

LEVEL

2

18 ESD-TR-81-111, Vol. 2

14 MTR-8102, Vol. 2

AD A 097 635

6 USER'S MANUAL FOR STRATEGIC SATELLITE SYSTEM
TERMINAL SEGMENT LIFE CYCLE COST MODEL.
Volume II.

10 JANE E. COX and DAVID B. PETERS

11 MARCH 1981

DTIC
ELECTED
APR 10 1981

9 [The initial report]

Prepared for

DEPUTY FOR COMMUNICATIONS AND INFORMATION SYSTEMS
ELECTRONIC SYSTEMS DIVISION
AIR FORCE SYSTEMS COMMAND
UNITED STATES AIR FORCE
Hanscom Air Force Base, Massachusetts

15 F19628-81-C-4441



DTIC FILE COPY

Approved for public release;
distribution unlimited.

Project No. 6340
Prepared by
THE MITRE CORPORATION
Bedford, Massachusetts
Contract No. F19628-81-C-0001

235.50

81 4 10 002

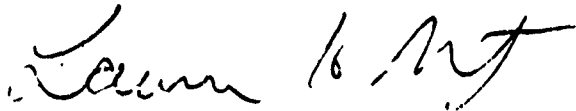
1002

When U.S. Government drawings, specifications, or other data are used for any purpose other than a definitely related government procurement operation, the government thereby incurs no responsibility nor any obligation whatsoever; and the fact that the government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data is not to be regarded by implication or otherwise, as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.

Do not return this copy. Retain or destroy.

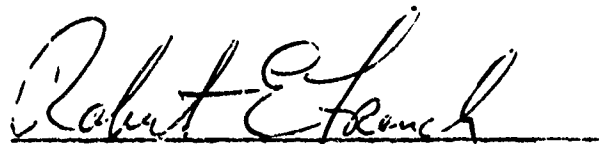
REVIEW AND APPROVAL

This technical report has been reviewed and is approved for publication.



LAWRENCE H. METZ, Major, USAF
Electronic Engineer
Logistics Division

FOR THE COMMANDER



ROBERT E. FRENCH, Lt Col, USAF
Chief, Acquisition Division
Satellite Communications Terminal SPO
Deputy for Communications and Information Systems

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER ESD-TR-81-111, Vol. 2	2. GOVT ACCESSION NO. AD-A097635	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) USER'S MANUAL FOR STRATEGIC SATELLITE SYSTEM TERMINAL SEGMENT LIFE CYCLE COST MODEL		5. TYPE OF REPORT & PERIOD COVERED
		6. PERFORMING ORG. REPORT NUMBER MTR-8102, Vol. 2
7. AUTHOR(s) Jane E. Cox David B. Peters		8. CONTRACT OR GRANT NUMBER(s) F19628-81-C-0001
9. PERFORMING ORGANIZATION NAME AND ADDRESS The MITRE Corporation P.O. Box 208 Bedford, MA 01730		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS Project No. 6340
11. CONTROLLING OFFICE NAME AND ADDRESS Deputy for Communications and Information Systems Electronic Systems Division, AFSC Hanscom AFB, MA 01731		12. REPORT DATE March 1981
		13. NUMBER OF PAGES 292
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		15. SECURITY CLASS. (of this report) UNCLASSIFIED
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE
16. DISTRIBUTION STATEMENT (of this Report) Approved for public release; distribution unlimited.		
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)		
18. SUPPLEMENTARY NOTES		
19. KEY WORDS (Continue on reverse side if necessary and identify by block number) COMPUTERIZED MODELS FORTRAN SOURCE CODE LIFE CYCLE COST STRATEGIC SATELLITE SYSTEM		
20. ABSTRACT (Continue on reverse side if necessary and identify by block number) > This volume of the User's Manual for the Strategic Satellite System Terminal Segment Life Cycle Cost Model contains listings of the FORTRAN source code for the three programs comprising the Model. These three programs are the Preprocessor, the LCC Program, and the RLA program.		

ACKNOWLEDGMENT

This report has been prepared by The MITRE Corporation under Project No. 6340. The contract is sponsored by the Electronic Systems Division, Air Force Systems Command, Hanscom Air Force Base, Massachusetts.

The authors wish to express their appreciation to the many people who played a role in the derivation of the Strategic Satellite System Terminal Segment Life Cycle Cost Model.

The Model is comprised of three programs: the Preprocessor, the LCC Program, and the RLA Program. The LCC Program, the heart of the Model, is actually a third generation AUTOLCC model. The original version was designed and implemented by Dick Moynihan and Joyce Calabro. The second version was modified by Josh Seeger and C. C. Cho. Mary Jean Hayes implemented this version.

The third and current version was designed by the authors and coded by Joyce Calabro, Sharon Rawls, and Lucille Record. The RLA Program was designed and implemented by C. C. Cho and Mary Jean Hayes.

Accession For	
NTIS GRA&I	☒
DTIC TAB	☐
Unannounced	☐
Justification	
By	
Distribution/	
Availability Codes	
Dist	Avail and/or Special
A	

TABLE OF CONTENTS

<u>Section</u>	<u>Page</u>
APPENDIX D PREPROCESSOR FORTRAN SOURCE CODE	7
MAIN PROCEDURE	7
SUBROUTINE ERROR	24
SUBROUTINE CLRYET	26
SUBROUTINE CHECK1	27
SUBROUTINE CHECK2	28
APPENDIX E LCC PROGRAM FORTRAN SOURCE CODE	29
MAIN PROCEDURE	29
SUBROUTINE READ1	35
SUBROUTINE READ2	38
SUBROUTINE READ3	39
SUBROUTINE READ4	40
SUBROUTINE READ5	42
SUBROUTINE READ6	43
SUBROUTINE READ7	44
SUBROUTINE READ8	46
SUBROUTINE READ9A	48
SUBROUTINE READ9B	50
SUBROUTINE READ10	52
SUBROUTINE READ11	54
SUBROUTINE INITAX	56
SUBROUTINE ERRCK1	61
SUBROUTINE RLCOMP	63
SUBROUTINE OTABST	65
SUBROUTINE ITAB1A	67
SUBROUTINE ITAB1B	70
SUBROUTINE ITAB1C	73
SUBROUTINE ITAB2	77
SUBROUTINE ITAB3	78
SUBROUTINE ITAB4	80
SUBROUTINE ITAB5	82
SUBROUTINE ITAB6	83
SUBROUTINE ITAB7	84
SUBROUTINE ITAB8	86
SUBROUTINE ITAB9A	88
SUBROUTINE ITAB9B	90
SUBROUTINE ITB10A	92

TABLE OF CONTENTS
(continued)

<u>Section</u>	<u>Page</u>
SUBROUTINE ITB10B	94
SUBROUTINE ITB10C	96
SUBROUTINE ITB10D	98
SUBROUTINE ITAB11	100
SUBROUTINE ZFAIL	101
SUBROUTINE ZNFB	102
SUBROUTINE ZERHB	104
SUBROUTINE ZERHSE	106
SUBROUTINE ZISET	109
SUBROUTINE ZUSE	111
SUBROUTINE ZTYPE	112
SUBROUTINE ZTFR	113
SUBROUTINE ZSECI	115
SUBROUTINE ZPMEQ	119
SUBROUTINE ZTISQ	120
SUBROUTINE ZYRSQ	121
SUBROUTINE ZTOTPQ	122
SUBROUTINE ZLC	123
FUNCTION U	124
FUNCTION F	125
FUNCTION XLEARN	126
SUBROUTINE COST1	127
SUBROUTINE COST2	129
SUBROUTINE COST3	131
SUBROUTINE COST4	133
SUBROUTINE COST5	135
SUBROUTINE COST6	137
SUBROUTINE COST7	139
SUBROUTINE COST8	142
SUBROUTINE COST9	145
SUBROUTINE COST10	147
SUBROUTINE COST11	152
FUNCTION CHLCC	155
SUBROUTINE DPIUP	162
SUBROUTINE DDMF	163
SUBROUTINE DRM	165
SUBROUTINE DXRM	166
SUBROUTINE DXUC	167
SUBROUTINE DUP	168
SUBROUTINE DFR	170

TABLE OF CONTENTS
(continued)

<u>Section</u>	<u>Page</u>
SUBROUTINE DXFR	173
SUBROUTINE DFPR	174
SUBROUTINE DXFPR	178
SUBROUTINE DRTS	179
SUBROUTINE DNRTS	181
SUBROUTINE DCOND	183
SUBROUTINE DSRU	185
SUBROUTINE DXMIL	191
SUBROUTINE OTAB1	193
SUBROUTINE OTAB2	196
SUBROUTINE OTAB3A	199
SUBROUTINE OTAB3B	202
SUBROUTINE OTAB3C	208
SUBROUTINE OTAB4A	214
SUBROUTINE OTAB4B	217
SUBROUTINE OTAB4C	219
SUBROUTINE OTAB5	221
SUBROUTINE OTAB6	224
SUBROUTINE OTAB7	226
SUBROUTINE RLAPRT	230
SUBROUTINE OSENS	231
SUBROUTINE INITAL	240
SUBROUTINE TITLE	244
SUBROUTINE TDSORT	245
SUBROUTINE SSETXX	246
SUBROUTINE PRMPT1	247
SUBROUTINE PRMPT2	250
SUBROUTINE PRMPT3	254
SUBROUTINE PRMPT4	258
SUBROUTINE PRMPT5	259
SUBROUTINE PRMPT6	260

TABLE OF CONTENTS
(concluded)

<u>Section</u>	<u>Page</u>
APPENDIX F RLA PROGRAM FORTRAN SOURCE CODE	261
MAIN PROCEDURE	261
SUBROUTINE READ1	264
SUBROUTINE READ2	266
SUBROUTINE READ3	268
SUBROUTINE ITAB1	269
SUBROUTINE ITAB2	271
SUBROUTINE ZTRAN	272
SUBROUTINE ZISINO	274
SUBROUTINE INITAX	275
SUBROUTINE STEPO	277
SUBROUTINE STEP1	278
SUBROUTINE STEP2	279
SUBROUTINE STEP3	280
SUBROUTINE STEP4	282
SUBROUTINE OUT9A	285
SUBROUTINE OTAB1	287
SUBROUTINE INITIAL	289

APPENDIX D

PREPROCESSOR FORTRAN SOURCE CODE

```

C *****
C *
C *          SSS LCC MODEL PREPROCESSOR
C *
C *      THIS PREPROCESSOR PERFORMS TWO BASIC FUNCTIONS:
C *
C *      1) CREATING AN ITEM-IN-PLATFORM MATRIX IN UNIT 21 GIVEN:
C *          LRU-IN-PLATFORM MATRIX (UNIT 25) AND
C *          SRU-IN-LRU MATRIX (UNIT 26).
C *      2) CHECKING UNITS 18,19,20,22,25 AND 26 FOR INPUT FILE
C *          FORMAT ERRORS.  IN PARTICULAR, THE FOLLOWING ERROR
C *          CONDITIONS ARE CHECKED FOR:
C *
C *          - MULTIPLE CARDS FOR AN ITEM NUMBER
C *          - MORE ITEMS IN A FILE THAN IN THE INITIAL FILE
C *          - ITEMS NOT APPEARING IN THE INITIAL ITEM FILE BUT
C *            FOUND IN OTHER FILES
C *          - FEWER ITEMS IN A FILE THAN IN THE INITIAL FILE
C *          - ITEMS NOT APPEARING IN A FILE BUT NOT FOUND IN THE
C *            INITIAL FILE
C *          - SRU ITEMS IN LRU ITEM LISTS
C *          - LRU ITEMS IN SRU ITEM LISTS
C *          - ITEM NUMBERS OUT OF SEQUENCE IN A FILE
C *          - ITEM INDICES OUT OF RANGE
C *          - MORE THAN THE MAXIMUM NUMBER OF ITEMS IN THE INITIAL
C *            FILE
C *          - MORE THAN THE MAXIMUM NUMBER OF SRUS IN AN LRU IN
C *            THE LRU/SRU CROSS REFERENCE FILE (UNIT 12)
C *          - END OF FILE FOUND BEFORE THE END OF THE SRU LIST
C *            FOR AN LRU IN THE LRU/SRU CROSS REFERENCE FILE
C *          - MISSING END-OF-FILE MARKER IN A FILE
C *
C *****

```

```

COMMON ITEM(999,2),NERR,YET(999),LASTI,MAXI
LOGICAL ITEM,YET,WROTE
REAL NITEM(999,12),INPVEC(28),XCOL1,QPA(50)
INTEGER INITEM(12)
INTEGER I,IL,IJ,OOP,NDS,K,NP,LRU,NUMITM,NUMLRU,ISRU(50)
INTEGER NERR,LASTI,MAXI,MAXNDS,NRM,IRM,NRMM1,IPAGE
DATA STAR/1H*/,BLANK/1H /

```

```

1  FORMAT(A1,I3,39X,I2)
2  FORMAT(A1,I3,12F4.2)
3  FORMAT(A1)
4  FORMAT(A1,I3,I2,14(I3,F2.0))
5  FORMAT(A1,I3)
6  FORMAT(/24H PROCESSING COMPLETED. ,I3,17H ERRORS DETECTED.)
7  FORMAT(/49H OUTPUT FILE (ITEMS IN PLATFORMS) NOT WRITTEN DUE/
+17X,10HTO ERRORS./)
8  FORMAT(/53H OUTPUT FILE (ITEMS IN PLATFORMS) WRITTEN TO UNIT 21./)
9  FORMAT(/1H ,I3,10H LRUS AND ,I3,29H SRUS READ FROM INITIAL FILE.)
10 FORMAT(/41H SSS LCC PREPROCESSOR -- EXECUTION BEGINS)
11 FORMAT(A1,I3,12I4)
12 FORMAT(1H ,15,12F7.2)
13 FORMAT(1H1//////////)
+44X,44H*****/
+44X,1H*,42X,1H*/
+44X,44H*          SSS LIFE CYCLE COST MODEL          */
+44X,44H*          PREPROCESSOR                      */
+44X,1H*,42X,1H*/
+44X,44H*****/)
14 FORMAT(1H1,6X,44H PREPROCESSOR INPUT FILE: DATA FILE 11B -- ,
+35HLRU ITEM CONFIGURATION ON PLATFORMS///)
15 FORMAT(1H1,4X,45H PREPROCESSOR OUTPUT FILE: DATA FILE 11A -- ,
+31HITEM CONFIGURATION ON PLATFORMS///)
16 FORMAT(A1,I3,I2,I2)
17 FORMAT(A1,5X,14(I3,F2.0))
18 FORMAT(1H1/30X,70HPREPROCESSOR INPUT FILE: DATA FILE 8B -- LRU/SR
+U CROSS REFERENCE DATA)
19 FORMAT(59X,11H(CONTINUED)///)
20 FORMAT(/9X,4H#SRU,4X,12HSRU SRU ,3X,12HSRU SRU ,3X,12HSRU
+ SRU ,3X,12HSRU SRU ,3X,12HSRU SRU ,3X,12HSRU SRU
+,3X,12HSRU SRU /1X,3HLRU,5X,5HTYPES,3X,12HINDEX QUAN-,3X,12HI
+NDEX QUAN-,3X,12HINDEX QUAN-,3X,12HINDEX QUAN-,3X,12HINDEX QUA
+N-,3X,12HINDEX QUAN-,3X,12HINDEX QUAN-/1X,5HINDEX,3X,6HIN LRU,2X
+,12HNO. TITY ,3X,12HNO. TITY ,3X,12HNO. TITY ,3X,12HNO.
+ TITY ,3X,12HNO. TITY ,3X,12HNO. TITY ,3X,11HNO. TITY/1X
+,4H(IL),4X,5H(NDS),3X,12H(ISRU) (QPA),3X,12H(ISRU) (QPA),3X,12H(IS
+RU) (QPA),3X,12H(ISRU) (QPA),3X,12H(ISRU) (QPA),3X,12H(ISRU) (QPA)
+,3X,12H(ISRU) (QPA)///)
21 FORMAT(2X,I3,5X,I3,5X,I3,3X,F4.0,5X,I3,3X,F4.0,5X,I3,3X,F4.0,5X,I3
+,3X,F4.0,5X,I3,3X,F4.0,5X,I3,3X,F4.0,5X,I3,3X,F4.0/18X,I3,3X,F4.0,
+5X,I3,3X,F4.0,5X,I3,3X,F4.0,5X,I3,3X,F4.0,5X,I3,3X,F4.0,5X,I3,3X,F
+4.0,5X,I3,3X,F4.0)
22 FORMAT(18X,I3,3X,F4.0,5X,I3,3X,F4.0,5X,I3,3X,F4.0,5X,I3
+,3X,F4.0,5X,I3,3X,F4.0,5X,I3,3X,F4.0,5X,I3,3X,F4.0/18X,I3,3X,F4.0,
+5X,I3,3X,F4.0,5X,I3,3X,F4.0,5X,I3,3X,F4.0,5X,I3,3X,F4.0,5X,I3,3X,F
+4.0,5X,I3,3X,F4.0)

```

```

C    PLEASE ANNOUNCE US

    WRITE(7,13)
    WRITE(6,10)

C    READ IN THE INITIAL DATA FROM UNIT 18

    REWIND 18
    LASTI=0
    DO 50 ILOOP=1,MAXI
        READ(18,1) XCOL1,I,LRU
        IF(XCOL1.EQ.STAR) GO TO 100
        IF (I.GT.0.AND.I.LE.MAXI) GO TO 40

C    ITEM INDEX OUT OF RANGE.  NOTIFY AND SKIP TO NEXT CARD

        CALL ERROR(8,I,18)
        GO TO 50
40    CONTINUE

        ITEM(I,1)=.TRUE.

C    CHECK IF THIS ITEM IS AN LRU.  IF SO, MARK AND COUNT IT

        IF(LRU.NE.1) GO TO 45
        ITEM(I,2)=.TRUE.
        IF (.NOT.YET(I)) NUMLRU=NUMLRU+1

45    CONTINUE

        IF (.NOT.YET(I)) NUMITM=NUMITM+1
        CALL CHECK1(I,18)

50    CONTINUE

C    IF WE GET HERE THEN THERE WERE TOO MANY RECORDS IN THE FILE

        CALL ERROR(11,0,18)

100   CONTINUE

        I=NUMITM-NUMLRU
        WRITE(6,9) NUMLRU,I

C    READ IN THE LRU IN PLATFORM MATRIX

```

```

23  FORMAT(7H  ITEM,16X,40HAVERAGE NUMBER OF LRU ITEMS INSTALLED ON,
+14H PLATFORM TYPE/,8H  INDEX,15X,50(1H-)/,7H  (I),14,1117//)
24  FORMAT(7H  ITEM,16X,36HAVERAGE NUMBER OF ITEMS INSTALLED ON,
+14H PLATFORM TYPE/,8H  INDEX,15X,50(1H-)/,7H  (I),14,1117//)
25  FORMAT(/45H ***** THE REMAINDER OF FILE 8B NOT READ)

C    MAXI HOLDS THE MAXIMUM NUMBER AND INDEX OF ITEMS

      MAXI=999

C    MAXNDS HOLDS THE MAXIMUM NUMBER OF SRUS IN AN LRU

      MAXNDS=50

      NERR=0
      NUMLRU=0
      NUMITM=0
      WROTE=.FALSE.
      DO 30 NP=1,12

C        INITEM WILL HOLD THE ITEM-IN-PLATFORM DATA IN INTEGER FORMAT

          INITEM(NP)=0

30    CONTINUE

      DO 36 ILOOP=1,MAXI

C        NITEM(I,NP) WILL HOLD THE ITEM-IN-PLATFORM DATA

          DO 33 NP=1,12
            NITEM(ILOOP,NP)=0.
33      CONTINUE

C        ITEM(I,1) WILL INDICATE WHETHER ITEM(I) WAS IN INITIAL FILE

          ITEM(ILOOP,1)=.FALSE.

C        ITEM(I,2) IS .TRUE. IF ITEM(I) IS AN LRU

          ITEM(ILOOP,2)=.FALSE.

C        YET(I) WILL INDICATE WHETHER ITEM(I) WAS IN A PARTICULAR FILE

          YET(ILOOP)=.FALSE.

36    CONTINUE

```

```

        REWIND 25
        CALL CLRYET

C      LABEL THE INPUT TABLE

        WRITE(7,14)
        WRITE(7,23) (I,I=1,12)
        IPAGE=0
        LASTI=0
        DO 200 ILOOP=1,NUMLRU
            READ(25,2) XCOL1,IL,(NITEM(IL,NP),NP=1,12)
            IF (XCOL1.NE.STAR) GO TO 120

C      IF WE FOUND A STAR, THEN THERE ARE TOO FEW ITEMS IN THIS FILE

            CALL ERROR(13,IL,25)

C      NOW CHECK WHICH ITEMS ARE MISSING FROM THIS FILE

            DO 110 IL=1,MAXI
                IF (ITEM(IL,2).AND..NOT.YET(IL)) CALL ERROR(4,IL,25)
110          CONTINUE

C      ESCAPE.  JUMP TO NEXT FILE

            GO TO 210

120          CONTINUE

C      ECHO PRINT THE FILE
        WRITE(7,12) IL,(NITEM(IL,NP),NP=1,12)
        IPAGE=IPAGE+1
        IF (IPAGE.LT.40) GO TO 130
        WRITE(7,14)
        WRITE(7,19)
        WRITE(7,23) (J1,J1=1,12)
        IPAGE=0

130          CONTINUE

            IF (IL.GT.0.AND.IL.LE.MAXI) GO TO 140

C      ITEM INDEX OUT OF RANGE.  NOTIFY AND SKIP TO NEXT CARD

            CALL ERROR(8,IL,25)
            GO TO 200

140          CONTINUE

```

```

        CALL CHECK1(IL,25)

C      CHECK IF THIS LRU ITEM IS MARKED AS AN LRU IN INITIAL FILE
C      (THIS FILE SHOULD HAVE ONLY LRU ITEMS)

        IF (ITEM(IL,1).AND..NOT.ITEM(IL,2)) CALL ERROR(3,IL,25)

200    CONTINUE

C      WE SHOULD BE AT THE BOTTOM OF THE FILE.

        READ(25,2) XCOL1,IL,(INPVEC(J1),J1=1,12)
        IF (XCOL1.EQ.STAR) GO TO 210

C      WE WEREN'T.  READ ON, ERROR CHECKING AS BEFORE.

        CALL ERROR(12,0,25)
206    CONTINUE

        IF (IL.GT.0.AND.IL.LE.MAXI) GO TO 207

C      ITEM INDEX OUT OF RANGE.  NOTIFY AND SKIP TO NEXT RECORD

        CALL ERROR(8,IL,25)
        GO TO 208

207    CONTINUE

C      NOW THAT WE KNOW THAT WE HAVE A LEGAL SUBSCRIPT

        DO 998 J1=1,12
            NITEM(IL,J1)=INPVEC(J1)
998    CONTINUE

C      ECHO PRINT THE FILE

        IPAGE=IPAGE+1
        WRITE(7,12) IL,(NITEM(IL,NP),NP=1,12)
        IF (IPAGE.LT.40) GO TO 999
        WRITE(7,14)
        WRITE(7,19)
        WRITE(7,23) (J1,J1=1,12)
        IPAGE=0

999    CONTINUE

```

```

        CALL CHECK1(IL,25)
C      CHECK AGAIN IF THIS LRU ITEM IS MARKED AS AN LRU IN INITIAL
C      FILE (THIS FILE SHOULD HAVE ONLY LRU ITEMS)
        IF (ITEM(IL,1).AND..NOT.ITEM(IL,2)) CALL ERROR(3,IL,25)
208    CONTINUE
        READ(25,5) XCOL1,IL
C      CHECK IF WE ARE AT THE END-OF-FILE MARKER OR END OF DATA
        IF (XCOL1.NE.STAR.AND.IL.NE.0) GO TO 206
C      IF WE RAN OUT OF DATA WITHOUT A MARKER, NOTIFY
        IF (XCOL1.NE.STAR) CALL ERROR(9,0,25)
210    CONTINUE
C      READ IN THE LRU/SRU CROSS REFERENCE TABLE
        REWIND 26
        CALL CLRYET
C      LABEL THE OUTPUT
        WRITE(7,18)
        WRITE(7,20)
        IPAGE=0
        LASTI=0
        DO 300 ILOOP=1,NUMLRU
            READ(26,4) XCOL1,IL,NDS,(ISRU(K),QPA(K),K=1,14)
            IF (XCOL1.NE.STAR) GO TO 225
C      IF WE FOUND A STAR, THEN THERE ARE TOO FEW ITEMS HERE
        CALL ERROR(13,0,26)
C      ESCAPE. JUMP TO NEXT FILE
        GO TO 310
225    CONTINUE

```

```

C      ECHO PRINT THE DATA

      IPAGE=IPAGE+1
      WRITE(7,21) IL,NDS,(ISRU(J1),QPA(J1),J1=1,14)
      IF (IPAGE.LT.40) GO TO 226
        WRITE(7,18)
        WRITE(7,19)
        WRITE(7,20)
        IPAGE=0

226    CONTINUE

      IF (IL.GT.0.AND.IL.LE.MAXI) GO TO 227

C      INDEX IS OUT OF RANGE.  NOTIFY AND SKIP TO NEXT RECORD

      CALL ERROR(8,IL,26)
      GO TO 300

227    CONTINUE

      CALL CHECK1(IL,26)

C      CHECK IF THIS LRU ITEM IS MARKED AS AN LRU IN INITIAL FILE

      IF (ITEM(IL,1).AND..NOT.ITEM(IL,2)) CALL ERROR(3,IL,26)

C      CHECK IF THE NUMBER OF SRUS IN THIS LRU EXCEEDS THE LIMIT

      IF (NDS.LE.MAXNDS) GO TO 228

      CALL ERROR(15,IL,26)

C      SET NDS TO MAXNDS.  THIS WILL GIVE SOME ERROR CHECKING
C      OF THE SRU LIST.  NOTE THAT ADDITIONAL CARDS WILL BE READ
C      AS LISTS FOR AN ITEM INDEXED ZERO, GENERATING MESSAGES.

      NDS=MAXNDS

228    CONTINUE

C      READ IN THE REST OF THE SRU DATA, OFF OF SUBSEQUENT CARDS
C      IF NECESSARY

      J2=14
      J3=14

```

```

230      CONTINUE

C      CHECK IF THERE ARE MORE CARDS TO READ

      IF(.NOT.(NDS.GT.J3.AND.NDS.LE.MAXNDS)) GO TO 240
      J2=J3+1
      J3=J2+13
      READ(26,17) XCOL1,(ISRU(J1),QPA(J1),J1=J2,J3)
      WRITE(7,22) (ISRU(J1),QPA(J1),J1=J2,J3)
      IF (XCOL1.NE.STAR) GO TO 230

      CALL ERROR(16,IL,26)

C      ESCAPE TO NEXT FILE

      GO TO 310

240      CONTINUE

C      THIS NEXT SET OF CODE IS THE ONLY CALCULATION DONE BY THIS
C      PROGRAM.  IF THERE ARE ANY SRUS IN THIS LRU, THEN THEY ARE
C      ADDED INTO THE ITEM-IN-PLATFORM MATRIX

      IF (NDS.EQ.0) GO TO 250
      DO 248 K=1,NDS
      IF (ISRU(K).GT.0.AND.ISRU(K).LE.MAXI) GO TO 244

C      INDEX OUT OF RANGE.  NOTIFY AND JUMP TO NEXT SRU

      CALL ERROR(6,IL,26)
      GO TO 248

244      CONTINUE

C      CHECK IF THIS SRU ITEM APPEARS IN THE INITIAL FILE

      IF (.NOT.ITEM(ISRU(K),1)) CALL ERROR(2,K,26)

C      CHECK IF THIS SRU ITEM IS MARKED AS AN SRU IN INITIAL FILE

      IF (ITEM(ISRU(K),1).AND.ITEM(ISRU(K),2)) CALL ERROR(7,K,26)

      DO 246 NP=1,12
      NITEM(ISRU(K),NP)=NITEM(ISRU(K),NP)+QPA(K)*NITEM(IL,NP)
246      CONTINUE

248      CONTINUE

```

```

250     CONTINUE
300     CONTINUE
C      THIS SHOULD BE THE END OF THE FILE
      READ(26,3) XCOL1
      IF (XCOL1.EQ.STAR) GO TO 305
C      IT WASN'T. NOTIFY, AND SKIP TO NEXT FILE
      CALL ERROR(12,0,26)
      WRITE(7,25)
305     CONTINUE
C      CHECK TO SEE IF WE MISSED ANY LRUS ALONG THE WAY
      DO 310 IL=1,MAXI
      IF (ITEM(IL,2).AND..NOT.YET(IL)) CALL ERROR(4,IL,26)
310     CONTINUE
C      WRITE OUT THE ITEM IN PLATFORM FILE TO UNIT 21 AND THE OFFLINE
C      PRINTER ONLY IF WE HAVE NOT ENCOUNTERED ANY ERRORS YET.
      IF (NERR.NE.0) GO TO 314
C      'WROTE' FLAGS THAT WE WROTE THE FILE
      WROTE=.TRUE.
      REWIND 18
C      LABEL THE OUTPUT
      WRITE(7,15)
      WRITE(7,24) (I,I=1,12)
      IPAGE=0
      DO 313 ILOOP=1,NUMITM
      READ(18,5) XCOL1,I
C      PRINT WITHOUT DECIMAL POINT IN A FIELD OF WIDTH FOUR
      DO 312 NP=1,12
      INITEM(NP)=INT(0.5+100*NITEM(I,NP))
312     CONTINUE

```

```

        IPAGE=IPAGE+1
        WRITE(21,11) BLANK,I,(INITEM(NP),NP=1,12)
        WRITE(7,12) I,(NITEM(I,NP),NP=1,12)
        IF (IPAGE.LT.40) GO TO 313
        WRITE(7,15)
        WRITE(7,19)
        WRITE(7,24) (J1,J1=1,12)
        IPAGE=0

313      CONTINUE

        WRITE(21,3) STAR

314      CONTINUE

C        CHECK THIS SUPPORT EQUIPMENT PER ITEM FILE FOR CONSISTENCY

        REWIND 20
        CALL CLRYET

        DO 400 ILOOP=1,NUMITM
          READ(20,16) XCOL1,I,NRM,IRM
          IF(XCOL1.NE.STAR) GO TO 320

C          IF WE FOUND A STAR THEN THERE ARE TOO FEW ITEMS HERE

          CALL ERROR(13,0,20)

C          ESCAPE.  JUMP TO NEXT FILE

          GO TO 410

320      CONTINUE

        IF (I.GT.0.AND.I.LE.MAXI) GO TO 340

C          INDEX OUT OF RANGE.  NOTIFY AND GO TO NEXT INDEX RECORD

          CALL ERROR(8,I,20)
          GO TO 345

340      CONTINUE

          CALL CHECK1(I,20)

345      CONTINUE

```

```

C      SKIP NRM-1 LINES TO GET TO THE NEXT ITEM INDEX

C      THIS GOTO IS TO A POINT INSIDE OF THE FOLLOWING DO LOOP,
C      ONLY TO INSURE CHECKING OF THE CURRENT IRM VALUE.

      IF (NRM.EQ.1) GO TO 350

      NRMM1=NRM-1
      DO 360 K=1,NRMM1
        READ(20,16) XCOL1,IL,NRM,IRM

        IF (IL.EQ.0.AND.NRM.EQ.0) GO TO 350
        CALL ERROR(10,I,20)

C      IT IS TOO DIFFICULT TO ANTICIPATE EXACTLY WHAT THE
C      ERROR WAS AND TO LOCATE THE NEXT "CORRECT" RECORD,
C      SO THE REST OF THE FILE IS NOT READ.

      GO TO 415

350      CONTINUE
      IF (IRM.LT.1.OR.IRM.GT.4) CALL ERROR(14,I,20)

360      CONTINUE

400      CONTINUE
      READ(20,5) XCOL1,I

C      WE SHOULD BE AT THE END OF THE FILE

      IF (XCOL1.EQ.STAR) GO TO 410

C      WE WEREN'T. NOTIFY, AND READ ON, ERROR CHECKING

      CALL ERROR(12,0,20)

402      CONTINUE
      IF (I.GT.0.AND.I.LE.MAXI) GO TO 404

C      INDEX OUT OF RANGE. NOTIFY, AND GO TO NEXT RECORD

      CALL ERROR(8,I,20)
      GO TO 406

404      CONTINUE

```

```

CALL CHECK1(I,20)

406  CONTINUE

C    SKIP NRM-1 LINES TO GET TO THE NEXT ITEM INDEX

C    THIS GOTO IS TO A POINT INSIDE OF THE FOLLOWING DO LOOP,
C    ONLY TO INSURE CHECKING OF THE CURRENT IRM VALUE.

      IF (NRM.EQ.1) GO TO 408

      NRMM1=NRM-1
      DO 409 K=1,NRMM1
        READ(20,16) XCOL1,IL,NRM,IRM

        IF (.NOT.(IL.EQ.0.AND.NRM.EQ.0)) GO TO 408
        CALL ERROR(10,I,20)

C      IT IS TOO DIFFICULT TO ANTICIPATE EXACTLY WHAT THE
C      ERROR WAS AND TO LOCATE THE NEXT "CORRECT" RECORD,
C      SO THE REST OF THE FILE IS NOT READ.

      GO TO 510

408  CONTINUE
      IF (IRM.LT.1.OR.IRM.GT.4) CALL ERROR(14,I,20)

409  CONTINUE
      READ(20,5) XCOL1,I

C    CHECK IF AT END OF FILE OR END OF DATA

      IF (XCOL1.NE.STAR.AND.I.NE.0) GO TO 402

C    IF AT END OF DATA BUT NOT END OF FILE THEN MARKER MISSING

      IF (XCOL1.NE.STAR) CALL ERROR(9,0,20)

410  CONTINUE

C    CHECK IF WE MISSED ANY ITEMS ALONG THE WAY

      CALL CHECK2(20)

415  CONTINUE

```

```

C      CHECK THIS BY-ITEM FILE FOR CONSISTENCY

      REWIND 22
      CALL CLRDET

      DO 500 ILOOP=1,NUMITM
        READ(22,5) XCOL1,I
        IF(XCOL1.NE.STAR) GO TO 420

C          IF WE FOUND A STAR THEN THERE ARE TOO FEW ITEMS HERE

          CALL ERROR(13,0,22)

C          ESCAPE. JUMP TO NEXT FILE
          GO TO 510

420      CONTINUE
        IF (I.GT.0.AND.I.LE.MAXI) GO TO 440

C          INDEX OUT OF RANGE. NOTIFY AND GO TO NEXT RECORD

          CALL ERROR(8,I,22)
          GO TO 500

440      CONTINUE

          CALL CHECK1(I,22)

500      CONTINUE

          READ(22,5) XCOL1,I

C          WE SHOULD BE AT THE END OF THE FILE

          IF (XCOL1.EQ.STAR) GO TO 510

C          WE WEREN'T. NOTIFY, AND READ ON, ERROR CHECKING

          CALL ERROR(12,0,22)

505      CONTINUE
        IF (I.GT.0.AND.I.LE.MAXI) GO TO 507

C          INDEX OUT OF RANGE. NOTIFY AND GO TO NEXT RECORD

          CALL ERROR(8,I,22)
          GO TO 508

```

```

507      CONTINUE

        CALL CHECK1(I,22)

508      CONTINUE
        READ(22,5) XCOL1,I

C        CHECK IF AT END OF FILE OR END OF DATA

        IF (XCOL1.NE.STAR.AND.I.NE.0) GO TO 505

C        IF AT END OF DATA BUT NOT END OF FILE THEN MARKER MISSING

        IF (XCOL1.NE.STAR) CALL ERROR(9,0,22)

510      CONTINUE

C        CHECK IF WE MISSED ANY ITEMS ALONG THE WAY

        CALL CHECK2(22)

C        CHECK THIS BY-ITEM FILE FOR CONSISTENCY

        REWIND 19
        CALL CLRYET

        DO 600 ILOOP=1,NUMITM
          READ(19,5) XCOL1,I
          IF(XCOL1.NE.STAR) GO TO 520

C          IF WE FOUND A STAR THEN THERE ARE TOO FEW ITEMS HERE

          CALL ERROR(13,0,19)

C          ESCAPE.  JUMP TO NEXT FILE.

          GO TO 610

520      CONTINUE
        IF (I.GT.0.AND.I.LE.MAXI) GO TO 540

C        INDEX OUT OF RANGE.  NOTIFY AND GO TO NEXT FILE

        CALL ERROR(8,I,19)
        GO TO 600

```

```

540     CONTINUE
        CALL CHECK1(I,19)

600     CONTINUE
        READ (19,5) XCOL1,I

C      WE SHOULD BE AT THE END OF THE FILE
        IF (XCOL1.EQ.STAR) GO TO 610

C      WE WEREN'T. NOTIFY AND READ ON, ERROR CHECKING
        CALL ERROR(12,0,19)

605     CONTINUE
        IF (I.GT.0.AND.I.LE.MAXI) GO TO 607

C      INDEX OUT OF RANGE. NOTIFY AND GO TO NEXT RECORD
        CALL ERROR(8,I,19)
        GO TO 608

607     CONTINUE
        CALL CHECK1(I,19)

608     CONTINUE
        READ(19,5) XCOL1,I

C      CHECK IF AT END OF FILE OR END OF DATA
        IF (XCOL1.NE.STAR.AND.I.NE.0) GO TO 605

C      IF AT END OF DATA BUT NOT END OF FILE THEN MARKER MISSING
        IF (XCOL1.NE.STAR) CALL ERROR(9,0,19)

610     CONTINUE

C      CHECK IF WE MISSED ANY ITEMS ALONG THE WAY
        CALL CHECK2(19)

C      GIVE THE NEWS

```

WRITE(6,6) NERR

IF (WROTE) WRITE(6,8)

IF (.NOT.WROTE) WRITE(6,7)

STOP

END

SUBROUTINE ERROR(TYPE,LINE,FILE)

C THIS SUBROUTINE PRINTS OUT THE ERROR MESSAGES AND TALLIES
C THE NUMBER OF ERRORS.

COMMON ITEM(999,2),NERR,YET(999),LASTI,MAXI
INTEGER TYPE,LINE,FILE,MAXI,NERR
LOGICAL ITEM,YET

```

1  FORMAT(/24H *ERROR 1* ITEM NUMBER ,I3,23H HAS TWO CARDS IN UNIT ,
    +I2)
2  FORMAT(/31H *ERROR 2* A NEW ITEM INDEXED ,I3,9H APPEARS ,
    +8HIN UNIT ,I2)
3  FORMAT(/25H *ERROR 3* AN SRU (ITEM ,I3,21H) APPEARS IN THE LRU ,
    +13HLIST IN UNIT ,I2)
4  FORMAT(/17H *ERROR 4* ITEM ,I3,22H IS MISSING FROM UNIT ,I2)
5  FORMAT(/17H *ERROR 5* ITEM ,I3,28H IS OUT OF SEQUENCE IN UNIT ,
    +I2)
6  FORMAT(/51H *ERROR 6* INVALID SRU INDEX IN LIST FOR LRU ITEM ,I3,
    +9H IN UNIT ,I2)
7  FORMAT(/25H *ERROR 7* AN LRU (ITEM ,I3,20H) APPEARS IN THE SRU,
    +14H LIST IN UNIT ,I2)
8  FORMAT(/32H *ERROR 8* INVALID ITEM INDEX (,I4,10H) APPEARS ,
    +8HIN UNIT ,I2)
9  FORMAT(/53H *ERROR 9* END OF FILE MARKER (*) MISSING FROM UNIT ,
    +I2)
10 FORMAT(/38H *ERROR 10* FILE FORMAT ERROR IN UNIT ,I2,5H: NRM/
    +17X,15HVALUE FOR ITEM ,I3,17H IS INCONSISTENT./
    +17X,39HTHE REMAINDER OF THIS FILE IS NOT READ.)
11 FORMAT(/53H *ERROR 11* ITEM DESCRIPTION FILE CONTAINS MORE THAN /
    +17X,29HTHE ALLOWABLE NUMBER OF ITEMS)
12 FORMAT(/17H *ERROR 12* UNIT ,I2,26H CONTAINS MORE ITEMS THAN ,
    +12HINITIAL FILE)
13 FORMAT(/17H *ERROR 13* UNIT ,I2,27H CONTAINS FEWER ITEMS THAN ,
    +12HINITIAL FILE)
14 FORMAT(/44H *ERROR 14* IRM VALUE OUT OF RANGE FOR ITEM ,I3/
    +17X,8HIN UNIT ,I2)
15 FORMAT(/16H *ERROR 15* LRU ,I3,24H CONTAINS TOO MANY SRUS ,
    +8HIN UNIT ,I2)
16 FORMAT(/52H *ERROR 16* END OF FILE FOUND BEFORE END OF SRU LIST/
    +17X,8HFOR LRU ,I3,9H IN UNIT ,I2)

```

```

IF (TYPE.EQ.1) WRITE(6,1) LINE,FILE
IF (TYPE.EQ.2) WRITE(6,2) LINE,FILE
IF (TYPE.EQ.3) WRITE(6,3) LINE,FILE
IF (TYPE.EQ.4) WRITE(6,4) LINE,FILE
IF (TYPE.EQ.5) WRITE(6,5) LINE,FILE

```

```
IF (TYPE.EQ.6) WRITE(6,6) LINE,FILE
IF (TYPE.EQ.7) WRITE(6,7) LINE,FILE
IF (TYPE.EQ.8) WRITE(6,8) LINE,FILE
IF (TYPE.EQ.9) WRITE(6,9) FILE
IF (TYPE.EQ.10) WRITE(6,10) FILE,LINE
IF (TYPE.EQ.11) WRITE(6,11)
IF (TYPE.EQ.12) WRITE(6,12) FILE
IF (TYPE.EQ.13) WRITE(6,13) FILE
IF (TYPE.EQ.14) WRITE(6,14) LINE,FILE
IF (TYPE.EQ.15) WRITE(6,15) LINE,FILE
IF (TYPE.EQ.16) WRITE(6,16) LINE,FILE
```

```
IF (TYPE.EQ.1) WRITE(7,1) LINE,FILE
IF (TYPE.EQ.2) WRITE(7,2) LINE,FILE
IF (TYPE.EQ.3) WRITE(7,3) LINE,FILE
IF (TYPE.EQ.4) WRITE(7,4) LINE,FILE
IF (TYPE.EQ.5) WRITE(7,5) LINE,FILE
IF (TYPE.EQ.6) WRITE(7,6) LINE,FILE
IF (TYPE.EQ.7) WRITE(7,7) LINE,FILE
IF (TYPE.EQ.8) WRITE(7,8) LINE,FILE
IF (TYPE.EQ.9) WRITE(7,9) FILE
IF (TYPE.EQ.10) WRITE(7,10) FILE,LINE
IF (TYPE.EQ.11) WRITE(7,11)
IF (TYPE.EQ.12) WRITE(7,12) FILE
IF (TYPE.EQ.13) WRITE(7,13) FILE
IF (TYPE.EQ.14) WRITE(7,14) LINE,FILE
IF (TYPE.EQ.15) WRITE(7,15) LINE,FILE
IF (TYPE.EQ.16) WRITE(7,16) LINE,FILE
```

NERR=NERR+1

RETURN

END

SUBROUTINE CLRDET

C THIS SUBROUTINE 'CLEARS OUT' YET(999) BY SETTING TO .FALSE.
C AND RESETS LASTI TO 0.

COMMON ITEM(999,2),NERR,YET(999),LASTI,MAXI
INTEGER I,NERR,MAXI, LASTI
LOGICAL YET,ITEM

DO 10 I=1,MAXI
YET(I)=.FALSE.
10 CONTINUE

LASTI=0

RETURN

END

SUBROUTINE CHECK1(I,FILE)

C THIS SUBROUTINE PERFORMS ERROR TESTS ON ITEM I IN FILE 'FILE'

COMMON ITEM(999,2),NERR,YET(999),LASTI,MAXI
LOGICAL ITEM,YET
INTEGER I,FILE,NERR,LASTI,MAXI

C CHECK IF THIS ITEM IS IN SEQUENCE
IF (I.LT.LASTI) CALL ERROR(5,I,FILE)
LASTI=I

C CHECK IF THIS ITEM APPEARED IN THE INITIAL FILE
IF (.NOT.ITEM(I,1)) CALL ERROR(2,I,FILE)

C CHECK IF THIS ITEM HAS ALREADY APPEARED IN THIS FILE
IF (YET(I)) CALL ERROR(1,I,FILE)
YET(I)=.TRUE.

RETURN

END

SUBROUTINE CHECK2(FILE)

C THIS SUBROUTINE CHECK WHICH ITEMS ARE MISSING FROM FILE 'FILE'.
C NOTE THAT CHECK2 ONLY TEST FOR "ITEMNESS" AND NOT "LRUNESS", SO
C THAT FILES INDEXED BY LRU ITEM ARE CHECKED IN THE MAIN PROGRAM.

COMMON ITEM(999,2),NERR,YET(999),LASTI,MAXI
INTEGER I,FILE,NERR,LASTI,MAXI
LOGICAL ITEM,YET

DO 10 I=1,MAXI
IF (ITEM(I,1).AND..NOT.YET(I)) CALL ERROR(4,I,FILE)
10 CONTINUE

RETURN

END

APPENDIX E

LCC PROGRAM FORTRAN SOURCE CODE

```

C*****
C
C
C*****
C
COMMON /LDFPR / LDFPR
COMMON /LDSRU / LDSRU
COMMON /LDFR / LDFR
COMMON /LDUP / LDUP
COMMON /LDCOND/ LDCOND
COMMON /LDNRTS/ LDNRTS
COMMON /LDRTS / LDRTS
COMMON /LDRM / LDRM
COMMON /LDERV / LDERV
COMMON /EXITXX/ EXITXX
INTEGER EXITXX
COMMON /ITERXX/ ITERXX
COMMON /PRNTXX/ PRNTXX
INTEGER PRNTXX
COMMON /RERDXX/ RERDXX
INTEGER RERDXX
COMMON /NERRXX/ NERRXX
COMMON /NERRYY/ NERRYY
COMMON /REDOXX/ REDOXX
INTEGER REDOXX
C
1 FORMAT(1H1//22H PROGRAM STOPS DUE TO ,I4,
+ 16H ERRORS ON INPUT)
C
C
C
C
C*****
C* INITIALIZE SENSITIVITY PRINT PARAMETERS *
C*****
C
CALL SSETXX
C
C
C
C*****
C* EACH PASS THROUGH THIS LOOP REPRESENTS A SINGLE LCC CALCULATION *
C*****

```

```

C
    DO 888 ITERXX=1,50
C
C
C
C*****
C*  PROMPT THE USER FOR CONTROL VARIABLES PRNTXX,MAXPMT,XTITLE,RERDXX, *
C*  AND FULLXX. *
C*****
C
    EXITXX=0
    CALL PRMPT1
C
C
C
C*****
C*  IF THIS IS THE FIRST ITERATION OR IF THE USER REQUESTED TO *
C*  REREAD THE FILES, INITIALIZE VARIABLES AND READ THE INPUT FILES *
C*****
C
    IF(PRNTXX.NE.0) CALL TITLE
    IF(ITERXX.NE.1.AND.RERDXX.NE.1) GO TO 2
    CALL INITAL
    NERRXX=0
    REWIND 11
    REWIND 12
    REWIND 13
    REWIND 14
    REWIND 15
    REWIND 16
    REWIND 17
    REWIND 18
    REWIND 19
    REWIND 20
    REWIND 21
    REWIND 22
    CALL READ1
    CALL READ2
    CALL READ3
    CALL READ4
    CALL READ5
    CALL READ6
    CALL READ7
    CALL READ8
    CALL READ9A
    CALL READ9B
    CALL READ10

```

```

        CALL READ11
        NERRYY=NERRXX
    2 CONTINUE
C
C
C
C*****
C*  READ NAMELISTS /GO1/, /GO2/ AND /SENS/ TO OVERRIDE      *
C*  VARIABLE VALUES WITH INTERACTIVE INPUTS AND TO SET SENSITIVITY  *
C*  OUTPUT FLAGS.                                           *
C*****
C
        CALL PRMPT2
        IF(EXITXX.EQ.1) GO TO 6
        CALL PRMPT3
        IF(EXITXX.EQ.1) GO TO 6
C
C
C
C*****
C*  PERFORM ANY ERROR CHECKS OR CALCULATIONS WHICH SHOULD      *
C*  BE PERFORMED BEFORE THE INPUT TABLES ARE PRINTED.      *
C*****
C
        CALL INITAX
C
C
C
C*****
C*  PRINT THE INPUT DATA VALUES.                             *
C*****
C
        CALL ERRCK1
        CALL RLCOMP
        CALL OTABST
        CALL ITAB1A
        CALL ITAB1B
        CALL ITAB1C
        CALL ITAB2
        CALL ITAB3
        CALL ITAB4
        CALL ITAB5
        CALL ITAB6
        CALL ITAB7
        CALL ITAB8
        CALL ITAB9A
        CALL ITAB9B

```

```

CALL ITB10A
CALL ITB10B
CALL ITB10C
CALL ITB10D
CALL ITAB11

C
C
C
C*****
C*  STOP IF ANY ERRORS WERE FOUND ON INPUT.          *
C*****
C
      NERRXX=NERRYY
      IF(NERRXX.EQ.0) GO TO 4
      WRITE(6,1) NERRXX
      IF(PRNTXX.NE.0) WRITE(7,1) NERRXX
      STOP
4 CONTINUE

C
C
C
C*****
C*  LCC CALCULATIONS                                *
C*****
C
      CALL ZFAIL
      CALL ZNFB
      CALL ZERHB
      CALL ZERHSE
      CALL ZISET
      CALL ZUSE
      CALL ZTYPE
      CALL ZTFR
      CALL ZSECI
      CALL ZPMEQ
      CALL ZTISQ
      CALL ZYRSQ
      CALL ZTOTPQ
      CALL ZLC
      CALL COST1
      CALL COST2
      CALL COST3
      CALL COST4
      CALL COST5
      CALL COST6
      CALL COST7
      CALL COST8

```

```

CALL COST9
CALL COST10
CALL COST11

C
C
C
C*****
C* SENSITIVITY CALCULATIONS *
C*****
C
CALL DPIUP
CALL DDMF
CALL DRM
CALL DXRM
CALL DXUC
IF (LDUP .NE.0.OR.(PRNTXX.NE.0.AND.LDERV .NE.0)) CALL DUP
CALL DFR
CALL DXFR
CALL DFPR
CALL DXFPR
IF (LDRTS .NE.0.OR.(PRNTXX.NE.0.AND.LDERV .NE.0)) CALL DRTS
IF (LDNRTS.NE.0.OR.(PRNTXX.NE.0.AND.LDERV .NE.0)) CALL DNRTS
IF (LDCOND.NE.0.OR.(PRNTXX.NE.0.AND.LDERV .NE.0)) CALL DCOND
IF (LDSRU .NE.0.OR.(PRNTXX.NE.0.AND.LDERV .NE.0)) CALL DSRU
CALL DXMIL

C
C
C
C*****
C* TELL THE USER TO ADJUST TERMINAL TO A NEW PAGE *
C*****
C
CALL PRMPT4
IF(EXITXX.EQ.1) GO TO 6

C
C
C
C*****
C* PRINT OUTPUT TABLES AT TERMINAL AND/OR OFFLINE PRINTER *
C*****
C
CALL OTAB1
CALL OTAB2
CALL OTAB3A
CALL OTAB3B
CALL OTAB3C
CALL OTAB4A

```

```

CALL OTAB4B
CALL OTAB4C
CALL OTAB5
CALL OTAB6
CALL OTAB7
CALL RLAPRT
CALL PRMPT5
IF(EXITXX.EQ.1) GO TO 6
C
C
C
C*****
C* PRINT SENSITIVITY TABLE AT TERMINAL AND/OR OFFLINE PRINTER *
C*****
C
CALL OSENS
C
C
C
C*****
C* ASK THE USER WHETHER A' OTHER RUN IS DESIRED. *
C*****
C
6 CALL PRMPT6
IF(REDOXX.EQ.0) GO TO 999
888 CONTINUE
C
999 STOP
C
END

```

SUBROUTINE READ1

```

C
C***** 800827 110323454 *****
C* SSS MOD LCR *
C* READS THE MISCELLANEOUS SCALAR DATA *
C* FILE FROM CHANNEL 11 *
C*****
C
COMMON /ACPP/ ACPP
COMMON /BAA/ BAA
COMMON /BDATA/ BDATA
INTEGER BDATA
COMMON /BF/ BF
COMMON /BIRD/ BIRD
COMMON /BLR/ BLR
COMMON /BMF/ BMF
COMMON /BRCT/ BRCT
COMMON /CFG/ CFG(3)
COMMON /CPD1/ CPD1
COMMON /CPD2/ CPD2
COMMON /CPPC/ CPPC
COMMON /CPPD/ CPPD(3)
COMMON /CRCT/ CRCT
COMMON /DAA/ DAA
COMMON /DAD/ DAD
COMMON /DDATA/ DDATA
INTEGER DDATA
COMMON /DLR/ DLR
COMMON /DMF/ DMF
COMMON /DRCT/ DRCT(3)
COMMON /FSED/ FSED
COMMON /HPD1/ HPD1
INTEGER HPD1
COMMON /HPD2/ HPD2
INTEGER HPD2
COMMON /IMC/ IMC
REAL IMC
COMMON /KFAC/ KFAC(4)
REAL KFAC
COMMON /MILR/ MILR(3)
REAL MILR
COMMON /MRF/ MRF
REAL MRF
COMMON /MRO/ MRO
REAL MRO
COMMON /MUSE/ MUSE
REAL MUSE

```

```

COMMON /NRUC/ NRUC
REAL NRUC
COMMON /OST/ OST(3)
COMMON /OSTC/ OSTC
COMMON /PAL1/ PAL1
COMMON /PAL2B/ PAL2B
COMMON /PAL2D/ PAL2D
COMMON /PIUP/ PIUP
COMMON /PMLR/ PMLR
COMMON /QTYP1/ QTYP1
INTEGER QTYP1
COMMON /QTYP2B/ QTYP2B
INTEGER QTYP2B
COMMON /QTYP2D/ QTYP2D
INTEGER QTYP2D
COMMON /R/ R
INTEGER R
COMMON /RCPP/ RCPP
COMMON /RMC/ RMC
COMMON /SA/ SA
COMMON /SPC1/ SPC1
INTEGER SPC1
COMMON /SPC2/ SPC2
INTEGER SPC2
COMMON /SR/ SR
COMMON /TEFM/ TEFM
COMMON /TNLR/ TNLR
COMMON /TORB/ TORB
COMMON /TORD/ TORD
COMMON /TR/ TR
COMMON /TRAVB/ TRAVB
COMMON /TRAV1D/ TRAV1D
COMMON /TYP2TF/ TYP2TF
COMMON /UCPP/ UCPP
COMMON /XFPR/ XFPR
COMMON /XFR/ XFR
COMMON /XMIL/ XMIL
COMMON /XUC/ XUC
NAMELIST /MISC/ BF,BAA,BLR,BMF,BRCT,CFG,CPPC,CPPD,CRCT,DAA,DAD,DLR
+,DMF,DRCT,IMC,KFAC,MILR,MUSE,NRUC,OST,OSTC,PIUP,PMLR,RMC,SA,TNLR,X
+XFR,XFR,XMIL,XUC,HPD2,TORB,TORD,MRO,MRF,SR,TR,PAL1,PAL2B,PAL2D,TRA
+V1D,TRAVB,ACPP,CPD2,RCPP,UCPP,BIRD,QTYP1,QTYP2B,QTYP2D,SPC2,TYP2TF
+,BDATA,CPD1,DDATA,FSADC,HPD1,R,SPC1,TEFM

```

C
C
C

```

READ(11,MISC)

```

RETURN
END

SUBROUTINE READ2

800827 110338677

```

C
C*****
C* SSS MOD LCR *
C* READS THE BASE CONFIGURATION FILE *
C* FROM CHANNEL 12 *
C*****
C
COMMON /BNOUN/ BNOUN(16,16)
COMMON /BPLAT/ bPLAT(16)
INTEGER bPLAT
COMMON /BSP/ BSP(16)
INTEGER BSP
COMMON /BTYPE/ BTYPE(16)
INTEGER BTYPE
COMMON /LO/ LO(16)
COMMON /MXNS/ MXNS
COMMON /NBC/ NBC(16)
REAL NBC
COMMON /NHB/ NHB(16)
COMMON /NS/ NS
COMMON /TNB/ TNB(16)
COMMON /NERRXX/ NERRXX
COMMON /PRNTXX/ PRNTXX
INTEGER PRNTXX
DATA XXSTAR/1H*/
1 FORMAT(A1,I2,16A1,F3.0,3I2,F3.0,2I2)
2 FORMAT(A1)
3 FORMAT(/49H UNIT 12 ERRGR: END OF FILE CARD NOT FOUND AFTER/
+17X,37HMAXIMUM NUMBER OF CARDS WERE READ IN.)
C
C
MXNS=0
DO 210 NS=1,16
  READ(12, 1) XXCOL1,NS1,(BNOUN(NS,I1),I1=1,16),TNB(NS),LO(NS),
+ BTYPE(NS),NHB(NS),NBC(NS),BPLAT(NS),BSP(NS)
  IF(XXCOL1.EQ.XXSTAR) RETURN
  MXNS=NS
210 CONTINUE
C
  READ(12, 2) XXCOL1
  IF(XXCOL1.EQ.XXSTAR) RETURN
  NERRXX=NERRXX+1
  WRITE(6, 3)
  IF(PRNTXX.NE.0) WRITE(7, 3)
C
  RETURN
END

```

SUBROUTINE READ3

800827 110350351

```
C*****
C* READS THE PLATFORM OPERATIONAL DATA FILE *
C* FROM CHANNEL 13 *
C*****
```

C

```
COMMON /AMPM/ AMPM(10,3)
COMMON /APFH/ APFH(10,3)
COMMON /FGH/ FGH(10)
COMMON /LE/ LE(10)
COMMON /M/ M
COMMON /MMPD/ MMPD(10,3)
REAL MMPD
COMMON /MMPM/ MMPM(10)
REAL MMPM
COMMON /MXNP/ MXNP
COMMON /NP/ NP
COMMON /PNOUN/ PNOUN(10,12)
COMMON /TFAC/ TFAC(10)
COMMON /THRS/ THRS(10)
COMMON /NERRXX/ NERRXX
COMMON /PRNTXX/ PRNTXX
INTEGER PRNTXX
DATA XXSTAR/1H*/
1 FORMAT(A1,I3,I2A1,I2,3F4.0,F3.2,4X,F3.1,3F4.1,3F5.1,F7.0,F5.0)
2 FORMAT(A1)
3 FORMAT(/49H UNIT 13 ERROR: END OF FILE CARD NOT FOUND AFTER/
+17X,37HMAXIMUM NUMBER OF CARDS WERE READ IN.)
```

C

C

```
MXNP=0
DO 210 NP=1,10
  READ(13, 1) XXCOL1,NP1,(PNOUN(NP,K1),K1=1,12),LE(NP),(APFH(NP,
+ M),M=1,3),TFAC(NP),MMPM(NP),(AMPM(NP,M),M=1,3),(MMPD(NP,M),
+ M=1,3),THRS(NP),FGH(NP)
  IF(XXCOL1.EQ.XXSTAR) RETURN
  MXNP=NP
210 CONTINUE
```

C

```
READ(13, 2) XXCOL1
IF(XXCOL1.EQ.XXSTAR) RETURN
NERRXX=NERRXX+1
WRITE(6, 3)
IF(PRNTXX.NE.0) WRITE(7, 3)
```

C

```
RETURN
END
```

SUBROUTINE READ4

```

C
C***** 800827 110358018 *****
C* SSS MOD LCR *
C* READS THE PLATFORM TERMINAL COST & NONRECURRING MOD/I *
C* DATA FILE FROM CHANNEL 14 *
C*****
C
COMMON /DRAG/ DRAG(10)
COMMON /FR/ FR(3,10)
COMMON /INTNR/ INTNR(10)
REAL INTNR
COMMON /INTR/ INTR(10)
REAL INTR
COMMON /K/ K(10)
REAL K
COMMON /M/ M
COMMON /MXNP/ MXNP
COMMON /NAE/ NAE(10)
REAL NAE
COMMON /NP/ NP
COMMON /NRMI/ NRMI(10)
REAL NRMI
COMMON /NTRMP/ NTRMP(10)
REAL NTRMP
COMMON /PDIV/ PDIV(10)
COMMON /NERRXX/ NERRXX
COMMON /PRNTXX/ PRNTXX
INTEGER PRNTXX
DATA XXSTAR/1H*/
1 FORMAT(A1,I3,F4.2,F9.0,F8.0,F9.0,F4.2,3F3.2,F3.0,2F4.2)
2 FORMAT(A1)
3 FORMAT(/49H UNIT 14 ERROR: END OF FILE CARD NOT FOUND AFTER/
+17X,37HMAXIMUM NUMBER OF CARDS WERE READ IN.)
C
C
MXNP=0
DO 210 NP=1,10
    READ(14, 1) XXCOL1,NP1,NTRMP(NP),INTNR(NP),INTR(NP),NRMI(NP),
+    PDIV(NP),(FR(M,NP),M=1,3),DRAG(NP),K(NP),NAE(NP)
    IF(XXCOL1.EQ.XXSTAR) RETURN
    MXNP=NP
210 CONTINUE
C
    READ(14, 2) XXCOL1
    IF(XXCOL1.EQ.XXSTAR) RETURN
    NERRXX=NERRXX+1
    WRITE(6, 3)

```

C IF (PRNTXX.NE.0) WRITE(7, 3)
 RETURN
 END

SUBROUTINE READ5

800827 110417429

```

C
C*****
C* SSS MOD LCR *
C* READS THE PLATFORM RECURRING MOD/INSTALLATION *
C* DATA FILE FROM CHANNEL 15 *
C*****
C
COMMON /AKIT/ AKIT(4,10)
COMMON /IA/ IA
COMMON /M/ M
COMMON /MIFIX/ MIFIX(3,10)
REAL MIFIX
COMMON /MIMH/ MIMH(4,3,10)
REAL MIMH
COMMON /MXNP/ MXNP
COMMON /NP/ NP
COMMON /NERRXX/ NERRXX
COMMON /PRNTXX/ PRNTXX
INTEGER PRNTXX
DATA XXSTAR/1H*/
1 FORMAT(A1,I3,7F5.0,12F3.0)
2 FORMAT(A1)
3 FORMAT(/49H UNIT 15 ERROR: END OF FILE CARD NOT FOUND AFTER/
+17X,37HMAXIMUM NUMBER OF CARDS WERE READ IN.)
C
C
MXNP=0
DO 210 NP=1,10
READ(15, 1) XXCOL1,NP1,(MIFIX(M,NP),M=1,3),(AKIT(IA,NP),IA=1,4),
+ ((MIMH(IA,M,NP),IA=1,4),M=1,3)
IF(XXCOL1.EQ.XXSTAR) RETURN
MXNP=NP
210 CONTINUE
C
READ(15, 2) XXCOL1
IF(XXCOL1.EQ.XXSTAR) RETURN
NERRXX=NERRXX+1
WRITE(6, 3)
IF(PRNTXX.NE.0) WRITE(7, 3)
C
RETURN
END

```

SUBROUTINE READ6

800827 110436281

```

C
C*****
C* READS THE PLATFORM DEPLOYMENT AT BASES *
C* DATA FILE FROM CHANNEL 16 *
C*****
C
COMMON /MXNP/ MXNP
COMMON /NP/ NP
COMMON /NPLT/ NPLT(10,16)
REAL NPLT
COMMON /NS/ NS
COMMON /NERRXX/ NERRXX
COMMON /PRNTXX/ PRNTXX
INTEGER PRNTXX
DATA XXSTAR/1H*/
1 FORMAT(A1,I3,16F4.2)
2 FORMAT(A1)
3 FORMAT(/49H UNIT 16 ERROR: END OF FILE CARD NOT FOUND AFTER/
+17X,37HMAXIMUM NUMBER OF CARDS WERE READ IN.)
C
C
MXNP=0
DO 210 NP=1,10
    READ(16, 1) XXCOL1,NP1,(NPLT(NP,NS),NS=1,16)
    IF(XXCOL1.EQ.XXSTAR) RETURN
    MXNP=NP
210 CONTINUE
C
    READ(16, 2) XXCOL1
    IF(XXCOL1.EQ.XXSTAR) RETURN
    NERRXX=NERRXX+1
    WRITE(6, 3)
    IF(PRNTXX.NE.0) WRITE(7, 3)
C
    RETURN
END

```

SUBROUTINE READ7

800827 110450953

C
 C*****
 C* SSS MOD LCR *
 C* READS THE SUPPORT EQUIPMENT DATA *
 C* FILE FROM CHANNEL 17 *
 C*****
 C

COMMON /CSE/ CSE(250)
 COMMON /DATAS/ DATAS(250)
 INTEGER DATAS
 COMMON /L/ L
 COMMON /MSE/ MSE(250)
 REAL MSE
 COMMON /MXL/ MXL
 COMMON /SEDEV/ SEDEV(250)
 COMMON /SEINO/ SEINO(250)
 INTEGER SEINO
 COMMON /SENOUN/ SENOUN(250,20)
 COMMON /SENUM/ SENUM(250,12)
 COMMON /SETYPE/ SETYPE(250)
 INTEGER SETYPE
 COMMON /NERRXX/ NERRXX
 COMMON /PRNTXX/ PRNTXX
 INTEGER PRNTXX
 DATA XXSTAR/1H*/
 1 FORMAT(A1,I3,20A1,12A1,F7.0,F4.3,I2,I3,F8.0)
 2 FORMAT(A1)
 3 FORMAT(/49H UNIT 17 ERROR: END OF FILE CARD NOT FOUND AFTER/
 +17X,37HMAXIMUM NUMBER OF CARDS WERE READ IN.)

