROC3    EQU   *                   No offerings for this teacher
         BAL   R10,TEACHLOW
PROCESSX EQU   *
         L     R10,SVPROC
         BR    R10
*****************************************************************
*        TEACHLOW - No courses for this teacher.                *
*                   This is NOT an error condition.             *
*                   Just skip this teacher and go to next.      *
*****************************************************************
TEACHLOW EQU   *
         ST    R10,SVTCHLOW
         BAL   R10,READTCH         Read next teacher record
         L     R10,SVTCHLOW
         BR    R10
*****************************************************************
*        MATCH - Course offering with teacher.                  *
*****************************************************************
MATCH    EQU   *
         ST    R10,SVMATCH
         MVC   HOLDTID,TTID        Current teacher
         MVC   HOLDNAME,TTNAME
         BAL   R10,DOTHIS1         Do this one (teacher)
         BAL   R10,READTCH         Read next teacher record
         L     R10,SVMATCH
         BR    R10
```

(continued)

```
*****************************************************************
*        OFFERLOW - Course offering without matching teacher    *
*****************************************************************
OFFERLOW EQU   *
         ST    R10,SVOFFLOW
         MVC   HOLDTID,OTID       Current teacher
         MVC   HOLDNAME,=CL15'Unknown'
         BAL   R10,DOTHIS1        Do this one (teacher)
         L     R10,SVOFFLOW
         BR    R10
*****************************************************************
*        DOTHIS1 - Do this one (teacher)                        *
*****************************************************************
DOTHIS1  EQU   *
         ST    R10,SVTHIS1
         MVC   RREC,BLANKS
         MVC   RTID,HOLDTID       First line for each teacher
         MVC   RTNAME,HOLDNAME     will show ID and name
DOTHIS2  EQU   *                  This loop for each offering
         BAL   R10,CHKLNS          for this teacher...
         CP    LNS,=P'0'
         BNE   DOTHIS3
         MVC   RTID,HOLDTID       Also shown on first line
         MVC   RTNAME,HOLDNAME     of each page
DOTHIS3  EQU   *
         MVC   RCID,OCID
         MVC   RSECT,OSECT
         MVC   RROOM,OROOM
         BAL   R10,WRITE
         MVC   RREC,BLANKS
         BAL   R10,READOFF        Read next offerings record
         CLI   EOFOFFER,C'Y'      If EOF then done
         BE    DOTHIS4
         CLC   HOLDTID,OTID       Still the same teacher?
         BE    DOTHIS2             Yes, continue this one
DOTHIS4  EQU   *
         MVC   RREC,BLANKS        Blank line between teachers
         BAL   R10,WRITE
         L     R10,SVTHIS1
         BR    R10
*****************************************************************
*        READTCH - Read a teacher record.                       *
*****************************************************************
READTCH  EQU   *
         ST    R10,SVREADT
         GET   TEACHERS,TREC      Read a single teacher record
         B     READTX
ATENDTCH EQU   *
         MVI   EOFTEACH,C'Y'
READTX   EQU   *
         L     R10,SVREADT
         BR    R10
*****************************************************************
*        READOFF - Read a course offerings record.              *
*****************************************************************
READOFF  EQU   *
         ST    R10,SVREADO
```

(continued)

```
READOFF2 EQU   *
         GET   OFFER,OREC         Read a single offerings record
         CLC   OSEM,=CL3'F92'     Process F92 records only
         BNE   READOFF2
         B     READOX
ATENDOFF EQU   *
         MVI   EOFOFFER,C'Y'
READOX   EQU   *
         L     R10,SVREADO
         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,RREC        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
*                                 At this point we know that
*                                 at least one of the input
*                                 files is at EOF. Process
*                                 other file as "unmatched"
*                                 until at EOF also.
WRAPUP2  EQU   *
         CLI   EOFOFFER,C'Y'
         BE    WRAPUP3
         BAL   R10,OFFERLOW
         B     WRAPUP2
WRAPUP3  EQU   *
         CLI   EOFTEACH,C'Y'
         BE    WRAPUP4
         BAL   R10,TEACHLOW
         B     WRAPUP3
WRAPUP4  EQU   *
         CLOSE TEACHERS
         CLOSE OFFER
         CLOSE REPORT
         WTO   'TEACH11A ... Teacher list on REPORT.TXT'
         L     R10,SVWRAP
         BR    R10
```

(continued)

