LABEL 000000000PRINTER00175100CC EX OBJECT/READ;FILE SOURCEFILE=SYMBOL/LIST;END←

OBJECT /READ

SYMBOL/LIST

```
BEGIN REAL COMMON;                                                          00010000
%% LIST/CANDE SOURCE PROGRAM. 11^72. %%%                                    00010100
  COMMENT: * TITLE: B5500/B5700 MARK XIV SYSTEM RELEASE              *      00010110
           * FILE ID: SYMBOL/LIST  TAPE ID: SYMBOL2/FILE000          *      00010111
           * THIS MATERIAL IS PROPRIETARY TO BURROUGHS CORPORATION    *      00010112
           * AND IS NOT TO BE REPRODUCED, USED, OR DISCLOSED          *      00010113
           * EXCEPT IN ACCORDANCE WITH PROGRAM LICENSE OR UPON        *      00010114
           * WRITTEN AUTHORIZATION OF THE PATENT DIVISION OF          *      00010115
           * BURROUGHS CORPORATION, DETROIT, MICHIGAN 48232           *      00010116
           *                                                          *      00010117
           * COPYRIGHT (C) 1971, 1972 BURROUGHS CORPORATION           *      00010118
           *  AA320206   AA332366   AA386657                          *;     00010119
%% RANDOM UPDATE VERSION. %%%                                               00010200
COMMENT PROGRAM EXECUTION PROCEEDS AS FOLLOWS:                              00010300
    1) READ "1P" FILE, PLACING TAB WORDS IN ARRAY "SORTA"                   00010400
       UNTIL EOFMARK ( 100000000 ) IS OBTAINED.                             00010500
       EACH "1P" TAB WORD IS CONSTRUCTED AS:                                00010600
        WD.[1:2]  = TYPE CODE                                               00010700
          CODE=0, INPUT RECORD, SEQ. NO. TYPED IN BY USER                   00010800
          CODE=2, INPUT RECORD, SEQ. NO. NOT TYPED IN BY USER               00010900
          CODE=1, SEQ. NO. FOLLOWED BY GROUP MARK ( DELETION )              00011000
          CODE=3, "FIX" INSTRUCTION                                         00011100
        WD.[ 4: 4]= WD. NO. OF "BASE"                                       00011200
          ("BASE" WORDS CONTAIN DISK ADDRESSES OF TANKFILE ROWS)            00011300
        WD.[ 8: 8]= RELATIVE SEG. NO. IN TANKFILE ROW                       00011400
        WD.[16: 5]= OFFSET (IN WDS.) OF INPUT RECORD IN SEGMENT             00011500
        WD.[21:27]= SEQUENCE NO. (IN OCTAL) ASSOCIATED WITH RECORD          00011600
    2) SORT ARRAY "SORTA" ACCORDING TO SEQ. NUMBER FIELD                    00011700
    3) MOVE SORTED "1P" WORDS TO ARRAY "SORTAA", 32 WDS/ROW                 00011800
    4) MERGE SORTED "1P" WORDS WITH "1I" WORDS.                             00011900
       EACH "1T" WORD IS CONSTRUCTED AS:                                    00012000
         WD.[ 4:16]= LOCATION (RECORD NO.) OF RECORD IN SOURCE FILE         00012100
         WD.[21:27]= SEQ.NO. ( OCTAL ) OF RECORD                            00012200
       ARRAY "CHNGS" IS USED TO STORE "CHANGE" INSTRUCTIONS                 00012300
       DURING THE MERGE, EACH "CHNGS" WORD IS CONSTRUCTED AS:               00012400
         WD.[32:16]=LOCATION (RECORD NO.) OF RECORD IN SOURCE FILE          00012500
         WD.[1:1]=1 IF A "DELETE" INSTRUCTION.                              00012600
         WD.[2:1]=1 IF A "REPLACEMENT" RECORD ( NOT INSERTION )             00012700
       THE ROUTINE SETS "COPYTOG" TO TRUE WHEN A COPY                       00012800
       (SOURCE FILE TO WRKFIL) IS REQUIRED. A COPY IS REQUIRED WHEN         00012900
         A) A RECORD HAS BEEN DELETED FROM THE SOURCE FILE                  00013000
         B) A RECORD MUST BE "INSERTED" INTO THE SOURCE FILE                00013100
       A COPY WILL -NOT- BE REQUIRED WHEN                                   00013200
         A) A FIX IS EXECUTED ON AN EXISTING RECORD                         00013300
         B) AN "EXISTING" RECORD IS "REPLACED" WITH A NEW RECORD            00013400
       IF PRINTING IS REQUESTED, THE REQUIRED RECORDS ARE                   00013500
       PRINTED DURING THE MERGE.                                            00013600