C
 C
 MXL=0
 DO 210 IXXX1=1,250
 READ(17, 1) XXCOL1,L,(SENOUN(L,I1),I1=1,20),(SENUM(L,I2),I2=1,
 + 12),CSE(L),MSE(L),SETYPE(L),DATAS(L),SEDEV(L)
 IF(XXCOL1.EQ.XXSTAR) RETURN
 MXL=IXXX1
 SEINO(IXXX1)=L
 210 CONTINUE

C
 READ(17, 2) XXCOL1
 IF(XXCOL1.EQ.XXSTAR) RETURN
 NERRXX=NERRXX+1
 WRITE(6, 3)
 IF(PRNTXX.NE.0) WRITE(7, 3)
 C

RETURN
END

SUBROUTINE READ8

800827 110458958

C
 C*****
 C* SSS MOD LCR *
 C* READS THE ITEM EQUIPMENT DATA FILE *
 C* FROM CHANNEL 18 *
 C*****

C

COMMON /GFE/ GFE(999)
 INTEGER GFE
 COMMON /I/ I
 COMMON /INO/ INO(999)
 COMMON /INOUN/ INOUN(999,24)
 REAL INOUN
 COMMON /INTEG/ INTEG(999)
 REAL INTEG
 COMMON /LFAC/ LFAC(999)
 REAL LFAC
 COMMON /LRU/ LRU(999)
 COMMON /MXI/ MXI
 COMMON /NHI/ NHI(999)
 COMMON /PA/ PA(999)
 COMMON /PTNUM/ PTNUM(999,12)
 COMMON /RM/ RM(999)
 COMMON /UP/ UP(999)
 COMMON /WT/ WT(999)
 COMMON /NERRXX/ NERRXX
 COMMON /PRNTXX/ PRNTXX
 INTEGER PRNTXX
 DATA XXSTAR/1H*/
 1 FORMAT(A1,I3,24A1,12A1,F3.2,I2,I3,2I2,F6.0,F4.3,F5.2,F4.2)
 2 FORMAT(A1)
 3 FORMAT(/49H UNIT 18 ERROR: END OF FILE CARD NOT FOUND AFTER/
 +17X,37HMAXIMUM NUMBER OF CARDS WERE READ IN.)

C

C

MXI=0
 DO 210 IXXX1=1,999
 READ(18, 1) XXCOL1,I,(INOUN(I,K1),K1=1,24),(PTNUM(I,K2),K2=1,
 + 12),LFAC(I),LRU(I),NHI(I),GFE(I),INTEG(I),UP(I),RM(I),WT(I),
 + PA(I)
 IF(XXCOL1.EQ.XXSTAR) RETURN
 MXI=IXXX1
 INO(IXXX1)=I
 210 CONTINUE

C

READ(18, 2) XXCOL1

```
IF(XXCOL1.EQ.XXSTAR) RETURN  
NERRXX=NERRXX+1  
WRITE(6, 3)  
IF(PRNTXX.NE.0) WRITE(7, 3)
```

C

```
RETURN  
END
```

SUBROUTINE READ9A

```

C
C***** 800827 110508404
C* READS THE ITEM MAINTENANCE DATA FILE *
C* FROM CHANNEL 19 *
C*****
C
COMMON /BCMH/ BCMH(999)
COMMON /BMH/ BMH(999)
COMMON /COND/ COND(999)
COMMON /DMH/ DMH(999)
COMMON /FPR/ FPR(999)
COMMON /I/ I
COMMON /INO/ INO(999)
COMMON /IPCF/ IPCF(999)
REAL IPCF
COMMON /MTBMI/ MTBMI(999,4)
REAL MTBMI
COMMON /MXI/ MXI
COMMON /NRTS/ NRTS(999)
REAL NRTS
COMMON /RIP/ RIP(999)
COMMON /RL/ RL(999)
INTEGER RL
COMMON /RMH/ RMH(999)
COMMON /RTS/ RTS(999)
COMMON /WEAR/ WEAR(999)
COMMON /NERRXX/ NERRXX
COMMON /PRNTXX/ PRNTXX
INTEGER PRNTXX
DATA XXSTAR/1H*/
1 FORMAT(A1,I3,4F8.0,F4.3,F3.2,F5.2,3F4.3,4F4.2,I1)
2 FORMAT(A1)
3 FORMAT(/49H UNIT 19 ERROR: END OF FILE CARD NOT FOUND AFTER/
+17X,37HMAXIMUM NUMBER OF CARDS WERE READ IN.)
4 FORMAT(1X,37HINPUT ERROR: INPUT FILE READ IN FROM/
+51HCHANNEL # 19 CONTAINS FEWER ITEMS THAN ITITIAL FILE)
5 FORMAT(1X,49HINPUT ERROR: INDEXING IN INPUT FILE READ IN FROM/
+46HCHANNEL # 19 IS INCONSISTENT WITH INITIAL FILE)

C
C
DO 210 IXXX1=1,MXI
READ(19, 1) XXCOL1,I,(MTBMI(I,K1),K1=1,4),FPR(I),RIP(I),IPCF(I),
+ RTS(I),NRTS(I),COND(I),RMH(I),BCMH(I),BMH(I),DMH(I),RL(I)
WEAR(I)=COND(I)
IF(XXCOL1.NE.XXSTAR) GO TO 200
NERRXX=NERRXX+1

```

```

        WRITE(6, 4)
        IF(PRNTXX.NE.0) WRITE(7, 4)
        RETURN
200    CONTINUE
        IF(I      .EQ.INO  (IXXX1))GO TO 210
        WRITE(6, 5)
        IF(PRNTXX.NE.0) WRITE(7, 5)
210    CONTINUE
C
        READ(19, 2) XXCOL1
        IF(XXCOL1.EQ.XXSTAR) RETURN
        NERRXX=NERRXX+1
        WRITE(6, 3)
        IF(PRNTXX.NE.0) WRITE(7, 3)
C
        RETURN
        END

```

SUBROUTINE READ9B

```

C
C*****800827 110521080*****
C* SSS MOD LCR *
C* READS THE ITEM MAINTENANCE DATA FILE PART 2 *
C* FROM CHANNEL 22 *
C*****
C
COMMON /DATAB/ DATAB(999)
INTEGER DATAB
COMMON /DATAD/ DATAD(999)
INTEGER DATAD
COMMON /I/ I
COMMON /INO/ INO(999)
COMMON /MXI/ MXI
COMMON /TIME1/ TIME1(999)
INTEGER TIME1
COMMON /UCTDEV/ UCTDEV(999)
COMMON /NERRXX/ NERRXX
COMMON /PRNTXX/ PRNTXX
INTEGER PRNTXX
DATA XXSTAR/1H*/
1 FORMAT(A1,4(I3),F7.0)
2 FORMAT(A1)
3 FORMAT(/49H UNIT 22 ERROR: END OF FILE CARD NOT FOUND AFTER/
+17X,37HMAXIMUM NUMBER OF CARDS WERE READ IN.)
4 FORMAT(1X,37HINPUT ERROR: INPUT FILE READ IN FROM/
+51HCHANNEL # 22 CONTAINS FEWER ITEMS THAN ITITIAL FILE)
5 FORMAT(1X,49HINPUT ERROR: INDEXING IN INPUT FILE READ IN FROM/
+46HCHANNEL # 22 IS INCONSISTENT WITH INITIAL FILE)
C
C
DO 210 IXXX1=1,MXI
READ(22, 1) XXCOL1,I,DATAD(I),DATAB(I),TIME1(I),UCTDEV(I)
IF(XXCOL1.NE.XXSTAR) GO TO 200
NERRXX=NERRXX+1
WRITE(6, 4)
IF(PRNTXX.NE.0) WRITE(7, 4)
RETURN
200 CONTINUE
IF(I .EQ. INO (IXXX1))GO TO 210
WRITE(6, 5)
IF(PRNTXX.NE.0) WRITE(7, 5)
210 CONTINUE
C
READ(22, 2) XXCOL1
IF(XXCOL1.EQ.XXSTAR) RETURN

```

```
NERRXX=NERRXX+1  
WRITE(6, 3)  
IF(PRNTXX.NE.0) WRITE(7, 3)
```

C

```
RETURN  
END
```

SUBROUTINE READ10

800827 110526034

C
 C*****
 C* SSS MOD LCR *
 C* READS THE ITEM/SUPPORT EQUIPMENT *
 C* CROSS-REFERENCE DATA FILE FROM CHANNEL 20 *
 C*****
 C

COMMON /A/ A(999,4,30)
 INTEGER A
 COMMON /I/ I
 COMMON /INO/ INO(999)
 COMMON /IRMIN/ IRMIN(999,4)
 COMMON /K1TEMP/ K1TEMP
 COMMON /MXI/ MXI
 COMMON /NJA/ NJA(999,4)
 COMMON /NRM/ NRM(999)
 COMMON /QSA/ QSA(999,4,30)
 COMMON /NERRXX/ NERRXX
 COMMON /PRNTXX/ PRNTXX
 INTEGER PRNTXX
 DATA XXSTAR/1H*/
 1 FORMAT(A1,I3,2I2,I3,9(I4,F3.0))
 2 FORMAT(A1)
 3 FORMAT(/49H UNIT 20 ERROR: END OF FILE CARD NOT FOUND AFTER/
 +17X,37HMAXIMUM NUMBER OF CARDS WERE READ IN.)
 4 FORMAT(1X,37HINPUT ERROR: INPUT FILE READ IN FROM/
 +51HCHANNEL # 20 CONTAINS FEWER ITEMS THAN ITITIAL FILE)
 5 FORMAT(1X,49HINPUT ERROR: INDEXING IN INPUT FILE READ IN FROM/
 +46HCHANNEL # 20 IS INCONSISTENT WITH INITIAL FILE)

C
 C

DO 230 IXX1=1,MXI
 READ(20, 1) XXCOL1,I,NRM(I),IRMIN(I,1),NJA(I,1),(A(I,1,K2),
 + QSA(I,1,K2),K2=1,9)
 IF(XXCOL1.NE.XXSTAR) GO TO 200
 NERRXX=NERRXX+1
 WRITE(6, 4)
 IF(PRNTXX.NE.0) WRITE(7, 4)
 RETURN
 200 CONTINUE
 IF(I .EQ. INO (IXX1))GO TO 210
 WRITE(6, 5)
 IF(PRNTXX.NE.0) WRITE(7, 5)
 IF(.NOT.(NJA(I,K1TEMP).GT.9)) GO TO 210
 ITER=(NJA(I,1)-1)/9

C

```

DO 205 K2TEMP=1,ITER
K3=MIN0(K2+9,30)
K4=K2+1
C
      READ(20,6)(A(I,1,K2),QSA(I,1,K2),K2=K4,K3)
6      FORMAT(11X,9(I4,F3.0))
205      CONTINUE
210      CONTINUE
      IF(.NOT.(NRM(I).GT.1)) GO TO 220
      INRM=NRM(I)
C
      DO 215 K1TEMP=2,INRM
      READ(20,7)IRMIN(I,K1TEMP),NJA(I,K1TEMP),
+      (A(I,K1TEMP,K2),QSA(I,K1TEMP,K2),K2=1,9)
7      FORMAT(6X,I2,I3,9(I4,F3.0))
      IF (.NOT.(NJA(I,K1TEMP).GT.9)) GO TO 214
      ITER=(NJA(I,K1TEMP)-1)/9
      DO 212 K2TEMP=1,ITER
      K3=MIN0(K2+9,30)
      K4=K2+1
      READ(20,8) (A(I,K1TEMP,K2),
+      QSA(I,K1TEMP,K2),K2=K4,K3)
8      FORMAT(11X,9(I4,F3.0))
212      CONTINUE
214      CONTINUE
215      CONTINUE
220      CONTINUE
230      CONTINUE
C
      READ(20, 2) XXCOL1
      IF(XXCOL1.EQ.XXSTAR) RETURN
      NERRXX=NERRXX+1
      WRITE(6, 3)
      IF(PRNTXX.NE.0) WRITE(7, 3)
C
      RETURN
      END

```

SUBROUTINE READ11

800827 110556202

C
 C*****
 C* BASELINE CHANGE *
 C* READS THE ITEM CONFIGURATION DATA *
 C* FILE FROM CHANNEL 21 *
 C*****

C
 COMMON /I/ I
 COMMON /INO/ INO(999)
 COMMON /MXI/ MXI
 COMMON /NITEM/ NITEM(999,10)
 REAL NITEM
 COMMON /NP/ NP
 COMMON /NERRXX/ NERRXX
 COMMON /PRNTXX/ PRNTXX
 INTEGER PRNTXX
 DATA XXSTAR/1H*/
 1 FORMAT(A1,I3,10F4.2)
 2 FORMAT(A1)
 3 FORMAT(/49H UNIT 21 ERROR: END OF FILE CARD NOT FOUND AFTER/
 +17X,37HMAXIMUM NUMBER OF CARDS WERE READ IN.)
 4 FORMAT(1X,37HINPUT ERROR: INPUT FILE READ IN FROM/
 +51HCHANNEL # 21 CONTAINS FEWER ITEMS THAN ITITIAL FILE)
 5 FORMAT(1X,49HINPUT ERROR: INDEXING IN INPUT FILE READ IN FROM/
 +46HCHANNEL # 21 IS INCONSISTENT WITH INITIAL FILE)

C
 C
 DO 210 IXXX1=1,MXI
 READ(21, 1) XXCOL1,I,(NITEM(I,NP),NP=1,10)
 IF(XXCOL1.NE.XXSTAR) GO TO 200
 NERRXX=NERRXX+1
 WRITE(6, 4)
 IF(PRNTXX.NE.0) WRITE(7, 4)
 RETURN
 200 CONTINUE
 IF(I .EQ. INO (IXXX1))GO TO 210
 WRITE(6, 5)
 IF(PRNTXX.NE.0) WRITE(7, 5)
 210 CONTINUE

C
 READ(21, 2) XXCOL1
 IF(XXCOL1.EQ.XXSTAR) RETURN
 NERRXX=NERRXX+1
 WRITE(6, 3)
 IF(PRNTXX.NE.0) WRITE(7, 3)

C

RETURN
END

SUBROUTINE INITAX

```
C
C***** 800827 110603116
C* SSS MODS *
C* INITIALIZATION OF MODEL PARAMETERS *
C*****
C
```

```
COMMON /AFC/ AFC
COMMON /AFMC/ AFMC
COMMON /BAFC/ BAFC(6)
COMMON /BIIMC/ BIIMC(6)
COMMON /BISC/ BISC(6)
COMMON /BMTRC/ BMTRC
COMMON /BOFMC/ BOFMC(6)
COMMON /BOLC/ BOLC(6)
COMMON /BONMC/ BONMC(6)
COMMON /BRSC/ BRSC(6)
COMMON /BS/ BS(999)
COMMON /BSECC/ BSECC(6)
COMMON /BSECP/ BSECP(6)
COMMON /BTCDI/ BTCDI
COMMON /BXTRC/ BXTRC
COMMON /DMTRC/ DMTRC
COMMON /ERHAB/ ERHAB(250,16)
COMMON /ERHAD/ ERHAD(250)
COMMON /ERHD/ ERHD(999)
COMMON /FAIL/ FAIL(999,16)
COMMON /FPLT/ FPLT(999)
COMMON /FPM/ FPM(999)
COMMON /I/ I
COMMON /IIMCA/ IIMCA(999)
REAL IIMCA
COMMON /IIMCB/ IIMCB
REAL IIMCB
COMMON /IIMCD/ IIMCD
REAL IIMCD
COMMON /IIMCI/ IIMCI
REAL IIMCI
COMMON /IIMCR/ IIMCR
REAL IIMCR
COMMON /IMTRC/ IMTRC
REAL IMTRC
COMMON /INO/ INO(999)
COMMON /ISCB/ ISCB
REAL ISCB
COMMON /ISCD/ ISCD
REAL ISCD
```

```

COMMON /ISET/ ISET(250,16)
REAL ISET
COMMON /ISETD/ ISETD(250)
REAL ISETD
COMMON /L/ L
COMMON /MTRC/ MTRC
REAL MTRC
COMMON /MTRCI/ MTRCI(999)
REAL MTRCI
COMMON /MXI/ MXI
COMMON /MXL/ MXL
COMMON /MXNS/ MXNS
COMMON /NFD/ NFD(999)
REAL NFD
COMMON /NS/ NS
COMMON /OFMCA/ OFMCA(999)
COMMON /OFMCB/ OFMCB
COMMON /OFMCD/ OFMCD
COMMON /OLCP/ OLCP
COMMON /OLCT/ OLCT
COMMON /ONMC/ ONMC
COMMON /ONMCA/ ONMCA(999)
COMMON /RSC/ RSC
COMMON /SECBC/ SECBC
COMMON /SECBP/ SECBP
COMMON /SECC/ SECC
COMMON /SECDC/ SECDC
COMMON /SECDP/ SECDP
COMMON /SECIC/ SECIC
COMMON /SECII/ SECII
COMMON /SEFCIP/ SECIP
COMMON /SECP/ SECP
COMMON /SECR/ SECR
COMMON /SECRC/ SECRC
COMMON /SECRP/ SECRP
COMMON /SEDC/ SEDC
COMMON /SEINO/ SEINO(250)
INTEGER SEINO
COMMON /SEPC/ SEPC
COMMON /SETDC/ SETDC(250)
COMMON /STDC/ STDC
COMMON /STDCI/ STDCI
COMMON /STDCR/ STDCR
COMMON /TDC/ TDC(999)
COMMON /TERHB/ TERHB(250)
COMMON /TERHD/ TERHD(250)
COMMON /TERMH/ TERMH

```

```

COMMON /TERMI/ TERMI
COMMON /TIAC/ TIAC(999)
COMMON /TUCTDC/ TUCTDC
COMMON /USE/ USE(250,16)
COMMON /USED/ USED(250)
COMMON /XITEMQ/ XITEMQ(999)

```

C
C

```

DO 220 IXXX1=1,MXL
  L=SEINO(IXXX1)
  TERHB(L)=0.
  TERHD(L)=0.
  SETDC(L)=0.
  ERHAD(L)=0.
  ISETD(L)=0.
  USED(L)=1.
  DO 210 NS=1,MXNS
    ERHAB(L,NS)=0.
    ISET(L,NS)=0.
    USE(L,NS)=1.
210  CONTINUE
220  CONTINUE
    DO 240 IXXX1=1,MXI
      I=INO(IXXX1)
      NFD(I)=0.
      ERHD(I)=0.
      XITEMQ(I)=0.
      BS(I)=0.
      TIAC(I)=0.
      TDC(I)=0.
      FPM(I)=0.
      MTRCI(I)=0.
      FPLT(I)=0.
      ONMCA(I)=0.
      OFMCA(I)=0.
      IIMCA(I)=0.
      DO 230 NS=1,MXNS
        FAIL(I,NS)=0.
230  CONTINUE
240  CONTINUE
      TERMH=0.
      TERMI=0.
      ISCB=0.
      ISCD=0.
      RSC=0.
      ONMC=0.
      OFMCB=0.

```

OFMCD=0.
SEPC=0.
SEDC=0.
TUCTDC=0.
SECI=0.
SECR=0.
IIMCB=0.
IIMCD=0.
IIMCI=0.
IIMCR=0.
OLCP=0.
OLCT=0.
AFC=0.
BSECC(1)=0.
BSECC(2)=0.
BSECC(3)=0.
BSECC(4)=0.
BSECC(5)=0.
BSECC(6)=0.
BSECP(1)=0.
BSECP(2)=0.
BSECP(3)=0.
BSECP(4)=0.
BSECP(5)=0.
BSECP(6)=0.
BOLC(1)=0.
BOLC(2)=0.
BOLC(3)=0.
BOLC(4)=0.
BOLC(5)=0.
BOLC(6)=0.
BAFC(1)=0.
BAFC(2)=0.
BAFC(3)=0.
BAFC(4)=0.
BAFC(5)=0.
BAFC(6)=0.
BISC(1)=0.
BISC(2)=0.
BISC(3)=0.
BISC(4)=0.
BISC(5)=0.
BISC(6)=0.
BRSC(1)=0.
BRSC(2)=0.
BRSC(3)=0.
BRSC(4)=0.

BRSC(5)=0.
BRSC(6)=0.
BONMC(1)=0.
BONMC(2)=0.
BONMC(3)=0.
BONMC(4)=0.
BONMC(5)=0.
BONMC(6)=0.
BOFMC(1)=0.
BOFMC(2)=0.
BOFMC(3)=0.
BOFMC(4)=0.
BOFMC(5)=0.
BOFMC(6)=0.
BIIMC(1)=0.
BIIMC(2)=0.
BIIMC(3)=0.
BIIMC(4)=0.
BIIMC(5)=0.
BIIMC(6)=0.
AFMC=0.
BMTRC=0.
BTCDI=0.
BXTRC=0.
DMTRC=0.
IMTRC=0.
MTRC=0.
SECBC=0.
SECBP=0.
SECC=0.
SECDC=0.
SECDP=0.
SECIC=0.
SECIP=0.
SECP=0.
SECRC=0.
SECRP=0.
STDC=0.
STDCI=0.
STDCR=0.

C

RETURN
END

SUBROUTINE ERRCK1

800827 110716780

C
C*****
C* PERFORM ERROR CHECKING ON INPUT DATA FILES *
C*****
C

COMMON /PRNTXX/ PRNTXX
INTEGER PRNTXX
COMMON /COND/ COND(999)
COMMON /I/ I
COMMON /INO/ INO(999)
COMMON /LE/ LE(10)
COMMON /LRU/ LRU(999)
COMMON /MTBMI/ MTBMI(999,4)
REAL MTBMI
COMMON /MXI/ MXI
COMMON /MXNP/ MXNP
COMMON /NERRYY/ NERRYY
COMMON /NHI/ NHI(999)
COMMON /NITEM/ NITEM(999,10)
REAL NITEM
COMMON /NP/ NP
COMMON /NRTS/ NRTS(999)
REAL NRTS
COMMON /RTS/ RTS(999)

- 1 FORMAT(/5X,49HINPUT ERROR: RTS + NRTS + COND /= 1 FOR ITEM TYPE,I5
+)
- 2 FORMAT(/5X,70HINPUT ERROR: INDENTURE LEVEL INDICATORS (LRU & NHI)
+NOT CONSISTENT FOR/7X,9HITEM TYPE,I5)
- 3 FORMAT(/5X,70HINPUT ERROR: INDENTURE LEVEL INDICATORS (LRU & NHI)
+NOT CONSISTENT FOR/7X,9HITEM TYPE,I5)
- 4 FORMAT(/5X,47HINPUT ERROR: MTBMI CANNOT BE ZERO FOR ITEM TYPE,I5/7
+X,25H IN OPERATING ENVIRONMENT,I5)

C
C
C

DO 290 IXXX1=1,MXI
I=INO(IXXX1)
CK1=RTS(I)+NRTS(I)+COND(I)
IF(.NOT.(CK1.GT.1.000001.OR.CK1.LT..999999)) GO TO 210
NERRYY=NERRYY+1
IF(PRNTXX.NE.0) WRITE(7, 1) I
IF(PRNTXX.NE.1) WRITE(06, 1) I
210 CONTINUE
IF(.NOT.(LRU(I).EQ.1)) GO TO 230
IF(.NOT.(NHI(I).NE.0)) GO TO 220
NERRYY=NERRYY+1

```

        IF(PRNTXX.NE.0) WRITE( 7, 2) I
        IF(PRNTXX.NE.1) WRITE(06, 2) I
220     CONTINUE
230     CONTINUE
        IF(.NOT.(LRU(I).NE.1)) GO TO 250
        IF(.NOT.(NHI(I).EQ.0)) GO TO 240
        NERRY=NERRY+1
        IF(PRNTXX.NE.0) WRITE( 7, 3) I
        IF(PRNTXX.NE.1) WRITE(06, 3) I
240     CONTINUE
250     CONTINUE
        DO 280 NP=1,MXNP
        IF(.NOT.(NITEM(I,NP).GT.0)) GO TO 270
        IF(.NOT.(MTBMI(I,LE(NP)).EQ.0)) GO TO 260
        NERRY=NERRY+1
        IF(PRNTXX.NE.0) WRITE( 7, 4) I,LE(NP)
        IF(PRNTXX.NE.1) WRITE(06, 4) I,LE(NP)
260     CONTINUE
270     CONTINUE
280     CONTINUE
290 CONTINUE
C
    RETURN
    END

```

SUBROUTINE RLCOMP

800827 110737131

```
C
C*****
C* BASELINE CHANGE *
C* SSS MOD FOR FLOAT *
C* SETS VALUES FOR RTS, NRTS, COND *
C* DEPENDING ON THE VALUE OF RL(I) *
C*****
```

```
C
COMMON /PRNTXX/ PRNTXX
INTEGER PRNTXX
COMMON /BIRD/ BIRD
COMMON /COND/ COND(999)
COMMON /I/ I
COMMON /INO/ INO(999)
COMMON /LRU/ LRU(999)
COMMON /MXI/ MXI
COMMON /NRTS/ NRTS(999)
REAL NRTS
COMMON /R/ R
INTEGER R
COMMON /RL/ RL(999)
INTEGER RL
COMMON /RTS/ RTS(999)
COMMON /WEAR/ WEAR(999)
```

C
C
C

```
DO 360 IXXX1=1,MXI
  I=INO(IXXX1)
  IF(.NOT.(R.EQ.1)) GO TO 210
  RL(I)=1
210 CONTINUE
  IF(.NOT.(R.EQ.2)) GO TO 240
  IF(.NOT.(LRU(I).EQ.1)) GO TO 220
  RL(I)=1
220 CONTINUE
  IF(.NOT.(LRU(I).NE.1)) GO TO 230
  RL(I)=2
230 CONTINUE
240 CONTINUE
  IF(.NOT.(R.EQ.3)) GO TO 270
  IF(.NOT.(LRU(I).EQ.1)) GO TO 250
  RL(I)=1
250 CONTINUE
  IF(.NOT.(LRU(I).NE.1)) GO TO 260
  RL(I)=3
```

```

260     CONTINUE
270     CONTINUE
      IF(.NOT.(R.EQ.4)) GO TO 280
      RL(I)=2
280     CONTINUE
      IF(.NOT.(R.EQ.5)) GO TO 310
      IF(.NOT.(LRU(I).EQ.1)) GO TO 290
      RL(I)=2
290     CONTINUE
      IF(.NOT.(LRU(I).NE.1)) GO TO 300
      RL(I)=3
300     CONTINUE
310     CONTINUE
      IF(.NOT.(R.EQ.6)) GO TO 320
      RL(I)=3
320     CONTINUE
      IF(.NOT.(RL(I).EQ.1)) GO TO 330
      COND(I)=WEAR(I)
      RTS(I)=(1.-COND(I))/(1.+BIRD)
      NRTS(I)=RTS(I)*BIRD
330     CONTINUE
      IF(.NOT.(RL(I).EQ.2)) GO TO 340
      COND(I)=WEAR(I)
      RTS(I)=0.
      NRTS(I)=1-COND(I)
340     CONTINUE
      IF(.NOT.(RL(I).EQ.3)) GO TO 350
      COND(I)=1.
      RTS(I)=0.
      NRTS(I)=0.
350     CONTINUE
360 CONTINUE
C
      RETURN
      END

```

SUBROUTINE OTABST

```

C
C*****800827 110807712
C* PRINTS SUMMARY TITLE PAGE*
C* SSS MOD LCR - 13 AUG 80*
C*****
C

```

```

COMMON /PRNTXX/ PRNTXX
INTEGER PRNTXX
COMMON /FULLXX/ FULLXX
INTEGER FULLXX
COMMON /FINC/ FINC
COMMON /PIUP/ PIUP
COMMON /R/ R
INTEGER R
COMMON /XFPR/ XFPR
COMMON /XFR/ XFR
COMMON /XMIL/ XMIL
COMMON /XUC/ XUC
1 FORMAT(1H1//////////44X,44(1H*)/44X,1H*,42X,1H*)
2 FORMAT(44X,1H*,5X,21HSYSTEM OPERATING LIFE,1X,F5.2,1X,5HYEARS,4X,1
+H*/44X,1H*,42X,1H*/44X,1H*,42X,1H*)
3 FORMAT(44X,1H*,5X,21HREPAIR LEVEL CASE RUN,2X,I3,11X,1H*/44X,1H*,4
+2X,1H*/44X,1H*,42X,1H*)
4 FORMAT(44X,1H*,5X,30HSENSITIVITY MULTIPLIER FACTORS,7X,1H*/44X,1H*
+,42X,1H*)
5 FORMAT(44X,1H*,9X,3HXUC,1X,F14.3,15X,1H*)
6 FORMAT(44X,1H*,9X,3HXFR,1X,F14.3,15X,1H*)
7 FORMAT(44X,1H*,9X,4HXFPR,F14.3,15X,1H*)
8 FORMAT(44X,1H*,9X,4HXMIL,F14.3,15X,1H*)
9 FORMAT(44X,1H*,9X,4HFINC,F14.3,15X,1H*/44X,1H*,42X,1H*/44X,44(1H*)
+)

```

```

C
C
C
C.....ONLY PRINT THIS TABLE IF FULL OFF-LINE OUTPUT WAS REQUESTED
IF(PRNTXX.EQ.0.OR.FULLXX.EQ.0) RETURN
C

```

```

WRITE( 7, 1)
WRITE( 7, 2) PIUP
WRITE( 7, 3) R
WRITE( 7, 4)
WRITE( 7, 5) XUC
WRITE( 7, 6) XFR
WRITE( 7, 7) XFPR
WRITE( 7, 8) XMIL
WRITE( 7, 9) FINC

```

C

RETURN
END

SUBROUTINE ITAB1A

800827 110809686

```
C
C*****
C* PRINTS THE SYSTEM-WIDE SCALAR PARAMETERS *
C* READ IN FROM THE MISC. DATA FILE :PART 1 *
C*****
C
```

```
COMMON /PRNTXX/ PRNTXX
INTEGER PRNTXX
COMMON /FULLXX/ FULLXX
INTEGER FULLXX
COMMON /BAA/ BAA
COMMON /BLR/ BLR
COMMON /BMF/ BMF
COMMON /DAA/ DAA
COMMON /DLR/ DLR
COMMON /DMF/ DMF
COMMON /HPD2/ HPD2
INTEGER HPD2
COMMON /MILR/ MILR(3)
REAL MILR
COMMON /MRF/ MRF
REAL MRF
COMMON /MRO/ MRO
REAL MRO
COMMON /PAL1/ PAL1
COMMON /PAL2B/ PAL2B
COMMON /PAL2D/ PAL2D
COMMON /PMLR/ PMLR
COMMON /SR/ SR
COMMON /TNLR/ TNLR
COMMON /TORB/ TORB
COMMON /TORD/ TORD
COMMON /TR/ TR
COMMON /TRAVB/ TRAVB
COMMON /TRAV1D/ TRAV1D
1 FORMAT(1H1/44X,44HINPUT TABLE 1: SYSTEM-WIDE SCALAR PARAMETERS///)
2 FORMAT(50X,30HGOVERNMENT-PROVIDED PARAMETERS/)
3 FORMAT(26X,13HLABOR FACTORS/)
4 FORMAT(29X,59H BAA - MONTHLY AVAILABLE WORKING HOURS PER MAINTEN
+ANCE /38X,50HMAN AT BASE LEVEL
+F14.2 )
5 FORMAT(29X,59H BMF - BASE MAINTENANCE FACTOR
+ ,F14.2 )
6 FORMAT(29X,59H DAA - MONTHLY AVAILABLE WORKING HOURS PER MAINTEN
+ANCE /38X,50HMAN AT DEPOT LEVEL
+F14.2 )
```

7 FORMAT(29X,59H DMF - DEPOT MAINTENANCE FACTOR
 + ,F14.2)
 8 FORMAT(29X,59H HPD2 - NUMBER OF HOURS SPENT BY A TYPE 2
 + /38X,50HTRAINEE IN CLASS PER DAY
 +I14)
 9 FORMAT(29X,59H MRO - AVERAGE MANHOURS PER FAILURE TO COMPLETE
 + /38X,50HON-EQUIPMENT MAINTENANCE RECORDS
 +F14.2)
 10 FORMAT(29X,59H MRF - AVERAGE MANHOURS PER FAILURE TO COMPLETE
 + /38X,50HOFF-EQUIPMENT MAINTENANCE RECORDS
 +F14.2)
 11 FORMAT(29X,59H SR - AVERAGE MANHOURS PER FAILURE TO COMPLETE
 + /38X,50HSUPPLY TRANSACTION RECORDS
 +F14.2)
 12 FORMAT(29X,59H TORB - TURNOVER RATE FOR BASE MAINT. PERSONNEL
 + ,F14.2)
 13 FORMAT(29X,59H TORD - TURNOVER RATE FOR DEPOT MAINT. PERSONNEL
 + ,F14.2)
 14 FORMAT(29X,59H TR - AVERAGE MANHOURS PER FAILURE TO COMPLETE
 + /38X,50HTRANSPORTATION TRANSACTION FORMS
 +F14.2)
 15 FORMAT(1X/26X,11HLABOR RATES/)
 16 FORMAT(29X,59H BLR - BASE MAINTENANCE LABOR RATE IN \$ PER HOUR
 + ,F14.2)
 17 FORMAT(29X,59H DLR - DEPOT MAINTENANCE LABOR RATE IN \$ PER HOUR
 + ,F14.2)
 18 FORMAT(28X,55HMILR(1) - MOD/INSTALLATION LABOR RATE DURING PRODUCT
 +ION/38X,13HIN \$ PER HOUR,37X,F14.2/28X,52HMILR(2) - MOD/INSTALLATI
 +ON LABOR RATE FOR FIELD MODS/38X,30HUSING DEPOT TEAM IN \$ PER HOUR
 +,20X,F14.2/28X,56HMILR(3) - MOD/INSTALLATION LABOR RATE FOR MODS P
 +ERFORMED/38X,26HAT THE DEPOT IN \$ PER HOUR,24X,F14.2)
 19 FORMAT(29X,59H PAL1 - AVERAGE DAILY PAY & ALLOWANCE FOR A
 + /38X,50HTYPE 1 TRAINEE
 +F14.2)
 20 FORMAT(29X,59H PAL2B - AVERAGE DAILY PAY & ALLOWANCE FOR A
 + /38X,50HTYPE 2 TRAINEE
 +F14.2)
 21 FORMAT(29X,59H PAL2D - AVERAGE DAILY PAY & ALLOWANCE FOR A
 + /38X,50HTYPE 2 DEPOT TRAINEE
 +F14.2)
 22 FORMAT(29X,59H PMLR - PRIME MISSION EQUIP OPER LABOR RATE IN \$ PE
 +R HOUR ,F14.2)
 23 FORMAT(29X,59HTRAV1D - AVERAGE TRAVEL EXPENSE FOR TYPE 1 AND TYPE
 +2 /38X,50HDEPOT TRAINEES
 +F14.2)
 24 FORMAT(29X,59H TRAVB - AVERAGE TRAVEL EXPENSE FOR TYPE 2 BASE TRAI
 +NEES ,F14.2)

25 FORMAT(29X,59H TNLR - TIMING NET OPERATOR LABOR RATE IN \$ PER HOU
+R ,F14.2)

C
C
C

C.....ONLY PRINT THIS TABLE IF FULL OFF-LINE OUTPUT WAS REQUESTED
IF(PRNTXX.EQ.0.OR.FULLXX.EQ.0) RETURN

C

WRITE(7, 1)
WRITE(7, 2)
WRITE(7, 3)
WRITE(7, 4) BAA
WRITE(7, 5) BMF
WRITE(7, 6) DAA
WRITE(7, 7) DMF
WRITE(7, 8) HPD2
WRITE(7, 9) MRO
WRITE(7,10) MRF
WRITE(7,11) SR
WRITE(7,12) TORB
WRITE(7,13) TORD
WRITE(7,14) TK
WRITE(7,15)
WRITE(7,16) BLR
WRITE(7,17) DLR
WRITE(7,18) (MILR(K1),K1=1,3)
WRITE(7,19) PAL1
WRITE(7,20) PAL2B
WRITE(7,21) PAL2D
WRITE(7,22) PMLR
WRITE(7,23) TRAV1D
WRITE(7,24) TRAVB
WRITE(7,25) TNLR

C

RETURN
END

SUBROUTINE ITAB1B

800827 110819685

```

C
C*****
C* PRINTS THE SYSTEM-WIDE SCALAR PARAMETERS *
C* READ IN FROM THE MISC. DATA FILE : PART 2 *
C*****
C
COMMON /PRNTXX/ PRNTXX
INTEGER PRNTXX
COMMON /FULLXX/ FULLXX
INTEGER FULLXX
COMMON /ACPP/ ACPP
COMMON /BRCT/ BRCT
COMMON /CFG/ CFG(3)
COMMON /CPD2/ CPD2
COMMON /CPPC/ CPPC
COMMON /CPPD/ CPPD(3)
COMMON /CRCT/ CRCT
COMMON /DAD/ DAD
COMMON /DRCT/ DRCT(3)
COMMON /IMC/ IMC
REAL IMC
COMMON /OST/ OST(3)
COMMON /OSTC/ OSTC
COMMON /RCPP/ RCPP
COMMON /RMC/ RMC
COMMON /SA/ SA
COMMON /UCPP/ UCPP
1 FORMAT(1H1/44X,44HINPUT TABLE 1: SYSTEM-WIDE SCALAR PARAMETERS/61X
+,11H(CONTINUED)///)
2 FORMAT(1X/26X,14HPIPELINE TIMES/)
3 FORMAT(29X,59H BRCT - BASE REPAIR CYCLE TIME IN MONTHS
+,F15.3 )
4 FORMAT(29X,59H CRCT - TIME FOR FAILURE AT SATELLITE BASE UNTIL RE
+PAIR /38X,50HAT CIMF BASE IN MONTHS ,
+F15.3 )
5 FORMAT(29X,59H DAD - TIME FROM FAILURE REMOVAL AT DEPOT UNTIL RE
+PAIR /38X,50HAT DEPOT IN MONTHS ,
+F15.3 )
6 FORMAT(28X,54HDRCT(1) - TIME FROM FAILURE AT CONUS BASE UNTIL REPA
+IR/38X,18HAT DEPOT IN MONTHS,32X,F15.3/28X,56HDRCT(2) - TIME FROM
+FAILURE AT PACIFIC BASE UNTIL REPAIR/38X,18HAT DEPOT IN MONTHS,32X
+,F15.3/28X,55HDRCT(3) - TIME FROM FAILURE AT EUROPE BASE UNTIL REP
+AIR/38X,18HAT DEPOT IN MONTHS,32X,F15.3/28X,52H OST(1) - ORDER AND
+ SHIPPING TIME FROM CONUS BASE TO/38X,15HDEPOT IN MONTHS,35X,F15.3
+/28X,54H OST(2) - ORDER AND SHIPPING TIME FROM PACIFIC BASE TO/38X
+,15HDEPOT IN MONTHS,35X,F15.3/28X,53H OST(3) - ORDER AND SHIPPING

```

```

+TIME FROM EUROPE BASE TO/38X,15HDEPOT IN MONTHS,35X,F15.3/28X,55H
+ OSTC - ORDER AND SHIPPING TIME FROM A SATELLITE BASE/38X,26HTO I
+TS CIMF BASE IN MONTHS,24X,F15.3/)
7 FORMAT(26X,17HUNIT COST FACTORS/)
8 FORMAT(29X,59H ACPP - ACQUISITION COST PER PAGE FOR ORIGINAL
+ /38X,50HNEGATIVES OF TECH. DATA
+F15.3 )
9 FORMAT(28X,53H CFG(1) - COST OF FUEL IN $ PER GALLON AT CONUS BASE
+S,7X,F15.3/28X,55H CFG(2) - COST OF FUEL IN $ PER GALLON AT PACIFI
+C BASES,5X,F15.3/28X,54H CFG(3) - COST OF FUEL IN $ PER GALLON AT
+EUROPE BASES,6X,F15.3/28X,52H CPD2 - COST PER CLASS PER DAY FOR
+TYPE 2 TRAINING,8X,F15.3/28X,55H CPPC - COST OF PACKING AND SHIP
+PING FROM A SATELLITE/38X,47HBASE TO ITS CIMF BASE IN $ PER NET WE
+IGHT POUND,3X,F15.3/28X,54HCPPD(1) - COST OF PACKING AND SHIPPING
+FROM CONUS BASE/38X,34HTO DEPOT IN $ PER NET WEIGHT POUND,16X,F15.
+3/28X,56HCPPD(2) - COST OF PACKING AND SHIPPING FROM PACIFIC BASE/
+38X,34HTO DEPOT IN $ PER NET WEIGHT POUND,16X,F15.3/28X,55HCPPD(3)
+ - COST OF PACKING AND SHIPPING FROM EUROPE BASE/38X,34HTO DEPOT I
+N $ PER NET WEIGHT POUND,16X,F15.3)
10 FORMAT(29X,59H RCPP - REPRODUCTION COST PER COPY PER PAGE OF
+ /38X,50HTECH. DATA
+F15.3 )
11 FORMAT(29X,59H IMC - INITIAL DEPOT INVENTORY MANAGEMENT COST PER
+ NEW /38X,50HPART IN $
+F15.3 )
12 FORMAT(29X,59H RMC - RECURRING DEPOT INVENTORY MANAGEMENT COST P
+ER NEW /38X,50HPART IN $ PER YEAR
+F15.3 )
13 FORMAT(29X,59H SA - BASE-LEVEL INVENTORY MANAGEMENT COST PER NE
+W PART /38X,50HIN $ PER YEAR
+F15.3 )
14 FORMAT(29X,59H UCPP - UPKEEP COST PER YEAR PER DISTINCT PAGE OF
+ /38X,50HTECH. DATA
+F15.3 )

```

C

C

C

C.....ONLY PRINT THIS TABLE IF FULL OFF-LINE OUTPUT WAS REQUESTED
IF(PRNTXX.EQ.0.OR.FULLXX.EQ.0) RETURN

C

```

WRITE( 7, 1)
WRITE( 7, 2)
WRITE( 7, 3) BRCT
WRITE( 7, 4) CRCT
WRITE( 7, 5) DAD
WRITE( 7, 6) (DRCT(K2),K2=1,3),(OST(K3),K3=1,3),OSTC
WRITE( 7, 7)

```

```
WRITE( 7, 8) ACPP  
WRITE( 7, 9) (CFG(K4),K4=1,3),CPD2,CPPC,(CPPD(K5),K5=1,3)  
WRITE( 7,10) RCPP  
WRITE( 7,11) IMC  
WRITE( 7,12) RMC  
WRITE( 7,13) SA  
WRITE( 7,14) UCPP
```

C

```
RETURN  
END
```

SUBROUTINE ITAB1C

800827 110825502

C

C*****

C* SSS MOD LCR *

C* PRINTS THE SYSTEM-WIDE SCALAR PARAMETERS *

C* READ IN FROM THE MISC. DATA FILE : PART 3 *

C*****

C

COMMON /PRNTXX/ PRNTXX

INTEGER PRNTXX

COMMON /FULLXX/ FULLXX

INTEGER FULLXX

COMMON /BDATA/ BDATA

INTEGER BDATA

COMMON /BF/ BF

COMMON /BIRD/ BIRD

COMMON /CPD1/ CPD1

COMMON /DDATA/ DDATA

INTEGER DDATA

COMMON /FSEDC/ FSEDC

COMMON /HPD1/ HPD1

INTEGER HPD1

COMMON /KFAC/ KFAC(4)

REAL KFAC

COMMON /MUSE/ MUSE

REAL MUSE

COMMON /NRUC/ NRUC

REAL NRUC

COMMON /PIUP/ PIUP

COMMON /QTYP1/ QTYP1

INTEGER QTYP1

COMMON /QTYP2B/ QTYP2B

INTEGER QTYP2B

COMMON /QTYP2D/ QTYP2D

INTEGER QTYP2D

COMMON /R/ R

INTEGER R

COMMON /SPC1/ SPC1

INTEGER SPC1

COMMON /SPC2/ SPC2

INTEGER SPC2

COMMON /TEFM/ TEFM

COMMON /TYP2TF/ TYP2TF

COMMON /XFPR/ XFPR

COMMON /XFR/ XFR

COMMON /XMIL/ XMIL

COMMON /XUC/ XUC

1 FORMAT(1H1/44X,44HINPUT TABLE 1: SYSTEM-WIDE SCALAR PARAMETERS/61X
+ ,11H(CONTINUED)///)
2 FORMAT(1X/26X,21HMISCELLANEOUS FACTORS/)
3 FORMAT(28X,41H BF - COEFFICIENT IN SPARING FUNCTION,19X,F14.2)
4 FORMAT(29X,59H BIRD - FRACTION OF BASE-REPAIR-INTENDED FAILURES
+ /38X,50HREQUIRING DEPOT REPAIR ,
+F14.2)
5 FORMAT(28X,54HKFAC(1) - FAILURE RATE EXPERIENCE FACTOR FOR AIRBORN
+E-/38X,19HFIGHTER ENVIRONMENT,31X,F14.2/28X,54HKFAC(2) - FAILURE R
+ATE EXPERIENCE FACTOR FOR AIRBORNE-/38X,17HCARGO ENVIRONMENT,33X,F
+14.2/28X,52HKFAC(3) - FAILURE RATE EXPERIENCE FACTOR FOR GROUND-/3
+8X,31HFIXED/TRANSPORTABLE ENVIRONMENT,19X,F14.2/28X,52HKFAC(4) - F
+AILURE RATE EXPERIENCE FACTOR FOR GROUND-/38X,18HMOBILE ENVIRONMEN
+T,32X,F14.2)
6 FORMAT(29X,59H MUSE - MINIMUM FRACTIONAL UTILIZATION FOR SENSITIV
+ITY /38X,50HCALCULATIONS ON SUPPORT EQUIPMENT COSTS ,
+F14.2)
7 FORMAT(29X,59H NRUC - NUMBER OF YEARS OF REPLACEMENT SPARES TO BE
+ /38X,50HPROVIDED UNDER THE SSS PRODUCTION CONTRACT(S) ,
+F14.2)
8 FORMAT(29X,59H PIUP - NUMBER OF SYSTEM OPERATING YEARS
+ ,F14.2)
9 FORMAT(29X,59H QTYP1 - NUMBER OF TYPE 1 TRAINEES
+ ,I14)
10 FORMAT(29X,59HQTYP2B - NUMBER OF TYPE 2 BASE TRAINEES
+ ,I14)
11 FORMAT(29X,59HQTYP2D - NUMBER OF TYPE 2 DEPOT TRAINEES
+ ,I14)
12 FORMAT(29X,59H R - REPAIR LEVEL CASE RUN NUMBER
+ ,I14)
13 FORMAT(29X,59H SPC2 - MAXIMUM NUMBER OF TYPE 2 TRAINEES PER CLASS
+ ,I14)
14 FORMAT(29X,59HTYP2TF - RATIO OF TYPE 2 TRAINING TIME TO TYPE 1
+ /38X,50HTRAINING TIME ,
+F14.2)
15 FORMAT(29X,59H XFPR - FALSE PULL RATE SENSITIVITY MUTIPLIER FACTO
+R ,F14.2)
16 FORMAT(29X,59H XFR - FAILURE RATE SENSITIVITY MULTIPLIER FACTOR
+ ,F14.2)
17 FORMAT(29X,59H XMIL - MOD/I LABOR HOURS SENSITIVITY MULTIPLIER FA
+CTOR ,F14.2)
18 FORMAT(29X,59H XUC - UNIT COST SENSITIVITY MULTIPLIER FACTOR
+ ,F14.2)
19 FORMAT(1H1/44X,44HINPUT TABLE 1: SYSTEM-WIDE SCALAR PARAMETERS/61X
+ ,11H(CONTINUED)///)
20 FORMAT(1X,50X,34HCONTRACTOR - DETERMINED PARAMETERS//)
21 FORMAT(29X,59H BDATA - NUMBER OF TECH. DATA PAGES FOR BASE MAINT.

```

+AND /38X,50HNOT ITEM OR SE SPECIFIC
+I14 )
22 FORMAT(29X,59H CPD1 - COST PER CLASS PER DAY FOR TYPE 1 TRAINING
+ ,F14.2 )
23 FORMAT(29X,59H DDATA - NUMBER OF TECH. DATA PAGES FOR DEPOT MAINT.
+ /38X,50HAND NOT ITEM OR SE SPECIFIC
+I14 )
24 FORMAT(29X,59H FSEDC - TOTAL COST OF FULL SCALE ENGINEERING
+ /38X,50HDEVELOPMENT PROGRAM
+F14.2 )
25 FORMAT(29X,59H HPD1 - NUMBER OF CLASS HOURS PER DAY FOR A TYPE 1
+ /38X,50HTRAINING CLASS
+I14 )
26 FORMAT(29X,59H SPC1 - MAXIMUM NUMBER OF TYPE 1 TRAINEES PER CLASS
+ ,I14 )
27 FORMAT(29X,59H TEFM - COST OF TRAINING EQUIPMENT, FACILITIES AND
+ /38X,50HMANUALS
+F14.2 )

```

C
C
C

C.....ONLY PRINT THIS TABLE IF FULL OFF-LINE OUTPUT WAS REQUESTED
IF(PRNTXX.EQ.0.OR.FULLXX.EQ.0) RETURN

C

```

WRITE( 7, 1)
WRITE( 7, 2)
WRITE( 7, 3) BF
WRITE( 7, 4) BIRD
WRITE( 7, 5) (KFAC(K2),K2=1,4)
WRITE( 7, 6) MUSE
WRITE( 7, 7) NRUC
WRITE( 7, 8) PIUF
WRITE( 7, 9) QTYP1
WRITE( 7,10) QTYP2B
WRITE( 7,11) QTYP2D
WRITE( 7,12) R
WRITE( 7,13) SPC2
WRITE( 7,14) TYP2TF
WRITE( 7,15) XFPR
WRITE( 7,16) XFR
WRITE( 7,17) XMIL
WRITE( 7,18) XUC
WRITE( 7,19)
WRITE( 7,20)
WRITE( 7,21) BDATA
WRITE( 7,22) CPD1
WRITE( 7,23) DDATA