```
*****************************************************************
*        Literals, if any, will go here                         *
*****************************************************************
         LTORG
*****************************************************************
*        File definitions                                       *
*****************************************************************
TEACHERS DCB   LRECL=29,RECFM=F,MACRF=G,EODAD=ATENDTCH,
               DDNAME='TEACHER.SRT'
OFFER    DCB   LRECL=18,RECFM=F,MACRF=G,EODAD=ATENDOFF,
               DDNAME='OFFER.SRT'
REPORT   DCB   LRECL=62,RECFM=F,MACRF=P,
               DDNAME='REPORT.TXT'
*****************************************************************
*        RETURN ADDRESSES                                       *
*****************************************************************
SVSETUP  DC    F'0'               SETUP
SVHDGS   DC    F'0'               HDGS
SVPROC   DC    F'0'               PROCESS
SVREADT  DC    F'0'               READTCH
SVREADO  DC    F'0'               READOFF
SVWRITE  DC    F'0'               WRITE
SVWRAP   DC    F'0'               WRAPUP
SVCHKLNS DC    F'0'               CHKLNS
SVMATCH  DC    F'0'               MATCH
SVOFFLOW DC    F'0'               OFFERLOW
SVTCHLOW DC    F'0'               TEACHLOW
SVTHIS1  DC    F'0'               DOTHIS1
*****************************************************************
*        Miscellaneous field definitions                        *
*****************************************************************
WCRLF    DC    X'0D25'            PC/370 ONLY - EBCDIC CR/LF
EOFTEACH DC    CL1'N'             End of teacher file? (Y/N)
EOFOFFER DC    CL1'N'             End of offerings file? (Y/N)
PGS      DC    PL2'0'             Nbr of pages printed.
LNS      DC    PL2'6'             Lines printed on this page.
MAXLNS   DC    PL2'6'             Max nbr lines per page.
*                                 My line counts exclude hdgs.
BLANKS   DS    0CL62
         DC    CL60' ',XL2'0D25'
HOLDTID  DC    CL3' '             Hold (current) teacher ID
HOLDNAME DC    CL15' '            Hold (current) teacher name
*****************************************************************
*        Input record definition - Teacher                      *
*****************************************************************
TREC     DS    0CL29      1-29    Teacher record
TTID     DS    CL3         1- 3   Teacher ID nbr
TTNAME   DS    CL15        4-18   Teacher name
TTDEG    DS    CL4        19-22   Highest degree
TTTEN    DS    CL1        23-23   Tenured?
TTPHONE  DS    CL4        24-27   Phone nbr
TTCRLF   DS    CL2        28-29   PC/370 only - CR/LF
```

(continued)

```
*****************************************************************
*        Input record definition - Offerings                    *
*****************************************************************
OREC     DS    0CL18      1-18    Offerings record
OSEM     DS    CL3        1- 3    Semester
OCID     DS    CL5        4- 8    Course ID
OSECT    DS    CL1        9- 9    Section number
OTID     DS    CL3       10-12    Teacher ID
OROOM    DS    CL4       13-16    Room number
OOCRLF   DS    CL2       17-18    PC/370 only - CR/LF
*****************************************************************
*        Report (line) definition                               *
*****************************************************************
RREC     DS    0CL62      1-62    Report record
RTID     DS    CL3        1- 3    Teacher ID nbr
         DC    CL2' '     4- 5
RTNAME   DS    CL15       6-20    Teacher name
         DC    CL3' '    21-23
RCID     DS    CL5       24-28    Course ID
         DC    CL6' '    29-34
RSECT    DS    CL1       35-35    Section number
         DC    CL5' '    36-40
RROOM    DS    CL4       41-44    Room number
         DC    CL16' '   45-60
RCRLF    DS    CL2       61-62    PC/370 only - CR/LF
*****************************************************************
*        Headings definitions                                   *
*****************************************************************
FORMFEED DS    0CL62              PC/370 only
*        DC    X'0C'              EBCDIC formfeed
*        DC    CL59' '
         DC    60C'_'             For testing...
         DC    X'0D25'            EBCDIC CR/LF
HD1      DS    0CL62
         DC    CL40'          F92 Teaching Assignments      '
         DC    CL4'Page'
HDPGS    DC    CL4'BZZ9'
         DC    CL12' '
         DC    XL2'0D25'
HD2      DS    0CL62
         DC    CL60' '
         DC    XL2'0D25'
HD3      DS    0CL62
         DC    CL40'TID       Name         Course  Section  '
         DC    CL20'Room'
         DC    XL2'0D25'
HD4      DS    0CL62
         DC    CL40'---  ---------------   ------  -------  '
         DC    CL20'----'
         DC    XL2'0D25'
         END   BEGIN
```

Exercises

1. True or false. When two files are processed using matching logic...

   T F a. one file can usually be thought of as the master file and the other as the transaction file.
   T F b. both files must be sequenced on the key field.
   T F c. DOS' SORT command creates a new file containing the sorted records.
   T F d. the DDNAME parameter of the DCB must refer to the sorted file.
   T F e. the MAINLINE logic will continue until both files are at EOF.
   T F f. the SETUP routine will contain a priming read for both files.
   T F g. the READ routine will read one record from each file.
   T F h. the PROCESS routine compares one record from each file based on the first field in the record.
   T F i. there are two possible conditions as a result of the compare in the PROCESS routine.
   T F j. unmatched records always indicate a fatal error.
   T F k. there will be a separate CHKLNS and HDGS routines for each input file.
   T F l. we read the next record of each file at the end of the PROCESS routine.
   T F m. when we reach WRAPUP, we know that both files are at EOF.

2. What changes would you make to TEACH11A.MLC to move the check for semester F92 records from the READOFF routine to the PROCESS routine? (Show both routines.)

3. (Refer to the Small Town Self-Storage database in More Datasets.) Produce a list of locker renters as follows:

