



Legac-E Education

## **Reference Modification**

---

The development of this program offering examples of Reference Modification came about as a consequence of being asked to offer advice on the use of the REXX Interpret statement.

The requirement was to check a large data set for a number of data strings, the number of which might vary run to run. The location of any given string was known, and indeed would be in the same place in each record irrespective of string length.

The advice requested was supplied but was taken further by producing a COBOL version in addition to the REXX solution. This was done because it was felt that COBOL would offer run time performance benefits and if coded correctly should be flexible enough to deal with rule changes over time without the need to re-compile the program.

This solution opted for:

- The start location of the strings to be supplied via the JCL EXEC statement PARM field
- The rules to be supplied via a data set known as RULEFILE
- The maximum rules allowed was to be 15 (customer specified.)
- Each rule record would be in the form:
  - First two bytes contain the rule length
  - Third byte is a comma
  - Fourth byte onwards contains the rule text to a maximum of 77 characters

The program below is a modified version of the program described above. It was modified to facilitate its use with the data supplied in a COBOL programming course.



Legac-E Education

## Reference Modification

---

```
CBL OFFSET
IDENTIFICATION DIVISION.
PROGRAM-ID.                M12EX4.
AUTHOR.                    T.R.SAMBROOKS.
    INSTALLATION.
    DATE-WRITTEN.          24th SEPT 2014.
*-----*
*   This program performs rule checking matching that done   *
*   by a REXX routine which was the subject of recent query. *
*-----*
*   INPUTS - 3 in Total                                     *
*   INPUT  - 1 (JCL PARM Field)                             *
*           Specifies the location of the target data in    *
*           the record to be searched.                       *
*   INPUT  - 2 (Rule file)                                  *
*           Contains up to 15 rules each in the form of a  *
*           2-byte length field separated from the rule by  *
*           a comma.                                         *
*   INPUT  - 3 (The file to be searched)                    *
*-----*
ENVIRONMENT DIVISION.
CONFIGURATION SECTION.
INPUT-OUTPUT SECTION.
FILE-CONTROL.
    SELECT MAIN-FILE      ASSIGN TO UT-S-MAINFILE.
    SELECT RULE-FILE      ASSIGN TO UT-S-RULEFILE.
DATA DIVISION.
FILE SECTION.
FD  MAIN-FILE            RECORDING MODE IS F
                          LABEL RECORDS ARE STANDARD
                          BLOCK CONTAINS 0 RECORDS
                          RECORD CONTAINS 80 CHARACTERS
                          DATA RECORD IS FB-LOCO-REC.

COPY FBREC.
FD  RULE-FILE            RECORDING MODE IS F
                          LABEL RECORDS ARE STANDARD
                          BLOCK CONTAINS 0 RECORDS
                          RECORD CONTAINS 80 CHARACTERS
                          DATA RECORD IS RULE-REC.

01  RULE-REC            PIC X(80).
```



Legac-E Education

## Reference Modification

---

```
WORKING-STORAGE SECTION.
77                                     PIC X(30) VALUE
   '** START OF WORKING STORAGE **'.
77 RULE-LOC                           PIC S9(4) COMP VALUE +1.
77 RULE-LEN                           PIC S9(4) COMP VALUE +1.
77 RULE-SUB                           PIC S9(4) COMP VALUE +1.
77 RT-SIZE                             PIC S9(4) COMP VALUE +0.
77 MAIN-EOF                           PIC X      VALUE 'R'.
   88 MAIN-END                         VALUE 'D'.
77 RULE-EOF                           PIC X      VALUE 'R'.
   88 RULES-READ                       VALUE 'D'.
01 RULE-TABLE.
   03 TB-RULE                           OCCURS 5 TO 15 TIMES
                                       DEPENDING ON RT-SIZE.
                                       PIC 99.
                                       PIC X.
                                       PIC X(77).
LINKAGE SECTION.
01 JCL-PARM.
   03 PARM-LEN                         PIC S9(4) COMP.
   03 DATA-POS                       PIC 99.
PROCEDURE DIVISION
000-MAIN-RTN.
   IF PARM-LEN = 0                     DISPLAY '** NO PARM FIELD **'
                                       GOBACK
   ELSE
   OPEN INPUT                          MOVE DATA-POS TO RULE-LOC.
   PERFORM 100-GET-RULES                RULE-FILE, MAIN-FILE.
   PERFORM 110-PROCESS-RULES           UNTIL RULES-READ.
   CLOSE                                UNTIL MAIN-END.
                                       RULE-FILE, MAIN-FILE.
*-----*
*   This is the logical end of the program.   *
*-----*
GOBACK.
100-GET-RULES.
   IF RULES-READ                       NEXT SENTENCE
   ELSE
   READ RULE-FILE                       INTO TB-RULE (RULE-SUB)
                                       AT END MOVE 'D' TO RULE-EOF
   END-READ
   IF NOT RULES-READ                   ADD +1 TO RT-SIZE, RULE-SUB
   ELSE                                 DISPLAY '**' RT-SIZE ' RULES LOAD **'
   END-IF
END-IF.
100-GET-RULES-EXIT.
EXIT.
```



Legac-E Education

## Reference Modification

---

```
110-PROCESS-RULES.
  PERFORM 200-GET-MAIN-REC.
  PERFORM                                VARYING RULE-SUB FROM 1 BY 1
                                          UNTIL RULE-SUB =
                                          (RT-SIZE + 1)
  MOVE RULE-SCOPE (RULE-SUB) TO RULE-LEN
  IF FB-LOCO-REC(RULE-LOC:RULE-LEN) =
      RULE-TXT(RULE-SUB) (1:RULE-LEN)
      DISPLAY '** ' FB-ENGINE-NO ' IS A ' FB-DESIGNER
              ' LOCO '
  END-IF
  END-PERFORM.
110-PROCESS-RULE-EXIT.
  EXIT.
200-GET-MAIN-REC.
  READ MAIN-FILE                          AT END MOVE 'D' TO MAIN-EOF.
200-GET-MAIN-REC-EXIT.
  EXIT.
*-----*
*   This is the physical end of program - M12EX4.   *
*-----*
```

## Invoking JCL

```
//          SET   MBR=M12EX4                /* PROGRAM TO BE COMPILED */
//          SET   STU=&SYSUID                /* DATA SET HLQ = USERID */
//S0020     EXEC  PROC=IGYWCLG,PARM.COBOL=LIB,LNGPRFX=IGY420,
//          PARM.GO=33
//COBOL.SYSLIB DD  DISP=SHR,DSN=&STU..COPY.BOOKS
//COBOL.SYSIN  DD  DISP=SHR,DSN=&STU..BATCH.COB(&MBR)
//GO.MAINFILE DD  DISP=SHR,DSN=&STU..BATCH.COB(ENGINES)
//GO.SYSOUT DD   SYSOUT=*
//GO.RULEFILE DD  *
06,FOWLER
05,IVATT
08,THOMPSON
```