```

WRITE(7,24) FSED
WRITE(7,25) HPD1
WRITE(7,26) SPC1
WRITE(7,27) TFM

C

RETURN
END

SUBROUTINE ITAB2

```

C
C***** 800827 110836539 *****
C* SSS MOD LCR *
C* PRINTS BASE CONFIGURATION DATA *
C*****
C
COMMON /PRNTXX/ PRNTXX
INTEGER PRNTXX
COMMON /FULLXX/ FULLXX
INTEGER FULLXX
COMMON /BNOUN/ BNOUN(16,16)
COMMON /BPLAT/ BPLAT(16)
INTEGER BPLAT
COMMON /BSP/ BSP(16)
INTEGER BSP
COMMON /BTYPE/ BTYPE(16)
INTEGER BTYPE
COMMON /LO/ LO(16)
COMMON /MXNS/ MXNS
COMMON /NBC/ NBC(16)
REAL NBC
COMMON /NHB/ NHB(16)
COMMON /NS/ NS
COMMON /TNB/ TNB(16)
1 FORMAT(1H1/28X,39HINPUT TABLE 2: BASE CONFIGURATION DATA/// 58X,4
+HNEXT,6X,3HNO.,5X,4HBASE,6X,4HBASE/2X,4HBASE,9X,4HBASE,10X,6HNO. 0
+F,3X,7HLOC. OF,4X,4HBASE,4X,6HHIGHER,4X,5HUNDER,4X,5HPLAT-,3X,7HSU
+PPORT/2X,5HINDEX,8X,4HNAME,10X,5HBASES,5X,4HBASE,6X,4HTYPE,5X,4HBA
+SE,6X,4HCIMF,4X,5HFORMS,2X,10HPHILOSOPHY/3X,4H(NS),7X,7H(BNOUN),8X
+,5H(TNB),5X,4H(LO),5X,7H(BTYPE),3X,5H(NHB),4X,5H(NBC),3X,7H(BPLAT)
+,3X,5H(BSP)/)
2 FORMAT(3X,I2,4X,16A1,4X,F4.0,7X,I2,8X,I2,7X,I2,6X,F4.1,6X,I2,8X,I2
+)
C
C
C
C.....ONLY PRINT THIS TABLE IF FULL OFF-LINE OUTPUT WAS REQUESTED
IF(PRNTXX.EQ.0.OR.FULLXX.EQ.0) RETURN
C
WRITE( 7, 1)
DO 210 NS=1,MXNS
WRITE( 7, 2) NS,(BNOUN(NS,I1),I1=1,16),TNB(NS),LO(NS),BTYPE(NS),
+ NHB(NS),NBC(NS),BPLAT(NS),BSP(NS)
210 CONTINUE
C
RETURN
END

```

SUBROUTINE ITAB3

```

C
C***** 800827 110846772
C* SSS MOD LCR *
C* PRINTS PLATFORM OPERATION DATA FILE *
C*****
C
COMMON /PRNTXX/ PRNTXX
INTEGER PRNTXX
COMMON /FULLXX/ FULLXX
INTEGER FULLXX
COMMON /AMPM/ AMPM(10,3)
COMMON /APFH/ APFH(10,3)
COMMON /FGH/ FGH(10)
COMMON /LE/ LE(10)
COMMON /M/ M
COMMON /MMPD/ MMPD(10,3)
REAL MMPD
COMMON /MMPM/ MMPM(10)
REAL MMPM
COMMON /MXNP/ MXNP
COMMON /NP/ NP
COMMON /PNOUN/ PNOUN(10,12)
COMMON /TFAC/ TFAC(10)
COMMON /THRS/ THRS(10)
1 FORMAT(1H1/44X,39HINPUT TABLE 3: PLATFORM OPERATION DATA///2X,5HP
+LAT-,16X,3HEN-,2X,20H*OPERATING HOURS IN*,6HUTILI-,1X,5HACTI-,1X,2
+OH*MISSIONS PER MONTH*,4X,11HOTHER HOURS,4X,1H*,6HTHRUST,2X,7HGALL
+ONS/2X,4HFORM,3X,8HPLATFORM,6X,6HVIRO*,18(1H-),7H*ZATION,1X,7HVAT
+ION*,18(1H-),1H*,19(1H-),1H*,1X,2HIN,6X,3HPER/2X,5HINDEX,2X,12HNOM
+ENCLATURE,2X,4HMENT,1X,20H*CONUS PACIF EUROPE*,6HFACTOR,1X,4HTIME,
+2X,20H*CONUS PACIF EUROPE*,20H CONUS PACIF EUROPE*,6HPOUNDS,2X,7HO
+PER HR/2X,4H(NP),4X,7H(PNOUN),6X,4H(LE),9X,6H(APFH),6X,6H(TFAC),1X
+,6H(MMPM),8X,6H(AMPM),13X,6H(MMPD),8X,6H(THRS),3X,5H(FGH)/)
2 FORMAT(2X,I3,3X,I2A1,3X,I2,3X,F5.0,1X,F5.0,1X,F5.0,4X,F4.2,2X,F4.1
+,3X,F5.1,1X,F5.1,1X,F5.1,1X,F6.1,1X,F6.1,1X,F6.1,1X,F8.0,1X,F6.0)
C
C
C
C.....ONLY PRINT THIS TABLE IF FULL OFF-LINE OUTPUT WAS REQUESTED
IF(PRNTXX.EQ.0.OR.FULLXX.EQ.0) RETURN
C
WRITE( 7, 1)
DO 210 NP=1,MXNP
WRITE( 7, 2) NP, (PNOUN(NP,K1),K1=1,12),LE(NP), (APFH(NP,M),M=1,
+ 3),TFAC(NP),MMPM(NP), (AMPM(NP,M),M=1,3), (MMPD(NP,M),M=1,3),
+ THRS(NP),FGH(NP)

```

210 CONTINUE

C

RETURN
END

SUBROUTINE ITAB4

C 800827 110850998
 C*****
 C* SSS MOD LCR *
 C* PRINTS PLATFORM TERMINAL COST AND INITIAL *
 C* MOD/INSTALLATION DATA FILE *
 C*****

C

COMMON /PRNTXX/ PRNTXX
 INTEGER PRNTXX
 COMMON /FULLXX/ FULLXX
 INTEGER FULLXX
 COMMON /DRAG/ DRAG(10)
 COMMON /FR/ FR(3,10)
 COMMON /INTNR/ INTNR(10)
 REAL INTNR
 COMMON /INTR/ INTR(10)
 REAL INTR
 COMMON /K/ K(10)
 REAL K
 COMMON /M/ M
 COMMON /MXNP/ MXNP
 COMMON /NAE/ NAE(10)
 REAL NAE
 COMMON /NP/ NP
 COMMON /NRMI/ NRMI(10)
 REAL NRMI
 COMMON /NTRMP/ NTRMP(10)
 REAL NTRMP
 COMMON /PDIV/ PDIV(10)

1 FORMAT(1H1/27X,76HINPUT TABLE 4: PLATFORM TERMINAL DATA & NON-REC
 +URRING MOD/INSTALLATION DATA//43X,23HIN THOUSANDS OF DOLLARS///10X
 +,3HNO.,43X,5HPLAT-,21X,3HNO.,5X,4HLBS.,4X,6HTHRUST/2X,5HPLAT-,3X,3
 +HPME,7X,4HNON-,32X,4HFORM,3X,18H*FRACTION MODS IN*,1X,5HADDED,3X,4
 +HDRAG,4X,5H-FUEL/2X,4HFORM,4X,4HTER-,6X,6HRECUR.,5X,6HRECUR.,5X,10
 +HNON-RECUR.,4X,6HDIVER-,1X,1H*,16(1H-),1H*,1X,5HANTEN,3X,3HPER,5X,
 +6HCNSMPT/2X,5HINDEX,3X,6HMINALS,4X,5HINTEG,6X,5HINTEG,6X,10HMOD/I
 +COST,4X,4HSITY,3X,18H*PROD FIELD DEPOT*,1X,3HNAS,5X,5HANTEN,3X,6HF
 +ACTOR/2X,4H(NP),4X,7H(NTRMP),3X,7H(INTNR),4X,6H(INTR),7X,6H(NRMI),
 +6X,6H(PDIV),8X,4H(FR),8X,5H(NAE),3X,6H(DRAG),3X,3H(K)/)
 2 FORMAT(2X,I3,5X,F5.2,3X,F10.3,1X,F9.3,2X,F10.3,5X,F5.2,3X,2(F4.2,2
 +X),F4.2,3X,F5.2,3X,F5.1,4X,F5.2)

C
 C
 C

C.....ONLY PRINT THIS TABLE IF FULL OFF-LINE OUTPUT WAS REQUESTED
 IF(PRNTXX.EQ.0.OR.FULLXX.EQ.0) RETURN

```

C
  WRITE( 7, 1)
  DO 210 NP=1,MXNP
    T1=INTNR(NP)/1000
    T2=INTR(NP)/1000
    T3=NRMI(NP)/1000
    WRITE( 7, 2) NP,NTRMP(NP),T1,T2,T3,PDIV(NP),(FR(M,NP),M=1,3),
+      NAE(NP),DRAG(NP),K(NP)
  210 CONTINUE
C
  RETURN
  END

```

SUBROUTINE ITAB5

800827 110902735

C

C*****

C* PRINTS PLATFORM RECURRING MOD/INSTALLATION LABOR HOURS BY MODE AND *

C* DATA FILE *

C*****

C

COMMON /PRNTXX/ PRNTXX

INTEGER PRNTXX

COMMON /FULLXX/ FULLXX

INTEGER FULLXX

COMMON /AKIT/ AKIT(4,10)

COMMON /IA/ IA

COMMON /M/ M

COMMON /MIFIX/ MIFIX(3,10)

REAL MIFIX

COMMON /MIMH/ MIMH(4,3,10)

REAL MIMH

COMMON /MXNP/ MXNP

COMMON /NP/ NP

1 FORMAT(1H1,35X,56HINPUT TABLE 5: PLATFORM RECURRING MOD/INSTALLAT
+ION DATA///57X,1H*,13X,45HMOD/INSTALLATION LABOR HOURS BY MODE AND
+ AREA,12X,1H*/2X,5HPLAT-,2X,19H*FIXED MOD/I COST *,7X,15HAKIT EQUI
+P COST,7X,1H*,70(1H-),1H*/2X,4HFORM,3X,1H*,17(1H-),1H*,2X,25(1H-),
+2X,1H*,5X,12H**PROD MOD**,5X,1H*,5X,13H**FIELD MOD**,5X,1H*,6X,13H
+**DEPOT MOD**,4X,1H*/2X,5HINDEX,2X,5H*PROD,1X,5HFIELD,1X,5HDEPOT,1
+X,1H*,2X,5HANTNA,1X,6HELBOX,1X,6HCNTLHD,1X,5HCABLE,2X,5H*ANTN,2X,
+4HELBX,2X,4HCTLH,2X,10HCBL * ANTN,2X,4HELBX,2X,4HCTLH,2X,10HCBL *
+ANTN,2X,4HELBX,2X,4HCTLH,2X,3HCBL,1X,1H*/2X,4H(NP),7X,8H(MIFIX)*,1
+9X,6H(AKIT),4X,6H(MIMH)/)
2 FORMAT(3X,13,2X,7(F6.0,1X),12F6.0)
3 FORMAT(//2X,25H* IN THOUSANDS OF DOLLARS)

C

C

C

C.....ONLY PRINT THIS TABLE IF FULL OFF-LINE OUTPUT WAS REQUESTED
IF(PRNTXX.EQ.0.OR.FULLXX.EQ.0) RETURN

C

WRITE(7, 1)

DO 210 NP=1,MXNP

WRITE(7, 2) NP,(MIFIX(M,NP),M=1,3),(AKIT(IA,NP),IA=1,4),

+ ((MIMH(IA,M,NP),IA=1,4),M=1,3)

210 CONTINUE

WRITE(7, 3)

C

RETURN

END

SUBROUTINE ITAB6

800827 110916757

```

C
C*****
C* PRINTS PLATFORM DEPLOYMENT AT BASES *
C* DATA FILE *
C*****
C
COMMON /PRNTXX/ PRNTXX
INTEGER PRNTXX
COMMON /FULLXX/ FULLXX
INTEGER FULLXX
COMMON /MXNP/ MXNP
COMMON /NP/ NP
COMMON /NPLT/ NPLT(10,16)
REAL NPLT
COMMON /NS/ NS
1 FORMAT(1H1,8X,58HINPUT TABLE 6: PLATFORM DEPLOYMENT AT BASES - NP
+LT(NP,NS)//1X,5HPLAT-/1X,4HFORM,3X,74HAVERAGE NUMBER OF PLATFORMS
+OF GIVEN TYPE AT EACH BASE WITHIN GROUPS BELOW/1X,5HINDEX,2X,122(1
+H-)/1X,4H(NP),16X,1H1,5X,1H2,5X,1H3,5X,1H4,5X,1H5,5X,1H6,5X,1H7,5X
+,1H8,5X,1H9,4X,2H10,4X,2H11,4X,2H12,4X,2H13,4X,2H14,4X,2H15,4X,2H1
+6/)
2 FORMAT(1X,I3,14X,16(F5.2,1X))
C
C
C
C....ONLY PRINT THIS TABLE IF FULL OFF-LINE OUTPUT WAS REQUESTED
IF(PRNTXX.EQ.0.OR.FULLXX.EQ.0) RETURN
C
WRITE( 7, 1)
DO 210 NP=1,MXNP
WRITE( 7, 2) NP,(NPLT(NP,NS),NS=1,16)
210 CONTINUE
C
RETURN
END

```

SUBROUTINE ITAB7

```

C
C***** 800827 110922178
C* SSS MOD LCR *
C* PRINTS SUPPORT EQUIPMENT DATA *
C*****
C
COMMON /PRNTXX/ PRNTXX
INTEGER PRNTXX
COMMON /FULLXX/ FULLXX
INTEGER FULLXX
COMMON /CSE/ CSE(250)
COMMON /DATAS/ DATAS(250)
INTEGER DATAS
COMMON /L/ L
COMMON /MSE/ MSE(250)
REAL MSE
COMMON /MXL/ MXL
COMMON /SEDEV/ SEDEV(250)
COMMON /SEINO/ SEINO(250)
INTEGER SEINO
COMMON /SENOUN/ SENOUN(250,20)
COMMON /SENUM/ SENUM(250,12)
COMMON /SETYPE/ SETYPE(250)
INTEGER SETYPE
1 FORMAT(1H1,44X,38HINPUT TABLE 7: SUPPORT EQUIPMENT DATA///58X,8HF
+RACTION,5X,14HCOM.ON-SITE(1),4X,9HNUMBER OF,5X,2HSE/3X,2HSE,42X,7H
+SE UNIT,4X,9HUNIT COST,4X,14HCOM.PROCUR.(2),4X,10HTECH ORDER,2X,6H
+DEVMNT/2X,5HINDEX,4X,15HSE NOMENCLATURE,7X,11HSE PART NO.,5X,4HCOS
+T,4X,11HTO MAINTAIN,3X,11HPECULIAR(3),8X,5HPAGES,8X,4HCOST/3X,3H(L
+),9X,8H(SENOUN),11X,7H(SENUM),7X,5H(CSE),7X,5H(MSE),7X,8H(SETYPE),
+9X,7H(DATAS),5X,7H(SEDEV)/)
2 FORMAT(3X,I3,3X,20A1,1X,12A1,4X,F7.0,7X,F5.3,10X,I2,14X,I3,4X,F8.0
+)
C
C
C
C.....ONLY PRINT THIS TABLE IF FULL OFF-LINE OUTPUT WAS REQUESTED
IF(PRNTXX.EQ.0.OR.FULLXX.EQ.0) RETURN
C
WRITE( 7, 1)
DO 210 IXXX1=1,MXL
L=SEINO(IXXX1)
WRITE( 7, 2) L,(SENOUN(L,I1),I1=1,20),(SENUM(L,I2),I2=1,12),
+ CSE(L),MSE(L),SETYPE(L),DATAS(L),SEDEV(L)
210 CONTINUE
C

```

RETURN
END

SUBROUTINE ITAB8

800827 110938835

C

C*****

C* SSS MOD LCR *

C* PRINTS ITEM EQUIPMENT DATA *

C*****

C

COMMON /PRNTXX/ PRNTXX

INTEGER PRNTXX

COMMON /FULLXX/ FULLXX

INTEGER FULLXX

COMMON /GFE/ GFE(999)

INTEGER GFE

COMMON /I/ I

COMMON /INO/ INO(999)

COMMON /INOUN/ INOUN(999,24)

REAL INOUN

COMMON /INTEG/ INTEG(999)

REAL INTEG

COMMON /LFAC/ LFAC(999)

REAL LFAC

COMMON /LRU/ LRU(999)

COMMON /MXI/ MXI

COMMON /NHI/ NHI(999)

COMMON /PA/ PA(999)

COMMON /PTNUM/ PTNUM(999,12)

COMMON /RM/ RM(999)

COMMON /UP/ UP(999)

COMMON /WT/ WT(999)

1 FORMAT(1H1/48X,35HINPUT TABLE 8: ITEM EQUIPMENT DATA)

2 FORMAT(61X,11H(CONTINUED))

3 FORMAT(/62X,6HLRU(1),3X,4HNEXT,5X,3HGFE,5X,5HINTE-,5X,4HITEM,5X,6
+HREPAIR,14X,5HPIECE/2X,4HITEM,47X,5HLEARN,6X,2HOR,4X,6HHIGHER,4X,5
+HINDI-,3X,7HGRATION,3X,4HUNIT,4X,9HMATERIALS,4X,4HITEM,4X,4HPART/2
+X,5HINDEX,9X,12HNOMENCLATURE,9X,11HPART NUMBER,6X,4HRATE,4X,6HSRU(
+0),3X,4HITEM,5X,5HCATOR,5X,5HITEMS,4X,4HCOST,5X,6HFACTOR,5X,6HWEIG
+HT,2X,5HCOUNT /3X,3H(I),12X,7H(INOUN),14X,7H(PTNUM),7X,6H(LFAC),3X
+,5H(LRU),4X,5H(NHI),4X,5H(GFE),3X,7H(INTEG),3X,4H(UP),6X,4H(RM),7X
+,4H(WT),4X,4H(PA)/)

4 FORMAT(4X,I3,3X,24A1,1X,12A1,6X,F4.2,6X,I2,6X,I3,7X,I2,7X,I2,4X,F7
+.0,6X,F4.3,5X,F6.2,3X,F5.2)

C

C

C

C.....ONLY PRINT THIS TABLE IF FULL OFF-LINE OUTPUT WAS REQUESTED
IF(PRNTXX.EQ.0.OR.FULLXX.EQ.0) RETURN

C

```

IPAGE=40
IFLAG=1
DO 230 IXXX1=1,MXI
  I=INO(IXXX1)
  IF(.NOT.(IPAGE.EQ.40)) GO TO 220
  WRITE( 7, 1)
  IPAGE=1
  IF(.NOT.(IFLAG.NE.1)) GO TO 210
  WRITE( 7, 2)
210  CONTINUE
  WRITE( 7, 3)
220  CONTINUE
  WRITE( 7, 4) I,(INOUN(I,K1),K1=1,24),(PTNUM(I,K2),K2=1,12),
+    LFAC(I),LRU(I),NHI(I),GFE(I),INTEG(I),UP(I),RM(I),WT(I),PA(I)
  IFLAG=0
  IPAGE=IPAGE+1
230 CONTINUE
C
  RETURN
END

```

SUBROUTINE ITAB9A

C 800827 111003895
 C*****
 C* PRINTS ITEM MAINTENANCE DATA *
 C*****
 C

```

COMMON /PRNTXX/ PRNTXX
INTEGER PRNTXX
COMMON /FULLXX/ FULLXX
INTEGER FULLXX
COMMON /BCMH/ BCMH(999)
COMMON /BMH/ BMH(999)
COMMON /COND/ COND(999)
COMMON /DMH/ DMH(999)
COMMON /FPR/ FPR(999)
COMMON /I/ I
COMMON /INO/ INO(999)
COMMON /IPCF/ IPCF(999)
REAL IPCF
COMMON /MTBMI/ MTBMI(999,4)
REAL MTBMI
COMMON /MXI/ MXI
COMMON /NRTS/ NRTS(999)
REAL NRTS
COMMON /RIP/ RIP(999)
COMMON /RL/ RL(999)
INTEGER RL
COMMON /RMH/ RMH(999)
COMMON /RTS/ RTS(999)
1 FORMAT(1H1/47X,39H INPUT TABLE 9A: ITEM MAINTENANCE DATA)
2 FORMAT(61X,11H(CONTINUED))
3 FORMAT(/58X,8HFRACTION,10X,19H,1X,7HFAILURE,1X
+,4HBASE,1X,2HLV/6X,1H*,5X,34HMEAN TIME BETWEEN MAINT. INCIDENTS,5X
+,6H*FALSE,1X,8HFAILURES,1X,8HCOST PER,2X,19HFRACTION FAILURES,6H
+REMOVE,2X,5HBENCH,3X,7HBASE LV,1X,5HDEPOT/1X,6HITEM *,1X,43(1H-),5
+H*PULL,2X,8HREPAIRED,1X,8HIN PLACE,2X,18H REPAIRED AT,1X,7HR
+EPLACE,1X,5HCHECK,3X,6HREPAIR,2X,6HREPAIR,1X,4HR.L./1X,5HINDEX,13H
+*AIR-FIGHTER*,10HAIR-CARGO*,11HGRND-FIXED*,12HGRND-MOBILE*,4HRATE,
+2X,8HIN PLACE,2X,6HREPAIR,2X,20H BASE DEPOT COND,7HMAN HRS,1X,
+7HMAN HRS,1X,7HMAN HRS,1X,6HMAN HR,1X,4HCODE/2X,3H(I),3X,8H(MTBMI1
+),3X,8H(MTBMI2),3X,8H(MTBMI3),3X,8H(MTBMI4),3X,5H(FPR),2X,5H(RIP),
+4X,6H(IPCF),2X,20H(RTS) (NRTS) (COND),2X,5H(RMH).2X,6H(BCMH),3X,5
+H(BMH),3X,5H(DMH),1X,4H(RL)/)
4 FORMAT(1X,I3,3X,F9.0,3(2X,F9.0),3X,F4.3,4X,F4.2,3X,F6.2,3X,F5.3,2X
+,F5.3,2X,F5.3,2X,3(F5.2,3X),F5.2,1X,I4)

```

C
 C

```

C
C.....ONLY PRINT THIS TABLE IF FULL OFF-LINE OUTPUT WAS REQUESTED
      IF(PRNTXX.EQ.0.OR.FULLXX.EQ.0) RETURN
C
      IPAGE=40
      IFLAG=1
      DO 230 IXXX1=1,MXI
        I=INO(IXXX1)
        IF(.NOT.(IPAGE.EQ.40)) GO TO 220
        WRITE( 7, 1)
        IPAGE=1
        IF(.NOT.(IFLAG.NE.1)) GO TO 210
        WRITE( 7, 2)
210      CONTINUE
        WRITE( 7, 3)
220      CONTINUE
        WRITE( 7, 4) I,(MTBMI(I,K1),K1=1,4),FPR(I),RIP(I),IPCF(I),
      +   RTS(I),NRTS(I),COND(I),RMH(I),BCMh(I),BMH(I),DMH(I),RL(I)
        IFLAG=0
        IPAGE=IPAGE+1
230 CONTINUE
C
      RETURN
      END

```

SUBROUTINE ITAB9B

800827 111005143

C
C*****
C* SSS MOD LCR *
C* PRINTS ITEM REPAIR HOURS,TECH ORDER AND TR. DATA *
C*****
C

COMMON /PRNTXX/ PRNTXX
INTEGER PRNTXX
COMMON /FULLXX/ FULLXX
INTEGER FULLXX
COMMON /DATAB/ DATAB(999)
INTEGER DATAB
COMMON /DATAD/ DATAD(999)
INTEGER DATAD
COMMON /I/ I
COMMON /INO/ INO(999)
COMMON /MXI/ MXI
COMMON /TIME1/ TIME1(999)
INTEGER TIME1
COMMON /UCTDEV/ UCTDEV(999)

1 FORMAT(1H1/5X,43HINPUT TABLE 9B: TECHNICAL ORDERS, TRAINING/22X,1
+9HAND UCT DEVELOPMENT)
2 FORMAT(28X,11H(CONTINUED))
3 FORMAT(/15X,32HNUMBER OF TECH. NO. OF HOURS,5X,3HUCT/5X,4HITE
+M,6X,30HDATA PAGES FOR FOR TYPE 1,6X,6HDEVMNT/5X,5HINDEX,5X,5
+HDEPOT,5X,4HBASE,6X,8HTRAINING,9X,4HCOST/6X,3H(I),5X,16H(DATAD) (
+DATAB),5X,7H(TIME1),8X,8H(UCTDEV)/)
4 FORMAT(6X,I3,7X,I3,6X,I3,9X,I3,9X,F7.0)

C
C
C

C.....ONLY PRINT THIS TABLE IF FULL OFF-LINE OUTPUT WAS REQUESTED
IF(PRNTXX.EQ.0.OR.FULLXX.EQ.0) RETURN

C

IPAGE=40
IFLAG=1
DO 230 IXXX1=1,MXI
I=INO(IXXX1)
IF(.NOT.(IPAGE.EQ.40)) GO TO 220
WRITE(7, 1)
IPAGE=1
IF(.NOT.(IFLAG.NE.1)) GO TO 210
WRITE(7, 2)
210 CONTINUE
WRITE(7, 3)
220 CONTINUE

```
        WRITE( 7, 4) I,DATAD(I),DATAB(I),TIME1(I),UCTDEV(I)
        IFLAG=0
        IPAGE=IPAGE+1
230    CONTINUE
C
        RETURN
        END
```

SUBROUTINE ITB10A

800827 111010553

C
C*****
C* SSS MOD LCR *
C* PRINTS ITEM/SUPPORT EQUIPMENT CROSS-REFERENCE *
C*****
C

COMMON /PRNTXX/ PRNTXX
INTEGER PRNTXX
COMMON /FULLXX/ FULLXX
INTEGER FULLXX
COMMON /A/ A(999,4,30)
INTEGER A
COMMON /I/ I
COMMON /INO/ INO(999)
COMMON /IRMIN/ IRMIN(999,4)
COMMON /IRMT/ IRMT
COMMON /MXI/ MXI
COMMON /MXIRMT/ MXIRMT
COMMON /NJA/ NJA(999,4)
COMMON /QSA/ QSA(999,4,30)
1 FORMAT(1H1/23X,66HINPUT TABLE 10A: ITEM/SE CROSS REFERENCE DATA -
+DEPOT LEVEL REPAIR)
2 FORMAT(41X,11H(CONTINUED))
3 FORMAT(/7X,5HNUMB./7X,5HOF SE,2X,2HSE,17(4X,2HSE)/1X,4HITEM,2X,4H
+TYPE, 9(1X,5HINDEX,1X,5HQUAN-)/1X,5HINDEX,1X,5HREQRD,2X,3HNO.,2X,4
+HTITY,8(3X,3HNO.,2X,4HTITY)/2X,3H(I),2X,5H(NJA), 9(2X,3H(A),2X,5H(
+QSA))/
4 FORMAT(2X,I3,3X,I3,28(9(2X,I4,2X,F4.0)/))

C
C
C

C.....ONLY PRINT THIS TABLE IF FULL OFF-LINE OUTPUT WAS REQUESTED
IF(PRNTXX.EQ.0.OR.FULLXX.EQ.0) RETURN

C

IFLAG=1
IPAGE=40
DO 240 IXXX1=1,MXI
I=INO(IXXX1)
IF(.NOT.(IPAGE.EQ.40)) GO TO 220
WRITE(7, 1)
IPAGE=1
IF(.NOT.(IFLAG.NE.1)) GO TO 210
WRITE(7, 2)
210 CONTINUE
WRITE(7, 3)
IPAGE=1

```

220  CONTINUE
      DO 230 IRMT=1,MXIRMT
        IF(.NOT.(IRMIN(I,IRMT).EQ.1)) GO TO 230
        N=NJA(I,IRMT)
        WRITE( 7, 4) I,NJA(I,IRMT),(A(I,IRMT,K2),QSA(I,IRMT,K2),K2=1,
+      N)
230  CONTINUE
      IFLAG=0
      IPAGE=IPAGE+1
240  CONTINUE
C
      RETURN
      END

```

SUBROUTINE ITB10B

```

C
C
C* SSS MOD LCR
C* PRINTS ITEM/SUPPORT EQUIPMENT CROSS-REFERENCE
C*****
C
COMMON /PRNTXX/ PRNTXX
INTEGER PRNTXX
COMMON /FULLXX/ FULLXX
INTEGER FULLXX
COMMON /A/ A(999,4,30)
INTEGER A
COMMON /I/ I
COMMON /INO/ INO(999)
COMMON /IRMIN/ IRMIN(999,4)
COMMON /IRMT/ IRMT
COMMON /MXI/ MXI
COMMON /MXIRMT/ MXIRMT
COMMON /NJA/ NJA(999,4)
COMMON /QSA/ QSA(999,4,30)
1 FORMAT(1H1/23X,68HINPUT TABLE 10B: ITEM/SE CROSS REFERENCE DATA -
+BASE LEVEL, PSE ONLY)
2 FORMAT(41X,11H(CONTINUED))
3 FORMAT(/7X,5HNUMB./7X,5HOF SE,2X,2HSE,17(4X,2HSE)/1X,4HITEM,2X,4H
+TYPE, 9(1X,5HINDEX,1X,5HQUAN-)/1X,5HINDEX,1X,5HREQRD,2X,3HNO.,2X,4
+HTITY,8(3X,3HNO.,2X,4HTITY)/2X,3H(I),2X,5H(NJA), 9(2X,3H(A),2X,5H(
+QSA))/)
4 FORMAT(2X,I3,3X,I3,28(9(2X,I4,2X,F4.0)/))
C
C
C
C.....ONLY PRINT THIS TABLE IF FULL OFF-LINE OUTPUT WAS REQUESTED
IF(PRNTXX.EQ.0.OR.FULLXX.EQ.0) RETURN
C
IFLAG=1
IPAGE=40
DO 240 IXXX1=1,MXI
I=INO(IXXX1)
IF(.NOT.(IPAGE.EQ.40)) GO TO 220
WRITE( 7, 1)
IPAGE=1
IF(.NOT.(IFLAG.NE.1)) GO TO 210
WRITE( 7, 2)
210 CONTINUE
WRITE( 7, 3)
IPAGE=1

```

```

220  CONTINUE
      DO 230 IRMT=1,MXIRMT
        IF(.NOT.(IRMIN(I,IRMT).EQ.2)) GO TO 230
        N=NJA(I,IRMT)
        WRITE( 7, 4) I,NJA(I,IRMT),(A(I,IRMT,K2),QSA(I,IRMT,K2),K2=1,
+          N)
230  CONTINUE
      IFLAG=0
      IPAGE=IPAGE+1
240  CONTINUE
C
      RETURN
      END

```

SUBROUTINE ITB10C

800827 111024229

```
C
C*****
C* SSS MOD LCR *
C* PRINTS ITEM/SUPPORT EQUIPMENT CROSS-REFERENCE *
C*****
C
```

```
COMMON /PRNTXX/ PRNTXX
INTEGER PRNTXX
COMMON /FULLXX/ FULLXX
INTEGER FULLXX
COMMON /A/ A(999,4,30)
INTEGER A
COMMON /I/ I
COMMON /INO/ INO(999)
COMMON /IRMIN/ IRMIN(999,4)
COMMON /IRMT/ IRMT
COMMON /MXI/ MXI
COMMON /MXIRMT/ MXIRMT
COMMON /NJA/ NJA(999,4)
COMMON /QSA/ QSA(999,4,30)
1 FORMAT(1H1/23X,63HINPUT TABLE 10C: ITEM/SE CROSS REFERENCE DATA -
+BASE LEVEL, MBS)
2 FORMAT(41X,11H(CONTINUED))
3 FORMAT(/7X,5HNUMB./7X,5HOF SE,2X,2HSE,17(4X,2HSE)/1X,4HITEM,2X,4H
+TYPE, 9(1X,5HINDEX,1X,5HQUAN-)/1X,5HINDEX,1X,5HREQRD,2X,3HNO.,2X,4
+HTITY,8(3X,3HNO.,2X,4HTITY)/2X,3H(I),2X,5H(NJA), 9(2X,3H(A),2X,5H(
+QSA)))
4 FORMAT(2X,I3,3X,I3,28(9(2X,I4,2X,F4.0)/))
```

```
C
C
C
C.....ONLY PRINT THIS TABLE IF FULL OFF-LINE OUTPUT WAS REQUESTED
IF(PRNTXX.EQ.0.OR.FULLXX.EQ.0) RETURN
```

```
C
IFLAG=1
IPAGE=40
DO 240 IXXX1=1,MXI
I=INO(IXXX1)
IF(.NOT.(IPAGE.EQ.40)) GO TO 220
WRITE( 7, 1)
IPAGE=1
IF(.NOT.(IFLAG.NE.1)) GO TO 210
WRITE( 7, 2)
210 CONTINUE
WRITE( 7, 3)
IPAGE=1
```

```

220  CONTINUE
      DO 230 IRMT=1,MXIRMT
        IF(.NOT.(IRMIN(I,IRMT).EQ.3)) GO TO 230
        N=NJA(I,IRMT)
        WRITE( 7, 4) I,NJA(I,IRMT),(A(I,IRMT,K2),QSA(I,IRMT,K2),K2=1,
+          N)
230  CONTINUE
      IFLAG=0
      IPAGE=IPAGE+1
240  CONTINUE
C
      RETURN
      END

```

SUBROUTINE ITB10D

800827 111030349

C
 C*****
 C* SSS MOD ICR *
 C* PRINTS ITEM/SUPPORT EQUIPMENT CROSS-REFERENCE *
 C*****
 C

COMMON /PRNTXX/ PRNTXX
 INTEGER PRNTXX
 COMMON /FULLXX/ FULLXX
 INTEGER FULLXX
 COMMON /A/ A(999,4,30)
 INTEGER A
 COMMON /I/ I
 COMMON /INO/ INO(999)
 COMMON /IRMIN/ IRMIN(999,4)
 COMMON /IRMT/ IRMT
 COMMON /MXI/ MXI
 COMMON /MXIRMT/ MXIRMT
 COMMON /NJA/ NJA(999,4)
 COMMON /QSA/ QSA(999,4,30)

1 FORMAT(1H1/23X,63HINPUT TABLE 10D: ITEM/SE CROSS REFERENCE DATA -
 +BASE LEVEL, UCT)
 2 FORMAT(41X,11H(CONTINUED))
 3 FORMAT(/7X,5HNUMB./7X,5HOF SE,2X,2HSE,17(4X,2HSE)/1X,4HITEM,2X,4H
 +TYPE, 9(1X,5HINDEX,1X,5HQUAN-)/1X,5HINDEX,1X,5HREQRD,2X,3HNO.,2X,4
 +HTITY,8(3X,3HNO.,2X,4HTITY)/2X,3H(I),2X,5H(NJA), 9(2X,3H(A),2X,5H(
 +QSA))/
 4 FORMAT(2X,I3,3X,I3,28(9(2X,I4,2X,F4.0)/))

C
 C
 C
 C.....ONLY PRINT THIS TABLE IF FULL OFF-LINE OUTPUT WAS REQUESTED
 IF(PRNTXX.EQ.0.OR.FULLXX.EQ.0) RETURN
 C

IFLAG=1
 IPAGE=40
 DO 240 IXX1=1,MXI
 I=INO(IXX1)
 IF(.NOT.(IPAGE.EQ.40)) GO TO 220
 WRITE(7, 1)
 IPAGE=1
 IF(.NOT.(IFLAG.NE.1)) GO TO 210
 WRITE(7, 2)
 210 CONTINUE
 WRITE(7, 3)
 IPAGE=1

```

220  CONTINUE
      DO 230 IRMT=1,MXIRMT
        IF(.NOT.(IRMIN(I,IRMT).EQ.4)) GO TO 230
        N=NJA(I,IRMT)
        WRITE( 7, 4) I,NJA(I,IRMT),(A(I,IRMT,K2),QSA(I,IRMT,K2),K2=1,
+          N\
230  CONTINUE
        IFLAG=C
        IPAGE=IPAGE+1
240  CONTINUE
C
      RETURN
      END

```

SUBROUTINE ITAB11

800827 111046239

C
C*****
C* PRINTS ITEM CONFIGURATIONS ON PLATFORMS DATA FILE *
C*****
C

COMMON /PRNTXX/ PRNTXX
INTEGER PRNTXX
COMMON /FULLXX/ FULLXX
INTEGER FULLXX
COMMON /I/ I
COMMON /INO/ INO(999)
COMMON /MXI/ MXI
COMMON /NITEM/ NITEM(999,10)
REAL NITEM
COMMON /NP/ NP

1 FORMAT(1H1/4X,64HINPUT TABLE 11: ITEM CONFIGURATIONS ON PLATFORMS
+ - NITEM(I,NP))
2 FORMAT(29X,11H(CONTINUED))
3 FORMAT(/2X,4HNITEM,15X,47HAVE. NUMBER OF ITEMS INSTALLED ON PLATFO
+RM TYPE/2X,5HINDEX,3X,69(1H-)/3X,3H(I), 12X,1H1,5X,1H2,5X,1H3,5X,1
+H4,5X,1H5,5X,1H6,5X,1H7,5X,1H8,5X,1H9,5X,2H10/)
4 FORMAT(3X,I3,9X,10(F5.2,1X))

C
C
C
C.....ONLY PRINT THIS TABLE IF FULL OFF-LINE OUTPUT WAS REQUESTED
IF(PRNTXX.EQ.0.OR.FULLXX.EQ.0) RETURN
C

IFLAG=1
IPAGE=40
DO 230 IXXX1=1,MXI
I=INO(IXXX1)
IF(.NOT.(IPAGE.EQ.40)) GO TO 220
WRITE(7, 1)
IPAGE=1
IF(.NOT.(IFLAG.NE.1)) GO TO 210
WRITE(7, 2)
210 CONTINUE
WRITE(7, 3)
220 CONTINUE
WRITE(7, 4) I,(NITEM(I,NP),NP=1,10)
IFLAG=0
IPAGE=IPAGE+1
230 CONTINUE

C
RETURN
END

SUBROUTINE ZFAIL

800827 111105266

C*****
 C* COMPUTES AUXILIARY VARIABLE FAIL(I,NS) *
 C*****

C
 COMMON /APFH/ APFH(10,3)
 COMMON /FAIL/ FAIL(999,16)
 COMMON /I/ I
 COMMON /INO/ INO(999)
 COMMON /KFAC/ KFAC(4)
 REAL KFAC
 COMMON /LE/ LE(10)
 COMMON /LO/ LO(16)
 COMMON /MTBMI/ MTBMI(999,4)
 REAL MTBMI
 COMMON /MXI/ MXI
 COMMON /MXNP/ MXNP
 COMMON /MXNS/ MXNS
 COMMON /NITEM/ NITEM(999,10)
 REAL NITEM
 COMMON /NP/ NP
 COMMON /NPLT/ NPLT(10,16)
 REAL NPLT
 COMMON /NS/ NS
 COMMON /RIP/ RIP(999)
 COMMON /TFAC/ TFAC(10)
 COMMON /XFR/ XFR

C
 C
 DO 230 NS=1,MXNS
 DO 220 IXXX2=1,MXI
 I=INO(IXXX2)
 DO 210 NP=1,MXNP
 IF(.NOT.(NITEM(I,NP).GT..000001)) GO TO 210
 FL=NITEM(I,NP)*(1.-RIP(I))*NPLT(NP,NS)*APFH(NP,LO(NS))*
 + TFAC(NP)*KFAC(LE(NP))*XFR/MTBMI(I,LE(NP))
 FAIL(I,NS)=FAIL(I,NS)+FL
 210 CONTINUE
 220 CONTINUE
 230 CONTINUE

C
 RETURN
 END

SUBROUTINE ZNFB

C

800827 111120316

C*****

C* COMPUTES PIPELINE SPARES NFB(I,NS) AT BASE NS *

C* AND NFD(I) AT THE DEPOT *

C*****

C

COMMON /B/ B
 INTEGER B
 COMMON /BRCT/ BRCT
 COMMON /BTYPE/ BTYPE(16)
 INTEGER BTYPE
 COMMON /COND/ COND(999)
 COMMON /CRCT/ CRCT
 COMMON /DAD/ DAD
 COMMON /DRCT/ DRCT(3)
 COMMON /FAIL/ FAIL(999,16)
 COMMON /FPR/ FPR(999)
 COMMON /I/ I
 COMMON /INO/ INO(999)
 COMMON /LO/ LO(16)
 COMMON /LRU/ LRU(999)
 COMMON /MXI/ MXI
 COMMON /MXNS/ MXNS
 COMMON /NBC/ NBC(16)
 REAL NBC
 COMMON /NFB/ NFB(999,16)
 REAL NFB
 COMMON /NFD/ NFD(999)
 REAL NFD
 COMMON /NHB/ NHB(16)
 COMMON /NHI/ NHI(999)
 COMMON /NRTS/ NRTS(999)
 REAL NRTS
 COMMON /NS/ NS
 COMMON /OST/ OST(3)
 COMMON /OSTC/ OSTC
 COMMON /RTS/ RTS(999)
 COMMON /TNB/ TNB(16)
 COMMON /XFPR/ XFPR
 REAL NHNRT
 REAL NHRT

C

C

DO 270 IXXX1=1,MXI
 I=INO(IXXX1)
 XF=XFPR*FPR(I)

```

      NHRT=0.
      NHNRT=0.
      IF(.NOT.(NHI(I).NE.0)) GO TO 210
      NHRT=RTS(NHI(I))
      NHNRT=NRTS(NHI(I))
210   CONTINUE
      DO 260 NS=1,MXNS
      SFL=0.
      IF(.NOT.(BTYPE(NS).EQ.3)) GO TO 220
C
C.....COMPUTE SATELLITE BASE FAILURES
      NFB(I,NS)=FAIL(I,NS)*FLOAT(LRU(I))*(1.+XF)*OSTC
220   CONTINUE
      IF(.NOT.(BTYPE(NS).EQ.2)) GO TO 240
C
C.....COMPUTE BASE FAILURES FROM SATELLITES
      TEM01=0.
      DO 230 B=1,MXNS
      IF(.NOT.(NHB(B).EQ.NS)) GO TO 230
      TEM01=TEM01+FAIL(I,B)*NBC(B)
230   CONTINUE
      SFL=TEM01
      SFL=SFL*(FLOAT(LRU(I))+NHRT)*((RTS(I)+XF)*CRCT+(NRTS(I)+
+      COND(I))*(OST(LO(NS))+U(XF)*CRCT))
240   CONTINUE
      IF(.NOT.(BTYPE(NS).LT.3)) GO TO 250
      NFB(I,NS)=FAIL(I,NS)*(FLOAT(LRU(I))+NHRT)*((RTS(I)+XF)*BRCT+
+      (NRTS(I)+COND(I))*OST(LO(NS)))+SFL
250   CONTINUE
      NFD(I)=NFD(I)+FAIL(I,NS)*TNB(NS)*((FLOAT(LRU(I))+NHRT)*NRTS(I)
+      *DRCT(LO(NS))+NHNRT*(1.-COND(I))*DAD)
260   CONTINUE
270   CONTINUE
C
      RETURN
      END

```

SUBROUTINE ZERHB

800827 111133346

C
 C*****
 C* COMPUTES ITEM INTERMEDIATE REPAIR HOURS PER MONTH, *
 C* ERHBI(I,NS) AT BASE NS AND ERHD(I) AT THE DEPOT. *
 C*****
 C

COMMON /B/ B
 INTEGER B
 COMMON /BCMh/ BCMH(999)
 COMMON /BMF/ BMF
 COMMON /BMH/ BMH(999)
 COMMON /BTYPE/ BTYPE(16)
 INTEGER BTYPE
 COMMON /COND/ COND(999)
 COMMON /DMF/ DMF
 COMMON /DMH/ DMH(999)
 COMMON /EBCBI/ EBCBI(999,16)
 COMMON /ERHBI/ ERHBI(999,16)
 COMMON /ERHD/ ERHD(999)
 COMMON /ERTBI/ ERTBI(999,16)
 COMMON /FAIL/ FAIL(999,16)
 COMMON /FPR/ FPR(999)
 COMMON /I/ I
 COMMON /INO/ INO(999)
 COMMON /LRU/ LRU(999)
 COMMON /MXI/ MXI
 COMMON /MXNS/ MXNS
 COMMON /NBC/ NBC(16)
 REAL NBC
 COMMON /NHB/ NHB(16)
 COMMON /NHI/ NHI(999)
 COMMON /NRTS/ NRTS(999)
 REAL NRTS
 COMMON /NS/ NS
 COMMON /RTS/ RTS(999)
 COMMON /TNB/ TNB(16)
 COMMON /XFPR/ XFPR
 REAL NHNRT
 REAL NHRT

C
 C

DO 260 IXXX1=1,MXI
 I=INO(IXXX1)
 NHRT=0.
 NHNRT=0.
 IF(.NOT.(NHI(I).NE.0)) GO TO 210

```

      NHRT=RTS(NHI(I))
      NHRNT=NRTS(NHI(I))
210  CONTINUE
      XF=XFPR*FPR(I)
      DO 250 NS=1,MXNS
        SFL1=0.
        SFL2=0.
        ERHBI(I,NS)=0.
        EBCBI(I,NS)=0.
        ERTBI(I,NS)=0.
        IF(.NOT.(BTYP(N).EQ.2)) GO TO 230
C
C.....INCLUDE REPAIRS FROM SATELLITE BASES
      TEM01=0.
      DO 220 B=1,MXNS
        IF(.NOT.(NHB(B).EQ.NS)) GO TO 220
        TEM01=TEM01+FAIL(I,B)*NBC(B)
220  CONTINUE
      SFL1=TEM01
      SFL2=SFL1*(FLOAT(LRU(I))+NHRT)*U(1.-COND(I))*RTS(I)*BMH(I)*
+      BMF
      SFL1=SFL1*(FLOAT(LRU(I))+NHRT)*((RTS(I)+NRTS(I)+XF+U(XF))*
+      COND(I))*BCM(H(I))*BMF
230  CONTINUE
      IF(.NOT.(BTYP(N).LT.3)) GO TO 240
      EBCBI(I,NS)=FAIL(I,NS)*(FLOAT(LRU(I))+NHRT)*((1.+XF)*BCM(H(I))
+      )*BMF+SFL1
      ERTBI(I,NS)=FAIL(I,NS)*(FLOAT(LRU(I))+NHRT)*U(1.-COND(I))*
+      KTS(I)*BMH(I)*BMF+SFL2
      ERHBI(I,NS)=EBCBI(I,NS)+ERTBI(I,NS)
240  CONTINUE
      ERHD(I)=ERHD(I)+FAIL(I,NS)*TNB(NS)*((FLOAT(LRU(I))+NHRT)*
+      NRTS(I)+NHRNT*(1.-COND(I)))*DMH(I)*DMF
250  CONTINUE
260  CONTINUE
C
      RETURN
      END

```

SUBROUTINE ZERHSE

C 800827 111214832

C*****

C* SSS MOD SLR - 21 MAY 80 *

C* CALCULATES EXPECTED MANHOURS PER MONTH *

C* THAT SUPPORT EQUIP. TYPE L IS UTILIZED, *

C* ERHAB(L,NS) AT BASE NS AND ERHAD(L) AT THE DEPOT. *

C*****

C

COMMON /A/ A(999,4,30)

INTEGER A

COMMON /BSP/ BSP(16)

INTEGER BSP

COMMON /EBCBI/ EBCBI(999,16)

COMMON /ERHAB/ ERHAB(250,16)

COMMON /ERHAD/ ERHAD(250)

COMMON /ERHD/ ERHD(999)

COMMON /ERTBI/ ERTBI(999,16)

COMMON /I/ I

COMMON /INO/ INO(999)

COMMON /IRMIN/ IRMIN(999,4)

COMMON /IRMT/ IRMT

COMMON /L/ L

COMMON /LT/ LT

COMMON /MXI/ MXI

COMMON /MXIRMT/ MXIRMT

COMMON /MXLT/ MXLT

COMMON /MXNS/ MXNS

COMMON /NJA/ NJA(999,4)

COMMON /NRM/ NRM(999)

COMMON /NS/ NS

COMMON /QSA/ QSA(999,4,30)

COMMON /RMI/ RMI(999,16)

INTEGER RMI

INTEGER FLAG

INTEGER FLAG1

INTEGER FLAG2

INTEGER FLAG3

INTEGER SECODE

C

C

DO 410 IXXX1=1,MXI

I=INO(IXXX1)

NXXX1=NRM(I)

IF(.NOT.(NXXX1.NE.0)) GO TO 400

DO 280 NS=1,MXNS

RMI(I,NS)=BSP(NS)+1

```

FLAG1=0
FLAG2=0
DO 240 IRMT=1,MXIRMT
  IF(.NOT.(IRMT.LE.NXXX1)) GO TO 250
  IF(.NOT.(IRMIN(I,IRMT).EQ.RMI(I,NS))) GO TO 210
  FLAG1=IRMT
210  CONTINUE
  IF(.NOT.(IRMIN(I,IRMT).EQ.2)) GO TO 220
  FLAG2=IRMT
220  CONTINUE
  IF(.NOT.(IRMIN(I,IRMT).EQ.1)) GO TO 230
  FLAG3=IRMT
230  CONTINUE
240  CONTINUE
250  CONTINUE
  FLAG=FLAG1
  IF(.NOT.(FLAG1.EQ.0)) GO TO 270
  FLAG=FLAG2
  RMI(I,NS)=2
  IF(.NOT.(FLAG2.EQ.0)) GO TO 260
  FLAG=FLAG3
  RMI(I,NS)=1
260  CONTINUE
270  CONTINUE
280  CONTINUE
DO 380 IRMT=1,MXIRMT
  IF(.NOT.(IRMT.LE.NXXX1)) GO TO 390
  NXXX2=NJA(I,IRMT)
  IF(.NOT.(NXXX2.NE.0)) GO TO 370
  DO 350 LT=1,MXLT
    IF(.NOT.(LT.LE.NXXX2)) GO TO 360
    L=A(I,IRMT,LT)
    IF(.NOT.(IRMIN(I,IRMT).EQ.1)) GO TO 300
    SECODE=INT(QSA(I,IRMT,LT)/100.)
    IF(.NOT.(SECODE.GT.0.AND.SECODE.LT.3)) GO TO 290
    ERHAD(L)=ERHAD(L)+ERHD(I)
290  CONTINUE
300  CONTINUE
DO 340 NS=1,MXNS
  IF(.NOT.(IRMIN(I,IRMT).EQ.RMI(I,NS))) GO TO 330
  SECODE=INT(QSA(I,IRMT,LT)/100.)
  IF(.NOT.(SECODE.GT.1)) GO TO 310
  ERHAB(L,NS)=ERHAB(L,NS)+EBCBI(I,NS)
310  CONTINUE
  IF(.NOT.(SECODE.GT.0.AND.SECODE.LT.3)) GO TO 320
  ERHAB(L,NS)=ERHAB(L,NS)+ERTBI(I,NS)
320  CONTINUE

```

```
330          CONTINUE
340          CONTINUE
350          CONTINUE
360          CONTINUE
370          CONTINUE
380          CONTINUE
390          CONTINUE
400          CONTINUE
410 CONTINUE
C
      RETURN
      END
```

SUBROUTINE ZISET

C 800827 111248356
 C*****
 C* SSS MOD SLR 21 MAY 80 *
 C* CALCULATES THE MAXIMUM NUMBER OF SUPPORT *
 C* EQUIP. OF TYPE L REQUIRED: ISET(L,NS) AT *
 C* BASE NS AND ISETD(L) AT THE DEPOT *
 C*****
 C

COMMON /A/ A(999,4,30)
 INTEGER A
 COMMON /ERHBI/ ERHBI(999,16)
 COMMON /ERHD/ ERHD(999)
 COMMON /I/ I
 COMMON /INO/ INO(999)
 COMMON /IRHIN/ IRMIN(999,4)
 COMMON /IRMT/ IRMT
 COMMON /ISET/ ISET(250,16)
 REAL ISET
 COMMON /ISETD/ ISETD(250)
 REAL ISETD
 COMMON /L/ L
 COMMON /LT/ LT
 COMMON /MXI/ MXI
 COMMON /MXIRMT/ MXIRMT
 COMMON /MXLT/ MXLT
 COMMON /MXNS/ MXNS
 COMMON /NJA/ NJA(999,4)
 COMMON /NRM/ NRM(999)
 COMMON /NS/ NS
 COMMON /QSA/ QSA(999,4,30)
 COMMON /RMI/ RMI(999,16)
 INTEGER RMI

C
 C

DO 360 IXXX1=1,MXI
 I=INO(IXXX1)
 NXXX1=NRM(I)
 IF(.NOT.(NXXX1.NE.0)) GO TO 350
 DO 270 NS=1,MXNS
 IF(.NOT.(ERHBI(I,NS).GT.0.000001)) GO TO 270
 DO 250 IRMT=1,MXIRMT
 IF(.NOT.(IRMT.LE.NXXX1)) GO TO 260
 IF(.NOT.(IRMIN(I,IRMT).EQ.RMI(I,NS))) GO TO 240
 NXXX2=NJA(I,IRMT)
 IF(.NOT.(NXXX2.NE.0)) GO TO 230
 DO 210 LT=1,MXLT

```

                IF(.NOT.(LT.LE.NXXX2)) GO TO 220
                L=A(I,IRMT,LT)
                TQSA=QSA(I,IRMT,LT)-AINT(QSA(I,IRMT,LT)/100.)*100.
                ISET(L,NS)=AMAX1(ISET(L,NS),TQSA)
210             CONTINUE
220             CONTINUE
230             CONTINUE
240             CONTINUE
250             CONTINUE
260             CONTINUE
270             CONTINUE
                IF(.NOT.(ERHD(I).GT.0.000001)) GO TO 340
                DO 320 IRMT=1,MXIRMT
                    IF(.NOT.(IRMT.LE.NXXX1)) GO TO 330
                    IF(.NOT.(IRMIN(I,IRMT).EQ.1)) GO TO 310
                    NXXX2=NJA(I,IRMT)
                    IF(.NOT.(NXXX2.GT.0)) GO TO 300
                    DO 280 LT=1,MXLT
                        IF(.NOT.(LT.LE.NXXX2)) GO TO 290
                        L=A(I,IRMT,LT)
                        TQSA=QSA(I,IRMT,LT)-AINT(QSA(I,IRMT,LT)/100.)*100.
                        ISETD(L)=AMAX1(ISETD(L),TQSA)
280                     CONTINUE
290                     CONTINUE
300                     CONTINUE
310                     CONTINUE
320                     CONTINUE
330                     CONTINUE
340                     CONTINUE
350                     CONTINUE
360 CONTINUE
C
    RETURN
    END

```

SUBROUTINE ZUSE

C

800827 111316209

C*****

C* COMPUTES UTILIZATION TO BE USED FOR *

C* SENSITIVITY BMF &DMF FACTORS *

C*****

C

```
COMMON /BAA/ BAA
COMMON /DAA/ DAA
COMMON /ERHAB/ ERHAB(250,16)
COMMON /ERHAD/ ERHAD(250)
COMMON /L/ L
COMMON /MUSE/ MUSE
REAL MUSE
COMMON /MXL/ MXL
COMMON /MXNS/ MXNS
COMMON /NS/ NS
COMMON /SEINO/ SEINO(250)
INTEGER SEINO
COMMON /SETYPE/ SETYPE(250)
INTEGER SETYPE
COMMON /USE/ USE(250,16)
COMMON /USED/ USED(250)
REAL MUSEB
REAL MUSED
```

C

C

```
MUSEB=MUSE*BAA
MUSED=MUSE*DAA
DO 250 IXXX1=1,MXL
  L=SEINO(IXXX1)
  IF(.NOT.(SETYPE(L).GE.2)) GO TO 240
  DO 220 NS=1,MXNS
    IF(.NOT.(ERHAB(L,NS).GT..000001.AND.ERHAB(L,NS).LT.
+      MUSEB)) GO TO 210
    USE(L,NS)=0.
210    CONTINUE
220    CONTINUE
    IF(.NOT.(ERHAD(L).GT..000001.AND.ERHAD(L).LT.MUSED)) GO TO 230
    USED(L)=0.
230    CONTINUE
240    CONTINUE
250 CONTINUE
```

C

```
RETURN
END
```

SUBROUTINE ZTYPE

```

C
C*****800827 111323268*****
C* COMPUTES SAT AND CIMF FACTORS ACCORDING *
C* TO BTYPE FOR SENSITIVITY CALCULATIONS *
C*****
C
COMMON /BTYPE/ BTYPE(16)
INTEGER BTYPE
COMMON /CIMF/ CIMF(16)
COMMON /MXNS/ MXNS
COMMON /NS/ NS
COMMON /SAT/ SAT(16)
C
C
DO 230 NS=1,MXNS
  SAT(NS)=0.
  CIMF(NS)=0.
  IF(.NOT.(BTYPE(NS).EQ.3)) GO TO 210
  SAT(NS)=1.
210  CONTINUE
  IF(.NOT.(BTYPE(NS).EQ.2)) GO TO 220
  CIMF(NS)=1.
220  CONTINUE
230  CONTINUE
C
RETURN
END