       ALL RECORDS ARE ASSUMED TO CONTAIN A SEQUENCE NUMBER IN              00013700
       FIELD POSITIONS 73 THRU 80 EXCEPT "TYPE DATA" FILES,                 00013800
       WHICH ARE ADDRESSED BY RELATIVE POSITION (STARTING WITH              00013900
       RECORD NUMBER 1) IN THE FILE;                                        00014000
FILE PFILE  DISK SERIAL (2,80); % "1P" FILE                                 00014100
FILE SOURCE DISK RANDOM (2,10,30); % RANDOM SOURCE FILE                     00014200
FILE SRCE   DISK SERIAL (2, 0, 0); % SERIAL SOURCE FILE                     00014300
FILE OLDTAB DISK SERIAL (2,30,300); % OLD "1T" FILE                        00014400
FILE PATCH  DISK SERIAL [20:600] (2,10,300); % SCRATCH PAD FILE            00014500
FILE NEWTAB DISK SERIAL [20:30] (2,30,300,SAVE 1); % NEW "1T" FILE         00014600
FILE WRKFIL DISK SERIAL [20:600] (2,10,300,SAVE 1); % NEW "1S" FILE        00014700
FILE MONFIL DISK SERIAL (2,10,300); % MONITOR FILE                         00014710
SAVE ARRAY  A,B[0:30],  SORTA,MERGE[0:512],                                 00014800
```

```
PARAMS[0:10], BASE[0:16], N,T[0:30];                                    00014900
ARRAY SORTAA,CHNGS[0:32,0:32];                                          00015000
REAL A1, A2, A3, A4, B1, B2, B3, B4, B5, EADRS,                         00015100
CODE, EOFMARK, HRANGE, I, LRANGE, LINE, MAX, N1, N2, NCHRS,             00015200
NCHG, NCT, NPARAMS, NP1R, NSEQ, NWORD, O1, O2,                          00015300
PC, PCT, PREV, PPTR, PSEQ, PWORD, RECSZ,                                00015400
SEQ, SZN, SZO, TANKADRS, TCT, TPTR, TSEQ, USER;                         00015500
BOOLEAN BFULL, BRAAK, CHANGES, COPYTOG, ENDTOG, EQLTOG,                 00015600
FINTOG, MONITR, MT, NUMBERED, PRNT, PRNTCHK, SFLG, SQUASHED;            00015700
DEFINE LEFTARROW = "←"#;                                                00015800
%***********************************************************************00016100
STREAM PROCEDURE MOVE(N,A,B); VALUE N;                                  00016200
%***********************************************************************00016300
   BEGIN LOCAL M;                                                       00016400
   SI:=LOC N; DI:=LOC M; DI:=DI+1; DS:=7 CHR;                           00016500
   SI:=A; DI:=B; DS:=N WDS; M(DS:=32 WDS; DS:=32 WDS);                  00016600
   END;                                                                 00016700
%***********************************************************************00016800
REAL STREAM PROCEDURE SIZEOFINPUT(A2); VALUE A2;                        00016900
%***********************************************************************00017000
   BEGIN                                                                00017100
   % ROUTINE OBTAINS NUMBER OF CHARACTERS IN USERS INPUT RECORD          00017200
   % FROM THE WORD WHICH PRECEEDS THE ACTUAL USERS INPUT.                00017300
   % THE ACTUAL USERS INPUT BEGINS AT ADDRESS "A2"                       00017400
   SI:=A2; SI:=SI-3; DI:=LOC SIZEOFINPUT; DI:=DI+5; DS:=3CHR;           00017500
   DI:=DI-3; DS:=3RESET;                                                00017600
   END STREAM PROCEDURE SIZEOFINPUT;                                    00017700
%***********************************************************************00017800
PROCEDURE SORT(L,U); % RECURSIVE SORT ROUTINE                           00017900
%***********************************************************************00018000
VALUE L,U;                                                              00018100
REAL L,U;                                                               00018200
   BEGIN REAL I,J,K,M;                                                  00018300
   LABEL AGAIN,TOP,BOTTOM,EXIT;                                         00018400
   IF L NEQ U THEN                                                      00018500
   IF L+1=U THEN                                                        00018600
      BEGIN IF SORTA[L].[21:27] GTR SORTA[U].[21:27] THEN               00018700