```
          1         2         3         4         5
12345678901234567890123456789012345678901234567890
              SMALL TOWN SELF-STORAGE    Page BZZ9
                  Current Renters

        Customer              Locker
--------------------------   --------
 ID#       Name              No  Type   Paid Thru
----- --------------------   --  ----   ---------
XXXXX XXXXXXXXXXXXXXXXXXX   XX    X    mm/dd/yy
XXXXX XXXXXXXXXXXXXXXXXXX   XX    X    mm/dd/yy
XXXXX XXXXXXXXXXXXXXXXXXX   XX    X    mm/dd/yy
```

   You will need the CUST and LOCKER files. In order to produce this report, both files must be in Customer ID sequence: type the following at the DOS prompt:

```
sort  /+1  <  cust.dat  >  cust.srt
sort  /+4  <  locker.dat  >  locker.srt
```

   Specify CUST.SRT and LOCKER.SRT as the DDNAME for the input DCBs.

Exercises

4. (Refer to the Small Town Self-Storage database in More Datasets.) Produce a locker availability summary report as follows:

```
         1         2         3         4
12345678901234567890123456789012345678901234567890
        SMALL TOWN SELF-STORAGE
      Locker Availability Summary

                                    Quantity
 Type    Size          Rate         Available
------   ----------   ------   ---------
  X      XXXXXXXXXX  BZZ9.99     BZZ9
  X      XXXXXXXXXX  BZZ9.99     BZZ9
  X      XXXXXXXXXX  BZZ9.99     BZZ9
```

You will need the LOCKER and RATE files. In order to produce this report, both files must be in Locker type sequence: type the following at the DOS prompt:

```
sort  /+3  <  locker.dat  >  locker.srt
sort  /+1  <  rate.dat  >  rate.srt
```

Specify LOCKER.SRT and RATE.SRT as the DDNAME for the input DCBs. Note this is a summary report, using control break logic. You are not listing each available locker, just the number of available lockers of each type. A locker is considered available if the Customer ID in the LOCKER file is blank.

5. (Refer to the Small Town Blood Bank database in More Datasets.) Produce a donor history summary as follows:

```
         1         2         3         4         5
12345678901234567890123456789012345678901234567890123456789012345678901234567890
              SMALL TOWN BLOOD BANK      Page BZZ9

                          First    Last   Number of
Donor ID  Donor Name   Donation Donation Donations
--------  ------------ -------- -------- ---------
  XXX     X XXXXXXXXXX mm/dd/yy mm/dd/yy   BZZ9
  XXX     X XXXXXXXXXX mm/dd/yy mm/dd/yy   BZZ9
  XXX     X XXXXXXXXXX mm/dd/yy mm/dd/yy   BZZ9
--------  ------------ -------- -------- ---------
 Total                                     BZZ9
```

Note: This program is similar to exercise 4 of chapter 10, except that the Donor name has been added to the report. See additional instructions in that chapter. The DONATION file will need to be sorted as shown there. This program will also read the DONOR file, but that file does not need to be sorted as it is already in Donor ID sequence. Show first initial and last name of donor.

Exercises

6. (Refer to the Small Town Hardware Store database in More Datasets.) Use the SALES file to update the quantity on hand field in the TOOL file. The quantity on hand (TQOH) will be reduced by the quantity sold (TSOLD). Do not update anyother fields.Write the updated TOOL file to NEWTOOL.DAT. (You are not producing a report.)

   If a tool ID is found on the SALES file without a matching tool ID on the TOOL file, then WTO an appropriate message and the record. (You will need to make another version of SALES.DAT to test this portion of your logic.)

   There is no need to sort the data since both files are already sequenced by tool ID.