```

SUBROUTINE ZTFR

C 800827 111333120
 C*****
 C* COMPUTES ITEM FAILURE RATE *
 C*****
 C

COMMON /APFH/ APFH(10,3)
 COMMON /FAIL/ FAIL(999,16)
 COMMON /FPLT/ FPLT(999)
 COMMON /FPM/ FPM(999)
 COMMON /I/ I
 COMMON /INO/ INO(999)
 COMMON /LO/ LO(16)
 COMMON /MXI/ MXI
 COMMON /MXNP/ MXNP
 COMMON /MXNS/ MXNS
 COMMON /NITEM/ NITEM(999,10)
 REAL NITEM
 COMMON /NP/ NP
 COMMON /NPLT/ NPLT(10,16)
 REAL NPLT
 COMMON /NS/ NS
 COMMON /PIUP/ PIUP
 COMMON /TFAC/ TFAC(10)
 COMMON /TFR/ TFR(999)
 COMMON /TNB/ TNB(16)

C
 C

DO 240 IXXX1=1,MXI
 I=INO(IXXX1)
 TEM01=0.
 DO 210 NS=1,MXNS
 TEM01=TEM01+TNB(NS)*FAIL(I,NS)
 210 CONTINUE
 FPM(I)=TEM01
 FPLT(I)=12.*PIUP*FPM(I)
 TEM03=0.
 DO 230 NS=1,MXNS
 TEM02=0.
 DO 220 NP=1,MXNP
 TEM02=TEM02+NITEM(I,NP)*NPLT(NP,NS)*APFH(NP,LO(NS))*TFAC(NP)
 220 CONTINUE
 TEM03=TEM03+TNB(NS)*TEM02
 230 CONTINUE
 TOO=TEM03
 TFR(I)=FPM(I)*1000000./TOO
 240 CONTINUE

C

RETURN
END

SUBROUTINE ZSECI

C

800827 111349246

C*****

C* COMPUTES SECI(I),THE PRO RATA PART OF SEC

*

C* SSS MOD JRC 2 JUN 80

*

C*****

C

```
COMMON /A/ A(999,4,30)
INTEGER A
COMMON /BAA/ BAA
COMMON /COND/ COND(999)
COMMON /CSE/ CSE(250)
COMMON /DAA/ DAA
COMMON /ERHA/ ERHA(250)
COMMON /ERHAB/ ERHAB(250,16)
COMMON /ERHAD/ ERHAD(250)
COMMON /ERHBI/ ERHBI(999,16)
COMMON /ERHD/ ERHD(999)
COMMON /I/ I
COMMON /INO/ INO(999)
COMMON /IRMIN/ IRMIN(999,4)
COMMON /IRMT/ IRMT
COMMON /ISET/ ISET(250,16)
REAL ISET
COMMON /ISETD/ ISETD(250)
REAL ISETD
COMMON /L/ L
COMMON /LT/ LT
COMMON /MSE/ MSE(250)
REAL MSE
COMMON /MXI/ MXI
COMMON /MXIRMT/ MXIRMT
COMMON /MXL/ MXL
COMMON /MXLT/ MXLT
COMMON /MXNS/ MXNS
COMMON /NJA/ NJA(999,4)
COMMON /NRM/ NRM(999)
COMMON /NS/ NS
COMMON /NSEB/ NSEB(250,16)
REAL NSEB
COMMON /NSED/ NSED(250)
REAL NSED
COMMON /PBDV/ PBDV(250)
COMMON /PDDV/ PDDV(250)
COMMON /PIUP/ PIUP
COMMON /QSA/ QSA(999,4,30)
COMMON /RMI/ RMI(999,16)
```

```

INTEGER RMI
COMMON /SECB/ SECB(250)
COMMON /SECD/ SECD(250)
COMMON /SECI/ SECI(999)
COMMON /SEDEV/ SEDEV(250)
COMMON /SEDV/ SEDV(250)
COMMON /SEINO/ SEINO(250)
INTEGER SEINO
COMMON /SETYPE/ SETYPE(250)
INTEGER SETYPE
COMMON /TERHB/ TERHB(250)
COMMON /TERHD/ TERHD(250)
COMMON /TNB/ TNB(16)
COMMON /TUCTDC/ TUCTDC
COMMON /UCTDC/ UCTDC(999)
COMMON /UCTDEV/ UCTDEV(999)
REAL NXXX1
REAL NXXX2
DIMENSION T2(999,250)
DIMENSION T4(999,250)

```

C
C

```

DO 210 IXXX1=1,MXI
  I=INO(IXXX1)
  UCTDC(I)=UCTDEV(I)*U(1.-COND(I))
  TUCTDC=TUCTDC+UCTDC(I)
210 CONTINUE
DO 270 IXXX1=1,MXL
  L=SEINO(IXXX1)
  DO 240 NS=1,MXNS
    IF(.NOT.(SETYPE(L).EQ.1)) GO TO 220
    NSEB(L,NS)=ERHAB(L,NS)/BAA*ISET(L,NS)
220    CONTINUE
    IF(.NOT.(SETYPE(L).GE.2)) GO TO 230
    NSEB(L,NS)=AINT(ERHAB(L,NS)/BAA+.9999)*ISET(L,NS)
230    CONTINUE
240    CONTINUE
    IF(.NOT.(SETYPE(L).EQ.1)) GO TO 250
    NSED(L)=ERHAD(L)/DAA*ISETD(L)
250    CONTINUE
    IF(.NOT.(SETYPE(L).GE.2)) GO TO 260
    NSED(L)=AINT(ERHAD(L)/DAA+.9999)*ISETD(L)
260    CONTINUE
270 CONTINUE
DO 430 IXXX1=1,MXI
  I=INO(IXXX1)
  DO 280 IXXX2=1,MXL

```

```

        L=SEINO(IXXX2)
        T2(I,L)=0.
        T4(I,L)=0.
280    CONTINUE
        NXXX1=NRM(I)
        IF(.NOT.(NXXX1.GT.0)) GO TO 360
        DO 350 NS=1,MXNS
            DO 330 IRMT=1,MXIRMT
                IF(.NOT.(IRMT.LE.NXXX1)) GO TO 340
                IF(.NOT.(IRMIN(I,IRMT).EQ.RMI(I,NS))) GO TO 320
                NXXX2=NJA(I,IRMT)
                IF(.NOT.(NXXX2.GT.0)) GO TO 310
                DO 290 LT=1,MXLT
                    IF(.NOT.(LT.LE.NXXX2)) GO TO 300
                    L=A(I,IRMT,LT)
                    T1=ERHBI(I,NS)*TNB(NS)*(QSA(I,IRMT,LT)-AINT(QSA(I,
+                    IRMT,LT)/100.)*100.)
                    TERHB(L)=TERHB(L)+T1
                    T2(I,L)=T2(I,L)+T1
290                CONTINUE
300                CONTINUE
310            CONTINUE
320        CONTINUE
330    CONTINUE
340    CONTINUE
350    CONTINUE
360    CONTINUE
        DO 410 IRMT=1,MXIRMT
            IF(.NOT.(IRMT.LE.NXXX1)) GO TO 420
            IF(.NOT.(IRMIN(I,IRMT).EQ.1)) GO TO 400
            NXXX2=NJA(I,IRMT)
            IF(.NOT.(NXXX2.GT.0)) GO TO 390
            DO 370 LT=1,MXLT
                IF(.NOT.(LT.LE.NXXX2)) GO TO 380
                L=A(I,IRMT,LT)
                T3=ERHD(I)*(QSA(I,IRMT,LT)-AINT(QSA(I,IRMT,LT)/100.)*
+                100.)
                TERHD(L)=TERHD(L)+T3
                T4(I,L)=T4(I,L)+T3
370            CONTINUE
380            CONTINUE
390        CONTINUE
400    CONTINUE
410    CONTINUE
420    CONTINUE
430    CONTINUE
        DO 470 IXXX1=1,MXL

```

```

L=SEINO(IXXX1)
SECD(L)=NSED(L)*CSE(L)*(1.+PIUP*MSE(L))
TEM01=0.
DO 440 NS=1,MXNS
    TEM01=TEM01+NSEB(L,NS)*TNB(NS)
440 CONTINUE
    SECB(L)=TEM01*CSE(L)*(1.+PIUP*MSE(L))
    PBDV(L)=0.
    PDDV(L)=0.
    TEM02=0.
    DO 450 NS=1,MXNS
        TEM02=TEM02+ERHAB(L,NS)
450 CONTINUE
    ERHA(L)=TEM02+ERHAD(L)
    SEDV(L)=U(ERHA(L))*SEDEV(L)
    IF(.NOT.(TERHB(L).GT.0.000001.OR.TERHD(L).GT.
+ 0.000001)) GO TO 460
    PBDV(L)=TERHB(L)*SEDEV(L)/(TERHB(L)+TERHD(L))
    PDDV(L)=TERHD(L)*SEDEV(L)/(TERHB(L)+TERHD(L))
460 CONTINUE
470 CONTINUE
    DO 510 IXX1=1,MXI
        I=INO(IXXX1)
        SECI(I)=UCTDC(I)
        DO 500 IXX2=1,MXL
            L=SEINO(IXXX2)
            IF(.NOT.(TERHB(L).GT.0.000001)) GO TO 480
            SECI(I)=SECI(I)+(SECB(L)+PBDV(L))*T2(I,L)/TERHB(L)
480 CONTINUE
            IF(.NOT.(TERHD(L).GT.0.000001)) GO TO 490
            SECI(I)=SECI(I)+(SECD(L)+PDDV(L))*T4(I,L)/TERHD(L)
490 CONTINUE
500 CONTINUE
510 CONTINUE
C
    RETURN
    END

```

SUBROUTINE ZPMEQ

800827 111440789

C

C*****

C* COMPUTES PRIME MISSION EQUIPMENT QUANTITIES *

C* FOR EACH LRU ITEM *

C* SSS MOD SLR - 15 MAY 80 *

C*****

C

COMMON /I/ I
COMMON /INO/ INO(999)
COMMON /LRU/ LRU(999)
COMMON /MXI/ MXI
COMMON /MXNP/ MXNP
COMMON /MXNS/ MXNS
COMMON /NITEM/ NITEM(999,10)
REAL NITEM
COMMON /NP/ NP
COMMON /NPLT/ NPLT(10,16)
REAL NPLT
COMMON /NS/ NS
COMMON /PMEQ/ PMEQ(999)
COMMON /TNB/ TNB(16)

C

C

DO 230 IXXX1=1,MXI
I=INO(IXXX1)
TEM02=0.
DO 220 NS=1,MXNS
TEM01=0.
DO 210 NP=1,MXNP
TEM01=TEM01+NPLT(NP,NS)*NITEM(I,NP)
210 CONTINUE
TEM02=TEM02+TEM01*TNB(NS)
220 CONTINUE
PMEQ(I)=TEM02*FLOAT(LRU(I))
230 CONTINUE

C

RETURN
END

SUBROUTINE ZTISQ

800827 111445592

```

C
C*****
C* COMPUTES TOTAL INVESTMENT SPARES QUANTITY *
C* FOR AN ITEM OF TYPE I *
C* SSS MOD SLR - 15 MAY 80 *
C*****
C
COMMON /DS/ DS(999)
COMMON /I/ I
COMMON /INO/ INO(999)
COMMON /MXI/ MXI
COMMON /MXNS/ MXNS
COMMON /NFB/ NFB(999,16)
REAL NFB
COMMON /NFD/ NFD(999)
REAL NFD
COMMON /NS/ NS
COMMON /TISQ/ TISQ(999)
COMMON /TNB/ TNB(16)

C
C
DO 220 IXXX1=1,MXI
  I=INO(IXXX1)
  TISQ(I)=0.
  DS(I)=F(NFD(I))
  DO 210 NS=1,MXNS
    BBS=F(NFB(I,NS))
    TISQ(I)=TISQ(I)+(TNB(NS)*BBS)
210  CONTINUE
    TISQ(I)=TISQ(I)+DS(I)
220  CONTINUE
C
RETURN
END

```

SUBROUTINE ZYRSQ

800827 111451867

C

C*****

C* COMPUTES YEARLY REPLACEMENT SPARES QUANTITY *

C* DUE TO AN INDIVIDUAL ITEM TYPE I *

C* SSS MOD SLR - 15 MAY 80 *

C*****

C

COMMON /COND/ COND(999)

COMMON /FAIL/ FAIL(999,16)

COMMON /I/ I

COMMON /INO/ INO(999)

COMMON /MXI/ MXI

COMMON /MXNS/ MXNS

COMMON /NHI/ NHI(999)

COMMON /NS/ NS

COMMON /TNB/ TNB(16)

COMMON /YRSQ/ YRSQ(999)

C

C

DO 230 IXXX1=1,MXI

I=INO(IXXX1)

CD=0.

IF(.NOT.(NHI(I).GT.0)) GO TO 210

CD=COND(NHI(I))

210 CONTINUE

TEM01=0.

DO 220 NS=1,MXNS

TEM01=TEM01+FAIL(I,NS)*TNB(NS)

220 CONTINUE

YRSQ(I)=12.*TEM01*(1.-CD)

230 CONTINUE

C

RETURN

END

SUBROUTINE ZTOTPQ

C 800827 111502753

C*****

C* COMPUTES THE TOTAL QUANTITIES OF *

C* EACH ITEM TO BE PROCURED *

C* SSS MOD SLR - 15 MAY 80 *

C*****

C

COMMON /I/ I

COMMON /INO/ INO(999)

COMMON /MXI/ MXI

COMMON /NRUC/ NRUC

REAL NRUC

COMMON /PMEQ/ PMEQ(999)

COMMON /TISQ/ TISQ(999)

COMMON /TOTPQ/ TOTPQ(999)

COMMON /YRSQ/ YRSQ(999)

C

C

DO 210 IXXX1=1,MXI

I=INO(IXXX1)

TOTPQ(I)=PMEQ(I)+TISQ(I)+NRUC*YRSQ(I)

210 CONTINUE

C

RETURN

END

SUBROUTINE ZLC

```
C
C***** 800827 111506859 *****
C* LEARNING CURVE EFFECTS FOR ALL ITEMS OF TYPE I *
C* SSS MOD SLR - 15 MAY 80 *
C*****
C
COMMON /I/ I
COMMON /INO/ INO(999)
COMMON /LC/ LC(999)
REAL LC
COMMON /MXI/ MXI
C
C
DO 210 IXXX1=1,MXI
I=INO(IXXX1)
LC(I)=XLEARN(I)
210 CONTINUE
C
RETURN
END
```

FUNCTION U(X)

800827 111510589

C
C
C

U=0.

IF(.NOT.(X.GT..000001)) GO TO 210

U=1.

210 CONTINUE

C

RETURN

END

```

      FUNCTION F(X)
C
      COMMON /BF/ BF
C
C
      F=0.
      IF(.NOT.(X.GT..000001)) GO TO 210
      F=X+BF*SQRT(X)
210 CONTINUE
C
      RETURN
      END

```

800827 111515246

```

      FUNCTION XLEARN(I)
C
C***** 800827 111520472 *****
C* LEARNING EFFECTS IN EQUATIONS *
C* SSS MOD SLR - 20 MAY 80 *
C*****
C
      COMMON /LFAC/ LFAC(999)
      REAL LFAC
      COMMON /TOTPQ/ TOTPQ(999)
      COMMON /XITEMQ/ XITEMQ(999)
      REAL N
C
C
      XLEARN=1.
      BI=ALOG10(LFAC(I))/ALOG10(2.)
      N=TOTPQ(I)+XITEMQ(I)
      IF(.NOT.(N.GT.0.000001)) GO TO 210
      XLEARN=1./N*(1./(BI+1.)*(N**(BI+1.)-1.)+.5*(N**BI+1.)+BI/12.*(N*
+      *(BI-1.)-1.))
      210 CONTINUE
C
      RETURN
      END

```

SUBROUTINE COST1

C 800827 111526680
 C*****
 C* COMPUTES PRODUCTION COST ELEMENT-PRODC *
 C* SSS MOD SLR - 20 MAY 80 *
 C*****
 C

COMMON /HDWRIT/ HDWRIT(999,10)
 COMMON /HDWRT/ HDWRT(10)
 COMMON /I/ I
 COMMON /INO/ INO(999)
 COMMON /INTNR/ INTNR(10)
 REAL INTNR
 COMMON /INTR/ INTR(10)
 REAL INTR
 COMMON /LC/ LC(999)
 REAL LC
 COMMON /LRU/ LRU(999)
 COMMON /LUP/ LUP(999)
 REAL LUP
 COMMON /MXI/ MXI
 COMMON /MXNP/ MXNP
 COMMON /MXNS/ MXNS
 COMMON /NITEM/ NITEM(999,10)
 REAL NITEM
 COMMON /NP/ NP
 COMMON /NPLT/ NPLT(10,16)
 REAL NPLT
 COMMON /NS/ NS
 COMMON /PRODC/ PRODC
 COMMON /TERMC/ TERMC(10)
 COMMON /TERMH/ TERMH
 COMMON /TERMI/ TERMI
 COMMON /TNB/ TNB(16)
 COMMON /TOTT/ TOTT(10)
 COMMON /UP/ UP(999)
 COMMON /XUC/ XUC

C
 C

DO 240 NP=1,MXNP
 HDWRT(NP)=0.
 DO 220 IXXX2=1,MXI
 I=INO(IXXX2)
 LUP(I)=UP(I)*LC(I)
 HDWRIT(I,NP)=NITEM(I,NP)*LUP(I)*XUC
 IF(.NOT.(LRU(I).EQ.1)) GO TO 210
 HDWRT(NP)=HDWRT(NP)+HDWRIT(I,NP)

```

210     CONTINUE
220     CONTINUE
      TEM01=0.
      DO 230 NS=1,MXNS
        T.M01=TEM01+TNB(NS)*NPLT(NP,NS)
230     CONTINUE
      TOT(T(NP))=TEM01
      TERMC(NP)=(INTNR(NP)/TOT(T(NP)))+INTR(NP)+HDWRT(NP)
      TERMH=TERMH+(TOT(T(NP))*HDWRT(NP))
      TERMI=TERMI+(TOT(T(NP))*INTR(NP))+INTNR(NP)
240     CONTINUE
      PRODC=TERMH+TERMI
C
      RETURN
      END

```

SUBROUTINE COST2

800827 111544007

C
 C*****
 C* COMPUTES MODIFICATION/INSTALLATION *
 C* COST ELEMENT-MIC *
 C*****

C
 COMMON /AKIT/ AKIT(4,10)
 COMMON /FR/ FR(3,10)
 COMMON /IA/ IA
 COMMON /IMICA/ IMICA(10)
 REAL IMICA
 COMMON /M/ M
 COMMON /MIC/ MIC
 REAL MIC
 COMMON /MIFIX/ MIFIX(3,10)
 REAL MIFIX
 COMMON /MILR/ MILR(3)
 REAL MILR
 COMMON /MIMH/ MIMH(4,3,10)
 REAL MIMH
 COMMON /MXM/ MXM
 COMMON /MXNP/ MXNP
 COMMON /MXNS/ MXNS
 COMMON /NIA/ NIA
 COMMON /NP/ NP
 COMMON /NPLT/ NPLT(10,16)
 REAL NPLT
 COMMON /NRMI/ NRMI(10)
 REAL NRMI
 COMMON /NS/ NS
 COMMON /PDIV/ PDIV(10)
 COMMON /RMICA/ RMICA(10)
 COMMON /TNB/ TNB(16)
 COMMON /XMIL/ XMIL
 REAL IMIC

C
 C
 C

C.....FIRST COMPUTE NON-RECURRING MOD/INSTALL COST
 IMIC=0.
 RMIC=0.
 DO 240 NP=1,MXNP
 IMICA(NP)=PDIV(NP)*NRMI(NP)
 IMIC=IMIC+IMICA(NP)

C

C.....NEXT COMPUTE RECURRING MOD/INSTALL COST

```

      TEM02=0.
      DO 220 M=1,MXM
        TEM01=0.
        DO 210 IA=1,NIA
          TEM01=TEM01+MIMH(IA,M,NP)*XMIL*MILR(M)+AKIT(IA,NP)
210      CONTINUE
          TEM02=TEM02+FR(M,NP)*((MIFIX(M,NP)*1000.)+TEM01)
220      CONTINUE
          RMICA(NP)=TEM02
          TEM03=0.
          DO 230 NS=1,MXNS
            TEM03=TEM03+TNB(NS)*NPLT(NP,NS)*RMICA(NP)
230      CONTINUE
            RMIC=RMIC+TEM03
240      CONTINUE
C
C.....TOTAL MOD/INSTALL COST IS THEN:
      MIC=IMIC+RMIC
C
      RETURN
      END

```

SUBROUTINE COST3

C 800827 111606484
 C*****
 C* COMPUTES OPERATIONS COST ELEMENT-OC *
 C*****
 C

COMMON /AFC/ AFC
 COMMON /AMPM/ AMPM(10,3)
 COMMON /APFH/ APFH(10,3)
 COMMON /BAFC/ BAFC(6)
 COMMON /BOLC/ BOLC(6)
 COMMON /BPLAT/ BPLAT(16)
 INTEGER BPLAT
 COMMON /BTYPE/ BTYPE(16)
 INTEGER BTYPE
 COMMON /CFG/ CFG(3)
 COMMON /DRAG/ DRAG(10)
 COMMON /FGH/ FGH(10)
 COMMON /K/ K(10)
 REAL K
 COMMON /LO/ LO(16)
 COMMON /MMPD/ MMPD(10,3)
 REAL MMPD
 COMMON /MMPM/ MMPM(10)
 REAL MMPM
 COMMON /MXNP/ MXNP
 COMMON /MXNS/ MXNS
 COMMON /NAE/ NAE(10)
 REAL NAE
 COMMON /NP/ NP
 COMMON /NPLT/ NPLT(10,16)
 REAL NPLT
 COMMON /NS/ NS
 COMMON /OC/ OC
 COMMON /OLC/ OLC
 COMMON /OLCP/ OLCP
 COMMON /OLCT/ OLCT
 COMMON /PIUP/ PIUP
 COMMON /PMLR/ PMLR
 COMMON /THRS/ THRS(10)
 COMMON /TNB/ TNB(16)
 COMMON /TNLR/ TNLR

C
 C
 C

C.....FIRST COMPUTE OPERATIONAL LABOR COST (OLC)
 DO 230 NS=1,MXNS

```

      TEM01=0.
      DO 210 NP=1,MXNP
        TEM01=TEM01+NPLT(NP,NS)*TNB(NS)*MMPD(NP,LO(NS))
210    CONTINUE
      DS1=TEM01*365.*PIUP*TNLR/60.
      TEM02=0.
      DO 220 NP=1,MXNP
        TEM02=TEM02+NPLT(NP,NS)*TNB(NS)*MMPM(NP)*AMPM(NP,LO(NS))
220    CONTINUE
      DS2=TEM02*12.*PIUP*PMLR/60.
      OLCT=OLCT+DS1
      OLCP=OLCP+DS2
      BOLC(BTYPE(NS))=BOLC(BTYPE(NS))+DS1+DS2
      IB=3+BPLAT(NS)
      BOLC(IB)=BOLC(IB)+DS1+DS2
230    CONTINUE
      OLC=OLCT+OLCP
C
C.....NEXT COMPUTE ADDED FUEL COST (AFC)
      DO 250 NS=1,MXNS
        TEM03=0.
        DO 240 NP=1,MXNP
          IF(.NOT.(K(NP).GT.0.0001.AND.THRS(NP).GT.0.0001)) GO TO 240
          TEM03=TEM03+NPLT(NP,NS)*TNB(NS)*APFH(NP,LO(NS))*FGH(NP)*
+          CFG(LO(NS))*NAE(NP)*DRAG(NP)/(K(NP)*THRS(NP))
240    CONTINUE
          DS3=TEM03*12.*PIUP
          AFC=AFC+DS3
          BAFC(BTYPE(NS))=BAFC(BTYPE(NS))+DS3
          IB=3+BPLAT(NS)
          BAFC(IB)=BAFC(IB)+DS3
250    CONTINUE
C
C.....TOTAL OPERATIONS COST (OC)
      OC=OLC+AFC
C
      RETURN
      END

```

SUBROUTINE COST4

```

C
C***** 800827 111646289
C* COMPUTES INVESTMENT SPARES COST ELEMENT-ISC *
C* SSS MOD SLR - 20 MAY 80 *
C*****

```

C

```

COMMON /BISC/ BISC(6)
COMMON /BPLAT/ BPLAT(16)
INTEGER BPLAT
COMMON /BS/ BS(999)
COMMON /BTYPE/ BTYPE(16)
INTEGER BTYPE
COMMON /DS/ DS(999)
COMMON /I/ I
COMMON /INO/ INO(999)
COMMON /ISC/ ISC
REAL ISC
COMMON /ISCA/ ISCA(999)
REAL ISCA
COMMON /ISCB/ ISCB
REAL ISCB
COMMON /ISCD/ ISCD
REAL ISCD
COMMON /LC/ LC(999)
REAL LC
COMMON /MXI/ MXI
COMMON /MXNS/ MXNS
COMMON /NFB/ NFB(999,16)
REAL NFB
COMMON /NFD/ NFD(999)
REAL NFD
COMMON /NS/ NS
COMMON /TISQ/ TISQ(999)
COMMON /TNB/ TNB(16)
COMMON /UP/ UP(999)
COMMON /XUC/ XUC

```

C

C

```

DO 220 IXXX1=1,MXI
  I=INO(IXXX1)
  ISCA(I)=TISQ(I)*UP(I)*LC(I)*XUC
DO 210 NS=1,MXNS
  BBSP=TNB(NS)*F(NFB(I,NS))
  BS(I)=BS(I)+BBSP
  BSPC=BBSP*UP(I)*XUC*LC(I)
  BISC(BTYPE(NS))=BISC(BTYPE(NS))+BSPC

```

```

        IB=3+BPLAT(NS)
        BISC(IB)=BISC(IB)+BSPC
        ISCB=ISCB+BSPC
210    CONTINUE
        DS(I)=F(NFD(I))
        ISCD=ISCD+DS(I)*UP(I)*XUC*LC(I)
220    CONTINUE
        TEM01=0.
        DO 230 IXXX1=1,MXI
            I=INO(IXXX1)
            TEM01=TEM01+ISCA(I)
230    CONTINUE
        ISC=TEM01
C
        RETURN
        END

```

SUBROUTINE COST5

800827 111657536

C
 C*****
 C* COMPUTES REPLACEMENT SPARES COST ELEMENT-RSC *
 C* SSS MOD SLR - 20 MAY 80 *
 C*****

C

COMMON /BPLAT/ BPLAT(16)
 INTEGER BPLAT
 COMMON /BRSC/ BRSC(6)
 COMMON /BTYP/ BTYP(16)
 INTEGER BTYP
 COMMON /COND/ COND(999)
 COMMON /FAIL/ FAIL(999,16)
 COMMON /I/ I
 COMMON /INO/ INO(999)
 COMMON /LC/ LC(999)
 REAL LC
 COMMON /LRU/ LRU(999)
 COMMON /MXI/ MXI
 COMMON /MXNS/ MXNS
 COMMON /NHI/ NHI(999)
 COMMON /NRUC/ NRUC
 REAL NRUC
 COMMON /NS/ NS
 COMMON /PIUP/ PIUP
 COMMON /RM/ RM(999)
 COMMON /RSC/ RSC
 COMMON /RSCA/ RSCA(999)
 COMMON /TNB/ TNB(16)
 COMMON /UP/ UP(999)
 COMMON /XUC/ XUC
 COMMON /YRSQ/ YRSQ(999)

C

C

DO 230 IXXX1=1,MXI
 I=INO(IXXX1)
 RSCA(I)=(NRUC*LC(I)+(PIUP-NRUC))*YRSQ(I)*(COND(I)+(1.-COND(I))*
 + RM(I))*UP(I)*XUC
 CD=0.
 IF(.NOT.(LRU(I).EQ.0)) GO TO 210
 CD=COND(NHI(I))
 210 CONTINUE
 DO 220 NS=1,MXNS
 TFL=TNB(NS)*FAIL(I,NS)
 BSPC=12.*TFL*(1.-CD)*(COND(I)+(1.-COND(I))*RM(I))*UP(I)*
 + XUC*(NRUC*LC(I)-(PIUP-NRUC))

```
        BRSC(BTYPE(NS))=BRSC(BTYPE(NS))+BSPC
        IB=3+BPLAT(NS)
        BRSC(IB)=BRSC(IB)+BSPC
220    CONTINUE
        RSC=RSC+RSCA(I)
230    CONTINUE
C
        RETURN
        END
```

SUBROUTINE COST6

```

C
C*****800827 111714868
C* COMPUTES ON-EQUIPMENT MAINTENANCE *
C* COST ELEMENT-ONMC *
C*****
C
COMMON /APFH/ APFH(10,3)
COMMON /BLR/ BLR
COMMON /BMF/ BMF
COMMON /BONMC/ BONMC(6)
COMMON /BPLAT/ BPLAT(16)
INTEGER BPLAT
COMMON /BTYP/ BTYP(16)
INTEGER BTYP
COMMON /FAIL/ FAIL(999,16)
COMMON /FPR/ FPR(999)
COMMON /I/ I
COMMON /INO/ INO(999)
COMMON /IPCF/ IPCF(999)
REAL IPCF
COMMON /KFAC/ KFAC(4)
REAL KFAC
COMMON /LE/ LE(10)
COMMON /LO/ LO(16)
COMMON /LRU/ LRU(999)
COMMON /MTBMI/ MTBMI(999,4)
REAL MTBMI
COMMON /MXI/ MXI
COMMON /MXNP/ MXNP
COMMON /MXNS/ MXNS
COMMON /NITEM/ NITEM(999,10)
REAL NITEM
COMMON /NP/ NP
COMMON /NPLT/ NPLT(10,16)
REAL NPLT
COMMON /NS/ NS
COMMON /ONMC/ ONMC
COMMON /ONMCA/ ONMCA(999)
COMMON /PIUP/ PIUP
COMMON /RIP/ RIP(999)
COMMON /RMH/ RMH(999)
COMMON /TFAC/ TFAC(10)
COMMON /TNB/ TNB(16)
COMMON /XFPR/ XFPR
COMMON /XFR/ XFR

```

C

```

C
DO 250 IXXX1=1,MXI
  I=INO(IXXX1)
  IF(.NOT.(LRU(I).EQ.1)) GO TO 250
  DO 240 NS=1,MXNS
    TFL=TNB(NS)*FAIL(I,NS)
    DS1=12.*PIUP*TFL*(1.+XFPR*FPR(I))*RMH(I)*BMF*BLR
    IF(.NOT.(RIP(I).LT..999)) GO TO 210
    DS2=12.*PIUP*TFL*RIP(I)*IPCF(I)*BMF/(1.-RIP(I))
210  CONTINUE
C
C.....IF RIP(I)=1 COMPUTE NO. OF NON-REMOVED FAILURES
  IF(.NOT.(RIP(I).GE..999)) GO TO 230
  TEM01=0.
  DO 220 NP=1,MXNP
    IF(.NOT.(NITEM(I,NP).GT..001)) GO TO 220
    TEM01=TEM01+NITEM(I,NP)*TFAC(NP)*KFAC(LE(NP))/MTBMI(I,
+    LE(NP))*NPLT(NP,NS)*APFH(NP,LO(NS))
220  CONTINUE
    TFL=XFR*TEM01*TNB(NS)
    DS2=12.*PIUP*TFL*IPCF(I)*BMF
230  CONTINUE
    ONMCA(I)=ONMCA(I)+DS1+DS2
    BONMC(BTYPE(NS))=BONMC(BTYPE(NS))+DS1+DS2
    IB=3+BPLAT(NS)
    BONMC(IB)=BONMC(IB)+DS1+DS2
240  CONTINUE
    ONMC=ONMC+ONMCA(I)
250  CONTINUE
C
  RETURN
  END

```

SUBROUTINE COST7

C

800827 111732926

C*****

C* COMPUTES OFF-EQUIPMENT MAINTENANCE

*

C* COST ELEMENT-OFMC

*

C*****

C

```
COMMON /BCMh/ BCMh(999)
COMMON /BLR/ BLR
COMMON /BMF/ BMF
COMMON /BMH/ BMH(999)
COMMON /BOFMC/ BOFMC(6)
COMMON /BPLAT/ BPLAT(16)
INTEGER BPLAT
COMMON /BTYPE/ BTYPE(16)
INTEGER BTYPE
COMMON /COND/ COND(999)
COMMON /CPPC/ CPPC
COMMON /CPPD/ CPPD(3)
COMMON /DLR/ DLR
COMMON /DMF/ DMF
COMMON /DMH/ DMH(999)
COMMON /FAIL/ FAIL(999,16)
COMMON /FPR/ FPR(999)
COMMON /I/ I
COMMON /INO/ INO(999)
COMMON /LO/ LO(16)
COMMON /LRU/ LRU(999)
COMMON /MRF/ MRF
REAL MRF
COMMON /MRO/ MRO
REAL MRO
COMMON /MXI/ MXI
COMMON /MXNS/ MXNS
COMMON /NHB/ NHB(16)
COMMON /NHI/ NHI(999)
COMMON /NRTS/ NRTS(999)
REAL NRTS
COMMON /NS/ NS
COMMON /OFMC/ OFMC
COMMON /OFMCA/ OFMCA(999)
COMMON /OFMCB/ OFMCB
COMMON /OFMCD/ OFMCD
COMMON /PIUP/ PIUP
COMMON /RIP/ RIP(999)
COMMON /RTS/ RTS(999)
COMMON /SR/ SR
```

```

COMMON /TCFB/ TCFB
COMMON /TCFD/ TCFD
COMMON /TNB/ TNB(16)
COMMON /TR/ TR
COMMON /WT/ WT(999)
COMMON /XFPR/ XFPR
REAL NHNRT
REAL NHRT

```

C
C

```

DO 260 IXXX1=1,MXI
  I=INO(IXXX1)
  NHRT=0.
  NHNRT=0.
  IF(.NOT.(LRU(I).EQ.0)) GO TO 210
    NHRT=RTS(NHI(I))
    NHNRT=NRTS(NHI(I))
210  CONTINUE
    XF=XFPR*FPR(I)
    DO 250 NS=1,MXNS
      SATLRU=0.
      IF(.NOT.(RTYPE(NS).EQ.3.AND.LRU(I).EQ.1)) GO TO 220
        SATLRU=1.
220  CONTINUE
      ACFB=(FLOAT(LRU(I))+NHRT)*((1.+XF)*BCMH(I)+RTS(I)*BMH(I))*BMF*
+      BLR+SATLRU*(1.+XF)*2.*CPPC*WT(I)
      T1=1.
      IF(.NOT.(RIP(I).NE.1.0)) GO TO 230
        T1=RIP(I)/(1.-RIP(I))
230  CONTINUE
      ACFB=ACFB+(T1*MRO+MRF+SR+TR)*BLR
      ACFD=(FLOAT(LRU(I))+NHRT)*(NRTS(I)*DMH(I)*DMF*DLR+(2.*NRTS(I)+
+      COND(I))*CPPD(LO(NS))*WT(I))+NHNRT*(1.-COND(I))*DMH(I)*DMF*
+      DLR
      TCFB=12.*PIUP*TNB(NS)*FAIL(I,NS)*ACFB
      TCFD=12.*PIUP*TNB(NS)*FAIL(I,NS)*ACFD
      OFMCA(I)=OFMCA(I)+TCFB+TCFD
      OFMCB=OFMCB+TCFB
      OFMCD=OFMCD+TCFD
      IB=BTYPE(NS)
      IB1=3+BPLAT(NS)
      IF(.NOT.(IB.EQ.3)) GO TO 240
        IB=2
        IB1=3+BPLAT(NHB(NS))
240  CONTINUE
      BOFMC(IB)=BOFMC(IB)+TCFB
      BOFMC(IB1)=BOFMC(IB1)+TCFB

```

250 CONTINUE
260 CONTINUE
OFMC=OFMCB+OFMCD

C

RETURN
END

SUBROUTINE COST8

```
C
C***** 800827 111810711
C* COMPUTES SUPPORT EQUIPMENT COST ELEMENT-SEC *
C* SSS MOD SLR - 27 MAY 80 *
C*****
C
```

```
COMMON /BPLAT/ BPLAT(16)
INTEGER BPLAT
COMMON /BSECC/ BSECC(6)
COMMON /BSECP/ BSECP(6)
COMMON /BTYP/ BTYP(16)
INTEGER BTYP
COMMON /CSE/ CSE(250)
COMMON /DUM/ DUM
INTEGER DUM
COMMON /L/ L
COMMON /MSE/ MSE(250)
REAL MSE
COMMON /MXI/ MXI
COMMON /MXL/ MXL
COMMON /MXNS/ MXNS
COMMON /NS/ NS
COMMON /NSEB/ NSEB(250,16)
REAL NSEB
COMMON /NSED/ NSED(250)
REAL NSED
COMMON /PIUP/ PIUP
COMMON /SECBC/ SECBC
COMMON /SECBP/ SECBP
COMMON /SECC/ SECC
COMMON /SECD/ SECD
COMMON /SECDP/ SECDP
COMMON /SECIC/ SECIC
COMMON /SECII/ SECII
COMMON /SECIP/ SECIP
COMMON /SECP/ SECP
COMMON /SECR/ SECR
COMMON /SECR/ SECR
COMMON /SECRP/ SECRP
COMMON /SEDC/ SEDC
COMMON /SEDV/ SEDV(250)
COMMON /SEINO/ SEINO(250)
INTEGER SEINO
COMMON /SEPC/ SEPC
COMMON /SETYPE/ SETYPE(250)
INTEGER SETYPE
```

```

COMMON /TNB/ TNB(16)
COMMON /TSEC/ TSEC
COMMON /TUCTDC/ TUCTDC
INTEGER DUMM

```

C
C

```

DO 220 IXXX1=1,MXL
  L=SEINO(IXXX1)
  TEM01=0.
  DO 210 NS=1,MXNS
    TEM01=TEM01+(NSEB(L,NS)*TNB(NS))
210  CONTINUE
    SEPC=(TEM01+NSED(L))*CSE(L)*(1.+PIUP*MSE(L))+SEPC
    SEDC=SEDC+SEDV(L)
220  CONTINUE
    SEDC=SEDC+TUCTDC
    TSEC=SEPC+SEDC
    DO 350 IXXX1=1,MXL
      L=SEINO(IXXX1)
      DO 230 NS=1,MXNS
        TCSEL=NSEB(L,NS)*TNB(NS)*CSE(L)
        SECII=SECII+TCSEL
        SECR=SECR+TCSEL*PIUP*MSE(L)
230  CONTINUE
        SECR=SECR+NSED(L)*CSE(L)*PIUP*MSE(L)
        TEM02=0.
        DO 240 NS=1,MXNS
          TEM02=TEM02+NSEB(L,NS)*TNB(NS)
240  CONTINUE
        SUMM=TEM02
        IF(.NOT.(SETYPE(L).NE.3)) GO TO 290
        SECDC=SECDC+NSED(L)*CSE(L)*(1.+PIUP*MSE(L))
        SECIC=SECIC+(SUMM+NSED(L))*CSE(L)
        SECRC=SECRC+(SUMM+NSED(L))*CSE(L)*PIUP*MSE(L)
        DO 260 DUM=1,3
          TEM03=0.
          DO 250 NS=1,MXNS
            IF(.NOT.(BTYPE(NS).EQ.DUM)) GO TO 250
            TEM03=TEM03+NSEB(L,NS)*TNB(NS)
250  CONTINUE
            BSECC(DUM)=BSECC(DUM)+(TEM03)*CSE(L)*(1.+PIUP*MSE(L))
260  CONTINUE
            DO 280 DUM=4,6
              DUMM=DUM-3
              TEM04=0.
              DO 270 NS=1,MXNS
                IF(.NOT.(BPLAT(NS).EQ.DUMM)) GO TO 270

```

```

      TEM04=TEM04+NSEB(L,NS)*TNB(NS)
270   CONTINUE
      BSECC(DUM)=BSECC(DUM)+(TEM04)*CSE(L)*(1.+PIUP*MSE(L))
280   CONTINUE
290   CONTINUE
      IF(.NOT.(SETYPE(L).EQ.3)) GO TO 340
      SECIP=SECIP+(SUMM+NSED(L))*CSE(L)
      SECRP=SECRP+(SUMM+NSED(L))*CSE(L)*PIUP*MSE(L)
      SECDP=SECDP+(NSED(L)*CSE(L)*(1.+PIUP*MSE(L)))
      DO 310 DUM=1,3
        TEM05=0.
        DO 300 NS=1,MXNS
          IF(.NOT.(BTYP(NS).EQ.DUM)) GO TO 300
          TEM05=TEM05+NSEB(L,NS)*TNB(NS)
300     CONTINUE
          BSECP(DUM)=BSECP(DUM)+TEM05*CSE(L)*(1.+PIUP*MSE(L))
310     CONTINUE
          DO 330 DUM=4,5
            DUMM=DUM-3
            TEM06=0.
            DO 320 NS=1,MXNS
              IF(.NOT.(BPLAT(NS).EQ.DUMM)) GO TO 320
              TEM06=TEM06+NSEB(L,NS)*TNB(NS)
320     CONTINUE
              BSECP(DUM)=BSECP(DUM)+(TEM06)*CSE(L)*(1.+PIUP*MSE(L))
330     CONTINUE
340     CONTINUE
350     CONTINUE
          TEM07=0.
          DO 360 DUM=1,3
            TEM07=TEM07+BSECC(DUM)
360     CONTINUE
            SECBC=TEM07
            TEM08=0.
            DO 370 DUM=1,3
              TEM08=TEM08+BSECP(DUM)
370     CONTINUE
            SECBP=TEM08
            SECII=SECII+SEDC
            SECC=SECIC+SECR
            SECP=SECIP+SECRP
C
      RETURN
      END

```

SUBROUTINE COST9

C 800827 111900777
 C*****
 C* COMPUTES ITEM INVENTORY MANAGEMENT *
 C* COST ELEMENT-IIMC *
 C*****
 C

COMMON /BIIMC/ BIIMC(6)
 COMMON /BPLAT/ BPLAT(16)
 INTEGER BPLAT
 COMMON /BTYPE/ BTYPE(16)
 INTEGER BTYPE
 COMMON /COND/ COND(999)
 COMMON /FAIL/ FAIL(999,16)
 COMMON /I/ I
 COMMON /IIMC/ IIMC
 REAL IIMC
 COMMON /IIMCA/ IIMCA(999)
 REAL IIMCA
 COMMON /IIMCB/ IIMCB
 REAL IIMCB
 COMMON /IIMCD/ IIMCD
 REAL IIMCD
 COMMON /IIMCI/ IIMCI
 REAL IIMCI
 COMMON /IIMCR/ IIMCR
 REAL IIMCR
 COMMON /IMC/ IMC
 REAL IMC
 COMMON /INO/ INO(999)
 COMMON /LRU/ LRU(999)
 COMMON /MXI/ MXI
 COMMON /MXNS/ MXNS
 COMMON /NFB/ NFB(999,16)
 REAL NFB
 COMMON /NHI/ NHI(999)
 COMMON /NS/ NS
 COMMON /PA/ PA(999)
 COMMON /PIUP/ PIUP
 COMMON /RMC/ RMC
 COMMON /RTS/ RTS(999)
 COMMON /SA/ SA
 COMMON /SAT/ SAT(16)
 COMMON /TNB/ TNB(16)
 REAL IUT
 REAL NHCD

C

C

```

DO 240 IXXX1=1,MXI
  I=INO(IXXX1)
  TEM01=0.
  DO 210 NS=1,MXNS
    TEM01=TEM01+FAIL(I,NS)
210  CONTINUE
    IUT=TEM01
    IUT=U(IUT)
    NHCD=0.
    IF(.NOT.(LRU(I).EQ.0)) GO TO 220
    NHCD=COND(NHI(I))
220  CONTINUE
    CPA=PA(I)*U(1.-COND(I))
    DO 230 NS=1,MXNS
      BIS=AMIN1(F(NFB(I,NS)),1.)*TNB(NS)
      BCIS=(1.-SAT(NS))*U(RTS(I)*NFB(I,NS))*TNB(NS)
      CB=PIUP*(BIS+BCIS*CPA)*SA
      IIMCA(I)=IIMCA(I)+CB
      IIMCB=IIMCB+CB
      BIIMC(BTYPE(NS))=BIIMC(BTYPE(NS))+CB
      IB=3+BPLAT(NS)
      BIIMC(IB)=BIIMC(IB)+CB
      IIMCR=IIMCR+CB
230  CONTINUE
    CD=IUT*(1.+CPA)*U(1.-NHCD)*(IMC+PIUP*RMC)
    IIMCA(I)=IIMCA(I)+CD
    IIMCD=IIMCD+CD
    IIMCI=IIMCI+IUT*(1.+CPA)*U(1.-NHCD)*IMC
    IIMCR=IIMCR+IUT*(1.+CPA)*U(1.-NHCD)*PIUP*RMC
240  CONTINUE
    IIMC=IIMCB+IIMCD

```

C

```

RETURN
END

```

SUBROUTINE COST10

C 800827 111925192
 C*****
 C* CALCULATES TECH ORDER COST - STDC *
 C* SSS MOD SLR - 28 MAY 80 *
 C*****
 C

```
COMMON /A/ A(999,4,30)
INTEGER A
COMMON /ACPP/ ACPP
COMMON /BDATA/ BDATA
INTEGER BDATA
COMMON /BTDC/ BTDC(16)
COMMON /BTYP/ BTYP(16)
INTEGER BTYP
COMMON /DATAB/ DATAB(999)
INTEGER DATAB
COMMON /DATAD/ DATAD(999)
INTEGER DATAD
COMMON /DATAS/ DATAS(250)
INTEGER DATAS
COMMON /DDATA/ DDATA
INTEGER DDATA
COMMON /EBCBI/ EBCBI(999,16)
COMMON /ERHAB/ ERHAB(250,16)
COMMON /ERHAD/ ERHAD(250)
COMMON /ERHD/ ERHD(999)
COMMON /ERTBI/ ERTBI(999,16)
COMMON /I/ I
COMMON /INO/ INO(999)
COMMON /IRMIN/ IRMIN(999,4)
COMMON /IRMT/ IRMT
COMMON /L/ L
COMMON /LT/ LT
COMMON /MXI/ MXI
COMMON /MXIRMT/ MXIRMT
COMMON /MXL/ MXL
COMMON /MXLT/ MXLT
COMMON /MXNS/ MXNS
COMMON /NJA/ NJA(999,4)
COMMON /NRM/ NRM(999)
COMMON /NS/ NS
COMMON /NSEB/ NSEB(250,16)
REAL NSEB
COMMON /NSED/ NSED(250)
REAL NSED
COMMON /OFMC/ OFMC
```

```

COMMON /OFMCA/ OFMCA(999)
COMMON /ONMC/ ONMC
COMMON /ONMCA/ ONMCA(999)
COMMON /PIUP/ PIUP
COMMON /PPSE/ PPSE(999,250)
COMMON /QSA/ QSA(999,4,30)
COMMON /RCPP/ RCPP
COMMON /RMI/ RMI(999,16)
INTEGER RMI
COMMON /SEINO/ SEINO(250)
INTEGER SEINO
COMMON /SETDC/ SETDC(250)
COMMON /STDC/ STDC
COMMON /STDCI/ STDCI
COMMON /STDCR/ STDCR
COMMON /TDC/ TDC(999)
COMMON /TNB/ TNB(16)
COMMON /TNSE/ TNSE(250)
COMMON /UCPP/ UCPP
INTEGER SECD1
DIMENSION SECODE(250)
DIMENSION SUM(250)
DIMENSION TERHAB(250)
DIMENSION TQSA(250)

```

C
C

```

SUM3=0.
SUM2=0.
TERM2=0.
DO 250 IXXX1=1,MXL
  L=SEINO(IXXX1)
  TEM01=0.
  DO 210 NS=1,MXNS
    IF(.NOT.(BTYPE(NS).EQ.1)) GO TO 210
    TEM01=TEM01+TNB(NS)*NSLB(L,NS)
210  CONTINUE
    T1=TEM01
    TEM02=0.
    DO 220 NS=1,MXNS
      IF(.NOT.(BTYPE(NS).EQ.2)) GO TO 220
      TEM02=TEM02+TNB(NS)*NSEB(L,NS)
220  CONTINUE
      T2=TEM02
      T7=T1+T2+NSED(L)
      TEM03=0.
      DO 230 NS=1,MXNS
        TEM03=TEM03+NSEB(L,NS)

```

```

230  CONTINUE
      TNSE(L)=TEM03+NSED(L)
      SUM3=SUM3+U(TNSE(L))*FLOAT(DATAS(L))
      TEM04=0.
      DO 240 NS=1,MXNS
        TEM04=TEM04+TNB(NS)*U(NSEB(L,NS))
240  CONTINUE
      SETDC(L)=U(T7)*FLOAT(DATAS(L))*(ACPP+RCPP+(PIUP-1.)*UCPP)+RCPP*
+    TEM04*FLOAT(DATAS(L))
250  CONTINUE
      DO 260 IXXX1=1,MXI
        I=INO(IXXX1)
        SUM2=SUM2+(FLOAT(DATAD(I))+FLOAT(DATAB(I)))
260  CONTINUE
      TERM1=(FLOAT(DDATA+BDATA)+SUM2+SUM3)*(ACPP+RCPP+(PIUP-1.)*UCPP)
      DO 290 NS=1,MXNS
        TEM05=0.
        DO 270 IXXX2=1,MXI
          I=INO(IXXX2)
          TEM05=TEM05+FLOAT(DATAB(I))
270  CONTINUE
        TEM06=0.
        DO 280 IXXX2=1,MXL
          L=SEINO(IXXX2)
          TEM06=TEM06+U(NSEB(L,NS))*FLOAT(DATAS(L))
280  CONTINUE
        BTDC(NS)=TNB(NS)*(FLOAT(BDATA)+TEM05+TEM06)*RCPP
        TERM2=TERM2+BTDC(NS)
290  CONTINUE
        STDC=TERM1+TERM2
        STDC1=STDC-(FLOAT(DDATA+BDATA)+SUM2+SUM3)*(PIUP-1.)*UCPP
        STDGR=STDC-STDCI
C
C.....CALCULATE TDC(I)
      DO 460 IXXX1=1,MXI
        I=INO(IXXX1)
        PPTM=(ONMCA(I)+OFMCA(I))/(ONMC+OFMC)
        DO 310 IXXX2=1,MXL
          L=SEINO(IXXX2)
          TEM07=0.
          DO 300 NS=1,MXNS
            TEM07=TEM07+TNB(NS)*ERHAB(L,NS)
300  CONTINUE
          TERHAB(L)=TEM07
          SECODE(I)=0.
          SUM(L)=0.
          TQSA(L)=0.

```

```

        PPSE(I,L)=0.
310    CONTINUE
        NXXX1=NRM(I)
        IF(.NOT.(NXXX1.GT.0)) GO TO 400
        DO 380 IRMT=1,MXIRMT
            IF(.NOT.(IRMT.LE.NXXX1)) GO TO 390
            NXXX2=NJA(I,IRMT)
            IF(.NOT.(NXXX2.GT.0)) GO TO 370
            DO 350 LT=1,MXLT
                IF(.NOT.(LT.LE.NXXX2)) GO TO 360
                L=A(I,IRMT,LT)
                IF(.NOT.(IRMIN(I,IRMT).EQ.1)) GO TO 320
                SECD1=3-INT(QSA(I,IRMT,LT)/100.)
                SECODE(L)=FLOAT(SECD1)
                TQSA(L)=QSA(I,IRMT,LT)-AINT(QSA(I,IRMT,LT)/100.)*100.
320        CONTINUE
            DO 340 NS=1,MXNS
                IF(.NOT.(IRMIN(I,IRMT).EQ.RMI(I,NS))) GO TO 330
                TQSA1=QSA(I,IRMT,LT)-AINT(QSA(I,IRMT,LT)/100.)*100.
                SECD1=(INT(QSA(I,IRMT,LT)/100.)-1)
                ST1=U(FLOAT(SECD1))*TNB(NS)*EBCBI(I,NS)
                SECD1=(3-INT(QSA(I,IRMT,LT)/100.))*INT(QSA(I,IRMT,
+                LT)/100.)
                ST2=U(FLOAT(SECD1))*TNB(NS)*ERTBI(I,NS)
                SUM(L)=SUM(L)+U(TQSA1)*(ST1+ST2)
330        CONTINUE
340        CONTINUE
350        CONTINUE
360        CONTINUE
370        CONTINUE
380        CONTINUE
390        CONTINUE
400    CONTINUE
        DO 420 IXXX2=1,MXL
            L=SEINO(IXXX2)
            T8=ERHAD(L)+TERHAB(L)
            IF(.NOT.(T8.GE.0.000001)) GO TO 410
            PPSE(I,L)=(U(TQSA(L))*U(SECODE(L))*ERHD(I)+SUM(L))/T8
410    CONTINUE
420    CONTINUE
        TEM08=0.
        DO 430 IXXX2=1,MXL
            L=SEINO(IXXX2)
            TEM08=TEM08+PPSE(I,L)*FLOAT(DATAS(L))
430    CONTINUE
        TEM10=0.
        DO 450 NS=1,MXNS

```

```

      TEM09=0.
      DO 440 IXXX3=1,MXL
        L=SEINO(IXXX3)
        TEM09=TEM09+PPSE(I,L)*U(NSEB(L,NS))*FLOAT(DATAS(L))
440    CONTINUE
        TEM10=TEM10+TNB(NS)*(PPTM*FLOAT(BDATA)+FLOAT(DATAB(I))+TEM09)
450    CONTINUE
        TDC(I)=(PPTM*FLOAT(DDATA+BDATA)+FLOAT(DATAD(I))+FLOAT(DATAB(I))+
      +      TEM08)*(ACPP+RCPP+(PIUP-1.)*UCPP)+RCPP*TEM10
460    CONTINUE
C
      RETURN
      END

```

SUBROUTINE COST11

C 800827 112028726
 C*****
 C* CALCULATES MAINTENANCE TRAINING COST - MTRC *
 C*****
 C

COMMON /BMTRC/ BMTRC
 COMMON /COND/ COND(999)
 COMMON /CPD1/ CPD1
 COMMON /CPD2/ CPD2
 COMMON /DMTRC/ DMTRC
 COMMON /ERTBI/ ERTBI(999,16)
 COMMON /HPD1/ HPD1
 INTEGER HPD1
 COMMON /HPD2/ HPD2
 INTEGER HPD2
 COMMON /I/ I
 COMMON /IMTRC/ IMTRC
 REAL IMTRC
 COMMON /INO/ INO(999)
 COMMON /MTRC/ MTRC
 REAL MTRC
 COMMON /MTRCI/ MTRCI(999)
 REAL MTRCI
 COMMON /MXI/ MXI
 COMMON /MXNS/ MXNS
 COMMON /NS/ NS
 COMMON /PAL1/ PAL1
 COMMON /PAL2B/ PAL2B
 COMMON /PAL2D/ PAL2D
 COMMON /PIUP/ PIUP
 COMMON /QTP1/ QTP1
 INTEGER QTP1
 COMMON /QTP2B/ QTP2B
 INTEGER QTP2B
 COMMON /QTP2D/ QTP2D
 INTEGER QTP2D
 COMMON /RMTRC/ RMTRC
 COMMON /SPC1/ SPC1
 INTEGER SPC1
 COMMON /SPC2/ SPC2
 INTEGER SPC2
 COMMON /TEFM/ TEFM
 COMMON /TIME1/ TIME1(999)
 INTEGER TIME1
 COMMON /TORB/ TORB
 COMMON /TORD/ TORD

```

COMMON /TRAVB/ TRAVB
COMMON /TRAV1D/ TRAV1D
COMMON /TYP2TF/ TYP2TF
COMMON /T2BA/ T2BA
COMMON /T2DA/ T2DA
REAL ITPF1
REAL ITPF2
REAL ITPF3
REAL ITPF4
REAL ITPF5

```

C
C

```

T1=0.
T2DA=0.
T2BA=0.
DO 220 IXXX1=1,MXI
  I=INO(IXXX1)
  T1=T1+TIME1(I)
  T2DA=TIME1(I)*U(1-COND(I))+T2DA
  TEM01=0.
  DO 210 NS=1,MXNS
    TEM01=TEM01+ERTBI(I,NS)
210  CONTINUE
    TEMP=TEM01
    T2BA=T2BA+TIME1(I)*U(TEMP)
220  CONTINUE
    T2DA=T2DA*TYP2TF
    T2BA=T2BA*TYP2TF
    TEMP=(T2DA*QTYP2D*(1.+(PIUP-1.)*TORD)+T2BA*QTYP2B*(1.+(PIUP-1.)*
+   TORB))/(HPD2*SPC2)
    MTRC=AIN(T1/HPD1+.9)*AIN(QTYP1/SPC1+.9)*CPD1+QTYP1*(AIN(T1/
+   HPD1+.9)*PAL1+TRAV1D*U(T1))+AIN(TEMP+.9)*CPD2+(1.+(PIUP-1.)*
+   TORD)*QTYP2D*(AIN(T2DA/HPD2+.9)*PAL2D+TRAV1D*U(T2DA))+(1.+
+   (PIUP-1.)*TORB)*QTYP2B*(AIN(T2BA/HPD2+.9)*PAL2B+TRAVB*U(T2BA))+
+   TEFM
    IMTRC=AIN(T1/HPD1+.9)*AIN(QTYP1/SPC1+.9)*CPD1+QTYP1*(AIN(T1/
+   HPD1+.9)*PAL1+TRAV1D*U(T1))+AIN((T2DA*QTYP2D+T2BA*QTYP2B)/
+   (HPD2*SPC2)+.9)*CPD2+QTYP2D*(AIN(T2DA/HPD2+.9)*PAL2D+TRAV1D*
+   U(T2DA))+QTYP2B*(AIN(T2BA/HPD2+.9)*PAL2B+TRAVB*U(T2BA))+TEFM
    RMTRC=MTRC-IMTRC
    TEMP=(T2DA*QTYP2D*(1.+(PIUP-1.)*TORD)+T2BA*QTYP2B*(1.+(PIUP-1.)*
+   TORB))/(HPD2*SPC2)
    BMTRC=0.
    T2=QTYP2D*T2DA+QTYP2B*T2BA
    IF(.NOT.(T2.GE.0.000001)) GO TO 230
    BMTRC=(QTYP2B*T2BA)/T2*AIN(TEMP+.9)*CPD2+(1.+(PIUP-1.)*TORB)*
+   QTYP2B*(AIN(T2BA/HPD2+.9)*PAL2B+TRAVB*U(T2BA))

```

```

230 CONTINUE
  DMTRC=MTRC-BMTRC
  DO 290 IXXX1=1,MXI
    I=INO(IXXX1)
    ITPF2=0.
    ITPF3=0.
    ITPF4=0.
    ITPF5=0.
    ITPF1=TIME1(I)/T1
    TEM02=0.
    DO 240 NS=1,MXNS
      TEM02=TEM02+ERTBI(I,NS)
240 CONTINUE
    TEMP=TEM02
    T2=QTYP2D*T2DA+QTYP2B*T2BA
    IF(.NOT.(T2.GE.0.000001)) GO TO 250
    ITPF2=TYP2TF*TIME1(I)*(QTYP2D*U(1.-COND(I))+QTYP2B*U(TEMP))*
+    1./T2
250 CONTINUE
    IF(.NOT.(T2DA.GE.0.000001)) GO TO 260
    ITPF3=TYP2TF*TIME1(I)*U(1.-COND(I))/T2DA
260 CONTINUE
    IF(.NOT.(T2BA.GE.0.000001)) GO TO 270
    ITPF4=TYP2TF*TIME1(I)*U(TEMP)/T2BA
270 CONTINUE
    T3=QTYP1*T1+QTYP2D*T2DA+QTYP2B*T2BA
    IF(.NOT.(T3.GE.0.000001)) GO TO 280
    ITPF5=TIME1(I)*(QTYP1+TYP2TF*(QTYP2D*U(1.-COND(I))+QTYP2B*
+    U(TEMP)))*1./T3
280 CONTINUE
    TEMP=(T2DA*QTYP2D*(1.+(PIUP-1.)*TORD)+T2BA*QTYP2B*(1.+(PIUP-1.)*
+    TORB))*1./(HPD2*SPC2)
    TERM1=ITPF1*(AINT(T1/HPD1+.9)*AINT(QTYP1/SPC1+.9)*CPD1+QTYP1*
+    (AINT(T1/HPD1+.9)*PAL1+TRAV1D*U(T1)))
    TERM2=ITPF2*AINT(TEMP+.9)*CPD2
    TERM3=ITPF3*(1.+(PIUP-1.)*TORD)*QTYP2D*(AINT(T2DA/HPD2+.9)*
+    PAL2D+TRAV1D*U(T2DA))
    TERM4=ITPF4*(1.+(PIUP-1.)*TORB)*QTYP2B*(AINT(T2BA/HPD2+.9)*
+    PAL2B+TRAVB*U(T2BA))
    TERM5=ITPF5*TEFM
    MTRCI(I)=TERM1+TERM2+TERM3+TERM4+TERM5
290 CONTINUE
C
  RETURN
  END

```

FUNCTION CHLCC(CC,CR,CN,I)

C

800827 112150107

C*****

C* SSS MOD SLR - 20 MAY 80 *

C* COMPUTES CHANGE IN LCC FOR REPAIR LEVEL SENSITIVITY *

C*****

C

COMMON /A/ A(999,4,30)
INTEGER A
COMMON /B/ B
INTEGER B
COMMON /BAA/ BAA
COMMON /BCMh/ BCMh(999)
COMMON /BLR/ BLR
COMMON /BMF/ BMF
COMMON /BMH/ BMH(999)
COMMON /BRCT/ BRCT
COMMON /BTYPE/ BTYPE(16)
INTEGER BTYPE
COMMON /CIMF/ CIMF(16)
COMMON /COND/ COND(999)
COMMON /CPPD/ CPPD(3)
COMMON /CRCT/ CRCT
COMMON /CSE/ CSE(250)
COMMON /DAA/ DAA
COMMON /DAD/ DAD
COMMON /DLR/ DLR
COMMON /DMF/ DMF
COMMON /DMH/ DMH(999)
COMMON /DRCT/ DRCT(3)
COMMON /ERHAB/ ERHAB(250,16)
COMMON /ERHAD/ ERHAD(250)
COMMON /FAIL/ FAIL(999,16)
COMMON /FPR/ FPR(999)
COMMON /HDWRIT/ HDWRIT(999,10)
COMMON /IRMIN/ IRMIN(999,4)
COMMON /IRMT/ IRMT
COMMON /ISCA/ ISCA(999)
REAL ISCA
COMMON /ISET/ ISET(250,16)
REAL ISET
COMMON /ISETD/ ISETD(250)
REAL ISETD
COMMON /L/ L
COMMON /LO/ LO(16)
COMMON /LRU/ LRU(999)
COMMON /LT/ LT

```

COMMON /MSE/ MSE(250)
REAL MSE
COMMON /MXIRMT/ MXIRMT
COMMON /MXL/ MXL
COMMON /MXLT/ MXLT
COMMON /MXNP/ MXNP
COMMON /MXNS/ MXNS
COMMON /NBC/ NBC(16)
REAL NBC
COMMON /NFB/ NFB(999,16)
REAL NFB
COMMON /NFD/ NFD(999)
REAL NFD
COMMON /NHB/ NHB(16)
COMMON /NHI/ NHI(999)
COMMON /NITEM/ NITEM(999,10)
REAL NITEM
COMMON /NJA/ NJA(999,4)
COMMON /NP/ NP
COMMON /NRM/ NRM(999)
COMMON /NRTS/ NRTS(999)
REAL NRTS
COMMON /NRUC/ NRUC
REAL NRUC
COMMON /NS/ NS
COMMON /NSEB/ NSEB(250,16)
REAL NSEB
COMMON /NSED/ NSED(250)
REAL NSED
COMMON /OST/ OST(3)
COMMON /PA/ PA(999)
COMMON /PIUP/ PIUP
COMMON /QSA/ QSA(999,4,30)
COMMON /RM/ RM(999)
COMMON /RMI/ RMI(999,16)
INTEGER RMI
COMMON /RSCA/ RSCA(999)
COMMON /RTS/ RTS(999)
COMMON /SA/ SA
COMMON /SAT/ SAT(16)
COMMON /SEINO/ SEINO(250)
INTEGER SEINO
COMMON /TNB/ TNB(16)
COMMON /TOTT/ TOTT(10)
COMMON /UP/ UP(999)
COMMON /USE/ USE(250,16)
COMMON /USED/ USED(250)

```

```

COMMON /WT/ WT(999)
COMMON /XFPR/ XFPR
COMMON /XITEMQ/ XITEMQ(999)
COMMON /XUC/ XUC
COMMON /YRSQ/ YRSQ(999)
DIMENSION CHSE(250,16)
DIMENSION CHSED(250)
DIMENSION CNFB(16)
DIMENSION CPP(16)
DIMENSION CRH(16)
REAL NHNRT
REAL NHRT

```

C
C
C

```

C..... CALCULATES CRH(NS),CRHD
      NHRT=0.
      NHNRT=0.
      IF(.NOT.(NHI(I).NE.0)) GO TO 210
      NHRT=RTS(NHI(I))
      NHNRT=NRTS(NHI(I))
210  CONTINUE
      DO 240 NS=1,MXNS
      CRH(NS)=0.
      IF(.NOT.(BTYPE(NS).NE.3)) GO TO 230
      TEM01=0.
      DO 220 B=1,MXNS
      IF(.NOT.(NHB(B).EQ.NS)) GO TO 220
      TEM01=TEM01+FAIL(I,B)*NBC(B)
220  CONTINUE
      CRH(NS)=FAIL(I,NS)*(FLOAT(LRU(I))+NHRT)*CR*BMH(I)*BMF+CIMF(NS)
      +      *TEM01*(FLOAT(LRU(I))+NHRT)*((CR+CN+U(XFPR*FPR(I))*CC)*
      +      BCMH(I)+CR*BMH(I))*BMF
230  CONTINUE
240  CONTINUE
      TEM02=0.
      DO 250 NS=1,MXNS
      TEM02=TEM02+FAIL(I,NS)*TNB(NS)
250  CONTINUE
      CRHD=TEM02*((FLOAT(LRU(I))+NHRT)*CN-NHNRT*CC)*DMH(I)*DMF

C
C..... CALCULATES CHSE(L,NS) AND CHSED(L)
      DO 270 IXXX1=1,MXL
      L=SEINO(IXXX1)
      CHSED(L)=0.
      DO 260 NS=1,MXNS
      CHSE(L,NS)=0.

```

```

260 CONTINUE
270 CONTINUE
DO 420 NS=1,MXNS
  NXXX1=NRM(I)
  IF(.NOT.(NXXX1.GT.0)) GO TO 410
  DO 390 IRMT=1,MXIRMT
    IF(.NOT.(IRMT.LE.NXXX1)) GO TO 400
    NXXX2=NJA(I,IRMT)
    IF(.NOT.(NXXX2.GT.0)) GO TO 380
    DO 360 LT=1,MXLT
      IF(.NOT.(LT.LE.NXXX2)) GO TO 370
      L=A(I,IRMT,LT)
      TQSA=0.
      IF(.NOT.(IRMIN(I,IRMT).EQ.RMI(I,NS))) GO TO 280
      TQSA=QSA(I,IRMT,LT)-AINT(QSA(I,IRMT,LT)/100.)*100.
280 CONTINUE
      IF(.NOT.(CRH(NS).GT..000001)) GO TO 290
      CHSE(L,NS)=USE(L,NS)*(CRH(NS)/BAA)*AMAX1(ISET(L,NS),
+      TQSA)
290 CONTINUE
      IF(.NOT.(CRH(NS).LT..000001)) GO TO 310
      IF(.NOT.(ERHAB(L,NS).GT..000001.OR.ERHAB(L,NS).LT.
+      -.000001)) GO TO 300
      CHSE(L,NS)=CRH(NS)/ERHAB(L,NS)*NSEB(L,NS)
300 CONTINUE
310 CONTINUE
      TQSA=0.
      IF(.NOT.(IRMIN(I,IRMT).EQ.1)) GO TO 320
      TQSA=QSA(I,IRMT,LT)-AINT(QSA(I,IRMT,LT)/100.)*100.
320 CONTINUE
      IF(.NOT.(CRHD.GT..000001)) GO TO 330
      CHSED(L)=USED(L)*CRHD/DAA*AMAX1(ISETD(L),TQSA)
330 CONTINUE
      IF(.NOT.(CRHD.LT..000001)) GO TO 350
      IF(.NOT.(ERHAD(L).GT..000001.OR.ERHAD(L).LT.
+      -.000001)) GO TO 340
      CHSED(L)=CRHD/ERHAD(L)*NSED(L)
340 CONTINUE
350 CONTINUE
360 CONTINUE
370 CONTINUE
380 CONTINUE
390 CONTINUE
400 CONTINUE
410 CONTINUE
420 CONTINUE
C

```

C..... CALCULATES CNFB(NS) AND CNFD

DO 440 NS=1,MXNS

TEM03=0.

DO 430 B=1,MXNS

IF(.NOT.(NHB(B).EQ.NS)) GO TO 430

TEM03=TEM03+(FAIL(I,B)*NBC(B))

430 CONTINUE

CNFB(NS)=(1.-SAT(NS))*FAIL(I,NS)*(FLOAT(LRU(I))+NHRT)*(CR*BRCT+

+ (CN+CC)*OST(LO(NS)))+CIMF(NS)*TEM03*(FLOAT(LRU(I))+NHRT)*(CR*

+ CRCT+(CN+CC)*(OST(LO(NS))+U(XFPR*FPR(I))*CRCT'))

440 CONTINUE

TEM04=0.

DO 450 NS=1,MXNS

TEM04=TEM04+FAIL(I,NS)*TNB(NS)*((FLOAT(LRU(I))+NHRT)*CN*

+ DRCT(LO(NS))-NHRT*CC*DAD)

450 CONTINUE

CNFD=TEM04

C

C..... CALCULATES CPP(NS)

DO 490 NS=1,MXNS

RTSNFB=RTS(I)*NFB(I,NS)

IF(.NOT.(RTSNFB.LT..000001)) GO TO 460

CPP(NS)=U(CR*(CNFB(NS)+NFB(I,NS)))

460 CONTINUE

IF(.NOT.(RTSNFB.GT..000001)) GO TO 480

CPP(NS)=0.

IF(.NOT.(CR.LT..000001)) GO TO 470

CPP(NS)=CR/RTS(I)

470 CONTINUE

480 CONTINUE

490 CONTINUE

TEM05=0.

DO 500 NS=1,MXNS

TEM05=TEM05+TNB(NS)*(F(CNFB(NS)+NFB(I,NS))-F(NFB(I,NS)))

500 CONTINUE

XITEMQ(I)=TEM05+(F(CNFD+NFD(I))-F(NFD(I)))

C

C..... CALCULATES CHLCC

CD=0.

IF(.NOT.(LRU(I).EQ.0)) GO TO 510

CD=COND(NHI(I))

510 CONTINUE

TEM06=0.

DO 520 NS=1,MXNS

TEM06=TEM06+TNB(NS)*(F(CNFB(NS)+NFB(I,NS)))

520 CONTINUE

CHLCCA=(TEM06+(F(CNFD+NFD(I))))*UP(I)*XLEARN(I)*XUC-ISCA(I)+(NRUC*

```

+ XLEARN(I)+(PIUP-NRUC))*YRSQ(I)*((COND(I)+CC)+(1.-(COND(I)+CC))*
+ RM(I))*UP(I)*XUC-RSCA(I)
TEM07=0.
DO 530 NS=1,MXNS
TEM07=TEM07+FAIL(I,NS)*TNB(NS)*((FLOAT(LRU(I))+NHRT)*(CR*BMH(I)*
+ BMF*BLR+CN*(DMH(I)*DMF*DLR+2.*CPPD(LO(NS))*WT(I))+CC*
+ CPPD(LO(NS))*WT(I))-NHNRT*CC*DMH(I)*DMF*DLR)
530 CONTINUE
CHLCCB=12.*PIUP*TEM07
CHLCCD=0.
NXXX1=NRM(I)
IF(.NOT.(NXXX1.GT.0)) GO TO 610
DO 600 IRMT=1,MXIRMT
NXXX2=NJA(I,IRMT)
IF(.NOT.(NXXX2.GT.0)) GO TO 590
DO 570 LT=1,MXLT
IF(.NOT.(LT.LE.NXXX2)) GO TO 580
L=A(I,IRMT,LT)
CHLCCC=0.
DO 550 NS=1,MXNS
IF(.NOT.(IRMT.EQ.RMI(I,NS))) GO TO 540
CHLCCC=CHLCCC+U(QSA(I,IRMT,LT)-AINT(QSA(I,IRMT,LT)/
+ 100.)*100.)*TNB(NS)*CHSE(L,NS)
540 CONTINUE
550 CONTINUE
CHLCCF=0.
IF(.NOT.(IRMT.EQ.1)) GO TO 560
CHLCCF=CHLCCF+CHSE(L)*U(QSA(I,IRMT,LT)-AINT(QSA(I,IRMT,
+ LT)/100.)*100.)
560 CONTINUE
CHLCCD=CHLCCD+((CHLCCC+CHLCCF)*CSE(L)*(1.+PIUP*MSE(L)))
570 CONTINUE
580 CONTINUE
590 CONTINUE
600 CONTINUE
610 CONTINUE
TEM08=0.
DO 620 NS=1,MXNS
TEM08=TEM08+(1.-SAT(NS))*TNB(NS)*PIUP*SA*(AMIN1(F(CNFB(NS)+
+ NFB(I,NS)),1.)-AMIN1(F(NFB(I,NS)),1.))+CPP(NS)*PA(I))
620 CONTINUE
TEM09=0.
DO 630 NP=1,MXNP
TEM09=TEM09+TOTT(NP)*(NITEM(I,NP)*UP(I)*XLEARN(I)*XUC-HDWRT(I,
+ NP))
630 CONTINUE
CHLCCE=TEM08+TEM09

```

CHLCC=CHLCCA+CHLCCB+CHLCCD+CHLCCE

C

RETURN
END

SUBROUTINE DPIUP

```

C
C***** 800827 112313688 *****
C* COMPUTES PROGRAM OPERATIONAL *
C* SSS MOD SLR-4 JUNE 1980 *
C* LIFETIME - PIUP FACTOR *
C*****
C
COMMON /COND/ COND(999)
COMMON /CPIUP/ CPIUP
COMMON /FINC/ FINC
COMMON /I/ I
COMMON /IIMCR/ IIMCR
REAL IIMCR
COMMON /INO/ INO(999)
COMMON /LDERV/ LDERV
COMMON /MXI/ MXI
COMMON /OC/ OC
COMMON /OFMC/ OFMC
COMMON /ONMC/ ONMC
COMMON /PIUP/ PIUP
COMMON /RM/ RM(999)
COMMON /RMTRC/ RMTRC
COMMON /SECR/ SECR
COMMON /STDCR/ STDCR
COMMON /TDPIUP/ TDPIUP
COMMON /UP/ UP(999)
COMMON /XUC/ XUC
COMMON /YRSQ/ YRSQ(999)

C
C
CPIUP=AIN(T(FINC*PIUP+.5)
TEM01=0.
DO 210 IXXX1=1,MXI
I=INO(IXXX1)
TEM01=TEM01+YRSQ(I)*(COND(I)+(1.-COND(I))*RM(I))*UP(I)
210 CONTINUE
T1=PIUP*TEM01*XUC
TDPIUP=CPIUP/PIUP*(OC+ONMC+OFMC+SECR+IIMCR+T1+PIUP/(PIUP-1.))*
+ (STDCR+RMTRC))/1000000.

C
RETURN
END

```

SUBROUTINE DLMF

C

800827 112329512

```
C*****
C* BASELINE CHANGE *
C* COMPUTES MAINTENANCE REPAIR TIME *
C* FACTORS - BMF AND DMF *
C*****
```

C

```
COMMON /BAA/ BAA
COMMON /BLR/ BLR
COMMON /CSE/ CSE(250)
COMMON /DAA/ DAA
COMMON /DLR/ DLR
COMMON /ERHAB/ ERHAB(250,16)
COMMON /ERHAD/ ERHAD(250)
COMMON /ERHBI/ ERHBI(999,16)
COMMON /ERHD/ ERHD(999)
COMMON /FINC/
COMMON 'I/ I
COMMON /INO/ INO(999)
COMMON /ISET/ ISET(250,16)
REAL ISET
COMMON /ISETD/ ISETD(250)
REAL ISETD
COMMON /L/ L
COMMON /LDERV/ LDERV
COMMON /MSE/ MSE(250)
REAL MSE
COMMON /MXI/ MXI
COMMON /MXL/ MXL
COMMON /MXNS/ MXNS
COMMON /NS/ NS
COMMON /ONMC/ ONMC
COMMON /PIUP/ PIUP
COMMON /SEINO/ SEINO(250)
INTEGER SEINO
COMMON /TDMF/ TDMF
COMMON /TNB/ TNB(16)
COMMON /USE/ USE(250,16)
COMMON /USED/ USED(250)
```

C

C

```
TDMFA=FINC*ONMC
DO 220 IXXX1=1,MXI
  I=INO(IXXX1)
  TEM01=0.
DO 210 NS=1,MXNS
```

```

      TEM01=TEM01+ERHBI(I,NS)*TNB(NS)
210  CONTINUE
      TDMFA=TDMFA+12.*PIUP*FINC*(TEM01*BLR+ERHD(I)*DLR)
220  CONTINUE
      TEM03=0.
      DO 240 IXXX1=1,MXL
        L=SEINO(IXXX1)
        TEM02=0.
        DO 230 NS=1,MXNS
          TEM02=TEM02+(USE(L,NS)*ERHAB(L,NS)*ISET(L,NS)*TNB(NS)/BAA)
230  CONTINUE
          TEM03=TEM03+(TEM02+USED(L)*ERHAD(L)*ISETD(L)/DAA)*CSE(L)*(1.+
+      PIUP*MSE(L))
240  CONTINUE
      TDMF=TDMFA+FINC*TEM03
      TDMF=TDMF/1000000.
C
      RETURN
      END

```

SUBROUTINE DRM

```

C                                                    800827 112337774
C*****
C* SSS MOD SLR - 5 JUN 80                                *
C* COMPUTES ITEM SPECIFIC REPAIR MATERIALS COST          *
C* FACTOR - RM(I)                                         *
C*****
C
COMMON /PRNTXX/ PRNTXX
INTEGER PRNTXX
COMMON /COND/ COND(999)
COMMON /FINC/ FINC
COMMON /I/ I
COMMON /IDRM/ IDRM(999)
COMMON /INO/ INO(999)
COMMON /LC/ LC(999)
REAL LC
COMMON /LDERV/ LDERV
COMMON /LDRM/ LDRM
COMMON /MXI/ MXI
COMMON /NRUC/ NRUC
REAL NRUC
COMMON /PIUP/ PIUP
COMMON /RM/ RM(999)
COMMON /TDRM/ TDRM(999)
COMMON /UP/ UP(999)
COMMON /XUC/ XUC
COMMON /YRSQ/ YRSQ(999)

C
C
DO 210 IXXX=1,MXI
    I=INO(IXXX)
    TDRM(I)=(NRUC*LC(I)+(PIUP-NRUC))*YRSQ(I)*((1.-COND(I))*FINC*
+    RM(I))*UP(I)*XUC/1000000.
210 CONTINUE

C
DO 999 IXXX=1,MXI
    IDRM (IXXX)=INO (IXXX)
999 CONTINUE
LD=LDRM
IF (PRNTXX.NE.0) LD=MAX0(LD,LDERV )
CALL TDSORT(TDRM ,IDRM ,LD,MXI )

C
RETURN
END

```

SUBROUTINE DXRM

```

C
C*****800827 112344184*****
C* COMPUTES GLOBAL REPAIR MATERIALS COST *
C* FACTOR - XRM *
C*****
C
COMMON /I/ I
COMMON /INO/ INO(999)
COMMON /LDERV/ LDERV
COMMON /MXI/ MXI
COMMON /TDRM/ TDRM(999)
COMMON /TDXRM/ TDXRM

C
C
TEM01=0.
DO 210 IXXX1=1,MXI
    I=INO(IXXX1)
    TEM01=TEM01+TDRM(I)
210 CONTINUE
TDXRM=TEM01

C
RETURN
END

```

SUBROUTINE DXUC

```
C
C*****800827 112344681*****
C* COMPUTES GLOBAL UNIT COST FACTOR - XUC *
C* SSS MOD SLR - 23 JUNE 80 *
C*****
C
COMMON /FINC/ FINC
COMMON /ISC/ ISC
REAL ISC
COMMON /LDERV/ LDERV
COMMON /RSC/ RSC
COMMON /TDXUC/ TDXUC
COMMON /TERMH/ TERMH
C
C
C      TDXUC=(FINC*(TERMH+ISC+RSC))/1000000.
C
RETURN
END
```

SUBROUTINE DUP

```

C
C***** 800827 112349357
C*****
C* SSS MOD SLR - 5 JUN 80 *
C* COMPUTES ITEM SPECIFIC UNIT COST *
C* FACTOR - UP(I) *
C*****
C
COMMON /PRNTXX/ PRNTXX
INTEGER PRNTXX
COMMON /FINC/ FINC
COMMON /I/ I
COMMON /IDUP/ IDUP(999)
COMMON /INO/ INO(999)
COMMON /ISCA/ ISCA(999)
REAL ISCA
COMMON /LC/ LC(999)
REAL LC
COMMON /LDERV/ LDERV
COMMON /LDUP/ LDUP
COMMON /MXI/ MXI
COMMON /MXNP/ MXNP
COMMON /NITEM/ NITEM(999,10)
REAL NITEM
COMMON /NP/ NP
COMMON /RSCA/ RSCA(999)
COMMON /TDUP/ TDUP(999)
COMMON /TOTT/ TOTT(10)
COMMON /UP/ UP(999)
COMMON /XUC/ XUC

C
C
DO 220 IXXX1=1,MXI
    I=INO(IXXX1)
    TEM01=0.
    DO 210 NP=1,MXNP
        TEM01=TEM01+TOTT(NP)*NITEM(I,NP)
210    CONTINUE
        TDUP(I)=FINC*(TEM01*UP(I)*LC(I)*XUC+(ISCA(I)+RSCA(I)))/1000000.
220    CONTINUE
C
DO 999 IXXX=1,MXI
    IDUP (IXXX)=INO (IXXX)
999    CONTINUE
    LD=LDUP
    IF (PRNTXX.NE.0) LD=MAX0(LD,LDERV )
    CALL TDSORT(TDUP ,IDUP ,LD,MXI )

```

C

RETURN
END

SUBROUTINE DFR

C
 C***** 800827 112356025
 C* SSS MOD SLR - 5 JUN 80 *
 C* COMPUTES ITEM SPECIFIC FAILURE RATE *
 C* - FR(I) *
 C*****
 C

COMMON /PRNTXX/ PRNTXX
 INTEGER PRNTXX
 COMMON /A/ A(999,4,30)
 INTEGER A
 COMMON /BAA/ BAA
 COMMON /COND/ COND(999)
 COMMON /CSE/ CSE(250)
 COMMON /DAA/ DAA
 COMMON /ERHBI/ ERHBI(999,16)
 COMMON /ERHD/ ERHD(999)
 COMMON /FINC/ FINC
 COMMON /I/ I
 COMMON /IDFR/ IDFR(999)
 COMMON /INO/ INO(999)
 COMMON /IRMIN/ IRMIN(999,4)
 COMMON /IRMT/ IRMT
 COMMON /ISCA/ ISCA(999)
 REAL ISCA
 COMMON /ISET/ ISET(250,16)
 REAL ISET
 COMMON /ISETD/ ISETD(250)
 REAL ISETD
 COMMON /L/ L
 COMMON /LDERV/ LDERV
 COMMON /LDFR/ LDFR
 COMMON /LT/ LT
 COMMON /MSE/ MSE(250)
 REAL MSE
 COMMON /MXI/ MXI
 COMMON /MXIRMT/ MXIRMT
 COMMON /MXLT/ MXLT
 COMMON /MXNS/ MXNS
 COMMON /NFB/ NFB(999,16)
 REAL NFB
 COMMON /NFD/ NFD(999)
 REAL NFD
 COMMON /NJA/ NJA(999,4)
 COMMON /NRM/ NRM(999)
 COMMON /NRUC/ NRUC

```

REAL NRUC
COMMON /NS/ NS
COMMON /OFMCA/ OFMCA(999)
COMMON /ONMCA/ ONMCA(999)
COMMON /PIUP/ PIUP
COMMON /QSA/ QSA(999,4,30)
COMMON /RM/ RM(999)
COMMON /RMI/ RMI(999,16)
INTEGER RMI
COMMON /RSCA/ RSCA(999)
COMMON /SA/ SA
COMMON /TDFR/ TDFR(999)
COMMON /TISQ/ TISQ(999)
COMMON /TISQN/ TISQN(999)
COMMON /TNB/ TNB(16)
COMMON /TQSA/ TQSA
COMMON /UP/ UP(999)
COMMON /USE/ USE(250,16)
COMMON /USED/ USED(250)
COMMON /XITEMQ/ XITEMQ(999)
COMMON /XUC/ XUC
COMMON /YRSQ/ YRSQ(999)

```

C
C

```

DO 320 IXXX1=1,MXI
  I=INO(IXXX1)
  TEM01=0.
  DO 210 NS=1,MXNS
    TEM01=TEM01+TNB(NS)*F((1.+FINC)*NFB(I,NS))
210  CONTINUE
    TISQN(I)=TEM01+F((1.+FINC)*NFD(I))
    XITEMQ(I)=(TISQN(I)-TISQ(I))+FINC*NRUC*YRSQ(I)
    TDFRA=(1.+FINC)*(NRUC*XLEARN(I)+(PIUP-NRUC))*YRSQ(I)*(COND(I)+
+    (1.-COND(I))*RM(I))*UP(I)*XUC-RSCA(I)+TISQN(I)*UP(I)*XLEARN(I)
+    *XUC-ISCA(I)+FINC*(ONMCA(I)+OFMCA(I))
    TDFRB=0.
    NXXX1=NRM(I)
    IF(.NOT.(NXXX1.GT.0)) GO TO 300
    DO 280 IRMT=1,MXIRMT
      IF(.NOT.(IRMT.LE.NXXX1)) GO TO 290
      NXXX2=NJA(I,IRMT)
      IF(.NOT.(NXXX2.GT.0)) GO TO 270
    DO 250 LT=1,MXLT
      IF(.NOT.(LT.LE.NXXX2)) GO TO 260
      L=A(I,IRMT,LT)
      TDFRB1=0.
    DO 230 NS=1,MXNS

```

```

        IF(.NOT.(IRMIN(I,IRMT).EQ.RMI(I,NS))) GO TO 220
        TQSA=QSA(I,IRMT,LT)-AINT(QSA(I,IRMT,LT)/100.)*100.
        TDFRB1=USE(L,NS)*ERHBI(I,NS)*U(TQSA)*ISET(L,NS)*
+       TNB(NS)/BAA+TDFRB1
220         CONTINUE
230         CONTINUE
        TDFRB2=0.
        IF(.NOT.(IRMIN(I,IRMT).EQ.1)) GO TO 240
        TQSA=QSA(I,IRMT,LT)-AINT(QSA(I,IRMT,LT)/100.)*100.
        TDFRB2=USED(L)*ERHD(I)*U(TQSA)*ISETD(L)/DAA
240         CONTINUE
        TDFRB=TDFRB+FINC*(TDFRB1+TDFRB2)*CSE(L)*(1.+PIUP*MSE(L))
250         CONTINUE
260         CONTINUE
270         CONTINUE
280         CONTINUE
290         CONTINUE
300         CONTINUE
        TEM02=0.
        DO 310 NS=1,MXNS
            TEM02=TEM02+(AMIN1(F((1.+FINC)*NFB(I,NS)),1.)-AMIN1(F(NFB(I,
+       NS)),1.))*TNB(NS)
310         CONTINUE
        TDFR(I)=(TDFRA+TDFRB+TEM02*PIUP*SA)/1000000.
320 CONTINUE
C
        DO 999 IXXX=1,MXI
            IDFR (IXXX)=INO (IXXX)
999 CONTINUE
        LD=LDFR
        IF (PRNTXX.NE.0) LD=MAX0(LD,LDERV )
        CALL TDSORT(TDFR ,IDFR ,LD,MXI )
C
        RETURN
        END

```

SUBROUTINE DXFR

800827 112421245

```

C
C*****
C* COMPUTES GLOBAL FAILURE RATE FACTOR - XFR *
C*****
C
COMMON /I/ I
COMMON /INO/ INO(999)
COMMON /LDERV/ LDERV
COMMON /MXI/ MXI
COMMON /TDFR/ TDFR(999)
COMMON /TDXFR/ TDXFR

C
C
TEM01=0.
DO 210 IXXX1=1,MXI
    I=INO(IXXX1)
    TEM01=TEM01+TDFR(I)
210 CONTINUE
TDXFR=TEM01

C
RETURN
END

```

SUBROUTINE DFPR

800827 112432081

C
C*****
C* COMPUTES ITEM FALSE PULL RATE - FPR(I) *
C*****
C

COMMON /PRNTXX/ PRNTXX
INTEGER PRNTXX
COMMON /A/ A(999,4,30)
INTEGER A
COMMON /B/ B
INTEGER B
COMMON /BAA/ BAA
COMMON /BCMh/ BCMH(999)
COMMON /BLR/ BLR
COMMON /BMF/ BMF
COMMON /BRCT/ BRCT
COMMON /BTYPE/ BTYPE(16)
INTEGER BTYPE
COMMON /CIMF/ CIMF(16)
COMMON /COND/ COND(999)
COMMON /CPPC/ CPPC
COMMON /CRCT/ CRCT
COMMON /CSE/ CSE(250)
COMMON /FAIL/ FAIL(999,16)
COMMON /FINC/ FINC
COMMON /FPR/ FPR(999)
COMMON /HDWRIT/ HDWRIT(999,10)
COMMON /I/ I
COMMON /IDFPR/ IDFPR(999)
COMMON /INO/ INO(999)
COMMON /IRMIN/ IRMIN(999,4)
COMMON /IRMT/ IRMT
COMMON /ISCA/ ISCA(999)
REAL ISCA
COMMON /ISET/ ISET(250,16)
REAL ISET
COMMON /L/ L
COMMON /LDERV/ LDERV
COMMON /LDFPR/ LDFPR
COMMON /LRU/ LRU(999)
COMMON /LT/ LT
COMMON /MSE/ MSE(250)
REAL MSE
COMMON /MXI/ MXI
COMMON /MXIRMT/ MXIRMT
COMMON /MXLT/ MXLT

```

COMMON /MXNP/ MXNP
COMMON /MXNS/ MXNS
COMMON /NBC/ NBC(16)
REAL NBC
COMMON /NFB/ NFB(999,16)
REAL NFB
COMMON /NHB/ NHB(16)
COMMON /NITEM/ NITEM(999,10)
REAL NITEM
COMMON /NJA/ NJA(999,4)
COMMON /NP/ NP
COMMON /NRM/ NRM(999)
COMMON /NRUC/ NRUC
REAL NRUC
COMMON /NS/ NS
COMMON /OSTC/ OSTC
COMMON /PIUP/ PIUP
COMMON /QSA/ QSA(999,4,30)
COMMON /RM/ RM(999)
COMMON /RMH/ RMH(999)
COMMON /RMI/ RMI(999,16)
INTEGER RMI
COMMON /RSCA/ RSCA(999)
COMMON /SA/ SA
COMMON /SAT/ SAT(16)
COMMON /TDFPR/ TDFPR(999)
COMMON /TEMP/ TEMP
COMMON /TISQ/ TISQ(999)
COMMON /TNB/ TNB(16)
COMMON /TOTT/ TOTT(10)
COMMON /TQSA/ TQSA
COMMON /UP/ UP(999)
COMMON /USE/ USE(250,16)
COMMON /WT/ WT(999)
COMMON /XFPR/ XFPR
COMMON /XITEMQ/ XITEMQ(999)
COMMON /XUC/ XUC
COMMON /YRSQ/ YRSQ(999)
DIMENSION CHNFB(999,16)

```

C
C

```

DO 270 NS=1,MXNS
  IF(.NOT.(BTYPE(NS).EQ.3)) GO TO 220
  DO 210 IXXX2=1,MXI
    I=INO(IXXX2)
    CHNFB(I,NS)=FAIL(I,NS)*FLOAT(LRU(I))*FINC*FPR(I)*XFPR*OSTC
210  CONTINUE

```

```

220  CONTINUE
      IF(.NOT.(BTYPE(NS).NE.3)) GO TO 260
      DO 250 IXXX2=1,MXI
        I=INO(IXXX2)
        CHNFB(I,NS)=FAIL(I,NS)*FLOAT(LRU(I))*FINC*FPR(I)*XFPR*BRCT
        IF(.NOT.(BTYPE(NS).EQ.2)) GO TO 240
        TEM01=0.
        DO 230 B=1,MXNS
          IF(.NOT.(NHB(B).EQ.NS)) GO TO 230
          TEM01=TEM01+FAIL(I,B)*NBC(B)
230  CONTINUE
        CHNFB(I,NS)=CHNFB(I,NS)+CIMF(NS)*TEM01*FLOAT(LRU(I))*FINC*
+      FPR(I)*XFPR*CRCT
240  CONTINUE
250  CONTINUE
260  CONTINUE
270  CONTINUE
      DO 430 IXXX1=1,MXI
        I=INO(IXXX1)
        TDFPR(I)=0.
        IF(.NOT.(LRU(I).NE.0)) GO TO 420
        TEM02=0.
        DO 280 NS=1,MXNS
          TEM02=TEM02+TNB(NS)*(F(NFB(I,NS)+CHNFB(I,NS))-F(NFB(I,NS)))
280  CONTINUE
        XITEMQ(I)=TEM02
        TEM03=0.
        DO 290 NS=1,MXNS
          TEM03=TEM03+TNB(NS)*FAIL(I,NS)
290  CONTINUE
        TDFPRA=(TISQ(I)+XITEMQ(I))*UP(I)*XLEARN(I)*XUC-ISCA(I)+12.*
+      PIUP*TEM03*FINC*FPR(I)*XFPR*RMH(I)*BMF*BLR
        TEM04=0.
        DO 300 NS=1,MXNS
          TEM04=TEM04+TNB(NS)*FAIL(I,NS)*(BCMH(I)*BMF*BLR+SAT(NS)*2.*
+      CPPC*WT(I))
300  CONTINUE
        TDFPRB=12.*PIUP*TEM04*FINC*XFPR*FPR(I)
        TDFPRC=0.
        DO 390 NS=1,MXNS
          TEMP=0.
          NXXX1=NRM(I)
          IF(.NOT.(NXXX1.GT.0)) GO TO 370
          DO 350 IRMT=1,MXIRMT
            IF(.NOT.(IRMT.LE.NXXX1)) GO TO 360
            IF(.NOT.(IRMIN(I,IRMT).EQ.RMI(I,NS))) GO TO 340
            NXXX2=NJA(I,IRMT)

```

```

      IF(.NOT.(NXXX2.GT.0)) GO TO 330
      DO 310 LT=1,MXLT
      IF(.NOT.(LT.LE.NXXX2)) GO TO 320
      TQSA=QSA(I,IRMT,LT)-AINT(QSA(I,IRMT,LT)/100.)*100.
      L=A(I,IRMT,LT)
      TEMP=TEMP+U(TQSA)*USE(L,NS)*ISET(L,NS)*CSE(L)*(1.+
+      PIUP*MSE(L))
310      CONTINUE
320      CONTINUE
330      CONTINUE
340      CONTINUE
350      CONTINUE
360      CONTINUE
370      CONTINUE
      TEM05=0.
      DO 380 B=1,MXNS
      IF(.NOT.(NHB(B).EQ.NS)) GO TO 380
      TEM05=TEM05+FAIL(I,B)*NBC(B)
380      CONTINUE
      TDFPRC=TDFPRC+(1.-SAT(NS))*(FAIL(I,NS)+CIMF(NS)*TEM05)*TEMP
390      CONTINUE
      TDFPRC=TDFPRC*FI(I)*FPR(I)*XFPR*BCMH(I)*BMF/BAA
      TEM06=0.
      DO 400 NS=1,MXNS
      TEM06=TEM06+TNB(NS)*(AMIN1(F(CHNFB(I,NS)+NFB(I,NS)),1.))-
+      AMIN1(F(NFB(I,NS)),1.))
400      CONTINUE
      TEM07=0.
      DO 410 NP=1,MXNP
      TEM07=TEM07+TOTT(NP)*(NITEM(I,NP)*UP(I)*XLEARN(I)*XUC-
+      HDWRIT(I,NP))
410      CONTINUE
      TDFPRD=TEM06*PIUP*SA+TEM07+(NRUC*XLEARN(I)+(PIUP-NRUC))*
+      YRSQ(I)*(COND(I)+(1.-COND(I))*RM(I))*UP(I)*XUC-RSCA(I)
      TDFPR(I)=(TDFPRA+TDFPRB+TDFPRC+TDFPRD)/1000000.
420      CONTINUE
430      CONTINUE
C      DO 999 IXXX=1,MXI
      IDFPR (IXXX)=INO (IXXX)
999      CONTINUE
      LD=LDFPR
      IF (PRNTXX.NE.0) LD=MAX0(LD,LDERV )
      CALL TDSORT(TDFPR ,IDFPR ,LD,MXI )
C
      RETURN
      END

```

SUBROUTINE DXFPR

800827 112507129

```

C
C*****
C* COMPUTES GLOBAL FALSE PULL RATE - *
C* XFPR FACTOR *
C*****
C
COMMON /I/ I
COMMON /INO/ INO(999)
COMMON /LDERV/ LDERV
COMMON /MXI/ MXI
COMMON /TDFPR/ TDFPR(999)
COMMON /TDXFPR/ TDXFPR

C
C
TEM01=0.
DO 210 IXXX1=1,MXI
    I=INO(IXXX1)
    TEM01=TEM01+TDFPR(I)
210 CONTINUE
TDXFPR=TEM01

C
RETURN
END

```

SUBROUTINE DRTS

```

C
C***** 800827 112511521 *****
C* COMPUTES ITEM BASE REPAIR FRACTION *
C* -RTS(I) FACTOR - DRTS *
C*****
C
COMMON /PRNTXX/ PRNTXX
INTEGER PRNTXX
COMMON /COND/ COND(999)
COMMON /FINC/ FINC
COMMON /I/ I
COMMON /IDRTS/ IDRTS(999)
COMMON /INO/ INO(999)
COMMON /LDERV/ LDERV
COMMON /LDRTS/ LDRTS
COMMON /MXI/ MXI
COMMON /NRTS/ NRTS(999)
REAL NRTS
COMMON /TDRTS/ TDRTS(999)

C
C
DO 250 IXXX=1,MXI
  I=INO(IXXX)
  TDRTS(I)=0.
  IF(.NOT.(COND(I).EQ.1.)) GO TO 210
  CR=AMIN1(FINC,1.)
  CN=0.
  CC=-1.*CR
210  CONTINUE
  IF(.NOT.(COND(I).LT.1.)) GO TO 220
  CR=AMIN1(FINC,NRTS(I))
  CN=-1.*CR
  CC=0.
220  CONTINUE
  IF(.NOT.(CR.GT.0.)) GO TO 230
  TDRTS(I)=(CHLCC(CC,CR,CN,I))/1000000.
230  CONTINUE
  IF(.NOT.(TDRTS(I).GT.-0.000001)) GO TO 240
  TDRTS(I)=0.
240  CONTINUE
250  CONTINUE

C
DO 999 IXXX=1,MXI
  IDRTS (IXXX)=INO (IXXX)
999  CONTINUE
LP=LDRTS

```

```
IF (PRNTXX.NE.0) LD=MAX0(LD,LDERV )  
CALL TDSORT(TDRTS ,IDRTS ,LD,MXI  )
```

C

```
RETURN  
END
```

SUBROUTINE DNRTS

800827 112521167

```
C
C*****
C* COMPUTES ITEM DEPOT REPAIR FRACTION - *
C* NRTS(I) FACTOR - DNRTS *
C*****
C
```

```
COMMON /PRNTXX/ PRNTXX
INTEGER PRNTXX
COMMON /COND/ COND(999)
COMMON /FINC/ FINC
COMMON /I/ I
COMMON /IDNRTS/ IDNRTS(999)
COMMON /INO/ INO(999)
COMMON /LDERV/ LDERV
COMMON /LDNRTS/ LDNRTS
COMMON /MXI/ MXI
COMMON /RTS/ RTS(999)
COMMON /TDNRTS/ TDNRTS(999)
```

C
C

```
DO 250 IXXX1=1,MXI
  I=INO(IXXX1)
  TDNRTS(I)=0.
  IF(.NOT.(COND(I).EQ.1.)) GO TO 210
  CN=AMIN1(FINC,1.)
  CR=0.
  CC=-1.*CN
210  CONTINUE
  IF(.NOT.(COND(I).LT.1.)) GO TO 220
  CN=AMIN1(FINC,RTS(I))
  CR=-1.*CN
  CC=0.
220  CONTINUE
  IF(.NOT.(CN.GT.0.)) GO TO 230
  TDNRTS(I)=(CHLCC(CC,CR,CN,I))/1000000.
230  CONTINUE
  IF(.NOT.(TDNRTS(I).GT.-0.000001)) GO TO 240
  TDNRTS(I)=0.
240  CONTINUE
250  CONTINUE
```

C

```
DO 999 IXXX=1,MXI
  IDNRTS(IXXX)=INO (IXXX)
999  CONTINUE
  LD=LDNRTS
  IF (PRNTXX.NE.0) LD=MAX0(LD,LDERV )
```

C CALL TDSORT(TDNRTS, IDNRTS, LD, MXI)
RETURN
END

SUBROUTINE DCOND

800827 112537576

```

C
C*****
C* COMPUTES ITEM CONDEMNATION RATE - *
C* COND(I) FACTOR - DCOND *
C*****
C
COMMON /PRNTXX/ PRNTXX
INTEGER PRNTXX
COMMON /COND/ COND(999)
COMMON /FINC/ FINC
COMMON /I/ I
COMMON /IDCOND/ IDCOND(999)
COMMON /INO/ INO(999)
COMMON /LDCOND/ LDCOND
COMMON /LDERV/ LDERV
COMMON /MXI/ MXI
COMMON /NRTS/ NRTS(999)
REAL NRTS
COMMON /RTS/ RTS(999)
COMMON /TDCOND/ TDCOND(999)

C
C
DO 250 IXXX1=1,MXI
  I=INO(IXXX1)
  TDCOND(I)=0.
  IF(.NOT.(COND(I).EQ.1.)) GO TO 210
  CC=0.
  CR=0.
  CN=0.
210 CONTINUE
  IF(.NOT.(COND(I).LT.1.)) GO TO 220
  CC=AMIN1(FINC,1.-COND(I))
  CR=-1.*RTS(I)/(RTS(I)+NRTS(I))*CC
  CN=-1.*NRTS(I)/(RTS(I)+NRTS(I))*CC
220 CONTINUE
  IF(.NOT.(CC.GT.0.)) GO TO 230
  TDCOND(I)=(CHLCC(CC,CR,CN,I))/1000000.
230 CONTINUE
  IF(.NOT.(TDCOND(I).GT.-0.000001)) GO TO 240
  TDCOND(I)=0.
240 CONTINUE
250 CONTINUE

C
DO 999 IXXX=1,MXI
  IDCOND(IXXX)=INO (IXXX)
999 CONTINUE

```

```
LD=LDCOND
IF (PRNTXX.NE.0) LD=MAX0(LD,LDERV )
CALL TDSORT(TDCOND,IDCOND,LD,MXI  )
```

C

```
RETURN
END
```

SUBROUTINE DSRU

```

C
C***** 800827 112549404
C* SSS MOD JRC - 5 JUN 80
C* COMPUTES SENSITIVITY ON WHICH SRUS
C* SHOULD BE LRUS - DSRU
C*****
C

```

```

COMMON /PRNTXX/ PRNTXX
INTEGER PRNTXX
COMMON /A/ A(999,4,30)
INTEGER A
COMMON /B/ B
INTEGER B
COMMON /BAA/ BAA
COMMON /BCMh/ BCMh(999)
COMMON /BLR/ BLR
COMMON /BMF/ BMF
COMMON /BMH/ BMH(999)
COMMON /BRCT/ BRCT
COMMON /BTYPE/ BTYPE(16)
INTEGER BTYPE
COMMON /CIMF/ CIMF(16)
COMMON /COND/ COND(999)
COMMON /CPPC/ CPPC
COMMON /CPPD/ CPPD(3)
COMMON /CRCT/ CRCT
COMMON /CSE/ CSE(250)
COMMON /DAA/ DAA
COMMON /DLR/ DLR
COMMON /DMF/ DMF
COMMON /DMH/ DMH(999)
COMMON /DRCT/ DRCT(3)
COMMON /ERHBI/ ERHBI(999,16)
COMMON /FAIL/ FAIL(999,16)
COMMON /FPR/ FPR(999)
COMMON /I/ I
COMMON /IDSRU/ IDSRU(999)
COMMON /INO/ INO(999)
COMMON /IRMIN/ IRMIN(999,4)
COMMON /IRMT/ IRMT
COMMON /ISCA/ ISCA(999)
REAL ISCA
COMMON /ISET/ ISET(250,16)
REAL ISET
COMMON /ISETD/ ISETD(250)
REAL ISETD

```

```

COMMON /L/ L
COMMON /LDERV/ LDERV
COMMON /LDSRU/ LDSRU
COMMON /LO/ LO(16)
COMMON /LRU/ LRU(999)
COMMON /LT/ LT
COMMON /MSE/ MSE(250)
REAL MSE
COMMON /MXI/ MXI
COMMON /MXIRMT/ MXIRMT
COMMON /MXLT/ MXLT
COMMON /MXNP/ MXNP
COMMON /MXNS/ MXNS
COMMON /NBC/ NBC(16)
REAL NBC
COMMON /NHB/ NHB(16)
COMMON /NHI/ NHI(999)
COMMON /NITEM/ NITEM(999,10)
REAL NITEM
COMMON /NJA/ NJA(999,4)
COMMON /NP/ NP
COMMON /NRM/ NRM(999)
COMMON /NRTS/ NRTS(999)
REAL NRTS
COMMON /NRUC/ NRUC
REAL NRUC
COMMON /NS/ NS
COMMON /OFMCA/ OFMCA(999)
COMMON /OST/ OST(3)
COMMON /OSTC/ OSTC
COMMON /PIUP/ PIUP
COMMON /QSA/ QSA(999,4,30)
COMMON /RM/ RM(999)
COMMON /RMI/ RMI(999,16)
INTEGER RMI
COMMON /RSCA/ RSCA(999)
COMMON /RTS/ RTS(999)
COMMON /SA/ SA
COMMON /SAT/ SAT(16)
COMMON /TDSRU/ TDSRU(999)
COMMON /TISQ/ TISQ(999)
COMMON /TNB/ TNB(16)
COMMON /TOTT/ TOTT(10)
COMMON /UP/ UP(999)
COMMON /USE/ USE(250,16)
COMMON /USED/ USED(250)
COMMON /WT/ WT(999)

```

```

COMMON /XFPR/ XFPR
COMMON /XITEMQ/ XITEMQ(999)
COMMON /XUC/ XUC
COMMON /YRSQ/ YRSQ(999)
DIMENSION DINFB(999,16)
DIMENSION DINFD(999)
DIMENSION DNHNFB(999,16)
DIMENSION DNHNFD(999)
REAL NXXX2
DIMENSION RNHMH(999,16)

```

C
C

```

DO 520 IXXX1=1,MXI
  I=INO(IXXX1)
  TDSRU(I)=0.
  IF(.NOT.(NHI(I).NE.0)) GO TO 510
  IF(.NOT.(LRU(I).EQ.0.AND.LRU(NHI(I)).EQ.1). GO TO 500
  DO 260 NS=1,MXNS
    IF(.NOT.(BTYPE(NS).EQ.3)) GO TO 210
    DNHNFB(I,NS)=(FAIL(NHI(I),NS)-FAIL(I,NS))*(1.+FPR(NHI(I)
+      ))*OSTC
    DINFB(I,NS)=FAIL(I,NS)*(1.+FPR(NHI(I)))*OSTC
210  CONTINUE
    IF(.NOT.(BTYPE(NS).NE.3)) GO TO 240
    TEM01=0.
    DO 220 B=1,MXNS
      IF(.NOT.(NHB(B).EQ.NS)) GO TO 220
      TEM01=TEM01+AMAX1((FAIL(NHI(I),B)-FAIL(I,B)),0.)*
+      NBC(B)
220  CONTINUE
      DNHNFB(I,NS)=AMAX1((FAIL(NHI(I),NS)-FAIL(I,NS)),0.)*
+      ((FPR(NHI(I))*XFPR+RTS(NHI(I)))*BRCT+(NRTS(NHI(I))+
+      COND(NHI(I))*OST(LO(NS)))+CIMF(NS)*TEM01*((RTS(NHI(I)
+      )+FPR(NHI(I))*XFPR)*CRCT+(NRTS(NHI(I))+COND(NHI(I))*
+      (OST(LO(NS))+U(FPR(NHI(I))*XFPR)*CRCT))
      TEM02=0.
      DO 230 B=1,MXNS
        IF(.NOT.(NHB(B).EQ.NS)) GO TO 230
        TEM02=TEM02+FAIL(I,B)*NBC(B)
230  CONTINUE
        DINFB(I,NS)=FAIL(I,NS)*((FPR(NHI(I))*XFPR+RTS(I))*BRCT+
+      (NRTS(I)+COND(I))*OST(LO(NS)))+CIMF(NS)*TEM02*((RTS(I)
+      +FPR(NHI(I))*XFPR)*CRCT+(NRTS(I)+COND(I))*OST(LO(NS))
+      +U(FPR(NHI(I))*XFPR)*CRCT))
240  CONTINUE
      RNHMH(I,NS)=0.
      IF(.NOT.(FAIL(NHI(I),NS).GT.0.001)) GO TO 250

```

```

+          RNHMH(I,NS)=AMIN1(FAIL(I,NS)/FAIL(NHI(I),NS),1.)*
250          ERHBI(NHI(I),NS)
+          CONTINUE
260          CONTINUE
          TEM03=0.
          DO 270 NS=1,MXNS
            TEM03=TEM03+AMAX1((FAIL(NHI(I),NS)-FAIL(I,NS)),0.)*TNB(NS)
+            *NRTS(NHI(I))*DRCT(LO(NS))
270          CONTINUE
          DNHNFD(I)=TEM03
          TEM04=0.
          DO 280 NS=1,MXNS
            TEM04=TEM04+FAIL(I,NS)*TNB(NS)*NRTS(I)*DRCT(LO(NS))
280          CONTINUE
          DINFD(I)=TEM04
          TEM05=0.
          DO 290 NP=1,MXNP
            TEM05=TEM05+TOTT(NP)*NITEM(I,NP)
290          CONTINUE
          TEM06=0.
          DO 300 NS=1,MXNS
            TEM06=TEM06+TNB(NS)*F(DINFB(I,NS))
300          CONTINUE
          TEM07=0.
          DO 310 NS=1,MXNS
            TEM07=TEM07+AMIN1(FAIL(I,NS),FAIL(NHI(I),NS))*TNB(NS)
310          CONTINUE
          XITEMQ(I)=TEM05+TEM06+F(DINFD(I))-TISQ(I)+12.*NRUC*TEM07*
+            (1.-COND(NHI(I)))-NRUC*YRSQ(I)
          TEM08=0.
          DO 320 NS=1,MXNS
            TEM08=TEM08+TNB(NS)*F(DNHNFB(I,NS))
320          CONTINUE
          TEM09=0.
          DO 330 NS=1,MXNS
            TEM09=TEM09+AMAX1((FAIL(NHI(I),NS)-FAIL(I,NS)),0.)*TNB(NS)
330          CONTINUE
          XITEMQ(NHI(I))=TEM08+F(DNHNFD(I))-TISQ(NHI(I))+12.*NRUC*
+            TEM09*(1.-COND(NHI(I)))-NRUC*YRSQ(NHI(I))
          TEM10=0.
          DO 340 NS=1,MXNS
            TEM10=TEM10+TNB(NS)*F(DNHNFB(I,NS))
340          CONTINUE
          TEM11=0.
          DO 350 NS=1,MXNS
            TEM11=TEM11+TNB(NS)*F(DINFB(I,NS))
350          CONTINUE

```

```

TEM12=0.
DO 360 NS=1,MXNS
    TEM12=TEM12+AMAX1((FAIL(NHI(I),NS)-FAIL(I,NS)),0.)*TNB(NS)
360 CONTINUE
    TEM13=0.
    DO 370 NS=1,MXNS
        TEM13=TEM13+AMIN1(FAIL(I,NS),FAIL(NHI(I),NS))*TNB(NS)
370 CONTINUE
    TDSRUA=(TEM10+F(DNHNFD(I)))*(UP(NHI(I))-UP(I))*XLEARN(NHI(I)
+      )*XUC-ISCA(NHI(I))+(TEM11+F(DINFD(I)))*UP(I)*XLEARN(I)*
+      XUC-ISCA(I)+(NRUC*XLEARN(NHI(I))+(PIUP-NRUC))*12.*TEM12*
+      (COND(NHI(I))+(1.-COND(NHI(I)))*RM(NHI(I)))*(UP(NHI(I))-
+      UP(I))*XUC-RSCA(NHI(I))+(NRUC*XLEARN(I)+(PIUP-NRUC))*12.*
+      TEM13*(1.-COND(NHI(I)))*(COND(I)+(1.-COND(I))*RM(I))*UP(I)
+      *XUC-RSCA(I)
    TEM14=0.
    DO 380 NS=1,MXNS
        TEM14=TEM14+AMAX1((FAIL(NHI(I),NS)-FAIL(I,NS)),0.)*TNB(NS)
+      *(((1.+FPR(NHI(I))*XFPR)*BCMh(NHI(I))+RTS(NHI(I))*
+      BMH(NHI(I))*BMF*BLR+NRTS(NHI(I))*(DMH(NHI(I))*DMF*DLR+
+      2.*CPPD(LO(NS))*(WT(NHI(I))-WT(I))+COND(NHI(I))*
+      CPPD(LO(NS))*(WT(NHI(I))-WT(I))+SAT(NS)*(1.+FPR(NHI(I))*
+      XFPR)*2.*CPPC*(WT(NHI(I))-WT(I)))
380 CONTINUE
    TEM15=0.
    DO 390 NS=1,MXNS
        TEM15=TEM15+AMIN1(FAIL(NHI(I),NS),FAIL(I,NS))*TNB(NS)*
+      (((1.+FPR(NHI(I))*XFPR)*BCMh(I)+RTS(I)*BMH(I))*BMF*BLR+
+      NRTS(I)*(DMH(I))*DMF*DLR+2.*CPPD(LO(NS))*WT(I))+COND(I)*
+      CPPD(LO(NS))*WT(I)+SAT(NS)*(1.+FPR(NHI(I))*XFPR)*2.*
+      CPPC*WT(I))
390 CONTINUE
    TDSRUB=12.*PIUP*TEM14-OFMCA(NHI(I))+12.*PIUP*TEM15-OFMCA(I)
    NXXX1=NRM(NHI(I))
    TDSRUC=0.
    IF(.NOT.(NXXX1.GT.0)) GO TO 470
    DO 450 IRMT=1,MXIRMT
        IF(.NOT.(IRMT.LE.NXXX1)) GO TO 460
        IF(.NOT.(IRMIN(NHI(I),IRMT).EQ.
+      RMI(NHI(I),NS))) GO TO 440
        NXXX2=NJA(NHI(I),IRMT)
        IF(.NOT.(NXXX2.GT.0)) GO TO 430
        DO 410 LT=1,MXLT
            IF(.NOT.(LT.LE.NXXX2)) GO TO 420
            L=A(NHI(I),IRMT,LT)
            DO 400 NS=1,MXNS
                TDSRUC=TDSRUC+U(QSA(NHI(I),IRMT,LT)-

```

```

+          AINT(QSA(NHI(I),IRMT,LT)/100.)*100.)*TNB(NS)*
+          (USE(L,NS)*RNHMH(I,NS)*ISET(L,NS)/BAA+USED(L)*
+          AMIN1(FAIL(I,NS),FAIL(NHI(I),NS))*NRTS(NHI(I))*
+          DMH(NHI(I))*DMF*ISETD(L)/DAA)*CSE(L)*(1.+PIUP*
+          MSE(L))
400          CONTINUE
410          CONTINUE
420          CONTINUE
430          CONTINUE
440          CONTINUE
450          CONTINUE
460          CONTINUE
          TDSRUC=-1.*TDSRUC
470          CONTINUE
          TEM16=0.
          DO 480 NS=1,MXNS
            TEM16=TEM16+U(FAIL(I,NS))*SAT(NS)*TNB(NS)
480          CONTINUE
          TDSRUD=PIUP*SA*TEM16
          TDSRU(I)=(TDSRUA+TDSRUB+TDSRUC+TDSRUD)/1000000.
          IF(.NOT.(TDSRU(I).GT.-0.000001)) GO TO 490
          TDSRU(I)=0.
490          CONTINUE
500          CONTINUE
510          CONTINUE
520          CONTINUE
C
          DO 999 IXXX=1,MXI
            IDSRU (IXXX)=INO  (IXXX)
999          CONTINUE
          LD=LDSRU
          IF (PRNTXX.NE.0) LD=MAX0(LD,LDERV )
          CALL TDSORT(TDSRU ,IDSRU ,LD,MXI  )
C
          RETURN
          END

```

SUBROUTINE DXMIL

C 800827 112702262
 C*****
 C* COMPUTES GLOBAL SENSITIVITY WITH RESPECT TO *
 C* MOD/I LABOR HOURS *
 C*****
 C

COMMON /FINC/ FINC
 COMMON /FR/ FR(3,10)
 COMMON /IA/ IA
 COMMON /LDERV/ LDERV
 COMMON /M/ M
 COMMON /MILR/ MILR(3)
 REAL MILR
 COMMON /MIMH/ MIMH(4,3,10)
 REAL MIMH
 COMMON /MXM/ MXM
 COMMON /MXNP/ MXNP
 COMMON /MXNS/ MXNS
 COMMON /NIA/ NIA
 COMMON /NP/ NP
 COMMON /NPLT/ NPLT(10,16)
 REAL NPLT
 COMMON /NS/ NS
 COMMON /TDMIL/ TDMIL
 COMMON /TDXMIL/ TDXMIL
 COMMON /TNB/ TNB(16)
 COMMON /XMIL/ XMIL

C
 C

TEM04=0.
 DO 240 NP=1,MXNP
 TEM01=0.
 DO 210 NS=1,MXNS
 TEM01=TEM01+TNB(NS)*NPLT(NP,NS)
 210 CONTINUE
 TEM03=0.
 DO 230 M=1,MXM
 TEM02=0.
 DO 220 IA=1,NIA
 TEM02=TEM02+MIMH(IA,M,NP)*XMIL
 220 CONTINUE
 TEM03=TEM03+FR(M,NP)*MILR(M)*TEM02
 230 CONTINUE
 TEM04=TEM04+TEM01*TEM03/1000000.
 240 CONTINUE
 TDMIL=TEM04

C

TDXMIL=FINC*TDMIL

RETURN

END

SUBROUTINE OTAB1

C

800827 112707950

C*****

C* PRINTS A SUMMARY BY TOP-LEVEL *

C* COST ELEMENTS *

C* SSS MOD LCR - 28 MAY 80 *

C*****

C

COMMON /PRNTXX/ PRNTXX

INTEGER PRNTXX

COMMON /FSED/ FSED

COMMON /IIMC/ IIMC

REAL IIMC

COMMON /ISC/ ISC

REAL ISC

COMMON /MIC/ MIC

REAL MIC

COMMON /MTRC/ MTRC

REAL MTRC

COMMON /OC/ OC

COMMON /OFMC/ OFMC

COMMON /ONMC/ ONMC

COMMON /RSC/ RSC

COMMON /SEDC/ SEDC

COMMON /SEPC/ SEPC

COMMON /STDC/ STDC

COMMON /TERMH/ TERMH

COMMON /TERMI/ TERMI

COMMON /TSEC/ TSEC

1 FORMAT(1H1,15X,50HOUTPUT TABLE 1: SUMMARY BY TOP-LEVEL COST ELEMEN
+TS/24X,33H(IN MILLIONS OF CONSTANT DOLLARS)//8X,12HPROGRAM COST,5X
+,11HDEVELOPMENT,3X,10HPRODUCTION,4X,7HSUPPORT,12X,5HTOTAL//)

2 FORMAT(2X,20HFULL SCALE ENG. DEVT,5X,F6.2,10X,F4.2,8X,F4.2,12X,F6.
+2)

3 FORMAT(2X,15HP.M.E.:HARDWARE,12X,F4.2,8X,F6.2,8X,F4.2,12X,F6.2)

4 FORMAT(9X,11HINTEGRATION,9X,F4.2,8X,F6.2,8X,F4.2,12X,F6.2)

5 FORMAT(2X,10HOPERATIONS,17X,F4.2,10X,F4.2,6X,F6.2,12X,F6.2)

6 FORMAT(2X,16HMOD/INSTALLATION,11X,F4.2,8X,F6.2,8X,F4.2,12X,F6.2)

7 FORMAT(2X,17HSPARES: INVESTMENT,10X,F4.2,8X,F6.2,8X,F4.2,12X,F6.2)

8 FORMAT(9X,11HREPLACEMENT,9X,F4.2,10X,F4.2,6X,F6.2,12X,F6.2)

9 FORMAT(2X,17HSUPPORT EQUIPMENT,8X,F6.2,8X,F6.2,8X,F4.2,12X,F6.2)

10 FORMAT(2X,12HON EQUIPMENT,15X,F4.2,10X,F4.2,6X,F6.2,12X,F6.2)

11 FORMAT(2X,13HOFF EQUIPMENT,14X,F4.2,10X,F4.2,6X,F6.2,12X,F6.2)

12 FORMAT(2X,8HTRAINING,19X,F4.2,10X,F4.2,6X,F6.2,12X,F6.2)

13 FORMAT(2X,20HINVENTORY MANAGEMENT,7X,F4.2,10X,F4.2,6X,F6.2,12X,F6.
+2)

14 FORMAT(2X,16HTECHNICAL ORDERS,11X,F4.2,8X,F6.2,8X,F4.2,12X,F6.2/)

15 FORMAT(12X,10HTOTAL COST,5X,F6.2,8X,F6.2,6X,F6.2,10X,F8.2)

C
C
C

```
TEMP1=TERMH/1000000.  
TEMP2=TERMI/1000000.  
TEMP3=OC/1000000.  
TEMP4=MIC/1000000.  
TEMP5=ISC/1000000.  
TEMP6=RSC/1000000.  
TEMP7=SEDC/1000000.  
TEMP8=SEPC/1000000.  
TEMP9=TSEC/1000000.  
TEMPA=ONMC/1000000.  
TEMPB=OFMC/1000000.  
TEMPC=MTRC/1000000.  
TEMPE=STDC/1000000.  
TEMPF=FSEDC/1000000.  
T1=TEMP1+TEMP2+TEMP4+TEMP5+TEMP8+TEMPE  
T2=TEMP3+TEMP6+TEMPA+TEMPB+TEMPC+TEMPE  
T4=TEMPF+TEMP7  
T3=T4+T1+T2  
ZERO=0.00  
IF(PRNTXX.NE.0) WRITE( 7, 1)  
IF(PRNTXX.NE.1) WRITE(06, 1)  
IF(PRNTXX.NE.0) WRITE( 7, 2) TEMPF,ZERO,ZERO,TEMPF  
IF(PRNTXX.NE.1) WRITE(06, 2) TEMPF,ZERO,ZERO,TEMPF  
IF(PRNTXX.NE.0) WRITE( 7, 3) ZERO,TEMP1,ZERO,TEMP1  
IF(PRNTXX.NE.1) WRITE(06, 3) ZERO,TEMP1,ZERO,TEMP1  
IF(PRNTXX.NE.0) WRITE( 7, 4) ZERO,TEMP2,ZERO,TEMP2  
IF(PRNTXX.NE.1) WRITE(06, 4) ZERO,TEMP2,ZERO,TEMP2  
IF(PRNTXX.NE.0) WRITE( 7, 5) ZERO,ZERO,TEMP3,TEMP3  
IF(PRNTXX.NE.1) WRITE(06, 5) ZERO,ZERO,TEMP3,TEMP3  
IF(PRNTXX.NE.0) WRITE( 7, 6) ZERO,TEMP4,ZERO,TEMP4  
IF(PRNTXX.NE.1) WRITE(06, 6) ZERO,TEMP4,ZERO,TEMP4  
IF(PRNTXX.NE.0) WRITE( 7, 7) ZERO,TEMP5,ZERO,TEMP5  
IF(PRNTXX.NE.1) WRITE(06, 7) ZERO,TEMP5,ZERO,TEMP5  
IF(PRNTXX.NE.0) WRITE( 7, 8) ZERO,ZERO,TEMP6,TEMP6  
IF(PRNTXX.NE.1) WRITE(06, 8) ZERO,ZERO,TEMP6,TEMP6  
IF(PRNTXX.NE.0) WRITE( 7, 9) TEMP7,TEMP8,ZERO,TEMP9  
IF(PRNTXX.NE.1) WRITE(06, 9) TEMP7,TEMP8,ZERO,TEMP9  
IF(PRNTXX.NE.0) WRITE( 7,10) ZERO,ZERO,TEMPA,TEMPA  
IF(PRNTXX.NE.1) WRITE(06,10) ZERO,ZERO,TEMPA,TEMPA  
IF(PRNTXX.NE.0) WRITE( 7,11) ZERO,ZERO,TEMPB,TEMPB  
IF(PRNTXX.NE.1) WRITE(06,11) ZERO,ZERO,TEMPB,TEMPB  
IF(PRNTXX.NE.0) WRITE( 7,12) ZERO,ZERO,TEMPC,TEMPC
```

```
IF(PRNTXX.NE.1) WRITE(06,12) ZERO,ZERO,TEMPC,TEMPC
IF(PRNTXX.NE.0) WRITE( 7,13) ZERO,ZERO,TEMPO,TEMPO
IF(PRNTXX.NE.1) WRITE(06,13) ZERO,ZERO,TEMPO,TEMPO
IF(PRNTXX.NE.0) WRITE( 7,14) ZERO,TEMPE,ZERO,TEMPE
IF(PRNTXX.NE.1) WRITE(06,14) ZERO,TEMPE,ZERO,TEMPE
IF(PRNTXX.NE.0) WRITE( 7,15) T4,T1,T2,T3
IF(PRNTXX.NE.1) WRITE(06,15) T4,T1,T2,T3
```

C

```
RETURN
END
```

SUBROUTINE OTAB2

C 800827 112802322

C*****

C* PRINTS PLATFORM MODIFICATION/INSTALLATION COSTS *

C*****

C

```
COMMON /PRNTXX/ PRNTXX
INTEGER PRNTXX
COMMON /AKIT/ AKIT(4,10)
COMMON /FR/ FR(3,10)
COMMON /IA/ IA
COMMON /M/ M
COMMON /MIFIX/ MIFIX(3,10)
REAL MIFIX
COMMON /MILR/ MILR(3)
REAL MILR
COMMON /MIMH/ MIMH(4,3,10)
REAL MIMH
COMMON /MXM/ MXM
COMMON /MXNP/ MXNP
COMMON /MXNS/ MXNS
COMMON /NIA/ NIA
COMMON /NP/ NP
COMMON /NPLT/ NPLT(10,16)
REAL NPLT
COMMON /NS/ NS
COMMON /PNOUN/ PNOUN(10,12)
COMMON /TMIL/ TMIL
COMMON /TNB/ TNB(16)
COMMON /XMIL/ XMIL
```

```
1 FORMAT(1H1/30X,56HOUTPUT TABLE 2: PLATFORM MODIFICATION/INSTALLATI
+ON COSTS/37X,33H(IN MILLIONS OF CONSTANT DOLLARS)////30X,45H***REC
+URRING MOD/I COST TOTALS BY PLATFORM***//26X,5HFIXED,22X,6HRETRO-,
+3X,5HPRDC-,10X,1H*/3X,5HPLAT-,18X,5HPLAT.,4X,5HA-KIT,13X,3HFIT.6X,
+4HTION,11X,1H*,1X,30HA-KIT PLUS LABOR COSTS BY AREA/3X,4HFORM,19X,
+5HPREP/,4X,6HEQUIP-,3X,5HMOD/I,4X,5HMOD/I,4X,5HMOD/I,10X,1H*,1X,35
+(1H-)/3X,5HINDEX,1X,13HPLATFORM NAME,4X,5HRSTR.,4X,4HMENT,5X,5HLAB
+OR,4X,5HTOTAL,4X,5HTOTAL,4X,5HTOTAL,1X,1H*,1X,7HANTENNA,2X,8HELEC.
+BOX,2X,7HCNTL.HD,2X,7HCABLING/3X,4H(NP)/)
2 FORMAT(3X,13,2X,12A1,3X,6(F8.3,1X),1X,F8.3,2X,F8.3,1X,F8.3,1X,F8.3
+)
3 FORMAT(/4X,18(1H-),1X,6(8(1H-),1X),1X,8(1H-),2X,8(1H-),1X,8(1H-),1
+X,8(1H-)//5X,11HCOST TOTALS,6X,6(F9.3),1X,F9.3,1X,3F9.3)
```

C

C

C

C.....ONLY PRINT THIS TABLE IF OFF-LINE OUTPUT WAS REQUESTED

```

IF (PRNTXX.EQ.0) RETURN
C
WRITE( 7, 1)
T11=0.
T12=0.
TMIL=0.
T14=0.
T15=0.
T16=0.
T17=0.
T18=0.
T19=0.
T20=0.
DO 330 NP=1,MXNP
  TEM01=0.
  DO 210 NS=1,MXNS
    TEM01=TEM01+TNB(NS)*NPLT(NP,NS)
210  CONTINUE
    TPLT=TEM01
    TEM02=0.
    DO 220 M=1,MXM
      TEM02=TEM02+FR(M,NP)*(MIFIX(M,NP)*1000.)
220  CONTINUE
      T1=TPLT*TEM02/1000000.
      TEM03=0.
      DO 230 IA=1,NIA
        TEM03=TEM03+AKIT(IA,NP)
230  CONTINUE
      T2=TPLT*TEM03/1000000.
      TEM05=0.
      DO 250 M=1,MXM
        TEM04=0.
        DO 240 IA=1,NIA
          TEM04=TEM04+MIMH(IA,M,NP)*XMIL
240  CONTINUE
          TEM05=TEM05+FR(M,NP)*MILR(M)*TEM04
250  CONTINUE
          T3=TPLT*TEM05/1000000.
          TEM06=0.
          DO 260 IA=1,NIA
            TEM06=TEM06+MIMH(IA,2,NP)*XMIL*MILR(2)+AKIT(IA,NP)
260  CONTINUE
            TEM07=0.
            DO 270 IA=1,NIA
              TEM07=TEM07+MIMH(IA,3,NP)*XMIL*MILR(3)+AKIT(IA,NP)
270  CONTINUE
            T4=TPLT*(FR(2,NP)*((MIFIX(2,NP)*1000.)+TEM06)+FR(3,NP)*

```

```

+ ((MIFIX(3,NP)*1000.)+TEM07))/1000000.
TEM08=0.
DO 280 IA=1,NIA
  TEM08=TEM08+MIMH(IA,1,NP)*XMIL*MILR(1)+AKIT(IA,NP)
280 CONTINUE
  T5=TPLT*FR(1,NP)*((MIFIX(1,NP)*1000.)+TEM08)/1000000.
  T6=T4+T5
  TEM09=0.
  DO 290 M=1,MXM
    TEM09=TEM09+FR(M,NP)*MIMH(1,M,NP)*XMIL*MILR(M)
290 CONTINUE
    T7=TPLT*(TEM09+AKIT(1,NP))/1000000.
    TEM10=0.
    DO 300 M=1,MXM
      TEM10=TEM10+FR(M,NP)*MIMH(2,M,NP)*XMIL*MILR(M)
300 CONTINUE
      T8=TPLT*(TEM10+AKIT(2,NP))/1000000.
      TEM11=0.
      DO 310 M=1,MXM
        TEM11=TEM11+FR(M,NP)*MIMH(3,M,NP)*XMIL*MILR(M)
310 CONTINUE
        T9=TPLT*(TEM11+AKIT(3,NP))/1000000.
        TEM12=0.
        DO 320 M=1,MXM
          TEM12=TEM12+FR(M,NP)*MIMH(4,M,NP)*XMIL*MILR(M)
320 CONTINUE
          T10=TPLT*(TEM12+AKIT(4,NP))/1000000.
          WRITE( 7, 2) NP,(PNOUN(NP,K1),K1=1,12),T1,T2,T3,T4,T5,T6,T7,T8,
+ T9,T10
          T11=T11+T1
          T12=T12+T2
          TMIL=TMIL+T3
          T14=T14+T4
          T15=T15+T5
          T16=T16+T6
          T17=T17+T7
          T18=T18+T8
          T19=T19+T9
          T20=T20+T10
330 CONTINUE
          WRITE( 7, 3) T11,T12,TMIL,T14,T15,T16,T17,T18,T19,T20
C
          RETURN
          END

```

SUBROUTINE OTAB3A

C 800827 112846709

C*****
 C* SSS MOD LCR - 21 MAY 80 *
 C* PRINTS OPERATION AND LOGISTICS SUPPORT *
 C* COST ELEMENTS - PART 1 *
 C*****

C

COMMON /PRNTXX/ PRNTXX
 INTEGER PRNTXX
 COMMON /AFC/ AFC
 COMMON /BAFC/ BAFC(6)
 COMMON /BISC/ BISC(6)
 COMMON /BOFMC/ BOFMC(6)
 COMMON /BOLC/ BOLC(6)
 COMMON /BONMC/ BONMC(6)
 COMMON /BRSC/ BRSC(6)
 COMMON /ISC/ ISC
 REAL ISC
 COMMON /ISCB/ ISCB
 REAL ISCB
 COMMON /ISCD/ ISCD
 REAL ISCD
 COMMON /OFMC/ OFMC
 COMMON /OFMCB/ OFMCB
 COMMON /OFMCD/ OFMCD
 COMMON /OLC/ OLC
 COMMON /ONMC/ ONMC
 COMMON /RSC/ RSC

1 FORMAT(1H1/35X,62HOUTPUT TABLE 3: OPERATION AND LOGISTICS SUPPORT
 + COST ELEMENTS/50X,33H(IN MILLIONS OF CONSTANT DOLLARS))/24X,1H|,2
 +7X,57H| INDEP CIMF SATEL BASE | AIR GROUND | DEPOT/11
 +X,98HCOST ELEMENT | INITIAL RECURRING TOTAL | BASES BASES BA
 +SES TOTAL | BASES BASES | TOTAL/24X,1H|,27X,1H|,31X,1H|,16X,
 +1H|/10X,110(1H-)/24X,1H|,27X,1H|,31X,1H|,16X,1H|)
 2 FORMAT(7X,18HOPERATIONS LABOR |,9X,2(F6.2,3X),1H|,3(F6.2,2X),F6.2,
 +1X,1H|,1X,2(F6.2,1X),1X,1H|)
 3 FORMAT(7X,10HADDED FUEL,7X,1H|,9X,2(F6.2,3X),1H|,3(F6.2,2X),F6.2,1
 +X,1H|,1X,2(F6.2,1X),1X,1H|)
 4 FORMAT(7X,18HINITIAL SPARES |,1X,F6.2,11X,F6.2,3X,1H|,3(F6.2,2X)
 +,F6.2,1X,1H|,1X,2(F6.2,1X),1X,1H|,1X,F6.2)
 5 FORMAT(7X,18HREPLACE. SPARES |,9X,2(F6.2,3X),1H|,3(F6.2,2X),F6.2,
 +1X,1H|,1X,2(F6.2,1X),1X,1H|)
 6 FORMAT(7X,18HON-EQUIP. MAINT. |,9X,2(F6.2,3X),1H|,3(F6.2,2X),F6.2,
 +1X,1H|,1X,2(F6.2,1X),1X,1H|)
 7 FORMAT(7X,18HOFF-EQUIP. MAINT. |,9X,2(F6.2,3X),1H|,3(F6.2,2X),F6.2,
 +1X,1H|,1X,2(F6.2,1X),1X,1H|,1X,F6.2)

```

C
C
C
C.....ONLY PRINT THIS TABLE IF OFF-LINE OUTPUT WAS REQUESTED
      IF(PRNTXX.EQ.0) RETURN
C
      WRITE( 7, 1)
      T1=OLC/1000000.
      T2=BOLC(1)/1000000.
      T3=BOIC(2)/1000000.
      T4=BOIC(3)/1000000.
      T5=BOIC(4)/1000000.
      T6=BOLC(5)/1000000.
      T7=BOLC(6)/1000000.
      WRITE( 7, 2) T1,T1,T2,T3,T4,T1,T5,T6
      T1=AFC/1000000.
      T2=BAFC(1)/1000000.
      T3=BAFC(2)/1000000.
      T4=BAFC(3)/1000000.
      T5=BAFC(4)/1000000.
      T6=BAFC(5)/1000000.
      T7=BAFC(6)/1000000.
      WRITE( 7, 3) T1,T1,T2,T3,T4,T1,T5,T6
      T1=ISC/1000000.
      T2=BISC(1)/1000000.
      T3=BISC(2)/1000000.
      T4=BISC(3)/1000000.
      T5=ISCB/1000000.
      T6=BISC(4)/1000000.
      T7=BISC(5)/1000000.
      T8=BISC(6)/1000000.
      T9=ISCD/1000000.
      WRITE( 7, 4) T1,T1,T2,T3,T4,T5,T6,T7,T9
      T1=RSC/1000000.
      T2=BRSC(1)/1000000.
      T3=BRSC(2)/1000000.
      T4=BRSC(3)/1000000.
      T5=BRSC(4)/1000000.
      T6=BRSC(5)/1000000.
      T7=BRSC(6)/1000000.
      WRITE( 7, 5) T1,T1,T2,T3,T4,T1,T5,T6
      T1=ONMC/1000000.
      T2=BONMC(1)/1000000.
      T3=BONMC(2)/1000000.
      T4=BONMC(3)/1000000.
      T5=BONMC(4)/1000000.
      T6=BONMC(5)/1000000.

```

```
T7=BONMC(6)/1000000.  
WRITE( 7, 6) T1,T1,T2,T3,T4,T1,T5,T6  
T1=OFMC/1000000.  
T2=BOFMC(1)/1000000.  
T3=BOFMC(2)/1000000.  
T4=BOFMC(3)/1000000.  
T5=OFMCB/1000000.  
T6=BOFMC(4)/1000000.  
T7=BOFMC(5)/1000000.  
T8=BOFMC(6)/1000000.  
T9=OFMCD/1000000.  
WRITE( 7, 7) T1,T1,T2,T3,T4,T5,T6,T7,T9
```

C

```
RETURN  
END
```

SUBROUTINE OTAB3B

C 800827 112923779
 C*****
 C* SSS MOD LCR - 21 MAY 80 *
 C* PRINTS OPERATION AND LOGISTICS SUPPORT *
 C* COST ELEMENTS - PART 2 *
 C*****

C
 COMMON /PRNTXX/ PRNTXX
 INTEGER PRNTXX
 COMMON /AFC/ AFC
 COMMON /BAFC/ BAFC(6)
 COMMON /BIIMC/ BIIMC(6)
 COMMON /BISC/ BISC(6)
 COMMON /BMTRC/ BMTRC
 COMMON /BOFMC/ BOFMC(6)
 COMMON /BOLC/ BOLC(6)
 COMMON /BONMC/ BONMC(6)
 COMMON /BPLAT/ BPLAT(16)
 INTEGER BPLAT
 COMMON /BRSC/ BRSC(6)
 COMMON /BSECC/ BSECC(6)
 COMMON /BSECP/ BSECP(6)
 COMMON /BTCDI/ BTCDI
 COMMON /BTDC/ BTDC(16)
 COMMON /BTDCA/ BTDCA
 COMMON /BTDCC/ BTDCC
 COMMON /BTDCG/ BTDCG
 COMMON /BTDCI/ BTDCI
 COMMON /BTDCM/ BTDCM
 COMMON /BTDCS/ BTDCS
 COMMON /BTDCT/ BTDCT
 COMMON /BTYP/ BTYP(16)
 INTEGER BTYP
 COMMON /BXTRC/ BXTRC
 COMMON /DMTRC/ DMTRC
 COMMON /DTDC/ DTDC
 COMMON /IIMC/ IIMC
 REAL IIMC
 COMMON /IIMCB/ IIMCB
 REAL IIMCB
 COMMON /IIMCD/ IIMCD
 REAL IIMCD
 COMMON /IIMCI/ IIMCI
 REAL IIMCI
 COMMON /IIMCR/ IIMCR
 REAL IIMCR

```

COMMON /IMTRC/ IMTRC
REAL IMTRC
COMMON /ISC/ ISC
REAL ISC
COMMON /ISCB/ ISCB
REAL ISCB
COMMON /ISCD/ ISCD
REAL ISCD
COMMON /MTRC/ MTRC
REAL MTRC
COMMON /MXNS/ MXNS
COMMON /NS/ NS
COMMON /OFMC/ OFMC
COMMON /OFMCB/ OFMCB
COMMON /OFMCD/ OFMCD
COMMON /OLC/ OLC
COMMON /ONMC/ ONMC
COMMON /RMTRC/ RMTRC
COMMON /RSC/ RSC
COMMON /SECB/ SECB
COMMON /SECBP/ SECBP
COMMON /SECC/ SECC
COMMON /SECD/ SECD
COMMON /SECDP/ SECDP
COMMON /SECIC/ SECIC
COMMON /SECIP/ SECIP
COMMON /SECP/ SECP
COMMON /SECRC/ SECRC
COMMON /SECRP/ SECRP
COMMON /STDC/ STDC
COMMON /STDCI/ STDCI
COMMON /STDCR/ STDCR
COMMON /TNB/ TNB(16)
1 FORMAT(7X,18HSUPPORT EQUIPMENT|,27X,1H|,31X,1H|,16X,1H|)
2 FORMAT(7X,18H COMMON |,1X,F6.2,2X,2(F6.2,3X),1H|,3(F6.2,2
+X),F6.2,1X,1H|,1X,2(F6.2,1X),1X,1H|,1X,F6.2)
3 FORMAT(7X,18H PECULIAR |,1X,F6.2,2X,2(F6.2,3X),1H|,3(F6.2,2
+X),F6.2,1X,1H|,1X,2(F6.2,1X),1X,1H|,1X,F6.2)
4 FORMAT(7X,18HINVENTORY MANAG. |,1X,F6.2,2X,2(F6.2,3X),1H|,3(F6.2,2
+X),F6.2,1X,1H|,1X,2(F6.2,1X),1X,1H|,1X,F6.2)
5 FORMAT(7X,18HMAINT. TRAINING* |,1X,F6.2,2X,2(F6.2,3X),1H|,3(6H --
÷ ,2X),F6.2,1X,1H|,1X,2(6H -- ,1X),1X,1H|,1X,F6.2)
6 FORMAT(7X,18HTECH. DATA |,1X,F6.2,2X,2(F6.2,3X),1H|,3(F6.2,2
+X),F6.2,1X,1H|,1X,2(F6.2,1X),1X,1H|,1X,F6.2)
7 FORMAT(10X,110(1H-))
8 FORMAT(24X,1H|,27X,1H|,31X,1H|,16X,1H|/12X,6HTOTALS,6X,1H|,1X,F6.2
+,2X,2(F6.2,3X),1H|,3(F6.2,2X),F6.2,1X,1H|,1X,F6.2,1X,F6.2,2X,1H|,1

```

```

+X,F6.2///)
9 FORMAT(2X,1H(,1H*,43H MAINT. TRAINING IS ALLOCATED TO BASE TOTAL,
+63H AND DEPOT TOTAL BUT IS NOT FURTHER ALLOCATED AMONG BASE TYPES,
+1H)///)
10 FORMAT(24X,16HNUMBER OF BASES:/24X,16(1H-)//28X,9HINDEP. = ,F5.0,9
+X,15HAIR BASES = ,F5.0//28X,9HCIMF = ,F5.0,9X,15HGROUND BASES
+ = ,F5.0/66X,5(1H-)/28X,9HSATEL. = ,F5.0,9X,15HTOTAL = ,F5.
+0/37X,5(1H-)/28X,9HTOTAL = ,F5.0)

```

C
C
C

C.....ONLY PRINT THIS TABLE IF OFF-LINE OUTPUT WAS REQUESTED
IF(PRNTXX.EQ.0) RETURN

C

```

WRITE( 7, 1)
T1=SECIC/1000000.
T2=SECRIC/1000000.
T3=SECC/1000000.
T4=BSECC(1)/1000000.
T5=BSECC(2)/1000000.
T6=BSECC(3)/1000000.
T7=SECBC/1000000.
T8=BSECC(4)/1000000.
T9=BSECC(5)/1000000.
T10=BSECC(6)/1000000.
T11=SECDIC/1000000.
WRITE( 7, 2) T1,T2,T3,T4,T5,T6,T7,T8,T9,T11
T1=SECIP/1000000.
T2=SECRP/1000000.
T3=SECP/1000000.
T4=BSECP(1)/1000000.
T5=BSECP(2)/1000000.
T6=BSECP(3)/1000000.
T7=SECBP/1000000.
T8=BSECP(4)/1000000.
T9=BSECP(5)/1000000.
T10=BSECP(6)/1000000.
T11=SECDP/1000000.
WRITE( 7, 3) T1,T2,T3,T4,T5,T6,T7,T8,T9,T11
T1=IIMCI/1000000.
T2=IIMCR/1000000.
T3=IIMC/1000000.
T4=BIIMC(1)/1000000.
T5=BIIMC(2)/1000000.
T6=BIIMC(3)/1000000.
T7=IIMCB/1000000.
T8=BIIMC(4)/1000000.

```

```

T9=BIIMC(5)/1000000.
T10=BIIMC(6)/1000000.
T11=IIMCD/1000000.
WRITE( 7, 4) T1,T2,T3,T4,T5,T6,T7,T8,T9,T11
T1=IMTRC/1000000.
T2=RMTRC/1000000.
T3=MTRC/1000000.
T7=BMTRC/1000000.
T11=DMTRC/1000000.
WRITE( 7, 5) T1,T2,T3,T7,T11
TEM01=0.
DO 210 NS=1,MXNS
  IF(.NOT.(BTYPE(NS).EQ.1)) GO TO 210
  TEM01=TEM01+BTDC(NS)
210 CONTINUE
  BTDC1=TEM01
  TEM02=0.
  DO 220 NS=1,MXNS
    IF(.NOT.(BTYPE(NS).EQ.2)) GO TO 220
    TEM02=TEM02+BTDC(NS)
220 CONTINUE
  BTDC2=TEM02
  TEM03=0.
  DO 230 NS=1,MXNS
    IF(.NOT.(BTYPE(NS).EQ.3)) GO TO 230
    TEM03=TEM03+BTDC(NS)
230 CONTINUE
  BTDC3=TEM03
  TEM04=0.
  DO 240 NS=1,MXNS
    TEM04=TEM04+BTDC(NS)
240 CONTINUE
  BTDC4=TEM04
  TEM05=0.
  DO 250 NS=1,MXNS
    IF(.NOT.(BPLAT(NS).EQ.1)) GO TO 250
    TEM05=TEM05+BTDC(NS)
250 CONTINUE
  BTDC5=TEM05
  TEM06=0.
  DO 260 NS=1,MXNS
    IF(.NOT.(BPLAT(NS).EQ.2)) GO TO 260
    TEM06=TEM06+BTDC(NS)
260 CONTINUE
  BTDC6=TEM06
  TEM07=0.
  DO 270 NS=1,MXNS

```

```

        IF(.NOT.(BPLAT(NS).EQ.3)) GO TO 270
        TEM07=TEM07+BTDC(NS)
270  CONTINUE
        BTDCM=TEM07
        DTDC=STDC-BTDCT
        T1=STDCI/1000000.
        T2=STDCR/1000000.
        T3=STDC/1000000.
        T4=BTDCI/1000000.
        T5=BTDC/1000000.
        T6=BTDCS/1000000.
        T7=BTDCI/1000000.
        T8=BTDC/1000000.
        T9=BTDCG/1000000.
        T10=BTDCM/1000000.
        T11=DTDC/1000000.
        WRITE( 7, 6) T1,T2,T3,T4,T5,T6,T7,T8,T9,T11
        WRITE( 7, 7)
        TEMP1=(ISC+SECIC+SECIP+IMTRC+STDCI+IIMCI)/1000000.
        TEMP2=(OLC+AFC+RSC+ONMC+OFMC+SECRC+SECRP+RMTRC+STDCR+IIMCR)/
+ 1000000.
        TEMP3=(OLC+AFC+ISC+RSC+ONMC+OFMC+SECC+SECP+MTRC+STDC+IIMC)/
+ 1000000.
        TEMP4=(BOLC(1)+BAFC(1)+BISC(1)+BRSC(1)+BONMC(1)+BOFMC(1)+BSECC(1)+
+ BSECP(1)+BIIMC(1)+BTDCI)/1000000.
        TEMP5=(BOLC(2)+BAFC(2)+BISC(2)+BRSC(2)+BONMC(2)+BOFMC(2)+BSECC(2)+
+ BSECP(2)+BIIMC(2)+BTDC/1000000.
        TEMP6=(BOLC(3)+BAFC(3)+BISC(3)+BRSC(3)+BONMC(3)+BOFMC(3)+BSECC(3)+
+ BSECP(3)+BIIMC(3)+BTDCS)/1000000.
        TEMP7=(OLC+AFC+ISCB+RSC+ONMC+OFMCB+SECB+SECBP+IIMCB+BMTRC+BTDCI)/
+ 1000000.
        TEMP8=(BOLC(4)+BAFC(4)+BISC(4)+BRSC(4)+BONMC(4)+BOFMC(4)+BSECC(4)+
+ BSECP(4)+BIIMC(4)+BTDC/1000000.
        TEMP9=(BOLC(5)+BAFC(5)+BISC(5)+BRSC(5)+BONMC(5)+BOFMC(5)+BSECC(5)+
+ BSECP(5)+BIIMC(5)+BTDCG)/1000000.
        TEMP10=(BOLC(6)+BAFC(6)+BISC(6)+BRSC(6)+BONMC(6)+BOFMC(6)+BSECC(6)+
+ BSECP(6)+BIIMC(6)+BTDCM)/1000000.
        TEMP11=(ISC+OFMCD+SECDC+SECDP+IIMCD+DMTRC+DTDC)/1000000.
        WRITE( 7, 8) TEMP1,TEMP2,TEMP3,TEMP4,TEMP5,TEMP6,TEMP7,TEMP8,
+ TEMP9,TEMP11
        WRITE( 7, 9)
        TEM08=0.
        DO 280 NS=1,MXNS
            IF(.NOT.(BTYP(NS).EQ.1)) GO TO 280
            TEM08=TEM08+TNB(NS)
280  CONTINUE
        TNIB=TEM08

```

```

      TEM09=0.
      DO 290 NS=1,MXNS
        IF(.NOT.(BTYPE(NS).EQ.2)) GO TO 290
        TEM09=TEM09+TNB(NS)
290  CONTINUE
      TNCB=TEM09
      TEM10=0.
      DO 300 NS=1,MXNS
        IF(.NOT.(BTYPE(NS).EQ.3)) GO TO 300
        TEM10=TEM10+TNB(NS)
300  CONTINUE
      TNSB=TEM10
      TEM11=0.
      DO 310 NS=1,MXNS
        IF(.NOT.(BPLAT(NS).EQ.1)) GO TO 310
        TEM11=TEM11+TNB(NS)
310  CONTINUE
      TNAB=TEM11
      TEM12=0.
      DO 320 NS=1,MXNS
        IF(.NOT.(BPLAT(NS).EQ.2)) GO TO 320
        TEM12=TEM12+TNB(NS)
320  CONTINUE
      TNGB=TEM12
      TEM13=0.
      DO 330 NS=1,MXNS
        IF(.NOT.(BPLAT(NS).EQ.3)) GO TO 330
        TEM13=TEM13+TNB(NS)
330  CONTINUE
      TNMB=TEM13
      TEMPA=TNIB+TNCB+TNSB
      TEMPB=TNAB+TNGB+TNMB
      WRITE( 7, 10) TNIB,TNAB,TNCB,TNGB,TNSB,TEMPB,TEMPA
C
      RETURN
      END

```

SUBROUTINE OTAB3C

800827 113046947

C

C*****

C* SSS MOD LCR - 21 MAY 80 *

C* PRINTS OPERATION AND LOGISTICS SUPPORT *

C* COST ELEMENTS - PART 3 - FOR TERMINAL OUTPUT ONLY *

C*****

C

```

COMMON /PRNTXX/ PRNTXX
INTEGER PRNTXX
COMMON /AFC/ AFC
COMMON /BAFC/ BAFC(6)
COMMON /BIIMC/ BIIMC(6)
COMMON /BISC/ BISC(6)
COMMON /BOFMC/ BOFMC(6)
COMMON /BOLC/ BOLC(6)
COMMON /BONMC/ BONMC(6)
COMMON /BPLAT/ BPLAT(16)
INTEGER BPLAT
COMMON /BRSC/ BRSC(6)
COMMON /BSECC/ BSECC(6)
COMMON /BSECP/ BSECP(6)
COMMON /BTDC/ BTDC(16)
COMMON /BTDCA/ BTDCA
COMMON /BTDCG/ BTDCG
COMMON /BTDCM/ BTDCM
COMMON /BTDCT/ BTDCT
COMMON /BTYP/ BTYP(16)
INTEGER BTYP
COMMON /DMTRC/ DMTRC
COMMON /DTDC/ DTDC
COMMON /IIMC/ IIMC
REAL IIMC
COMMON /IIMCD/ IIMCD
REAL IIMCD
COMMON /IIMCI/ IIMCI
REAL IIMCI
COMMON /IIMCR/ IIMCR
REAL IIMCR
COMMON /IMTRC/ IMTRC
REAL IMTRC
COMMON /ISC/ ISC
REAL ISC
COMMON /ISCD/ ISCD
REAL ISCD
COMMON /MTRC/ MTRC
REAL MTRC
    
```

COMMON /MXNS/ MXNS
 COMMON /NS/ NS
 COMMON /OFMC/ OFMC
 COMMON /OFMCD/ OFMCD
 COMMON /OLC/ OLC
 COMMON /ONMC/ ONMC
 COMMON /RMTRC/ RMTRC
 COMMON /RSC/ RSC
 COMMON /SECC/ SECC
 COMMON /SECDC/ SECDC
 COMMON /SECDP/ SECDP
 COMMON /SECIC/ SECIC
 COMMON /SECIP/ SECIP
 COMMON /SECP/ SECP
 COMMON /SECRC/ SECRC
 COMMON /SECRP/ SECRP
 COMMON /STDC/ STDC
 COMMON /STDCI/ STDCI
 COMMON /STDCR/ STDCR
 COMMON /TNB/ TNB(16)

REAL IT1
 REAL IT10
 REAL IT11
 REAL IT2
 REAL IT3
 REAL IT8
 REAL IT9
 REAL JT1
 REAL JT11
 REAL JT2
 REAL JT3
 REAL KT1
 REAL KT10
 REAL KT11
 REAL KT2
 REAL KT3
 REAL KT8
 REAL KT9

1 FORMAT(1H1//9X,62HOUTPUT TABLE 3: OPERATION AND LOGISTICS SUPPORT
 + COST ELEMENTS/23X,33H(IN MILLIONS OF CONSTANT DOLLARS)//18X,1H|,2
 +7X,25H| AIR GROUND | DEPOT/5X,66HCOST ELEMENT | INITIAL RECU
 +RRING TOTAL | EASES BASES | TOTAL/18X,1H|,27X,1H|,16X,1H|/4X,
 +67(1H-)/18X,1H|,27X,1H|,16X,1H|)
 2 FORMAT(1X,18HOPERATIONS LABOR |,9X,2(F6.2,3X),1H|,1X,2(F6.2,1X),1X
 +,1H|)
 3 FORMAT(1X,10HADDED FUEL,7X,1H|,9X,2(F6.2,3X),1H|,1X,2(F6.2,1X),1X,
 +1H|)

```

4 FORMAT(1X,13HINITIAL SPARES |,1X,F6.2,11X,F6.2,3X,1H|,1X,2(F6.2,
+1X),1X,1H|,1X,F6.2)
5 FORMAT(1X,18HREPLACE. SPARES |,9X,2(F6.2,3X),1H|,1X,2(F6.2,1X),1X
+,1H|)
6 FORMAT(1X,18HON-EQUIP. MAINT. |,9X,2(F6.2,3X),1H|,1X,2(F6.2,1X),1X
+,1H|)
7 FORMAT(1X,18HOFF-EQUIP. MAINT. |,9X,2(F6.2,3X),1H|,1X,2(F6.2,1X),1X
+,1H|,1X,F6.2)
8 FORMAT(1X,18HSUPPORT EQUIPMENT|,27X,1H|,16X,1H|)
9 FORMAT(1X,18H COMMON |,1X,F6.2,2X,2(F6.2,3X),1H|,1X,2(F6.
+2,1X),1X,1H|,1X,F6.2)
10 FORMAT(1X,18H PECULIAR |,1X,F6.2,2X,2(F6.2,3X),1H|,1X,2(F6.
+2,1X),1X,1H|,1X,F6.2)
11 FORMAT(1X,18HINVENTORY MANAG. |,1X,F6.2,2X,2(F6.2,3X),1H|,1X,2(F6.
+2,1X),1X,1H|,1X,F6.2)
12 FORMAT(1X,18HMAINT. TRAINING |,1X,F6.2,2X,2(F6.2,3X),1H|,16X,1H|,
+1X,F6.2)
13 FORMAT(1X,18HTECH. DATA |,1X,F6.2,2X,2(F6.2,3X),1H|,1X,2(F6.
+2,1X),1X,1H|,1X,F6.2)
14 FORMAT(4X,74(1H-))
15 FORMAT(6X,6HTOTALS,6X,1H|,1X,F6.2,2X,2(F6.2,3X),1H|,1X,2(F6.2,1X),
+1X,1H|,1X,F6.2///)
16 FORMAT(24X,16HNUMBER OF BASES:/24X,16(1H-)//28X,9HINDEP. = ,F5.0,9
+X,15HAIR BASES = ,F5.0//28X,9HCIMF = ,F5.0,9X,15HGROUND BASES
+ = ,F5.0/66X,5(1H-)/28X,9HSATEL. = ,F5.0,9X,15HTOTAL = ,F5.
+0/37X,5(1H-)/28X,9HTOTAL = ,F5.0)

```

C
C
C

```

IF(PRNTXX.NE.1) WRITE(06, 1)
AT1=OLC/1000000.
AT5=BOLC(4)/1000000.
AT6=BOLC(5)/1000000.
AT7=BOLC(6)/1000000.
IF(PRNTXX.NE.1) WRITE(06, 2) AT1,AT1,AT5,AT6
BT1=AFC/1000000.
BT5=BAFC(4)/1000000.
BT6=BAFC(5)/1000000.
BT7=BAFC(6)/1000000.
IF(PRNTXX.NE.1) WRITE(06, 3) BT1,BT1,BT5,BT6
CT1=ISC/1000000.
CT6=BISC(4)/1000000.
CT7=BISC(5)/1000000.
CT8=BISC(6)/1000000.
CT9=ISCD/1000000.
IF(PRNTXX.NE.1) WRITE(06, 4) CT1,CT1,CT6,CT7,CT9
DT1=RSC/1000000.

```

```

DT5=BRSC(4)/1000000.
DT6=BRSC(5)/1000000.
DT7=BRSC(6)/1000000.
IF(PRNTXX.NE.1) WRITE(06, 5) DT1,DT1,DT5,DT6
ET1=ONMC/1000000.
ET5=BONMC(4)/1000000.
ET6=BONMC(5)/1000000.
ET7=BONMC(6)/1000000.
IF(PRNTXX.NE.1) WRITE(06, 6) ET1,ET1,ET5,ET6
FT1=OFMC/1000000.
FT6=BOFMC(4)/1000000.
FT7=BOFMC(5)/1000000.
FT8=BOFMC(6)/1000000.
FT9=OFMCD/1000000.
IF(PRNTXX.NE.1) WRITE(06, 7) FT1,FT1,FT6,FT7,FT9
IF(PRNTXX.NE.1) WRITE(06, 8)
GT1=SECIC/1000000.
GT2=SECRC/1000000.
GT3=SECC/1000000.
GT8=BSECC(4)/1000000.
GT9=BSECC(5)/1000000.
GT10=BSECC(6)/1000000.
GT11=SECD/1000000.
IF(PRNTXX.NE.1) WRITE(06, 9) GT1,GT2,GT3,GT8,GT9,GT11
HT1=SECIP/1000000.
HT2=SECRP/1000000.
HT3=SECP/1000000.
HT8=BSECP(4)/1000000.
HT9=BSECP(5)/1000000.
HT10=BSECP(6)/1000000.
HT11=SECDP/1000000.
IF(PRNTXX.NE.1) WRITE(06,10) HT1,HT2,HT3,HT8,HT9,HT11
IT1=IIMCI/1000000.
IT2=IIMCR/1000000.
IT3=IIMC/1000000.
IT8=BIIMC(4)/1000000.
IT9=BIIMC(5)/1000000.
IT10=BIIMC(6)/1000000.
IT11=IIMCD/1000000.
IF(PRNTXX.NE.1) WRITE(06,11) IT1,IT2,IT3,IT8,IT9,IT11
JT1=IMTRC/1000000.
JT2=RMTRC/1000000.
JT3=MTRC/1000000.
JT11=DMTRC/1000000.
IF(PRNTXX.NE.1) WRITE(06,12) JT1,JT2,JT3,JT11
TEM01=0.
DO 210 NS=1,MXNS

```

```

        IF(.NOT.(BPLAT(NS).EQ.1)) GO TO 210
        TEM01=TEM01+BTDC(NS)
210  CONTINUE
        BTDCA=TEM01
        TEM02=0.
        DO 220 NS=1,MXNS
            IF(.NOT.(BPLAT(NS).EQ.2)) GO TO 220
            TEM02=TEM02+BTDC(NS)
220  CONTINUE
        BTDCG=TEM02
        TEM03=0.
        DO 230 NS=1,MXNS
            IF(.NOT.(BPLAT(NS).EQ.3)) GO TO 230
            TEM03=TEM03+BTDC(NS)
230  CONTINUE
        BTDCM=TEM03
        DTDC=STDC-BTDCT
        KT1=STDCI/1000000.
        KT2=STDCR/1000000.
        KT3=STDC/1000000.
        KT8=BTDCA/1000000.
        KT9=BTDCG/1000000.
        KT10=BTDCM/1000000.
        KT11=DTDC/1000000.
        IF(PRNTXX.NE.1) WRITE(06,13) KT1,KT2,KT3,KT8,KT9,KT11
        IF(PRNTXX.NE.1) WRITE(06,14)
        TEMP1=(ISC+SECIC+SECIP+IMTRC+STDCI+IIMCI)/1000000.
        TEMP2=(OLC+AFC+RSC+ONMC+OFMC+SECR+SECRP+RMTRC+STDCR+IIMCR)/
+ 1000000.
        TEMP3=(OLC+AFC+ISC+RSC+ONMC+OFMC+SECC+SECP+MTRC+STDC+IIMC)/
+ 1000000.
        TEMP8=(BOLC(4)+BAFC(4)+BISC(4)+BRSC(4)+BONMC(4)+BOFMC(4)+BSECC(4)+
+ BSECP(4)+BIIMC(4)+BTDCA)/1000000.
        TEMP9=(BOLC(5)+BAFC(5)+BISC(5)+BRSC(5)+BONMC(5)+BOFMC(5)+BSECC(5)+
+ BSECP(5)+BIIMC(5)+BTDCG)/1000000.
        TEMP10=(BOLC(6)+BAFC(6)+BISC(6)+BRSC(6)+BONMC(6)+BOFMC(6)+BSECC(6)+
+ BSECP(6)+BIIMC(6)+BTDCM)/1000000.
        TEMP11=(ISCD+OFMCD+SECD+SECDP+IIMCD+DMTRC+DTDC)/1000000.
        IF(PRNTXX.NE.1) WRITE(06,15) TEMP1,TEMP2,TEMP3,TEMP8,TEMP9,TEMP11
        TEM04=0.
        DO 240 NS=1,MXNS
            IF(.NOT.(BTYPE(NS).EQ.1)) GO TO 240
            TEM04=TEM04+TNB(NS)
240  CONTINUE
        TNIB=TEM04
        TEM05=0.
        DO 250 NS=1,MXNS

```

```

        IF(.NOT.(BTYPE(NS).EQ.2)) GO TO 250
        TEM05=TEM05+TNB(NS)
250 CONTINUE
        TNCB=TEM05
        TEM06=0.
        DO 260 NS=1,MXNS
            IF(.NOT.(BTYPE(NS).EQ.3)) GO TO 260
            TEM06=TEM06+TNB(NS)
260 CONTINUE
        TNSB=TEM06
        TEM07=0.
        DO 270 NS=1,MXNS
            IF(.NOT.(BPLAT(NS).EQ.1)) GO TO 270
            TEM07=TEM07+TNB(NS)
270 CONTINUE
        TNAB=TEM07
        TEM08=0.
        DO 280 NS=1,MXNS
            IF(.NOT.(BPLAT(NS).EQ.2)) GO TO 280
            TEM08=TEM08+TNB(NS)
280 CONTINUE
        TNGB=TEM08
        TEM09=0.
        DO 290 NS=1,MXNS
            IF(.NOT.(BPLAT(NS).EQ.3)) GO TO 290
            TEM09=TEM09+TNB(NS)
290 CONTINUE
        TNMB=TEM09
        TEMPA=TNIB+TNCB+TNSB
        TEMPB=TNAB+TNGB+TNMB
        IF(PRNTXX.NE.1) WRITE(06,16) TNIB,TNAB,TNCB,TNGB,TNSB,TEMPB,TEMPA
C
        RETURN
        END

```

SUBROUTINE OTAB4A

```

C
C***** 800827 113223743 *****
C* SSS MOD LCR - 21 MAY 80 *
C* PRINTS ITEM-SPECIFIC COSTS AND *
C* MAINTENANCE CHARACTERISTICS *
C*****
C
COMMON /PRNTXX/ PRNTXX
INTEGER PRNTXX
COMMON /FULLXX/ FULLXX
INTEGER FULLXX
COMMON /FAIL/ FAIL(999,16)
COMMON /FPLT/ FPLT(999)
COMMON /FPM/ FPM(999)
COMMON /I/ I
COMMON /IIMCA/ IIMCA(999)
REAL IIMCA
COMMON /INO/ INO(999)
COMMON /INOUN/ INOUN(999,24)
REAL INOUN
COMMON /ISCA/ ISCA(999)
REAL ISCA
COMMON /LRU/ LRU(999)
COMMON /MTRCI/ MTRCI(999)
REAL MTRCI
COMMON /MXI/ MXI
COMMON /MXNP/ MXNP
COMMON /MXNS/ MXNS
COMMON /NITEM/ NITEM(999,10)
REAL NITEM
COMMON /NP/ NP
COMMON /NPLT/ NPLT(10,16)
REAL NPLT
COMMON /NS/ NS
COMMON /OFMCA/ OFMCA(999)
COMMON /ONMCA/ ONMCA(999)
COMMON /PIUP/ PIUP
COMMON /RSCA/ RSCA(999)
COMMON /SECI/ SECI(999)
COMMON /TDC/ TDC(999)
COMMON /TIAC/ TIAC(999)
COMMON /TNB/ TNB(16)
1 FORMAT(1H1/32X,68HOUTPUT TABLE 4A: ITEM-SPECIFIC MAINTENANCE AND C
+OSTS CHARACTERISTICS//46X,40H(COSTS IN THOUSANDS OF CONSTANT DOLLA
+RS))
2 FORMAT(55X,11H(CONTINUED)//)

```

```

3  FORMAT(98X,5HTOTAL,17X,5HTOTAL/29X,3HLRU,22X,3HON-,6X,4HOFF-,22X,3
+OHITEM      ITEM      CORR. MAINT.,1X,7HSUPPORT/2X,4HITEM,23X,5HINDI
+-,2X,7HINITIAL,2X,5HREPL.,4X,6HEQUIP.,3X,6HEQUIP.,3X,6HMAINT.,4X,5
+HTECH.,2X,7HINVENT.,2X,7HSUPPORT,2X,9HCOST/FAIL,4X,6HCOST +/2X,5HI
+NDEX,2X,9HITEM NAME,11X,5HCATOR,2X,6HSPARES,3X,6HSPARES,3X,6HMAINT
+.,3X,6HMAINT.,3X,8HTRAINING,2X,6HORDERS,1X,5HMGMT.,4X,4HCOST,5X,11
+H(RSCA+ONMCA,2X,4HSECI/2X,3H(I),24X,5H(LRU),2X,6H(ISCA),3X,6H(RSCA
+),3X,7H(ONMCA),2X,7H(OFMCA),2X,7H(MTRCI),3X,5H(TDC),2X,7H(IIMCA),1
+2X,7H(OFMCA),5X,6H(TIAC)/)
4  FORMAT(1X,I3,3X,20A1,3X,I2,2X,8(F8.1,1X),3X,F6.3,3X,F8.1)
5  FORMAT(3X,22(1H-),9X,8(8(1H-),1X),14X,8(1H-)/3X,22HCOST TOTALS OVE
+R ITEMS,9X,8(F8.1,1X),10X,F11.1)

```

C

C

C

C.....ONLY PRINT THIS TABLE IF FULL OFF-LINE OUTPUT WAS REQUESTED
IF(PRNTXX.EQ.0.OR.FULLXX.EQ.0) RETURN

C

TEMP6=0.

TEMP7=0.

TEMP8=0.

TEMP9=0.

TEMP9A=0.

TEMP9B=0.

TEMP10=0.

TEMP11=0.

TEMP16=0.

IPAGE=40

IFLAG=1

DO 260 IXXX1=1,MXI

I=INO(IXXX1)

IF(.NOT.(IPAGE.EQ.40)) GO TO 220

WRITE(7, 1)

IPAGE=1

IF(.NOT.(IFLAG NE.1)) GO TO 210

WRITE(7, 2)

210 CONTINUE

WRITE(7, 3)

220 CONTINUE

TEMP1=ISCA(I)/1000.

TEMP2=RSCA(I)/1000.

TEMP3=ONMCA(I)/1000.

TEMP4=OFMCA(I)/1000.

TEMP4A=MTRCI(I)/1000.

TEMP4B=TDC(I)/1000.

TEMP5=IIMCA(I)/1000.

TIC=TEMP1+TEMP2+TEMP3+TEMP4+TEMP4A+TEMP4B+TEMP5

```

TEM01=0.
DO 230 NS=1,MXNS
    TEM01=TEM01+TNB(NS)*FAIL(I,NS)
230 CONTINUE
    FPM(I)=TEM01
    FPLT(I)=12.*PIUP*FPM(I)
    CMCF=(RSCA(I)+ONMCA(I)+OFMCA(I))/1000./FPLT(I)
    TEM03=0.
    DO 250 NS=1,MXNS
        TEM02=0.
        DO 240 NP=1,MXNP
            TEM02=TEM02+NPLT(NP,NS)*NITEM(I,NP)
240 CONTINUE
            TEM03=TEM03+TNB(NS)*TEM02
250 CONTINUE
        STNI=TEM03
        TIAC(I)=TIC+SECI(I)/1000.
        WRITE( 7, 4) I,(INOUN(I,K1),K1=1,20),LRU(I),TEMP1,TEMP2,TEMP3,
+    TEMP4,TEMP4A,TEMP4B,TEMP5,TIC,CMCF,TIAC(I)
        TEMP6=TEMP6+TEMP1
        TEMP7=TEMP7+TEMP2
        TEMP8=TEMP8+TEMP3
        TEMP9=TEMP9+TEMP4
        TEMP9A=TEMP9A+TEMP4A
        TEMP9B=TEMP9B+TEMP4B
        TEMP10=TEMP10+TEMP5
        TEMP11=TEMP11+TIC
        TEMP16=TEMP16+TIAC(I)
        IPAGE=IPAGE+1
        IFLAG=0
260 CONTINUE
        WRITE( 7, 5) TEMP6,TEMP7,TEMP8,TEMP9,TEMP9A,TEMP9B,TEMP10,TEMP11,
+    TEMP16
C
    RETURN
    END

```

SUBROUTINE OTAB4B

C

800827 113314325

C*****

C* SSS MOD LCR - 21 MAY 80 *

C* PRINTS ITEM-SPECIFIC COSTS AND *

C* MAINTENANCE CHARACTERISTICS *

C*****

C

COMMON /PENTXX/ PRNTXX

INTEGER PRNTXX

COMMON /FULLXX/ FULLXX

INTEGER FULLXX

COMMON /BS/ BS(999)

COMMON /DS/ DS(999)

COMMON /FAIL/ FAIL(999,16)

COMMON /FPLT/ FPLT(999)

COMMON /FPM/ FPM(999)

COMMON /I/ I

COMMON /INO/ INO(999)

COMMON /LUP/ LUP(999)

REAL LUP

COMMON /MXI/ MXI

COMMON /MXNP/ MXNP

COMMON /MXNS/ MXNS

COMMON /NITEM/ NITEM(999,10)

REAL NITEM

COMMON /NP/ NP

COMMON /NPLT/ NPLT(10,16)

REAL NPLT

COMMON /NS/ NS

COMMON /PIUP/ PIUP

COMMON /TNB/ TNB(16)

COMMON /TOTPQ/ TOTPQ(999)

1 FORMAT(1H1/37X,71HOUTPUT TABLE 4B: SYSTEM-WIDE ITEM COSTS AND MAIN
+TENANCE CHARACTERISTICS/)

2 FORMAT(58X,11H(CONTINUED)/)

3 FORMAT(11X,6HSYSTEM,5X,9HTOTAL NO.,3X,9HTOTAL NO.,11X,6HNO. OF,5X,
+11HNO. OF LIFE,17X,10HPRODUCTION/11X,6HNO. OF,5X,10HOF INITIAL,2X,
+10HOF INITIAL,2X,6HSYSTEM,2X,4HITEM,7X,11HCYCLE FAILS,5X,7HLEARNED
+,5X,8HCONTRACT/2X,4HITEM,5X,9HINSTALLED,2X,4HBASE,8X,5HDEPOT,7X,6H
+NO. OF,2X,9HFAILS PER,2X,10H(NO RIP OR,6X,4HUNIT,8X,11HPROCUREMENT
+/2X,5HINDEX,4X,5HITEMS,6X,6HSPARES,6X,6HSPARES,6X,5HITEMS,3X,6HMON
+TH ,5X,12HFALSE PULLS),4X,4HCOST,8X,8HQUANTITY//)4 FORMAT(3X,13,5X,F7.0,4X,F5.0,7X,F5.0,6X,F7.0,2X,F7.2,5X,F7.0,8X,F6
+.0,9X,F7.0)

C

C

```

C
C.....ONLY PRINT THIS TABLE IF FULL OFF-LINE OUTPUT WAS REQUESTED
      IF(PRNTXX.EQ.0.OR.FULLXX.EQ.0) RETURN
C
      IPAGE=40
      IFLAG=1
      DO 260 IXXX1=1,MXI
        I=INO(IXXX1)
        IF(.NOT.(IPAGE.EQ.40)) GO TO 220
        WRITE( 7, 1)
        IPAGE=1
        IF(.NOT.(IFLAG.NE.1)) GO TO 210
        WRITE( 7, 2)
210      CONTINUE
        WRITE( 7, 3)
220      CONTINUE
        TEM01=0.
        DO 230 NS=1,MXNS
          TEM01=TEM01+TNB(NS)*FAIL(I,NS)
230      CONTINUE
        FPM(I)=TEM01
        FPLT(I)=12.*PIUP*FPM(I)
        TEM03=0.
        DO 250 NS=1,MXNS
          TEM02=0.
          DO 240 NP=1,MXNP
            TEM02=TEM02+NPLT(NP,NS)*NITEM(I,NP)
240          CONTINUE
          TEM03=TEM03+TNB(NS)*TEM02
250          CONTINUE
          STNI=TEM03
          SNOI=STNI+BS(I)+DS(I)
          WRITE( 7, 4) I,STNI,BS(I),DS(I),SNOI,FPM(I),FPLT(I),LUP(I),
+          TOTPQ(I)
          IPAGE=IPAGE+1
          IFLAG=0
260      CONTINUE
C
      RETURN
      END

```

SUBROUTINE OTAB4C

```

C
C*****800827 113346937*****
C* PRINTS ITEM-SPECIFIC COSTS AND *
C* MAINTENANCE CHARACTERISTICS *
C*****
C
COMMON /PRNTXX/ PRNTXX
INTEGER PRNTXX
COMMON /FULLXX/ FULLXX
INTEGER FULLXX
COMMON /FPLT/ FPLT(999)
COMMON /FPM/ FPM(999)
COMMON /I/ I
COMMON /INO/ INO(999)
COMMON /LRU/ LRU(999)
COMMON /MXI/ MXI
COMMON /OFMCA/ OFMCA(999)
COMMON /ONMCA/ ONMCA(999)
COMMON /RSCA/ RSCA(999)
1 FORMAT(1H1/32X,56HOUTPUT TABLE 4C: SYSTEM-WIDE MAINTENANCE CHARACT
+ERISTICS///)
2 FORMAT(23X,35HAVERAGE CORRECTIVE MAINTENANCE COST,10X,30HTOTAL NUM
+BERS OF LRU FAILURES:/26X,20HPER FAILURE ($K) FOR//28X,4HLRUS,8X,4
+HSRUS,27X,7HMONTHLY,3X,8HLIFETIME//26X,F6.3,6X,F6.3,27X,F7.0,2X,F9
+.0)
C
C
C
C.....ONLY PRINT THIS TABLE IF FULL OFF-LINE OUTPUT WAS REQUESTED
IF(PRNTXX.EQ.0.OR.FULLXX.EQ.0) RETURN
C
WRITE( 7, 1)
TEM01=0.
DO 210 IXXX1=1,MXI
I=INO(IXXX1)
IF(.NOT.(LRU(I).EQ.1)) GO TO 210
TEM01=TEM01+FPM(I)
210 CONTINUE
TEMP14=TEM01
TEM02=0.
DO 220 IXXX1=1,MXI
I=INO(IXXX1)
IF(.NCT.(LRU(I).EQ.1)) GO TO 220
TEM02=TEM02+FPLT(I)
220 CONTINUE
TEMP15=TEM02

```

```

TEM03=0.
DO 230 IXXX1=1,MXI
  I=INO(IXXX1)
  IF(.NOT.(LRU(I).EQ.1)) GO TO 230
  TEM03=TEM03+RSCA(I)+ONMCA(I)+OFMCA(I)
230 CONTINUE
  TEMP12=TEM03
  TEM04=0.
  DO 240 IXXX1=1,MXI
    I=INO(IXXX1)
    IF(.NOT.(LRU(I).EQ.1)) GO TO 240
    TEM04=TEM04+FPLT(I)
240 CONTINUE
  TEMP12=TEMP12/(1000.*TEM04)
  TEM05=0.
  DO 250 IXXX1=1,MXI
    I=INO(IXXX1)
    IF(.NOT.(LRU(I).EQ.0)) GO TO 250
    TEM05=TEM05+RSCA(I)+ONMCA(I)+OFMCA(I)
250 CONTINUE
  TEMP13=TEM05
  TEM06=0.
  DO 260 IXXX1=1,MXI
    I=INO(IXXX1)
    IF(.NOT.(LRU(I).EQ.0)) GO TO 260
    TEM06=TEM06+FPLT(I)
260 CONTINUE
  TEMP13=TEMP13/(1000.*TEM06)
  WRITE( 7, 2) TEMP12,TEMP13,TEMP14,TEMP15
C
  RETURN
  END

```

SUBROUTINE OTAB5

C 800827 113410567
 C*****
 C* SSS MOD LCR - 21 MAY 80 *
 C* PRINTS SUPPORT EQUIPMENT REQUIREMENTS AND COSTS *
 C*****
 C

COMMON /PRNTXX/ PRNTXX
 INTEGER PRNTXX
 COMMON /FULLXX/ FULLXX
 INTEGER FULLXX
 COMMON /BPLAT/ BPLAT(16)
 INTEGER BPLAT
 COMMON /BTYPE/ BTYPE(16)
 INTEGER BTYPE
 COMMON /CSE/ CSE(250)
 COMMON /L/ L
 COMMON /MSE/ MSE(250)
 REAL MSE
 COMMON /MXL/ MXL
 COMMON /MXNS/ MXNS
 COMMON /NS/ NS
 COMMON /NSEB/ NSEB(250,16)
 REAL NSEB
 COMMON /NSED/ NSED(250)
 REAL NSED
 COMMON /PIUP/ PIUP
 COMMON /SEDO/ SEDV(250)
 COMMON /SEINO/ SEINO(250)
 INTEGER SEINO
 COMMON /SENOUN/ SENOUN(250,20)
 COMMON /SETDC/ SETDC(250)
 COMMON /TNB/ TNB(16)
 COMMON /TUCTDC/ TUCTDC

1 FORMAT(1H1,27X,56HOUTPUT TABLE 5: SUPPORT EQUIPMENT REQUIREMENTS A
 +ND COSTS/43X,18H(COSTS IN DOLLARS)///30X,93H* SUPPORT EQUIPMENT UN
 +ITS REQUIRED AT: * SYSTEM * UNIT * TECH. * SE * SYSTEM
 + * /30X,1H*,38(1H-),54H* TOTAL * LIFE- * ORDER * DEVMT * LIF
 +E- * /3X,2HSE,25X,93H* INDEP CIMF * AIR GROUND * THE
 + * REQUIRED * TIME * COST * COST * TIME * /2X,5HINDE
 +X,2X,20HSUPPORT EQUIP. NAME ,1X,93H* BASES BASES * BASES BASES *
 +DEPOT * UNITS * COST * * * COST (\$K) * /3X,
 +3H(L),24X,93H* * * * (1) *
 +(2) * (3) * (4) * (1)*(2)+(3)+(4)*///)
 2 FORMAT(2X,I3,3X,20A1,2X,F6.0,1X,F6.0,1X,F6.0,1X,F6.0,2X,F6.1,5X,F7
 +.0,2X,F8.0,1X,F8.0,1X,F8.0,4X,F10.0)
 3 FORMAT(///95X,11HSUBTOTAL = ,4X,F10.0)

```

4 FORMAT(//74X,32HUCT SOFTWARE DEVELOPMENT COST = ,4X,F10.0)
5 FORMAT(109X,14(1H-)/)
6 FORMAT(75X,31HTOTAL SUPPORT EQUIPMENT COST = ,1X,F13.0)

```

C
C
C
C
C

```

C.....ONLY PRINT THIS TABLE IF FULL OFF-LINE OUTPUT WAS REQUESTED
IF(PRNTXX.EQ.0.OR.FULLXX.EQ.0) RETURN

```

```

C
  T10=0.
  WRITE( 7, 1)
  DO 260 IXXX1=1,MXL
    L=SEINO(IXXX1)
    TEM01=0.
    DO 210 NS=1,MXNS
      IF(.NOT.(BTYPE(NS).EQ.1)) GO TO 210
      TEM01=TEM01+TNB(NS)*NSEB(L,NS)
210  CONTINUE
      T1=TEM01
      TEM02=0.
      DO 220 NS=1,MXNS
        IF(.NOT.(BTYPE(NS).EQ.2)) GO TO 220
        TEM02=TEM02+TNB(NS)*NSEB(L,NS)
220  CONTINUE
        T2=TEM02
        TEM03=0.
        DO 230 NS=1,MXNS
          IF(.NOT.(BPLAT(NS).EQ.1)) GO TO 230
          TEM03=TEM03+TNB(NS)*NSEB(L,NS)
230  CONTINUE
          T3=TEM03
          TEM04=0.
          DO 240 NS=1,MXNS
            IF(.NOT.(BPLAT(NS).EQ.2)) GO TO 240
            TEM04=TEM04+TNB(NS)*NSEB(L,NS)
240  CONTINUE
            T4=TEM04
            TEM05=0.
            DO 250 NS=1,MXNS
              IF(.NOT.(BPLAT(NS).EQ.3)) GO TO 250
              TEM05=TEM05+TNB(NS)*NSEB(L,NS)
250  CONTINUE
            T5=TEM05
            T6=NSED(L)
            T7=T1+T2+T6
            T8=CSE(L)*(1.+PIUP*MSE(L))
            T8A=SETDC(L)

```

```

      T8B=SEDV(L)
      T9=(T7*T8+T8A+T8B)/1000.
      T10=T10+T9
      T11=T10+TUCTDC/1000.
      WRITE( 7, 2) L,(SENOUN(L,K1),K1=1,20),T1,T2,T3,T4,T6,T7,T8,T8A,
+      T8B,T9
260 CONTINUE
      WRITE( 7, 3) T10
      T12=TUCTDC/1000.
      WRITE( 7, 4) T12
      WRITE( 7, 5)
      WRITE( 7, 6) T11
C
      RETURN
      END

```

SUBROUTINE OTAB6

```

C
C***** 800827 113442053
C* SSS MOD LCR - 21 MAY 80 *
C* PRINTS PLATFORM/TERMINAL FAILURE RATE *
C*****
C
COMMON /PRNTXX/ PRNTXX
INTEGER PRNTXX
COMMON /FULLXX/ FULLXX
INTEGER FULLXX
COMMON /APFH/ APFH(10,3)
COMMON /I/ I
COMMON /INO/ INO(999)
COMMON /KFAC/ KFAC(4)
REAL KFAC
COMMON /LE/ LE(10)
COMMON /LO/ LO(16)
COMMON /LRU/ LRU(999)
COMMON /MTBMI/ MTBMI(999,4)
REAL MTBMI
COMMON /MXI/ MXI
COMMON /MXNP/ MXNP
COMMON /MXNS/ MXNS
COMMON /NITEM/ NITEM(999,10)
REAL NITEM
COMMON /NP/ NP
COMMON /NPLT/ NPLT(10,16)
REAL NPLT
COMMON /NS/ NS
COMMON /NTRMP/ NTRMP(10)
REAL NTRMP
COMMON /PNOUN/ PNOUN(10,12)
COMMON /TERMC/ TERMC(10)
COMMON /TFAC/ TFAC(10)
COMMON /TNB/ TNB(16)
COMMON /XFR/ XFR
1 FORMAT(1H1,22X,51HOUTPUT TABLE 6: PLATFORM/TERMINAL FAILURE RATE D
+ATA ///24X,55HSYSTEM PME FAILS*/ FAILS*/ FAILS*/
+FAILS*/ /88H PLAT- NO. OF TERM. MONTH M
+ONTH MIL.HRS MIL.HRS PROD. COST/86H FORM
+PLAT- PER PER PER PER PER PER PLAT /8
+7H INDEX PLATFORM NAME FORMS PLAT PLAT TERM. PLAT
+ TERMINAL TYPE ($K) /38H (NP) (NTRMP)
+ /)
2 FORMAT(3X,I2,4X,I2A1,3X,F6.0,2X,F5.2,3X,F6.3,2X,F6.3,2X,F7.0,2X,F7
+.0,5X,F8.0)

```

3 FORMAT(////4X,65H* THESE FAILURES INCLUDE EVERY EVENT REQUIRING MA
 +INTENANCE ACTION/6X,65H(INCLUDING REPAIR-IN-PLACE). THEY DO NOT I
 +NCLUDE FALSE PULLS.)

C

C

C

C.....ONLY PRINT THIS TABLE IF FULL OFF-LINE OUTPUT WAS REQUESTED
 IF(PRNTXX.EQ.0.OR.FULLXX.EQ.0) RETURN

C

```

WRITE( 7, 1)
DO 250 NP=1,MXNP
  TEM01=0.
  DO 210 NS=1,MXNS
    TEM01=TEM01+TNB(NS)*NPLT(NP,NS)
210  CONTINUE
    TPLT=TEM01
    PFPM=0.
    TEM02=0.
    DO 220 IXXX2=1,MXI
      I=INO(IXXX2)
      IF(.NOT.(LRU(I).EQ.1.AND.NITEM(I,NP).GT..000001)) GO TO 220
      TEM02=TEM02+NITEM(I,NP)/MTBMI(I,LE(NP))
220  CONTINUE
    TEM03=0.
    DO 230 NS=1,MXNS
      TEM03=TEM03+NPLT(NP,NS)*TNB(NS)*APFH(NP,LO(NS))
230  CONTINUE
    PFPM=TEM02*TEM03*TFAC(NP)*KFAC(LE(NP))*XFR/TPLT
    TFPM=PFPM/NTRMP(NP)
    PFPMH=0.
    TEM04=0.
    DO 240 IXXX2=1,MXI
      I=INO(IXXX2)
      IF(.NOT.(LRU(I).EQ.1.AND.NITEM(I,NP).GT..000001)) GO TO 240
      TEM04=TEM04+NITEM(I,NP)/MTBMI(I,LE(NP))
240  CONTINUE
    PFPMH=TEM04*TFAC(NP)*KFAC(LE(NP))*1000000.*XFR
    TFPMH=PFPMH/NTRMP(NP)
    T1=TERMC(NP)/1000.
    WRITE( 7, 2) NP,(PNOUN(NP,K1),K1=1,12),TPLT,NTRMP(NP),PFPM,TFPM,
      + PFPMH,TFPMH,T1
250  CONTINUE
    WRITE( 7, 3)
C
RETURN
END

```

SUBROUTINE OTAB7

C 800827 113508624
 C*****
 C* BASELINE CHANGES *
 C* PRINTS MANPOWER REQUIREMENTS *
 C*****
 C

COMMON /PRNTXX/ PRNTXX
 INTEGER PRNTXX
 COMMON /FULLXX/ FULLXX
 INTEGER FULLXX
 COMMON /ABMHYD/ ABMHYD(16)
 COMMON /ABMHYD/ ABMHYD(16)
 COMMON /ABMP/ ABMP(16)
 COMMON /ADMP/ ADMP
 COMMON /BCMh/ BCMH(999)
 COMMON /BMF/ BMF
 COMMON /BMH/ BMH(999)
 COMMON /COND/ COND(999)
 COMMON /DMF/ DMF
 COMMON /DMH/ DMH(999)
 COMMON /FAIL/ FAIL(999,16)
 COMMON /FPR/ FPR(999)
 COMMON /HPD1/ HPD1
 INTEGER HPD1
 COMMON /HPD2/ HPD2
 INTEGER HPD2
 COMMON /I/ I
 COMMON /INO/ INO(999)
 COMMON /LRU/ LRU(999)
 COMMON /MRF/ MRF
 REAL MRF
 COMMON /MRO/ MRO
 REAL MRO
 COMMON /MXI/ MXI
 COMMON /MXNS/ MXNS
 COMMON /NHI/ NHI(999)
 COMMON /NRTS/ NRTS(999)
 REAL NRTS
 COMMON /NS/ NS
 COMMON /QTP1/ QTP1
 INTEGER QTP1
 COMMON /QTP2B/ QTP2B
 INTEGER QTP2B
 COMMON /QTP2D/ QTP2D
 INTEGER QTP2D
 COMMON /RIP/ RIP(999)

```

COMMON /RTS/ RTS(999)
COMMON /SAMHY/ SAMHY
COMMON /SAMP/ SAMP
COMMON /SR/ SR
COMMON /TABMHY/ TABMHY(16)
COMMON /TABMP/ TABMP(16)
COMMON /TIME1/ TIME1(999)
INTEGER TIME1
COMMON /TMPYT1/ TMPYT1
COMMON /TMPYT2/ TMPYT2
COMMON /TNB/ TNB(16)
COMMON /TORB/ TORB
COMMON /TORD/ TORD
COMMON /TR/ TR
COMMON /T2BA/ T2BA
COMMON /T2DA/ T2DA
COMMON /XFPR/ XFPR
REAL NHNRT
REAL NHRT

```

```

1 FORMAT(1H1,22X,38HOUTPUT TABLE 7:  MANPOWER REQUIREMENTS//2X,4HBA
+SE,5X,12HMANHOURS PER,10X,14HTOTAL MANYEARS,6X,14HTOTAL MANHOURS,6
+X,14HTOTAL MANYEARS/2X,4HTYPE,5X,13HYEAR PER BASE,7X,17HPER YEAR P
+ER BASE,3X,18HPER YEAR/BASE TYPE,2X,18HPER YEAR/BASE TYPE/2X,4H(NS
+)/9X,20HMAINT.  MGMT. DATA//)
2 FORMAT(3X,12,4X,F6.0,4X,F6.0,11X,F4.1,15X,F8.0,13X,F6.1)
3 FORMAT(/2X,4HBASE/2X,5HTOTAL,4X,3H- -,8X,3H- -,12X,3H- -,15X,F8.0,
+13X,F6.1)
4 FORMAT(/2X,5HDEPOT/2X,5HTOTAL,4X,3H- -,8X,3H- -,12X,3H- -,15X,F8.0
+,13X,F6.1/)
5 FORMAT(2X,5HTOTAL,48X,F8.0,13X,F6.1////)
6 FORMAT(31X,23HTOTAL MANYEARS PER YEAR/35X,23H  IN TRAINING
+ //9X,20HFIRST YEAR  ,8X,F6.1//9X,20HEACH SUBSEQUENT YEAR
+,8X,F6.1)

```

C
C
C

C.....ONLY PRINT THIS TABLE IF FULL OFF-LINE OUTPUT WAS REQUESTED
IF(PRNTXX.EQ.0.OR.FULLXX.EQ.0) RETURN

C

```

TEM01=0.
DO 210 IXXX1=1,MXI
  I=INO(IXXX1)
  TEM01=TEM01+TIME1(I)
210 CONTINUE
T1=TEM01
ABMHY=0.
ABMPY=0.

```

```

ADMHY=0.
WRITE( 7, 1)
DO 250 NS=1,MXNS
  SUM1=0.
  SUM2=0.
  SUM3=0.
  DO 240 IXXX2=1,MXI
    I=INO(IXXX2)
    NHRT=0.
    NHNRT=0.
    IF(.NOT.(LRU(I).EQ.0)) GO TO 220
    NHRT=RTS(NHI(I))
    NHNRT=NRTS(NHI(I))
220  CONTINUE
    ABMHFM=(FLOAT(LRU(I))+NHRT)*(((1.+FPR(I)*XFPR)*BCMHI(I)+RTS(I)*
+    BMHI(I))*BMF)
    ADMHF=((FLOAT(LRU(I))+NHRT)*NRTS(I)+NHNRT*(1.-COND(I)))*DMHI(I)
+    *DMF
    SUM1=SUM1+ABMHFM*FAIL(I,NS)
    SUM2=SUM2+ADMHF*FAIL(I,NS)*TNB(NS)
    T2=1.0
    IF(.NOT.(RIP(I).NE.1.0)) GO TO 230
    T2=RIP(I)/(1.0-RIP(I))
230  CONTINUE
    SUM3=SUM3+(T2*MRO+MRF+SR+TR)*FAIL(I,NS)
240  CONTINUE
    ABMHY(NS)=12.*SUM1
    ABMHYD(NS)=12.*SUM3
    ABMP(NS)=(ABMHY(NS)+ABMHYD(NS))/1728.
    TABMHY(NS)=(ABMHY(NS)+ABMHYD(NS))*TNB(NS)
    TABMP(NS)=TABMHY(NS)/1728.
    ABMPY=ABMPY+TABMP(NS)
    ABMHY=ABMHY+TABMHY(NS)
    ADMHY=ADMHY+12.*SUM2
    WRITE( 7, 2) NS,ABMHY(NS),ABMHYD(NS),ABMP(NS),TABMHY(NS),
+    TABMP(NS)
250  CONTINUE
    ADMP=ADMHY/1728.
    SAMHY=ABMHY+ADMHY
    SAMP=ABMPY+ADMP
    TMPYT1=FLOAT(QTYP1)*AINT(T1/FLOAT(HPD1)+.5)/216.+FLOAT(QTYP2D)*
+    AINT(T2DA/FLOAT(HPD2)+.5)/216.+FLOAT(QTYP2B)*AINT(T2BA/
+    FLOAT(HPD2)+.5)/216.
    TMPYT2=FLOAT(QTYP2D)*TORD*AINT(T2DA/FLOAT(HPD2)+.5)/216.+
+    FLOAT(QTYP2B)*TORB*AINT(T2BA/FLOAT(HPD2)+.5)/216.
    WRITE( 7, 3) ABMHY,ABMPY
    WRITE( 7, 4) ADMHY,ADMP

```

```
WRITE( 7, 5) SAMHY,SAMP  
WRITE( 7, 6) TMPYT1,TMPYT2
```

C

```
RETURN  
END
```

SUBROUTINE RLAPRT

800827 113546619

C
C*****
C* PRINTS TIAC TO A FILE OF LATER USE IN RLA *
C*****
C

COMMON /PRNTXX/ PRNTXX
INTEGER PRNTXX
COMMON /FULLXX/ FULLXX
INTEGER FULLXX
COMMON /DUM/ DUM
INTEGER DUM
COMMON /I/ I
COMMON /INO/ INO(999)
COMMON /MXI/ MXI
COMMON /TIAC/ TIAC(999)
INTEGER FINISH
INTEGER START
1 FORMAT(I3)
2 FORMAT(6(I3,1X,F8.1,1X))
3 FORMAT(1H\$)

C
C
C
C

WRITE(23, 1) MXI
DO 220 DUM=1,MXI,6
START=DUM
FINISH=DUM+5
IF(.NOT.(FINISH.GT.MXI)) GO TO 210
FINISH=MXI
210 CONTINUE
WRITE(23, 2) (INO(I),TIAC(INO(I)),I=START,FINISH)
220 CONTINUE
WRITE(23, 3)

C

RETURN
END

SUBROUTINE OSENS

800827 113554443

```
C
C*****
C* PRINTS THE RESULTS OF THE SENSITIVITY ANALYSIS ON THE      *
C* OFF-LINE PRINTER AND/OR AT THE TERMINAL.                  *
C* IF PRNTXX=1 OR 2, OUTPUT GOES TO THE PRINTER.             *
C* IF PRNTXX=0 OR 2, OUTPUT GOES TO THE TERMINAL.            *
C*****
C
```

```

DIMENSION XXTEM1( 999),XXTEM2( 999)
COMMON /PRNTXX/ PRNTXX
INTEGER PRNTXX
COMMON /COND/ COND(999)
COMMON /CPIUP/ CPIUP
COMMON /FINC/ FINC
COMMON /FPR/ FPR(999)
COMMON /I/ I
COMMON /IDCOND/ IDCOND(999)
COMMON /IDFPR/ IDFPR(999)
COMMON /IDFR/ IDFR(999)
COMMON /IDNRTS/ IDNRTS(999)
COMMON /IDRM/ IDRM(999)
COMMON /IDRTS/ IDRTS(999)
COMMON /IDSRU/ IDSRU(999)
COMMON /IDUP/ IDUP(999)
COMMON /INO/ INO(999)
COMMON /LDCOND/ LDCOND
COMMON /LDERV/ LDERV
COMMON /LDFPR/ LDFPR
COMMON /LDFR/ LDFR
COMMON /LDNRTS/ LDNRTS
COMMON /LDRM/ LDRM
COMMON /LDRTS/ LDRTS
COMMON /LDSRU/ LDSRU
COMMON /LDUP/ LDUP
COMMON /LRU/ LRU(999)
COMMON /NRTS/ NRTS(999)
REAL NRTS
COMMON /RM/ RM(999)
COMMON /RTS/ RTS(999)
COMMON /TDCOND/ TDCOND(999)
COMMON /TDFPR/ TDFPR(999)
COMMON /TDFR/ TDFR(999)
COMMON /TDMF/ TDMF
COMMON /TDNRTS/ TDNRTS(999)
COMMON /TDPIUP/ TDPIUP
COMMON /TDRM/ TDRM(999)

```

COMMON /TDRTS/ TDRTS(999)
 COMMON /TDSRU/ TDSRU(999)
 COMMON /TDUP/ TDUP(999)
 COMMON /TDXFPR/ TDXFPR
 COMMON /TDXFR/ TDXFR
 COMMON /TDXMIL/ TDXMIL
 COMMON /TDXRM/ TDXRM
 COMMON /TDXUC/ TDXUC
 COMMON /TFR/ TFR(999)
 COMMON /UP/ UP(999)

C

1 FORMAT(1H1/25X,30HLCC SENSITIVITY ANALYSIS TABLE/)
 2 FORMAT(30H CHANGE IN LCC (\$M) ,12F7.3)
 3 FORMAT(1X/41H CHANGE IN LCC (\$M) DUE TO AN INCREASE OF,F5.1,25H %
 +IN:)
 4 FORMAT(46H GLOBAL UNIT COST (XUC FACTOR) - ,F12.3)
 5 FORMAT(46H GLOBAL FAILURE RATE (XFR FACTOR) - ,F12.3)
 6 FORMAT(46H GLOBAL FALSE PULL RATE (XFPR FACTOR) - ,F12.3)
 7 FORMAT(46H MAINTENANCE REPAIR TIMES (BMF/DMF FACTOR) -,F12.3)
 8 FORMAT(46H REPAIR MATERIALS COST (XRM FACTOR) - ,F12.3)
 9 FORMAT(46H MOD/I LABOR HOURS (XMIL FACTOR) - ,F12.3)
 10 FORMAT(1X/41H CHANGE IN LCC (\$M) DUE TO AN INCREASE OF,F5.1,25H YE
 +ARS IN:)
 11 FORMAT(46H SYSTEM LIFETIME (PIUP FACTOR) - ,F12.3)
 12 FORMAT(1X/71H ITEM FAILURE RATE (ORDERED BY SIGNIFICANCE)
 +)
 13 FORMAT(/30H ITEM INDEX ,3X,12(I4,3X))
 14 FORMAT(30H CHANGE IN FR (PPM) ,12F7.0)
 15 FORMAT(1X/71H ITEM UNIT COST (ORDERED BY SIGNIFICANCE)
 +)
 16 FORMAT(/30H ITEM INDEX ,3X,12(I4,3X))
 17 FORMAT(30H CHANGE IN UP ,12F7.0)
 18 FORMAT(1X/71H ITEM FALSE PULL RATE (ORDERED BY SIGNIFICANCE)
 +)
 19 FORMAT(/30H ITEM INDEX ,1X,12(I4,3X))
 20 FORMAT(30H CHANGE IN FPR ,12F7.3)
 21 FORMAT(1X/71H ITEM REPAIR MATERIALS COST (ORDERED BY SIGNIFICANCE)
 +)
 22 FORMAT(/30H ITEM INDEX ,3X,12(I4,3X))
 23 FORMAT(30H CHANGE IN RM (COST) ,12F7.0)
 24 FORMAT(1X/71H ITEM INTERMEDIATE REPAIR FRACTION (ORDERED BY SIGNIF
 +ICANCE))
 25 FORMAT(/30H ITEM INDEX ,1X,12(I4,3X))
 26 FORMAT(30H CHANGE IN RTS ,12F7.2)
 27 FORMAT(1X/71H ITEM DEPOT REPAIR FRACTION (ORDERED BY SIGNIFICANCE)
 +)
 28 FORMAT(/30H ITEM INDEX ,1X,12(I4,3X))

```

29 FORMAT(30H  CHANGE IN NRTS                ,12F7.2)
30 FORMAT(1X/71H ITEM CONDEMNATION RATE (ORDERED BY SIGNIFICANCE)
+
)
31 FORMAT(/30H  ITEM INDEX                    ,1X,12(I4,3X))
32 FORMAT(30H  CHANGE IN COND                ,12F7.2)
33 FORMAT(1X/71H LCC SENSITIVITY ON WHICH SRUS SHOULD BE LRU
+
)
34 FORMAT(/30H  ITEM INDEX                    ,3X,12(I4,3X))
35 FORMAT(30H  CHANGE IN SRU                  ,12F7.0)

```

C
C
C

```

IF(PRNTXX.NE.1)WRITE(6,1)
IF(PRNTXX.NE.0)WRITE(7,1)
TEMXXX=FINC*100.
IF(PRNTXX.NE.1)WRITE(6, 3) TEMXXX
IF(PRNTXX.NE.0)WRITE(7, 3) TEMXXX
IF(PRNTXX.NE.1)WRITE(6, 4) TDXUC
IF(PRNTXX.NE.0)WRITE(7, 4) TDXUC
IF(PRNTXX.NE.1)WRITE(6, 5) TDXFR
IF(PRNTXX.NE.0)WRITE(7, 5) TDXFR
IF(PRNTXX.NE.1)WRITE(6, 6) TDXFPR
IF(PRNTXX.NE.0)WRITE(7, 6) TDXFPR
IF(PRNTXX.NE.1)WRITE(6, 7) TDMF
IF(PRNTXX.NE.0)WRITE(7, 7) TDMF
IF(PRNTXX.NE.1)WRITE(6, 8) TDXRM
IF(PRNTXX.NE.0)WRITE(7, 8) TDXRM
IF(PRNTXX.NE.1)WRITE(6, 9) TDXMIL
IF(PRNTXX.NE.0)WRITE(7, 9) TDXMIL
TEMXXX=CPIUP
IF(PRNTXX.NE.1)WRITE(6,10) TEMXXX
IF(PRNTXX.NE.0)WRITE(7,10) TEMXXX
IF(PRNTXX.NE.1)WRITE(6,11) TDPIUP
IF(PRNTXX.NE.0)WRITE(7,11) TDPIUP

```

C

```

C*****
C* SECTION 1: ITEM FAILURE RATE (ORDERED BY SIGNIFICANCE) *
C*****

```

C

```

IF(LDFR .EQ.0.AND.LDERV .EQ.0) GO TO 305
IXXX=MAX0(LDERV ,LDFR )
DO 301 JXXX=1,IXXX
  XXTEM1(JXXX)=FINC *TFR(IDFR (JXXX))
  XXTEM2(JXXX)=TDFR (IDFR (JXXX))
301 CONTINUE

```

C

```

C.....PRINT OUT DERIVATIVES AT TERMINAL SIX AT A TIME.

```

```

      IF(LDFR .EQ.0.OR.PRNTXX.EQ.1) GO TO 303
      WRITE(6, 12)
      DO 302 JXXX=1,LDFR ,6
        MMHI=MINO(LDFR ,JXXX+5)
        WRITE(6, 13) (IDFR (KXXX),KXXX=JXXX,MMHI)
        WRITE(6, 14) (XXTEM1(KXXX),KXXX=JXXX,MMHI)
        WRITE(6,2) (XXTEM2(KXXX),KXXX=JXXX,MMHI)
302 CONTINUE
C
C.....PRINT OUT DERIVATIVES ON OFF-LINE PRINTER 12 AT A TIME.
303 IF(PRNTXX.EQ.0) GO TO 305
      WRITE(7, 12)
      DO 304 JXXX=1,IXXX,12
        MMHI=MINO(IXXX,JXXX+11)
        WRITE(7, 13) (IDFR (KXXX),KXXX=JXXX,MMHI)
        WRITE(7, 14) (XXTEM1(KXXX),KXXX=JXXX,MMHI)
        WRITE(7,2) (XXTEM2(KXXX),KXXX=JXXX,MMHI)
304 CONTINUE
305 CONTINUE
C
C*****
C* SECTION 2: ITEM UNIT COST (ORDERED BY SIGNIFICANCE) *
C*****
C
      IF(LDUP .EQ.0.AND.LDERV .EQ.0) GO TO 310
      IXXX=MAXO(LDERV ,LDUP )
      DO 306 JXXX=1,IXXX
        XXTEM1(JXXX)=FINC *UP(IDUP (JXXX))
        XXTEM2(JXXX)=TDUP (IDUP (JXXX))
306 CONTINUE
C
C.....PRINT OUT DERIVATIVES AT TERMINAL SIX AT A TIME.
      IF(LDUP .EQ.0.OR.PRNTXX.EQ.1) GO TO 308
      WRITE(6, 15)
      DO 307 JXXX=1,LDUP ,6
        MMHI=MINO(LDUP ,JXXX+5)
        WRITE(6, 16) (IDUP (KXXX),KXXX=JXXX,MMHI)
        WRITE(6, 17) (XXTEM1(KXXX),KXXX=JXXX,MMHI)
        WRITE(6,2) (XXTEM2(KXXX),KXXX=JXXX,MMHI)
307 CONTINUE
C
C.....PRINT OUT DERIVATIVES ON OFF-LINE PRINTER 12 AT A TIME.
308 IF(PRNTXX.EQ.0) GO TO 310
      WRITE(7, 15)
      DO 309 JXXX=1,IXXX,12
        MMHI=MINO(IXXX,JXXX+11)
        WRITE(7, 16) (IDUP (KXXX),KXXX=JXXX,MMHI)

```

```

        WRITE(7, 17) (XXTEM1(KXXX),KXXX=JXXX,MMHI)
        WRITE(7,2) (XXTEM2(KXXX),KXXX=JXXX,MMHI)
309 CONTINUE
310 CONTINUE
C
C*****
C* SECTION 3: ITEM FALSE PULL RATE (ORDERED BY SIGNIFICANCE) *
C*****
C
        IF(LDFPR .EQ.0.AND.LDERV .EQ.0) GO TO 315
        IXXX=MAX0(LDERV ,LDFPR )
        DO 311 JXXX=1,IXXX
            XXTEM1(JXXX)=FINC *FPR(IDFPR (JXXX))
            XXTEM2(JXXX)=TDFPR (IDFPR (JXXX))
311 CONTINUE
C
C.....PRINT OUT DERIVATIVES AT TERMINAL SIX AT A TIME.
        IF(LDFPR .EQ.0.OR.PRNTXX.EQ.1) GO TO 313
        WRITE(6, 18)
        DO 312 JXXX=1,LDFPR ,6
            MMHI=MINO(LDFPR ,JXXX+5)
            WRITE(6, 19) (IDFPR (KXXX),KXXX=JXXX,MMHI)
            WRITE(6, 20) (XXTEM1(KXXX),KXXX=JXXX,MMHI)
            WRITE(6,2) (XXTEM2(KXXX),KXXX=JXXX,MMHI)
312 CONTINUE
C
C.....PRINT OUT DERIVATIVES ON OFF-LINE PRINTER 12 AT A TIME.
313 IF(PRNTXX.EQ.0) GO TO 315
        WRITE(7, 18)
        DO 314 JXXX=1,IXXX,12
            MMHI=MINO(IXXX,JXXX+11)
            WRITE(7, 19) (IDFPR (KXXX),KXXX=JXXX,MMHI)
            WRITE(7, 20) (XXTEM1(KXXX),KXXX=JXXX,MMHI)
            WRITE(7,2) (XXTEM2(KXXX),KXXX=JXXX,MMHI)
314 CONTINUE
315 CONTINUE
C
C*****
C* SECTION 4: ITEM REPAIR MATERIALS COST (ORDERED BY SIGNIFICANC *
C*****
C
        IF(LDRM .EQ.0.AND.LDERV .EQ.0) GO TO 320
        IXXX=MAX0(LDERV ,LDRM )
        DO 316 JXXX=1,IXXX
            XXTEM1(JXXX)=FINC *RM(IDRM (JXXX))*UP(IDRM (JXXX))
            XXTEM2(JXXX)=TDRM (IDRM (JXXX))
316 CONTINUE

```

```

C
C.....PRINT OUT DERIVATIVES AT TERMINAL SIX AT A TIME.
IF(LDRM .EQ.0.OR.PRNTXX.EQ.1) GO TO 318
WRITE(6, 21)
DO 317 JXXX=1,LDRM ,6
  MMHI=MINO(LDRM ,JXXX+5)
  WRITE(6, 22) (IDRM (KXXX),KXXX=JXXX,MMHI)
  WRITE(6, 23) (XXTEM1(KXXX),KXXX=JXXX,MMHI)
  WRITE(6,2) (XXTEM2(KXXX),KXXX=JXXX,MMHI)
317 CONTINUE
C
C.....PRINT OUT DERIVATIVES ON OFF-LINE PRINTER 12 AT A TIME.
318 IF(PRNTXX.EQ.0) GO TO 320
WRITE(7, 21)
DO 319 JXXX=1,IXXX,12
  MMHI=MINO(IXXX,JXXX+11)
  WRITE(7, 22) (IDRM (KXXX),KXXX=JXXX,MMHI)
  WRITE(7, 23) (XXTEM1(KXXX),KXXX=JXXX,MMHI)
  WRITE(7,2) (XXTEM2(KXXX),KXXX=JXXX,MMHI)
319 CONTINUE
320 CONTINUE
C
C*****
C* SECTION 5: ITEM INTERMEDIATE REPAIR FRACTION (ORDERED BY SIGN *
C*****
C
IF(LDRTS .EQ.0.AND.LDERV .EQ.0) GO TO 325
IXXX=MAX0(LDERV ,LDRTS )
DO 321 JXXX=1,IXXX
  XXTEM1(JXXX)=FINC *AMIN1(FINC,NRTS(IDRTS (JXXX)))/FINC
  XXTEM2(JXXX)=TDRTS (IDRTS (JXXX))
321 CONTINUE
C
C.....PRINT OUT DERIVATIVES AT TERMINAL SIX AT A TIME.
IF(LDRTS .EQ.0.OR.PRNTXX.EQ.1) GO TO 323
WRITE(6, 24)
DO 322 JXXX=1,LDRTS ,6
  MMHI=MINO(LDRTS ,JXXX+5)
  WRITE(6, 25) (IDRTS (KXXX),KXXX=JXXX,MMHI)
  WRITE(6, 26) (XXTEM1(KXXX),KXXX=JXXX,MMHI)
  WRITE(6,2) (XXTEM2(KXXX),KXXX=JXXX,MMHI)
322 CONTINUE
C
C.....PRINT OUT DERIVATIVES ON OFF-LINE PRINTER 12 AT A TIME.
323 IF(PRNTXX.EQ.0) GO TO 325
WRITE(7, 24)
DO 324 JXXX=1,IXXX,12

```

```

        MMHI=MINO(IXXX,JXXX+11)
        WRITE(7, 25) (IDRTS (KXXX),KXXX=JXXX,MMHI)
        WRITE(7, 26) (XXTEM1(KXXX),KXXX=JXXX,MMHI)
        WRITE(7,2) (XXTEM2(KXXX),KXXX=JXXX,MMHI)
324 CONTINUE
325 CONTINUE
C
C*****
C* SECTION 6: ITEM DEPOT REPAIR FRACTION (ORDERED BY SIGNIFICANC *
C*****
C
        IF(LDNRTS.EQ.0.AND.LDERV .EQ.0) GO TO 330
        IXXX=MAX0(LDERV ,LDNRTS)
        DO 326 JXXX=1,IXXX
            XXTEM1(JXXX)=FINC *AMIN1(FINC,RTS(IDNRTS(JXXX)))/FINC
            XXTEM2(JXXX)=TDNRTS(IDNRTS(JXXX))
326 CONTINUE
C
C.....PRINT OUT DERIVATIVES AT TERMINAL SIX AT A TIME.
        IF(LDNRTS.EQ.0.OR.PRNTXX.EQ.1) GO TO 328
        WRITE(6, 27)
        DO 327 JXXX=1,LDNRTS,6
            MMHI=MINO(LDNRTS,JXXX+5)
            WRITE(6, 28) (IDNRTS(KXXX),KXXX=JXXX,MMHI)
            WRITE(6, 29) (XXTEM1(KXXX),KXXX=JXXX,MMHI)
            WRITE(6,2) (XXTEM2(KXXX),KXXX=JXXX,MMHI)
327 CONTINUE
C
C.....PRINT OUT DERIVATIVES ON OFF-LINE PRINTER 12 AT A TIME.
328 IF(PRNTXX.EQ.0) GO TO 330
        WRITE(7, 27)
        DO 329 JXXX=1,IXXX,12
            MMHI=MINO(IXXX,JXXX+11)
            WRITE(7, 28) (IDNRTS(KXXX),KXXX=JXXX,MMHI)
            WRITE(7, 29) (XXTEM1(KXXX),KXXX=JXXX,MMHI)
            WRITE(7,2) (XXTEM2(KXXX),KXXX=JXXX,MMHI)
329 CONTINUE
330 CONTINUE
C
C*****
C* SECTION 7: ITEM CONDEMNATION RATE (ORDERED BY SIGNIFICANCE) *
C*****
C
        IF(LDCOND.EQ.0.AND.LDERV .EQ.0) GO TO 335
        IXXX=MAX0(LDERV ,LDCOND)
        DO 331 JXXX=1,IXXX
            XXTEM1(JXXX)=FINC *AMIN1(FINC,1.-COND(IDCOND(JXXX)))/FINC

```

```

        XXTEM2(JXXX)=TDCOND(IDCOND(JXXX))
331 CONTINUE
C
C.....PRINT OUT DERIVATIVES AT TERMINAL SIX AT A TIME.
        IF(LDCOND.EQ.0.OR.PRNTXX.EQ.1) GO TO 333
        WRITE(6, 30)
        DO 332 JXXX=1,LDCOND,6
            MMHI=MINO(LDCOND,JXXX+5)
            WRITE(6, 31) (IDCOND(KXXX),KXXX=JXXX,MMHI)
            WRITE(6, 32) (XXTEM1(KXXX),KXXX=JXXX,MMHI)
            WRITE(6,2) (XXTEM2(KXXX),KXXX=JXXX,MMHI)
332 CONTINUE
C
C.....PRINT OUT DERIVATIVES ON OFF-LINE PRINTER 12 AT A TIME.
333 IF(PRNTXX.EQ.0) GO TO 335
        WRITE(7, 30)
        DO 334 JXXX=1,IXXX,12
            MMHI=MINO(IXXX,JXXX+11)
            WRITE(7, 31) (IDCOND(KXXX),KXXX=JXXX,MMHI)
            WRITE(7, 32) (XXTEM1(KXXX),KXXX=JXXX,MMHI)
            WRITE(7,2) (XXTEM2(KXXX),KXXX=JXXX,MMHI)
334 CONTINUE
335 CONTINUE
C
C*****
C* SECTION 8: LCC SENSITIVITY ON WHICH SRUS SHOULD BE LRU *
C*****
C
        IF(LDSRU .EQ.0.AND.LDERV .EQ.0) GO TO 340
        IXXX=MAXO(LDERV ,LDSRU )
        DO 336 JXXX=1,IXXX
            XXTEM1(JXXX)=FINC *FLOAT((1-LRU(IDSRU (JXXX))))/FINC
            XXTEM2(JXXX)=TDSRU (IDSRU (JXXX))
336 CONTINUE
C
C.....PRINT OUT DERIVATIVES AT TERMINAL SIX AT A TIME.
        IF(LDSRU .EQ.0.OR.PRNTXX.EQ.1) GO TO 338
        WRITE(6, 33)
        DO 337 JXXX=1,LDSRU ,6
            MMHI=MINO(LDSRU ,JXXX+5)
            WRITE(6, 34) (IDSRU (KXXX),KXXX=JXXX,MMHI)
            WRITE(6, 35) (XXTEM1(KXXX),KXXX=JXXX,MMHI)
            WRITE(6,2) (XXTEM2(KXXX),KXXX=JXXX,MMHI)
337 CONTINUE
C
C.....PRINT OUT DERIVATIVES ON OFF-LINE PRINTER 12 AT A TIME.
338 IF(PRNTXX.EQ.0) GO TO 340

```

```
WRITE(7, 33)
DO 339 JXXX=1, IXXX, 12
  MMHI=MINO(IXXX, JXXX+11)
  WRITE(7, 34) (IDSRU (KXXX), KXXX=JXXX, MMHI)
  WRITE(7, 35) (XITEM1(KXXX), KXXX=JXXX, MMHI)
  WRITE(7, 2) (XITEM2(KXXX), KXXX=JXXX, MMHI)
339 CONTINUE
340 CONTINUE
C
  RETURN
END
```

SUBROUTINE INITIAL

C
C.....INITIALIZES VARIABLES TO DEFAULT VALUES.
C

800827 113608063

```
COMMON /NTABXX/ NTABXX
COMMON /NERRXX/ NERRXX
COMMON /B/ B
INTEGER B
COMMON /BINO/ BINO(16)
INTEGER BINO
COMMON /BXREF/ BXREF(1)
INTEGER BXREF
COMMON /DIXREF/ DIXREF(1)
INTEGER DIXREF
COMMON /DUINO/ DUINO(999)
INTEGER DUINO
COMMON /DUM/ DUM
INTEGER DUM
COMMON /I/ I
COMMON /IA/ IA
COMMON /IAINO/ IAINO(4)
COMMON /IAXREF/ IAXREF(1)
COMMON /INO/ INO(999)
COMMON /IRM/ IRM
COMMON /IRMINO/ IRMINO(4)
COMMON /IRMT/ IRMT
COMMON /IRMTNO/ IRMTNO(4)
COMMON /IRMTXR/ IRMTXR(1)
COMMON /IRMXRF/ IRMXRF(1)
COMMON /IXREF/ IXREF(999)
COMMON /K1TEMP/ K1TEMP
COMMON /K1TNO/ K1TNO(4)
COMMON /K2TEMP/ K2TEMP
COMMON /K2TNO/ K2TNO(30)
COMMON /L/ L
COMMON /LT/ LT
COMMON /LTINO/ LTINO(30)
COMMON /LTXREF/ LTXREF(1)
COMMON /LXREF/ LXREF(250)
COMMON /M/ M
COMMON /MINO/ MINO(3)
COMMON /MXI/ MXI
COMMON /MXIRM/ MXIRM
COMMON /MXIRMT/ MXIRMT
COMMON /MXKT/ MXKT
COMMON /MXKTE/ MXKTE
COMMON /MXL/ MXL
```

```

COMMON /MXLT/ MXLT
COMMON /MXM/ MXM
COMMON /MXNP/ MXNP
COMMON /MXNS/ MXNS
COMMON /MXREF/ MXREF(1)
COMMON /NIA/ NIA
COMMON /NP/ NP
COMMON /NPINO/ NPINO(10)
COMMON /NPXREF/ NPXREF(1)
COMMON /NS/ NS
COMMON /NSINO/ NSINO(16)
COMMON /NSXREF/ NSXREF(1)
COMMON /SEINO/ SEINO(250)
INTEGER SEINO
COMMON /XK1TNO/ XK1TNO(1)
INTEGER XK1TNO
COMMON /XK2TNO/ XK2TNO(1)
INTEGER XK2TNO

```

C

```

NTABXX=0
NERRXX=0

```

C

```

DO 10 IXXX1=1,1
  IAXREF(IXXX1)=IXXX1
  XK2TNO(IXXX1)=IXXX1
  NSXREF(IXXX1)=IXXX1
  NPXREF(IXXX1)=IXXX1
  IRMXRF(IXXX1)=IXXX1
  IRMTXR(IXXX1)=IXXX1
  DIXREF(IXXX1)=IXXX1
  LTXREF(IXXX1)=IXXX1
  BXREF(IXXX1)=IXXX1
  XK1TNO(IXXX1)=IXXX1
  MXREF(IXXX1)=IXXX1
10 CONTINUE

```

C

```

NIA=4
MXKT=4
MXL=250
MXNP=10
MXNS=16
MXI=999
MXLT=30
MXKTE=30
MXIRM=4
MXM=3
MXIRMT=4

```

```

C      DO      30 B=1,16
          BINO(B)=B
30 CONTINUE
C      DO      40 IRM=1,4
          IRMINO(IRM)=IRM
40 CONTINUE
C      DO      50 IRMT=1,4
          IRMTNO(IRMT)=IRMT
50 CONTINUE
C      DO      60 IXXX1=1,999
          IXREF(IXXX1)=IXXX1
60 CONTINUE
C      DO      70 IA=1,4
          IAINO(IA)=IA
70 CONTINUE
C      DO      80 DUM=1,999
          DUINO(DUM)=DUM
80 CONTINUE
C      DO      90 I=1,999
          INO(I)=I
90 CONTINUE
C      DO     100 NS=1,16
          NSINO(NS)=NS
100 CONTINUE
C      DO     110 IXXX1=1,250
          LXREF(IXXX1)=IXXX1
110 CONTINUE
C      DO     120 NP=1,10
          NPINO(NP)=NP
120 CONTINUE
C      DO     130 K1TEMP=1,4
          K1TNO(K1TEMP)=K1TEMP
130 CONTINUE
C      DO     140 M=1,3
          MINO(M)=M

```

```
140 CONTINUE
C
    DO 150 L=1,250
        SEINO(L)=L
150 CONTINUE
C
    DO 160 LT=1,30
        LTINO(LT)=LT
160 CONTINUE
C
    DO 170 K2TEMP=1,30
        K2TNO(K2TEMP)=K2TEMP
170 CONTINUE
C
    RETURN
    END
```

SUBROUTINE TITLE

800827 113630740

```

C
C*****
C* SUBROUTINE TO PRINT A TITLE PAGE FOR OFF-LINE OUTPUT.      *
C*****
C
C      COMMON /XTITLE/ XTITLE(30)
C
C      1 FORMAT(1H1////////////////////////////////////
+44X,44H*****/
+44X,1H*,42X,1H*/
+44X,44H*   STRATEGIC SATELLITE SYSTEM LCC MODEL   */
+44X,1H*,42X,1H*/
+44X,10H*   RUN:   ,30A1,4H   */
+44X,1H*,42X,1H*/
+44X,44H*****/)
C
C      WRITE(7,1) (XTITLE(L),L=1,30)
C
C      RETURN
C      END

```

```

SUBROUTINE TDSORT(TD,ID,LD,N)
C
C***** 800827 113644704 *****
C* THIS SUBROUTINE 'BUBBLES UP' TO THE FIRST -LD- POSITIONS IN ARRAY *
C* -ID- THE INDEX NUMBERS CORRESPONDING TO THE -LD- HIGHEST *
C* VALUES OF ARRAY -TD-. *
C*****
C
C      DIMENSION TD(N),ID(N)
C
C      DO 7 L=1,LD
C        MA=N-L
C        DO 6 M=1,MA
C          MB=N-M
C          IF (ABS(TD(ID(MB+1))).LT.ABS(TD(ID(MB)))) GO TO 6
C            IDD=ID(MB+1)
C            ID(MB+1)=ID(MB)
C            ID(MB)=IDD
C        6 CONTINUE
C      7 CONTINUE
C
C      RETURN
C      END

```

SUBROUTINE SSETXX

```

C
C***** 800827 113644790
C*****
C* THIS SUBROUTINE INITIALIZES SENSITIVITY ANALYSIS VARIABLES TO *
C* DEFAULT VALUES. *
C*****
C
COMMON /FINC/ FINC
COMMON /LDCOND/ LDCOND
COMMON /LDERV/ LDERV
COMMON /LDFPR/ LDFPR
COMMON /LDFR/ LDFR
COMMON /LDNRTS/ LDNRTS
COMMON /LDRM/ LDRM
COMMON /LDRTS/ LDRTS
COMMON /LDSRU/ LDSRU
COMMON /LDUP/ LDUP

C
FINC =.25
LDCOND=0
LDERV =12
LDFPR =0
LDFR =0
LDNRTS=0
LDRM =0
LDRTS =0
LDSRU =0
LDUP =0

C
RETURN
END

```

SUBROUTINE PRMPT1

```

C
C***** 800827 113645002 *****
C* FIRST OF FOUR PROMPTING SUBROUTINES TO READ IN USER INPUTS FROM *
C* THE TERMINAL. IF THIS IS THE FIRST CALL OF THE LCC: *
C* 1- PRINT TITLE *
C* 2- ASK USER WHERE HE WANTS HIS OUTPUT. (PRNTXX=0 TERMINAL ONLY; *
C* PRNTXX=1 OFF-LINE ONLY; PRNTXX=2 BOTH PLACES.) *
C* 3- ASK USER FOR MIN OR MAX PROMPTING (MAX=LONG PROMPT COMMENTS) *
C* 4- IF USER REQUEST OFF-LINE OUTPUT, GET A NAME FOR THE RUN. *
C* IF THIS IS A SUBSEQUENT CALL FOR THE LCC: *
C* 1- NOTIFY USER THAT VARIABLES ARE AS THEY WERE AFTER LAST *
C* NAMELISTS WERE SUBMITTED. *
C* 2- ASK USER IF HE WANTS TO REREAD INPUT FILES (RERDXX=1). *
C* 3- ASK MIN OR MAX PROMPTING ONLY IF LAST RUN WAS MAX PROMPTING. *
C* 4- ASK USER WHERE HE WANTS HIS OUTPUT. (PRNTXX=0 TERMINAL ONLY; *
C* PRNTXX=1 OFF-LINE ONLY; PRNTXX=2 BOTH PLACES) *
C* 5- IF USER REQUEST OFF-LINE OUTPUT, GET A NAME FOR THE RUN. *
C*****

```

C

```

COMMON /EXITXX/ EXITXX
INTEGER EXITXX
COMMON /FULLXX/ FULLXX
INTEGER FULLXX
COMMON /ITERXX/ ITERXX
COMMON /MAXPMT/ MAXPMT
COMMON /PRNTXX/ PRNTXX
INTEGER PRNTXX
COMMON /RERDXX/ RERDXX
INTEGER RERDXX
COMMON /XTITLE/ XTITLE(30)
DATA BK/1H /,CHM/1HM/,CHI/1HI/,CHN/1HN/,
+CHA/1HA/,CHX/1HX/,CHY/1HY/,CHP/1HP/,CHF/1HF/

```

C

```

1 FORMAT(1X/47H STRATEGIC SATELLITE SYSTEM LCC MODEL )
2 FORMAT(1X/50H AT THIS POINT, VARIABLE VALUES ARE AS THEY WERE A,
+13HFTER THE LAST/
+53H NAMELISTS WERE SUBMITTED. DO YOU WISH TO RESET NAME,
+15HLIST GO1 OR GO2/
+54H VARIABLES TO THE VALUES FOUND IN THE INPUT FILES (Y O,
+6HR N)-?)
3 FORMAT(1X/44H MINIMUM OR MAXIMUM PROMPTING (MIN OR MAX)-?)
4 FORMAT(3A1)
5 FORMAT(1X/50H SUBMIT 'MIN' OR 'MAX' STARTING IN COLUMN 1. NOTH,
+15HING ELSE WORKS.)
6 FORMAT(30H OUTPUT AT TERMINAL (Y OR N)-?)
7 FORMAT(A1)

```

```

8 FORMAT(49H SUBMIT 'Y' OR 'N' STARTING IN COLUMN 1. NOTHING,
+12H ELSE WORKS.)
9 FORMAT(55H OFF-LINE OUTPUT: FULL, PARTIAL, OR NONE (F, P, OR N)-?)
10 FORMAT(29H SUBMIT A TITLE FOR THIS RUN:)
11 FORMAT(30A1)
12 FORMAT(/50H SET EXITXX=1 IN ANY NAMELIST IF YOU WANT TO EXIT.)
13 FORMAT(37H SUBMIT 'F', 'P', OR 'N' IN COLUMN 1.,
+21H NOTHING ELSE WORKS.)

```

C

```

      IF(ITERXX.NE.1) GO TO 16
      WRITE(6,1)
      GO TO 18
16 CONTINUE
      WRITE(6,2)
      RERDXX=2
      STR1=BK
17 READ(5,7) STR1
      IF(STR1.EQ.CHY) RERDXX=1
      IF(STR1.EQ.CHN) RERDXX=0
      IF(RERDXX.NE.2) GO TO 18
      WRITE(6,8)
      GO TO 17
18 CONTINUE
      IF(ITERXX.GT.1.AND.MAXPMT.NE.1) GO TO 20
      WRITE(6,3)
      MAXPMT=2
19 READ(5,4) STR1,STR2,STR3
      IF(STR1.EQ.CHM.AND.STR2.EQ.CHI.AND.STR3.EQ.CHN) MAXPMT=0
      IF(STR1.EQ.CHM.AND.STR2.EQ.CHA.AND.STR3.EQ.CHX) MAXPMT=1
      IF(MAXPMT.NE.2) GO TO 20
      WRITE(6,5)
      GO TO 19
20 CONTINUE
      MM1=2
      WRITE(6,6)
21 READ(5,7) STR1
      IF(STR1.EQ.CHY) MM1=1
      IF(STR1.EQ.CHN) MM1=0
      IF(MM1.NE.2) GO TO 22
      WRITE(6,8)
      GO TO 21
22 NM2=3
      STR1=BK
      WRITE(6,9)
23 READ(5,7) STR1
      IF(STR1.EQ.CHF) MM2=2
      IF(STR1.EQ.CHP) MM2=1

```

```

      IF(STR1.EQ.CHN) MM2=0
      IF(MM2.NE.3) GO TO 24
      WRITE(6,13)
      GO TO 23
24  CONTINUE
      IF(MM2.EQ.0) PRNTXX=0
      IF(MM1.EQ.0.AND.MM2.GE.1) PRNTXX=1
      IF(MM1.EQ.1.AND.MM2.GE.1) PRNTXX=2
      FULLXX=0
      IF(MM2.EQ.2) FULLXX=1
      IF(PRNTXX.EQ.0) GO TO 88
      WRITE(6,10)
      READ(5,11) (XTITLE(L),L=1,30)
88  IF(MAXPMT.EQ.1.OR.ITERXX.EQ.1) WRITE(6,12)
C
      RETURN
      END

```

SUBROUTINE PRMPT2

C 800827 113645395
 C*****
 C* SECOND OF THE PROMPTING ROUTINES. PROMPTS THE USER FOR NAMELIST *
 C* /GO1/(WHICH CONTAINS ALL A-M VARIABLES FROM THE INPUT *
 C* FILES AND ALLOWS THE USER TO OVERRIDE THOSE VALUES IN REAL TIME). *
 C*****
 C

```
COMMON /EXITXX/ EXITXX
INTEGER EXITXX
COMMON /ITERXX/ ITERXX
COMMON /MAXPMT/ MAXPMT
COMMON /RERDXX/ RERDXX
INTEGER RERDXX
COMMON /LDERV / LDERV
COMMON /FINC / FINC
COMMON /A/ A(999,4,30)
INTEGER A
COMMON /ACPP/ ACPP
COMMON /AKIT/ AKIT(4,10)
COMMON /AMPM/ AMPM(10,3)
COMMON /APFH/ APFH(10,3)
COMMON /BAA/ BAA
COMMON /BCMH/ BCMH(999)
COMMON /BDATA/ BDATA
INTEGER BDATA
COMMON /BF/ BF
COMMON /BIRD/ BIRD
COMMON /BLR/ BLR
COMMON /BMF/ BMF
COMMON /BMH/ BMH(999)
COMMON /BNOUN/ BNOUN(16,16)
COMMON /BPLAT/ BPLAT(16)
INTEGER BPLAT
COMMON /BRCT/ BRCT
COMMON /BSP/ BSP(16)
INTEGER BSP
COMMON /BTYP/ BTYP(16)
INTEGER BTYP
COMMON /CFG/ CFG(3)
COMMON /COND/ COND(999)
COMMON /CPD1/ CPD1
COMMON /CPD2/ CPD2
COMMON /CPPC/ CPPC
COMMON /CPPD/ CPPD(3)
COMMON /CRCT/ CRCT
COMMON /CSE/ CSE(250)
```

```

COMMON /DAA/ DAA
COMMON /DAD/ DAD
COMMON /DATAB/ DATAB(999)
INTEGER DATAB
COMMON /DATAD/ DATAD(999)
INTEGER DATAD
COMMON /DATAS/ DATAS(250)
INTEGER DATAS
COMMON /DDATA/ DDATA
INTEGER DDATA
COMMON /DLR/ DLR
COMMON /DMF/ DMF
COMMON /DMH/ DMH(999)
COMMON /DRAG/ DRAG(10)
COMMON /DRCT/ DRCT(3)
COMMON /FGH/ FGH(10)
COMMON /FPR/ FPR(999)
COMMON /FR/ FR(3,10)
COMMON /FSEDG/ FSEDG
COMMON /GFE/ GFE(999)
INTEGER GFE
COMMON /HPD1/ HPD1
INTEGER HPD1
COMMON /HPD2/ HPD2
INTEGER HPD2
COMMON /I/ I
COMMON /IMC/ IMC
REAL IMC
COMMON /INOUN/ INOUN(999,24)
REAL INOUN
COMMON /INTEG/ INTEG(999)
REAL INTEG
COMMON /INTNR/ INTNR(10)
REAL INTNR
COMMON /INTR/ INTR(10)
REAL INTR
COMMON /IPCF/ IPCF(999)
REAL IPCF
COMMON /IRMIN/ IRMIN(999,4)
COMMON /K/ K(10)
REAL K
COMMON /KFAC/ KFAC(4)
REAL KFAC
COMMON /L/ L
COMMON /LE/ LE(10)
COMMON /LFAC/ LFAC(999)
REAL LFAC

```

```

COMMON /LO/ LO(16)
COMMON /LRU/ LRU(999)
COMMON /MIFIX/ MIFIX(3,10)
REAL MIFIX
COMMON /MILR/ MILR(3)
REAL MILR
COMMON /MIMH/ MIMH(4,3,10)
REAL MIMH
COMMON /MMPD/ MMPD(10,3)
REAL MMPD
COMMON /MMPM/ MMPM(10)
REAL MMPM
COMMON /MRF/ MRF
REAL MRF
COMMON /MRO/ MRO
REAL MRO
COMMON /MSE/ MSE(250)
REAL MSE
COMMON /MTBMI/ MTBMI(999,4)
REAL MTBMI
COMMON /MUSE/ MUSE
REAL MUSE

```

C

```

NAMELIST /GO1/ EXITXX,A,ACPP,AKIT,AMPM,APFH,BAA,BCMH,BDATA,BF,
+   BIRD,BLR,BMF,BMH,BNOUN,BPLAT,BRCT,BSP,BTYPE,CFG,COND,CPD1,
+   CPD2,CPPC,CPPD,CRCT,CSE,DAA,DAD,DATAB,LATAD,DATAS,DDATA,DLR,
+   DMF,DMH,DRAG,DRCT,FGH,FPR,FR,FSEDC,GFE,HPD1,HPD2,I,IMC,
+   INOUN,INTEG,INTNR,INTR,IPCF,IRMIN,K,KFAC,L,LE,LFAC,LO,LRU,
+   MIFIX,MILR,MIMH,MMPD,MMPM,MRF,MRO,MSE,MTBMI,MUSE

```

C

```

1 FORMAT(1X)
2 FORMAT(52H NAMELIST /GO1/ CONTAINS ALL VARIABLES FOUND IN THE,
+13H INPUT FILES /
+40H THAT BEGIN WITH THE LETTERS A TO M. )
3 FORMAT(53H AT THIS POINT, NAMELIST /GO1/ VARIABLES CONTAIN VAL,
+13HUES AS IN THE/
+14H INPUT FILES.)
4 FORMAT(53H AT THIS POINT, NAMELIST /GO1/ VARIABLES ARE AS THEY,
+11H WERE AFTER/
+40H THE LAST NAMELIST /GO1/ WAS SUBMITTED.)
5 FORMAT(54H TO USE THESE VALUES, SUBMIT AN EMPTY NAMELIST /GO1./
+50H TO OVERRIDE ANY OF THESE VALUES, SUBMIT A NON-EMP,
+18HTY NAMELIST /GO1/.)
6 FORMAT(42H SUBMIT NAMELIST /GO1/ IN NAMELIST FORMAT:)

```

C

```

WRITE(6,1)
IF(MAXPMT.NE.1) GO TO 30

```

```

WRITE(6,2)
IF(ITERXX.EQ.1.OR.RERDXX.EQ.1) WRITE(6,3)
IF(ITERXX.GT.1.AND.RERDXX.NE.1) WRITE(6,4)
WRITE(6,5)
30 WRITE(6,6)
READ(5,G01)
IF(EXITXX.EQ.1) RETURN
C
RETURN
END

```

SUBROUTINE PRMPT3

800827 113650974

C

C*****

C: THIRD OF THE PROMPTING ROUTINES. PROMPTS THE USER FOR NAMELISTS *

C* /GO2/(WHICH CONTAINS ALL N-Z VARIABLES FROM THE INPUT *)

C* FILES AND ALLOWS THE USER TO OVERRIDE THOSE VALUES IN REAL TIME) *

C* AND /SENS/ (WHICH CONTAINS SENSITIVITY ANALYSIS PRINT PARAMETERS). *

C*****

C

COMMON /EXITXX/ EXITXX

INTEGER EXITXX

COMMON /ITERXX/ ITERXX

COMMON /MAXPMT/ MAXPMT

COMMON /RERDXX/ RERDXX

INTEGER RERDXX

COMMON /LDERV / LDERV

COMMON /FINC / FINC

COMMON /LDCOND/ LDCOND

COMMON /LDFPR/ LDFPR

COMMON /LDFR/ LDFR

COMMON /LDNRTS/ LDNRTS

COMMON /LDRM/ LDRM

COMMON /LDRTS/ LDRTS

COMMON /LDSRU/ LDSRU

COMMON /LDUP/ LDUP

COMMON /NAE/ NAE(10)

REAL NAE

COMMON /NBC/ NBC(16)

REAL NBC

COMMON /NHB/ NHB(16)

COMMON /NHI/ NHI(999)

COMMON /NITEM/ NITEM(999,10)

REAL NITEM

COMMON /NJA/ NJA(999,4)

COMMON /NPLT/ NPLT(10,16)

REAL NPLT

COMMON /NRM/ NRM(999)

COMMON /NRMI/ NRMI(10)

REAL NRMI

COMMON /NRTS/ NRTS(999)

REAL NRTS

COMMON /NRUC/ NRUC

REAL NRUC

COMMON /NTRMP/ NTRMP(10)

REAL NTRMP

COMMON /OST/ OST(3)

COMMON /OSTC/ OSTC

COMMON /PA/ PA(999)
COMMON /PAL1/ PAL1
COMMON /PAL2B/ PAL2B
COMMON /PAL2D/ PAL2D
COMMON /PDIV/ PDIV(10)
COMMON /PIUP/ PIUP
COMMON /PMLR/ PMLR
COMMON /PNOUN/ PNOUN(10,12)
COMMON /PTNUM/ PTNUM(999,12)
COMMON /QSA/ QSA(999,4,30)
COMMON /QTYP1/ QTYP1
INTEGER QTYP1
COMMON /QTYP2B/ QTYP2B
INTEGER QTYP2B
COMMON /QTYP2D/ QTYP2D
INTEGER QTYP2D
COMMON /R/ R
INTEGER R
COMMON /RCPP/ RCPP
COMMON /RIP/ RIP(999)
COMMON /RL/ RL(999)
INTEGER RL
COMMON /RM/ RM(999)
COMMON /RMC/ RMC
COMMON /RMH/ RMH(999)
COMMON /RTS/ RTS(999)
COMMON /SA/ SA
COMMON /SEDEV/ SEDEV(250)
COMMON /SENOUN/ SENOUN(250,20)
COMMON /SENUM/ SENUM(250,12)
COMMON /SETYPE/ SETYPE(250)
INTEGER SETYPE
COMMON /SPC1/ SPC1
INTEGER SPC1
COMMON /SPC2/ SPC2
INTEGER SPC2
COMMON /SR/ SR
COMMON /TEFM/ TEFM
COMMON /TFAC/ TFAC(10)
COMMON /THRS/ THRS(10)
COMMON /TIME1/ TIME1(999)
INTEGER TIME1
COMMON /TNB/ TNB(16)
COMMON /TNLR/ TNLR
COMMON /TORB/ TORB
COMMON /TORD/ TORD
COMMON /TR/ TR

```

COMMON /TRAVB/ TRAVB
COMMON /TRAV1D/ TRAV1D
COMMON /TYP2TF/ TYP2TF
COMMON /UCPP/ UCPP
COMMON /UCTDEV/ UCTDEV(999)
COMMON /UP/ UP(999)
COMMON /WT/ WT(999)
COMMON /XFPR/ XFPR
COMMON /XFR/ XFR
COMMON /XMIL/ XMIL
COMMON /XUC/ XUC

```

C

```

NAMELIST /SENS/ EXITXX, LDERV , FINE , LDCOND, LDFPR , LDFR , LDNRTS,
+   LDRM , LDRTS , LDSRU , LDUP
NAMELIST /GO2/ EXITXX, NAE, NBC, NHB, NHI, NITEM, NJA, NPLT, NRM, NRMI,
+   NRTS, NRUC, NTRMP, OST, OSTC, PA, PAL1, PAL2B, PAL2D, PDIV, PIUP, PMLR,
+   PNOUN, PTNUM, QSA, QTYP1, QTYP2B, QTYP2D, R, RCPP, RIP, RL, RM, RMC,
+   RMH, RTS, SA, SEDEV, SENOUN, SENUM, SETYPE, SPC1, SPC2, SR, TEFM, TFAC,
+   THRS, TIME1, TNB, TNL, TORB, TORD, TR, TRAVB, TRAV1D, TYP2TF, UCPP,
+   UCTDEV, UP, WT, XFPR, XFR, XMIL, XUC

```

C

```

1 FORMAT(1X)
2 FORMAT(52H NAMELIST /GO2/ CONTAINS ALL VARIABLES FOUND IN THE,
+13H INPUT FILES /
+40H THAT BEGIN WITH THE LETTERS N TO Z. )
3 FORMAT(53H AT THIS POINT, NAMELIST /GO2/ VARIABLES CONTAIN VAL,
+13HUES AS IN THE/
+14H INPUT FILES.)
4 FORMAT(53H AT THIS POINT, NAMELIST /GO2/ VARIABLES ARE AS THEY,
+11H WERE AFTER/
+40H THE LAST NAMELIST /GO2/ WAS SUBMITTED.)
5 FORMAT(54H TO USE THESE VALUES, SUBMIT AN EMPTY NAMELIST /GO2./
+50H TO OVERRIDE ANY OF THESE VALUES, SUBMIT A NON-EMP,
+18HTY NAMELIST /GO2/.)
6 FORMAT(42H SUBMIT NAMELIST /GO2/ IN NAMELIST FORMAT:)
7 FORMAT(53H NAMELIST /SENS/ CONTAINS VARIABLES THAT CONTROL THE/
+53H DISPLAY OF THE SENSITIVITY ANALYSIS. )
8 FORMAT(53H AT THIS POINT, THE TERMINAL DISPLAYS SENSITIVITY WI,
+18HTH RESPECT TO ONLY/
+63H GLOBAL SENSITIVITY VARIABLES.
+)
9 FORMAT(52H AT THIS POINT, THE TERMINAL SENSITIVITY DISPLAY IS/
+32H AS IT WAS ON THE PREVIOUS RUN.)
10 FORMAT(53H FOR THE SAME TERMINAL DISPLAY, SUBMIT AN EMPTY NAME,
+12HLIST /SENS./
+54H FOR A DIFFERENT TERMINAL DISPLAY, SUBMIT A NON-EMPTY,
+17H NAMELIST /SENS/.)

```

11 FORMAT(43H SUBMIT NAMELIST /SENS/ IN NAMELIST FORMAT:)

C

WRITE(6,1)

IF(MAXPMT.NE.1) GO TO 30

WRITE(6,2)

IF(ITERXX.EQ.1.OR.RERDXX.EQ.1) WRITE(6,3)

IF(ITERXX.GT.1.AND.RERDXX.NE.1) WRITE(6,4)

WRITE(6,5)

30 WRITE(6,6)

READ(5,GO2)

IF(EXITXX.EQ.1) RETURN

WRITE(6,1)

IF(MAXPMT.NE.1) GO TO 31

LDTOT=0+LDFPR +LDSRU +LDFR +LDUP +LDCOND+LDNRTS+LDRTS +LDRM

WRITE(6,7)

IF(LDTOT.EQ.0) WRITE(6,8)

IF(LDTOT.NE.0) WRITE(6,9)

WRITE(6,10)

31 WRITE(6,11)

READ(5,SENS)

C

RETURN

END

SUBROUTINE PRMPT4

```

C                                                    800827 113653779
C*****
C* 4TH OF PROMPTING ROUTINES THAT TELLS USER THAT LCC HAS BEEN *
C* COMPLETED AND THAT GIVES THE USER A CHANCE TO EXIT. *
C*****
C
COMMON /MAXPMT/ MAXPMT
COMMON /EXITXX/ EXITXX
INTEGER EXITXX

C
DATA ECHAR/1HE/,BK/1H /

C
2 FORMAT(1X/15H LCC COMPLETED.)
3 FORMAT(49H IF YOU WISH TO EXIT, HIT -E-, THEN HIT -RETURN-;,
+12H OTHERWISE,)
4 FORMAT(46H ADJUST TERMINAL TO NEW PAGE AND HIT -RETURN-.)
5 FORMAT(A1)

C
STR=BK
WRITE(6,2)
IF(MAXPMT.EQ.1) WRITE(6,3)
WRITE(6,4)
READ(5,5) STR
IF(STR.EQ.ECHAR) EXITXX=1

C
RETURN
END

```

SUBROUTINE PRMPT5

800827 113653826

```

C
C*****
C* 5TH OF PROMPTING ROUTINES. *
C* GIVES THE USER A CHANCE TO EXIT. *
C*****
C
COMMON /MAXPMT/ MAXPMT
COMMON /EXITXX/ EXITXX
INTEGER EXITXX

C
DATA ECHAR/1HE/,BK/1H /

C
3 FORMAT(49H IF YOU WISH TO EXIT, HIT -E-, THEN HIT -RETURN-;,
+12H OTHERWISE,)
4 FORMAT(46H ADJUST TERMINAL TO NEW PAGE AND HIT -RETURN-.)
5 FORMAT(A1)

C
STR=BK
IF(MAXPMT.EQ.1) WRITE(6,3)
WRITE(6,4)
READ(5,5) STR
IF(STR.EQ.ECHAR) EXITXX=1

C
RETURN
END

```

APPENDIX F

RLA PROGRAM FORTRAN SOURCE CODE

```

C*****
C
C
C*****
COMMON /EXITXX/ EXITXX
INTEGER EXITXX
COMMON /ITERXX/ ITERXX
COMMON /PRNTXX/ PRNTXX
INTEGER PRNTXX
COMMON /RERDXX/ RERDXX
INTEGER RERDXX
COMMON /NERRXX/ NERRXX
COMMON /NERRYY/ NERRYY
COMMON /REDOXX/ REDOXX
INTEGER REDOXX

C
1 FORMAT(1H1//22H PROGRAM STOPS DUE TO ,I4,
+ 16H ERRORS ON INPUT)

C
C
C
C
C*****
C* INITIALIZE SENSITIVITY PRINT PARAMETERS *
C*****
C
CALL INITAL
NERRXX=0
REWIND 11
REWIND 12
REWIND 13
CALL READ1
CALL READ2
CALL READ3
NERRYY=NERRXX
2 CONTINUE

C
C
C
C*****
C*****
C* PRINT THE INPUT DATA VALUES. *
```

```

C*****
C
    CALL ITAB1
    CALL ITAB2
    CALL ZTRAN
C
C
C
C*****
C*  STOP IF ANY ERRORS WERE FOUND ON INPUT.          *
C*****
C
    NERRXX=NERRYY
    IF(NERRXX.EQ.0) GO TO 4
    WRITE(7,1)NERRXX
    STOP
4 CONTINUE
C
C
C
C*****
C*  LCC CALCULATIONS                                *
C*****
C
    CALL ZISINO
    CALL INITAX
    CALL STEP0
    CALL STEP1
    CALL STEP2
    CALL STEP3
    CALL STEP4
C
C
C
C*****
C
C*****
C*  PRINT OUTPUT TABLES AT TERMINAL AND/OR OFFLINE PRINTER  *
C*****
C
    CALL OUT9A
    CALL OTAB1
C
C
C
C*****

```

888 CONTINUE
C
999 STOP
C
END

SUBROUTINE READ1

```

C
C***** 800620 115546520 *****
C* READS THE LRU/SRU CROSS REFERENCE DATA *
C* FILE FROM CHANNEL 11 *
C*****
C
COMMON /IL/ IL
COMMON /ILINO/ ILINO(120)
COMMON /ISRU/ ISRU(120,30)
COMMON /MXIL/ MXIL
COMMON /NDS/ NDS(120)
COMMON /QPA/ QPA(120,30)
COMMON /NERRXX/ NERRXX
COMMON /PRNTXX/ PRNTXX
INTEGER PRNTXX
DATA XXSTAR/1H*/
1 FORMAT(A1,I3,I2,14(I3,F2.0))
2 FORMAT(A1,5X,14(I3,F2.0))
3 FORMAT(A1)
4 FORMAT(/49H UNIT 11 ERROR: END OF FILE CARD NOT FOUND AFTER/
+17X,37HMAXIMUM NUMBER OF CARDS WERE READ IN.)
C
C
MXIL=0
DO 220 IXXX1=1,120
READ(11, 1) XXCOL1,IL,NDS(IL),(ISRU(IL,K1),QPA(IL,K1),K1=1,14)
IF(XXCOL1.EQ.XXSTAR) RETURN
MXIL=IXXX1
ILINO(IXXX1)=IL
J2=14
J3=14
C
211 CONTINUE
IF(.NOT.(NDS(IL).GT.J3.AND.NDS(IL).LE.30)) GO TO 210
J2=J3+1
J3=J2+13
READ(11, 2) XXCOL,(ISRU(IL,K1),QPA(IL,K1),K1=J2,J3)
IF(XXCOL.NE.XXSTAR) GO TO 211
210 CONTINUE
220 CONTINUE
C

```

```
READ(11, 3) XXCOL1  
IF(XXCOL1.EQ.XXSTAR) RETURN  
NERRXX=NERRXX+1  
WRITE(7,4)
```

C

```
RETURN  
END
```

SUBROUTINE READ2

800620 115551971

```
C
C*****
C* READS THE ITEM MAINTENANCE DATA FILE *
C* FROM CHANNEL 19 *
C*****
C
```

```
COMMON /BCMH/ BCMH(999)
COMMON /BMH/ BMH(999)
COMMON /COND/ COND(999)
COMMON /DMH/ DMH(999)
COMMON /FPR/ FPR(999)
COMMON /I/ I
COMMON /INO/ INO(999)
COMMON /IPCF/ IPCF(999)
REAL IPCF
COMMON /LRU/ LRU(999)
COMMON /MTBMI/ MTBMI(999,4)
REAL MTBMI
COMMON /MXI/ MXI
COMMON /NRTS/ NRTS(999)
REAL NRTS
COMMON /RIP/ RIP(999)
COMMON /RL/ RL(999)
INTEGER RL
COMMON /RMH/ RMH(999)
COMMON /RTS/ RTS(999)
COMMON /NERRXX/ NERRXX
COMMON /PRNTXX/ PRNTXX
INTEGER PRNTXX
DATA XXSTAR/1H*/
```

```
1 FORMAT(A1,I3,4F8.0,F4.3,F3.2,F5.2,3F4.3,4F4.2,I1)
2 FORMAT(A1)
3 FORMAT(/49H UNIT 13 ERROR: END OF FILE CARD NOT FOUND AFTER/
+17X,37HMAXIMUM NUMBER OF CARDS WERE READ IN.)
```

C
C

```
MXI=0
DO 210 IXXX1=1,999
  LRU(I)=-1
  READ(13, 1) XXCOL1,I,(MTBMI(I,K1),K1=1,4),FPR(I),RIP(I),IPCF(I),
+   RTS(I),NRTS(I),COND(I),RMH(I),BCMH(I),BMH(I),DMH(I),RL(I)
  IF(XXCOL1.EQ.XXSTAR) RETURN
  MXI=IXXX1
  INO(IXXX1)=I
```

210 CONTINUE

C

```
READ(13, 2) XXCOL1  
IF(XXCOL1.EQ.XXSTAR) RETURN  
NERRXX=NERRXX+1  
WRITE(7,3)
```

C

```
RETURN  
END
```

SUBROUTINE READ3

800620 115554898

C

C*****

C* READS THE TIAC(I,6) FILE *

C* FROM CHANNEL 12 *

C*****

C

```
COMMON /DUM/ DUM
INTEGER DUM
COMMON /DUMM/ DUMM
INTEGER DUMM
COMMON /I/ I
COMMON /INO/ INO(999)
COMMON /MXD/ MXD
COMMON /MXDD/ MXDD
COMMON /MXI/ MXI
COMMON /TIAC/ TIAC(999,6)
INTEGER FINISH
INTEGER START
INTEGER XI
```

```
1 FORMAT(I3)
2 FORMAT(6(I3,1X,F8.1,1X))
3 FORMAT(A1)
```

C

C

```
MXDD=0
DO 230 DUMM=1,6
  READ(12, 1) XI
  MXDD=DUMM
  MXD=0
  DO 220 DUM=1,XI,6
    START=DUM
    FINISH=DUM+5
    IF(.NOT.(FINISH.GT.XI)) GO TO 210
    FINISH=XI
210  CONTINUE
    READ(12, 2) (INO(K1),TIAC(INO(K1),DUMM),K1=START,FINISH)
220  CONTINUE
    READ(12, 3) XXSIGN
    MXDD=DUMM
230 CONTINUE
```

C

```
RETURN
END
```

SUBROUTINE ITAB1

C 800620 115556296
 C*****
 C* LRU/SRU CROSS REFERENCE DATA *
 C*****
 C

COMMON /PRNTXX/ PRNTXX
 INTEGER PRNTXX
 COMMON /FULLXX/ FULLXX
 INTEGER FULLXX
 COMMON /IL/ IL
 COMMON /ILINO/ ILINO(120)
 COMMON /ISRU/ ISRU(120,30)
 COMMON /MXIL/ MXIL
 COMMON /NDS/ NDS(120)
 COMMON /QPA/ QPA(120,30)
 COMMON /XXCOL1/ XXCOL1
 DATA XXSTAR/1H*/

1 FORMAT(1H1/30X,59HINPUT TABLE 1: LRU/SRU CROSS REFERENCE DATA (FR
 +OM FILE 8B))
 2 FORMAT(59X,11H(CONTINUED)//)
 3 FORMAT(/9X,4H#SRU,4X,12HSRU SRU ,3X,12HSRU SRU ,3X,12HSRU
 + SRU ,3X,12HSRU SRU ,3X,12HSRU SRU ,3X,12HSRU SRU
 +,3X,12HSRU SRU /1X,3HLRU,5X,5HTYPES,3X,12HINDEX QUAN-,3X,12HI
 +NDEX QUAN-,3X,12HINDEX QUAN-,3X,12HINDEX QUAN-,3X,12HINDEX QUA
 +N-,3X,12HINDEX QUAN-,3X,12HINDEX QUAN-/1X,5HINDEX,3X,6HIN LRU,2X
 +,12HNO. TITY ,3X,12HNO. TITY ,3X,12HNO. TITY ,3X,12HNO.
 + TITY ,3X,12HNO. TITY ,3X,12HNO. TITY ,3X,11HNO. TITY/1X
 +,4H(IL),4X,5H(NDS),3X,12H(ISRU) (QPA),3X,12H(ISRU) (QPA),3X,12H(IS
 +RU) (QPA),3X,12H(ISRU) (QPA),3X,12H(ISRU) (QPA),3X,12H(ISRU) (QPA)
 +,3X,12H(ISRU) (QPA)//)
 4 FORMAT(2X,I3,5X,I3,5X,I3,3X,F4.0,5X,I3,3X,F4.0,5X,I3,3X,F4.0,5X,I3
 +,3X,F4.0,5X,I3,3X,F4.0,5X,I3,3X,F4.0,5X,I3,3X,F4.0/18X,I3,3X,F4.0,
 +5X,I3,3X,F4.0,5X,I3,3X,F4.0,5X,I3,3X,F4.0,5X,I3,3X,F4.0,5X,I3,3X,F
 +4.0,5X,I3,3X,F4.0)
 5 FORMAT(18X,I3,3X,F4.0,5X,I3,3X,F4.0,5X,I3,3X,F4.0,5X,I3
 +,3X,F4.0,5X,I3,3X,F4.0,5X,I3,3X,F4.0,5X,I3,3X,F4.0/18X,I3,3X,F4.0,
 +5X,I3,3X,F4.0,5X,I3,3X,F4.0,5X,I3,3X,F4.0,5X,I3,3X,F4.0,5X,I3,3X,F
 +4.0,5X,I3,3X,F4.0)

C
 C
 C
 C

IPAGE=40
 IFLAG=1
 DO 240 IXXX1=1,MXIL
 IL=ILINO(IXXX1)

```

        IF(.NOT.(IPAGE.EQ.40)) GO TO 220
        WRITE( 7, 1)
        IPAGE=1
        IF(.NOT.(IFLAG.NE.1)) GO TO 210
        WRITE( 7, 2)
210     CONTINUE
        WRITE( 7, 3)
220     CONTINUE
        WRITE( 7, 4) IL,NDS(IL),(ISRU(IL,K1),QPA(IL,K1),K1=1,14)
        J2=14
        J3=14
C
241     CONTINUE
        IF(.NOT.(NDS(IL).GT.J3.AND.NDS(IL).LE.30)) GO TO 230
        J2=J3+1
        J3=J2+13
        WRITE( 7, 5) (ISRU(IL,K1),QPA(IL,K1),K1=J2,J3)
        IF(XXCOL.NE.XXSTAR) GO TO 241
230     CONTINUE
        IFLAG=0
        IPAGE=IPAGE+1
240 CONTINUE
C
        RETURN
        END

```

SUBROUTINE ITAB2

800620 115602424

C
C*****
C* PRINTS ITEM REPAIR-LEVEL-DEVELOPMENT COSTS *
C*****
C

COMMON /PRNTXX/ PRNTXX
INTEGER PRNTXX
COMMON /FULLXX/ FULLXX
INTEGER FULLXX
COMMON /I/ I
COMMON /INO/ INO(999)
COMMON /MXI/ MXI
COMMON /TIAC/ TIAC(999,6)
1 FORMAT(1H1/38X,50HINPUT TABLE 2: TOTAL ITEM SUPPORT COST - TIAC(I,
+R)/48X,31H(COSTS IN THOUSANDS OF DOLLARS))
2 FORMAT(58X,11H(CONTINUED)//)
3 FORMAT(/49X,31HGLOBAL MAINTENANCE STRATEGY - R/26X,4HITEM,7X,68H-
+-----
+--/26X,5HINDEX,9X,1H1,11X,1H2,11X,1H3,11X,1H4,11X,1H5,11X,1H6//)
4 FORMAT(27X,I3,7X,6(F8.1,4X))

C
C
C
C
C

IPAGE=40
IFLAG=1
DO 230 IXXX1=1,MXI
I=INO(IXXX1)
IF(.NOT.(IPAGE.EQ.40)) GO TO 220
WRITE(7, 1)
IPAGE=1
IF(.NOT.(IFLAG.NE.1)) GO TO 210
WRITE(7, 2)
210 CONTINUE
WRITE(7, 3)
220 CONTINUE
WRITE(7, 4) I,(TIAC(I,K1),K1=1,6)
IFLAG=0
IPAGE=IPAGE+1
230 CONTINUE

C

RETURN
END

SUBROUTINE ZTRAN

800620 115614729

C
 C*****
 C* CALCULATES NTL(IS),TQPA, AND LRU(I) *
 C*****
 C

COMMON /PRNTXX/ PRNTXX
 INTEGER PRNTXX
 COMMON /IL/ IL
 COMMON /ILINO/ ILINO(120)
 COMMON /IS/ IS
 COMMON /ISRU/ ISRU(120,30)
 COMMON /ISXREF/ ISXREF(999)
 COMMON /LRU/ LRU(999)
 COMMON /MXIL/ MXIL
 COMMON /MXIS/ MXIS
 COMMON /NDL/ NDL(999)
 COMMON /NDS/ NDS(120)
 COMMON /NERRYY/ NERRYY
 COMMON /NTL/ NTL(999)
 COMMON /QPA/ QPA(120,30)
 COMMON /TQPA/ TQPA(999,120)
 INTEGER TQPA

1 FORMAT(/1X,17H*** ERROR - ITEM ,I3,39H IS LISTED AS BOTH AN LRU A
 +ND AN SRU IN,16H THE QPA MATRIX.)
 2 FORMAT(/1X,17H*** ERROR - ITEM ,I3,39H IS LISTED AS BOTH AN LRU A
 +ND AN SRU IN,16H THE QPA MATRIX.)

C
 C
 C

DO 270 IXXX1=1,MXIL
 IL=ILINO(IXXX1)
 IF(.NOT.(LRU(IL).EQ.0)) GO TO 210
 WRITE(7,1)IL
 NERRYY=NERRYY+1
 210 CONTINUE
 IF(.NOT.(LRU(IL).EQ.-1)) GO TO 220
 LRU(IL)=1
 220 CONTINUE
 C.....THESE 4 STMTS IMPLEMENT THE POINTER MATRIX
 NXXX1=NDS(IL)
 IF(NXXX1.EQ.0) GO TO 260
 DO 250 JXXX1=1,NXXX1
 IS=ISRU(IL,JXXX1)
 IF(.NOT.(LRU(IS).EQ.1)) GO TO 230
 WRITE(7,2)IS
 NERRYY=NERRYY+1

```
230    CONTINUE
      IF(.NOT.(LRU(IS).EQ.-1)) GO TO 240
      LRU(IS)=0
240    CONTINUE
      NTL(IS)=NTL(IS)+QPA(IL,JXXX1)
      NDL(IS)=NDL(IS)+1
      TQPA(IS,NDL(IS))=IL
250    CONTINUE
260    CONTINUE
270    CONTINUE
C      RETURN
      END
```

SUBROUTINE ZISINO

800620 115622448

```

C
C*****
C* CALCULATES ISINO(999)
C*****
C
COMMON /I/ I
COMMON /INO/ INO(999)
COMMON /ISINO/ ISINO(999)
COMMON /LRU/ LRU(999)
COMMON /MXI/ MXI
COMMON /MXIS/ MXIS

C
C
MXIS=0
JO 220 IXXX1=1,MXI
I=INO(IXXX1)
IF(.NOT.(LRU(I).EQ.0)) GO TO 210
MXIS=MXIS+1
ISINO(MXIS)=I
210 CONTINUE
220 CONTINUE

C
RETURN
END

```

SUBROUTINE INITAX

```

C
C***** 800620 115622777 *****
C* INITIALIZES *
C*****
C
COMMON /AIAC/ AIAC(999,3)
COMMON /I/ I
COMMON /IL/ IL
COMMON /ILINO/ ILINO(120)
COMMON /INO/ INO(999)
COMMON /IS/ IS
COMMON /ISINO/ ISINO(999)
COMMON /J/ J
COMMON /LCRL/ LCRL(999)
COMMON /LCRLS/ LCRLS(120,999)
COMMON /LCRS/ LCRS(999,3)
COMMON /MXI/ MXI
COMMON /MXIL/ MXIL
COMMON /MXIS/ MXIS
COMMON /MXJ/ MXJ
COMMON /TLAC/ TLAC(120)
COMMON /TSAC/ TSAC(120,3)

C
C
DO 230 IXXX1=1,MXIL
  IL=ILINO(IXXX1)
  TLAC(IL)=0.
  DO 210 IXXX2=1,MXIS
    IS=ISINO(IXXX2)
    LCRLS(IL,IS)=0
210  CONTINUE
  DO 220 J=1,MXJ
    TSAC(IL,J)=0.
220  CONTINUE
C
230 CONTINUE
DO 250 IXXX1=1,MXIS
  IS=ISINO(IXXX1)
  DO 240 J=1,MXJ
    AIAC(IS,J)=0.
    LCRS(IS,J)=0
C
240 CONTINUE
250 CONTINUE

```

```
DO 260 IXXX1=1,MXI  
  I=INO(IXXX1)  
  LCRL(I)=0  
260 CONTINUE  
C  
  RETURN  
  END
```

SUBROUTINE STEPO

800620 115623469

C
 C*****
 C* SETS SOME LCRL'S *
 C*****
 C

COMMON /COND/ COND(999)
 COMMON /IL/ IL
 COMMON /ILINO/ ILINO(120)
 COMMON /IS/ IS
 COMMON /ISINO/ ISINO(999)
 COMMON /ISRU/ ISRU(120,30)
 COMMON /ISXREF/ ISXREF(999)
 COMMON /LCRL/ LCRL(999)
 COMMON /MXIL/ MXIL
 COMMON /MXIS/ MXIS
 COMMON /NDS/ NDS(120)

C
 C

DO 240 IXXX1=1,MXIL
 IL=ILINO(IXXX1)
 IF(.NOT.(COND(IL).EQ.1)) GO TO 230
 LCRL(IL)=3

C.....THESE 4 STMTS IMPLEMENT THE POINTER MATRIX

NXXX1=NDS(IL)
 IF(NXXX1.EQ.0) GO TO 220
 DO 210 JXXX1=1,NXXX1
 IS=ISRU(IL,JXXX1)
 LCRL(IS)=3

210 CONTINUE
 220 CONTINUE
 230 CONTINUE
 240 CONTINUE

DO 260 IXXX1=1,MXIS
 IS=ISINO(IXXX1)
 IF(.NOT.(COND(IS).EQ.1)) GO TO 250
 LCRL(IS)=3

250 CONTINUE
 260 CONTINUE

C

RETURN
 END

SUBROUTINE STEP1

```

C
C***** 800620 115627248 *****
C* CALCULATES TSAC(IS,J) AND LCRS(IS,J) *
C*****
C
COMMON /DUM/ DUM
INTEGER DUM
COMMON /IS/ IS
COMMON /ISINO/ ISINO(999)
COMMON /LCRS/ LCRS(999,3)
COMMON /LRU/ LRU(999)
COMMON /MXD/ MXD
COMMON /MXIS/ MXIS
COMMON /TIAC/ TIAC(999,6)
COMMON /TSAC/ TSAC(120,3)

C
C
DO 260 IXXX1=1,MXIS
  IS=ISINO(IXXX1)
  TSAC(IS,1)=AMIN1(TIAC(IS,1),TIAC(IS,2),TIAC(IS,3))
  TSAC(IS,2)=AMIN1(TIAC(IS,4),TIAC(IS,5))
  TSAC(IS,3)=TIAC(IS,6)
  IF(.NOT.(LRU(IS).EQ.0)) GO TO 250
  DO 220 DUM=1,3
    IF(.NOT.(TIAC(IS,DUM).EQ.TSAC(IS,1))) GO TO 210
    LCRS(IS,1)=DUM
  210 CONTINUE
  220 CONTINUE
  DO 240 DUM=4,5
    IF(.NOT.(TIAC(IS,DUM).EQ.TSAC(IS,2))) GO TO 230
    LCRS(IS,2)=DUM
  230 CONTINUE
  240 CONTINUE
  LCRS(IS,3)=6
  250 CONTINUE
  260 CONTINUE
C
RETURN
END

```

SUBROUTINE STEP2

800620 115632728

```

C
C*****
C*  CALCULATES TSAC(IL,J) AND TLAC(IL)
C*****
C
COMMON /IL/ IL
COMMON /ILINO/ ILINO(120)
COMMON /IS/ IS
COMMON /ISRU/ ISRU(120,30)
COMMON /ISXREF/ ISXREF(999)
COMMON /J/ J
COMMON /LCRS/ LCRS(999,3)
COMMON /MXIL/ MXIL
COMMON /MXIS/ MXIS
COMMON /MXJ/ MXJ
COMMON /NDS/ NDS(120)
COMMON /NTL/ NTL(999)
COMMON /QPA/ QPA(120,30)
COMMON /TLAC/ TLAC(120)
COMMON /TSAC/ TSAC(120,3)
COMMON /TIAC/ TIAC(999,6)

C
DO 240 J=1,MXJ
  DO 230 IXXX2=1,MXIL
    IL=ILINO(IXXX2)
C.....THESE 4 STMTS IMPLEMENT THE POINTER MATRIX
    NXXX1=NDS(IL)
    IF(NXXX1.EQ.0) GO TO 220
    DO 210 JXXX1=1,NXXX1
      IS=ISRU(IL,JXXX1)
      TSAC(IL,J)=TSAC(IL,J)+(TSAC(IS,J)*QPA(IL,JXXX1)/NTL(IS))
      +
      +(TIAC(IL,LCRS(IS,J))/NDS(IL))
210    CONTINUE
      GO TO 230
220    CONTINUE
      IF(J.EQ.1) K1=1
      IF(J.NE.1) K1=2*J
      TSAC(IL,J)=TIAC(IL,K1)
230    CONTINUE
240    CONTINUE
      DO 250 IXXX1=1,MXIL
        IL=ILINO(IXXX1)
        TLAC(IL)=AMIN1(TSAC(IL,1),TSAC(IL,2),TSAC(IL,3))
250    CONTINUE
C
RETURN
END

```

SUBROUTINE STEP3

800620 115636632

C
 C*****
 C* CALCULATES LCRL(IL) AND LCRS(IL,IS) *
 C*****
 C

COMMON /IL/ IL
 COMMON /ILINO/ ILINO(120)
 COMMON /IS/ IS
 COMMON /ISINO/ ISINO(999)
 COMMON /ISRU/ ISRU(120,30)
 COMMON /J/ J
 COMMON /LCRL/ LCRL(999)
 COMMON /LCRLS/ LCRLS(120,999)
 COMMON /LCRS/ LCRS(999,3)
 COMMON /MXIL/ MXIL
 COMMON /MXIS/ MXIS
 COMMON /MXJ/ MXJ
 COMMON /NDS/ NDS(120)
 COMMON /TLAC/ TLAC(120)
 COMMON /TSAC/ TSAC(120,3)

C
 C

DO 230 IXXX1=1,MXIL
 IL=ILINO(IXXX1)
 DO 220 J=1,MXJ
 IF(.NOT.((TSAC(IL,J).EQ.TLAC(IL)).AND.(LCRL(IL).EQ.
 + 0))) GO TO 210
 LCRL(IL)=J
 210 CONTINUE
 220 CONTINUE
 230 CONTINUE
 DO 280 IXXX1=1,MXIL
 IL=ILINO(IXXX1)
 C.....THESE 4 STATEMENTS IMPLEMENT THE POINTER MATRIX
 NXXX1=NDS(IL)
 IF(NXXX1.EQ.0) GO TO 275
 DO 270 IXXX2=1,NXXX1
 IS=ISRU(IL,IXXX2)
 IF(.NOT.(LCRS(IS,LCRL(IL)).EQ.1)) GO TO 240
 LCRLS(IL,IS)=1
 240 CONTINUE
 IF(.NOT.((LCRS(IS,LCRL(IL)).EQ.3).OR.(LCRS(IS,LCRL(IL)).EQ.
 + 5).OR.(LCRS(IS,LCRL(IL)).EQ.6))) GO TO 260
 LCRLS(IL,IS)=3
 260 CONTINUE

```

      IF(.NOT.((LCRS(IS,LCRL(IL)).EQ.2).OR.(LCRS(IS,LCRL(IL)).EQ.
+      4))) GO TO 250
      LCRLS(IL,IS)=2
250   CONTINUE
270   CONTINUE
275   CONTINUE
280 CONTINUE
C
      RETURN
      END

```

SUBROUTINE STEP4

```

C
C***** 800620 115640822 *****
C* CALCULATES LCRL(I) *
C*****
C
COMMON /AIAC/ AIAC(999,3)
COMMON /DUM/ DUM
INTEGER DUM
COMMON /IL/ IL
COMMON /ILINO/ ILINO(120)
COMMON /ILXREF/ ILXREF(120)
COMMON /IS/ IS
COMMON /ISINO/ ISINO(999)
COMMON /ISRU/ ISRU(120,30)
COMMON /ISXREF/ ISXREF(999)
COMMON /LCRL/ LCRL(999)
COMMON /LCRLS/ LCRLS(120,999)
COMMON /MXD/ MXD
COMMON /MXIL/ MXIL
COMMON /MXIS/ MXIS
COMMON /NDL/ NDL(999)
COMMON /NDS/ NDS(120)
COMMON /NTL/ NTL(999)
COMMON /QPA/ QPA(120,30)
COMMON /TIAC/ TIAC(999,6)
COMMON /TQPA/ TQPA(999,120)
INTEGER TQPA
INTEGER COUNT
INTEGER R2
INTEGER TEST1
INTEGER TEST1A
INTEGER TEST1B
INTEGER TEST2
INTEGER TEST3
INTEGER TEST4
INTEGER TEMP

C
DO 250 DUM=2,3
  DO 240 IXXX2=1,MXIL
    IL=ILINO(IXXX2)
    R2=DUM
    IF(.NOT.(LCRL(IL).EQ.2)) GO TO 210
    R2=R2+2
  210 CONTINUE
C.....THESE 4 STMTS IMPLEMENT THE POINTER MATRIX
      NXXX1=NDS(IL)

```

```

        IF(NXXX1.EQ.0) GO TO 230
        DO 220 JXXX1=1,NXXX1
            IS=ISRU(IL,JXXX1)
            AIAC(IS,DUM)=AIAC(IS,DUM)+TIAC(IS,R2)*QPA(IL,JXXX1)/NTL(IS)
220     CONTINUE
230     CONTINUE
240     CONTINUE
250     CONTINUE
        DO 430 IXXX1=1,MXIS
            TEST1=0
            TEST1A=1
            TEST1B=1
            TEST2=0
            TEST3=0
            TEST4=0
            IS=ISINO(IXXX1)
            COUNT=0
            IF(.NOT.(NDL(IS).GT.1)) GO TO 310
            TEST1A=0
C.....THESE 4 STMTS IMPLEMENT THE POINTER MATRIX
            NXXX1=NTL(IS)
            IF(NXXX1.EQ.0) GO TO 300
            DO 290 JXXX1=1,NXXX1
                IL=TQPA(IS,JXXX1)
                COUNT=COUNT+1
                IF(.NOT.(COUNT.EQ.1)) GO TO 260
                TEMP=LCRLS(IL,IS)
260         CONTINUE
                IF(.NOT.((COUNT.GT.1).AND.(TEST1B.EQ.1))) GO TO 280
                IF(.NOT.(LCRLS(IL,IS).NE.TEMP)) GO TO 270
                TEST1B=0
270         CONTINUE
                TEMP=LCRLS(IL,IS)
280         CONTINUE
290         CONTINUE
300         CONTINUE
310         CONTINUE
            IF(.NOT.(NDL(IS).EQ.1)) GO TO 311
            IL=TQPA(IS,1)
            TEMP=LCRLS(IL,IS)
311         CONTINUE
            IF(.NOT.((TEST1A.EQ.1).OR.(TEST1B.EQ.1))) GO TO 320
            TEST1=1
320         CONTINUE
            IF(.NOT.((TEST1A.EQ.0).AND.(TEST1B.EQ.0))) GO TO 360
C.....THESE 4 STMTS IMPLEMENT THE POINTER MATRIX
            NXXX1=NTL(IS)

```

```

        IF(NXXX1.EQ.0) GO TO 350
        DO 340 JXXX1=1,NXXX1
            IL=TQPA(IS,JXXX1)
            IF(.NOT.(LCRL(IL).EQ.3)) GO TO 330
                TEST2=1
330         CONTINUE
340         CONTINUE
350         CONTINUE
360         CONTINUE
        IF(.NOT.(((TEST1A.EQ.0).AND.(TEST1B.EQ.0)).AND.(TEST2.EQ.
+         0))) GO TO 390
            IF(.NOT.(AIAC(IS,3).LE.AIAC(IS,2))) GO TO 370
                TEST3=1
370         CONTINUE
            IF(.NOT.(TEST3.EQ.0)) GO TO 380
                TEST4=1
380         CONTINUE
390         CONTINUE
            IF(.NOT.((TEST1.EQ.1).AND.(LCRL(IS).EQ.0))) GO TO 400
                LCRL(IS)=TEMP
400         CONTINUE
            IF(.NOT.(((TEST2.EQ.1).OR.(TEST3.EQ.1)).AND.(LCRL(IS).EQ.
+         0))) GO TO 410
                LCRL(IS)=3
410         CONTINUE
            IF(.NOT.((TEST4.EQ.1).AND.(LCRL(IS).EQ.0))) GO TO 420
                LCRL(IS)=2
420         CONTINUE
430         CONTINUE
C
        RETURN
        END

```

SUBROUTINE OUT9A

C 800620 115652880
 C*****
 C* WRITES THE ITEM MAINTENANCE DATA FILE *
 C* TO CHANNEL 14 *
 C*****
 C

COMMON /PRNTXX/ PRNTXX
 INTEGER PRNTXX
 COMMON /FULLXX/ FULLXX
 INTEGER FULLXX
 COMMON /BCMH/ BCMH(999)
 COMMON /BMH/ BMH(999)
 COMMON /COND/ COND(999)
 COMMON /DMH/ DMH(999)
 COMMON /FPR/ FPR(999)
 COMMON /I/ I
 COMMON /INO/ INO(999)
 COMMON /IPCF/ IPCF(999)
 REAL IPCF
 COMMON /LCRL/ LCRL(999)
 COMMON /MTBMI/ MTBMI(999,4)
 REAL MTBMI
 COMMON /MXI/ MXI
 COMMON /NRTS/ NRTS(999)
 REAL NRTS
 COMMON /RIP/ RIP(999)
 COMMON /RMH/ RMH(999)
 COMMON /RTS/ RTS(999)
 1 FORMAT(1H ,I3,4I8,I4,I3,I5,3I4,4I4,I1)
 2 FORMAT(1H*)

C
 C
 C
 C

DO 210 IXXX1=1,MXI
 I=INO(IXXX1)
 I1=MTBMI(I,1)
 I2=MTBMI(I,2)
 I3=MTBMI(I,3)
 I4=MTBMI(I,4)
 I5=INT(FPR(I)*1000+.99)
 I6=INT(RIP(I)*100+.99)
 I7=INT(IPCF(I)*100+.99)
 I8=INT(RTS(I)*1000+.99)
 I9=INT(NRTS(I)*1000+.99)
 I10=INT(COND(I)*1000+.99)

```

      I11=INT(RMH(I)*100+.99)
      I12=INT(BCMh(I)*100+.99)
      I13=INT(BMH(I)*100+.99)
      I14=INT(DMH(I)*100+.99)
      WRITE(14, 1) I, I1, I2, I3, I4, I5, I6, I7, I8, I9, I10, I11, I12, I13,
+I14, LCRL(I)
210 CONTINUE
      WRITE(14, 2)
C
      RETURN
      END

```

SUBROUTINE OTAB1

C 800620 11565511
 C*****
 C* REPAIR LEVEL ANALYSIS *
 C*****
 C

```

COMMON /PRNTXX/ PRNTXX
INTEGER PRNTXX
COMMON /FULLXX/ FULLXX
INTEGER FULLXX
COMMON /I/ I
COMMON /INO/ INO(999)
COMMON /LCRL/ LCRL(999)
COMMON /LRU/ LRU(999)
COMMON /MXI/ MXI
COMMON /RL/ RL(999)
INTEGER RL
DATA XXBL/1H /
DATA XXSTAR/1H*/
1 FORMAT(1H1/46X,37HOUTPUT TABLE 1: REPAIR LEVEL ANALYSIS)
2 FORMAT(59X,11H(CONTINUED))
3 FORMAT(/40X,7HLRU (1),9X,17HREPAIR LEVEL - RL,10X,7HRESULTS/29X,4
+HITEM,7X,2HOR,11X,24H-----,6X,10HDIFF. FROM/29X
+,5HINDEX,6X,7HSRU (0),6X,4HBASE,4X,5HDEPOT,4X,7HDISCARD,6X,16HCONT
+RATOR INPUT/30X,3H(1),20X,3H(1),6X,3H(2),7X,3H(3))
4 FORMAT(30X,I3,9X,I1,11X,A1,8X,A1,9X,A1,16X,A1)

```

C
 C
 C
 C

```

IPAGE=40
IFLAG=1
XXR=XXBL
XXB=XXBL
XXF=XXBL
XXD=XXBL
DO 270 IXXX1=1,MXI
  I=INO(IXXX1)
  IF(.NOT.(RL(I).NE.LCRL(I))) GO TO 210
  XXR=XXSTAR
210 CONTINUE
  IF(.NOT.(LCRL(I).EQ.1)) GO TO 220
  XXB=XXSTAR
220 CONTINUE
  IF(.NOT.(LCRL(I).EQ.2)) GO TO 230
  XXD=XXSTAR
230 CONTINUE

```

```

        IF(.NOT.(LCRL(I).EQ.3)) GO TO 240
        XXF=XXSTAR
240    CONTINUE
        IF(.NOT.(IPAGE.EQ.40)) GO TO 260
        WRITE( 7, 1)
        IPAGE=1
        IF(.NOT.(IFLAG.NE.1)) GO TO 250
        WRITE( 7, 2)
250    CONTINUE
        WRITE( 7, 3)
260    CONTINUE
        WRITE( 7, 4) I,LRU(I),XXB,XXD,XXF,XXR
        IFLAG=0
        IPAGE=IPAGE+1
        XXR=XXBL
        XXB=XXBL
        XXF=XXBL
        XXD=XXBL
270 CONTINUE
C
    RETURN
    END

```

SUBROUTINE INITAL

800620 115702567

C
C.....INITIALIZES VARIABLES TO DEFAULT VALUES.
C

```
COMMON /NTABXX/ NTABXX
COMMON /NERRXX/ NERRXX
COMMON /DINO/ DINO(6)
INTEGER DINO
COMMON /DIXREF/ DIXREF(1)
INTEGER DIXREF
COMMON /DUINO/ DUINO(6)
INTEGER DUINO
COMMON /DUM/ DUM
INTEGER DUM
COMMON /DUMM/ DUMM
INTEGER DUMM
COMMON /DXREF/ DXREF(1)
INTEGER DXREF
COMMON /I/ I
COMMON /IL/ IL
COMMON /ILINO/ ILINO(120)
COMMON /ILXREF/ ILXREF(120)
COMMON /INO/ INO(999)
COMMON /IS/ IS
COMMON /ISINO/ ISINO(999)
COMMON /ISRU/ ISRU(120,30)
COMMON /ISXREF/ ISXREF(999)
COMMON /IXREF/ IXREF(999)
COMMON /J/ J
COMMON /JINO/ JINO(3)
COMMON /JXREF/ JXREF(1)
COMMON /MXD/ MXD
COMMON /MXDD/ MXDD
COMMON /MXI/ MXI
COMMON /MXIL/ MXIL
COMMON /MXIS/ MXIS
COMMON /MXJ/ MXJ
COMMON /NDS/ NDS(120)
COMMON /NTL/ NTL(999)
COMMON /QPA/ QPA(120,30)
COMMON /TPA/ TPA(999,120)
COMMON /TQPA/ TQPA(999,120)
INTEGER TQPA
```

C
NTABXX=0
NERRXX=0
C

```

DO    10 DUMM=1,6
      DINO(DUMM)=DUMM
10 CONTINUE
C
MXI=999
MXJ=3
MXD=6
MXIL=120
MXIS=999
MXDD=6
C
DO    30 DUM=1,6
      DUINO(DUM)=DUM
30 CONTINUE
C
DO    40 IXXX1=1,1
      DXREF(IXXX1)=IXXX1
      DIXREF(IXXX1)=IXXX1
      JXREF(IXXX1)=IXXX1
40 CONTINUE
C
DO    50 I=1,999
      INO(I)=I
50 CONTINUE
C
DO    60 IL=1,120
      NDS(IL)=0
      DO    60 IXXX=1,30
          ISRU(IL,IXXX)=0
          QPA(IL,IXXX)=0.00000
60 CONTINUE
C
DO    70 IS=1,999
      NTL(IS)=0
      DO    70 IXXX=1,120
          TQPA(IS,IXXX)=0
          TPA(IS,IXXX)=0.00000
70 CONTINUE
C
DO    80 IL=1,120
      ILINO(IL)=IL
80 CONTINUE
C
DO    90 IXXX1=1,999
      IXREF(IXXX1)=IXXX1
      ISXREF(IXXX1)=IXXX1
90 CONTINUE

```

```
C      DO 100 IS=1,999
        ISINO(IS)=IS
100 CONTINUE
C      DO 110 IXXX1=1,120
        ILXREF(IXXX1)=IXXX1
110 CONTINUE
C      DO 120 J=1,3
        JINO(J)=J
120 CONTINUE
C      RETURN
      END
```