      DOUBLE(SORTA[L],SORTA[U],←,SORTA[U],SORTA[L]);                    00018800
      END                                                               00018900
   ELSE                                                                 00019000
      BEGIN M:=(U+L) DIV 2;                                             00019100
      SORT(L,M); SORT(M+1,U);                                           00019200
      J:=M+1;                                                           00019300
      FOR L:=L STEP 1 WHILE SORTA[L].[21:27] LSS SORTA[J].[21:27] DO;   00019400
      IF L LEQ M THEN                                                   00019500
         BEGIN I:=K:=L;                                                 00019600
AGAIN:   IF I GTR M THEN GO TO TOP;                                     00019700
         IF J GTR U THEN GO TO BOTTOM;                                  00019800
         IF SORTA[I].[21:27] LEQ SORTA[J].[21:27] THEN GO TO BOTTOM;    00019900
TOP:     MERGE[K]:=SORTA[J];                                            00020000
         J:=J+1;                                                        00020100
         IF K:=K+1 LEQ U THEN GO TO AGAIN ELSE GO EXIT;                 00020200
BOTTOM:  MERGE[K]:=SORTA[I];                                            00020300
         I:=I+1;                                                        00020400
         IF K:=K+1 LEQ U THEN GO TO AGAIN;                              00020500
EXIT:    MOVE(U-L+1,MERGE[L],SORTA[L]);                                 00020600
         END                                                            00020700
      END                                                               00020800
   END SORT;                                                            00020900
%***********************************************************************00021000
```

```
PROCEDURE DISKWAIT(I,A,S,D);                                                    00021100
VALUE I,S,D; REAL I,S,D; ARRAY A[*]; COMMUNICATE(-8);                           00021200
%***************************************************************************00021300
PROCEDURE TWXOUT(A,N,T);                                                        00021400
%***************************************************************************00021500
VALUE N,T;                                                                      00021600
REAL A,N,T;                                                                     00021700
   BEGIN  COMMUNICATE(-11);                                                     00021800
   BRAAK := BRAAK OR BOOLEAN(T); % MCP RETURNS 1 IF BREAK OCCURRED.             00021900
   END;                                                                         00022000
%***************************************************************************00022100
REAL STREAM PROCEDURE EOFCOUNT(F);                                              00022200
%***************************************************************************00022300
   BEGIN                                                                        00022400
   SI:=F;  3(SI:=SI-8); DI:=LOC F; DS:=WDS;                                     00022500
   SI:=F; 14(SI:=SI+8); DI:=LOC F; DS:=WDS;                                     00022600
   SI:=F;  7(SI:=SI+8); DI:=LOC EOFCOUNT; DS:=WDS;                              00022700
   END STREAM PROCEDURE EOFCOUNT;                                               00022800
%***************************************************************************00022900
PROCEDURE MSG(NO,SEQ); VALUE NO,SEQ; REAL NO,SEQ;                               00023000
%***************************************************************************00023100
   BEGIN REAL SIZ,SZ;                                                           00023200
   REAL STREAM PROCEDURE NCONV(N,SIZ); VALUE N;                                 00023300
      BEGIN                                                                     00023400
      SI:=LOC N; DI:=LOC NCONV; DS:=8DEC; DI:=DI-8; DS:=7FILL;                  00023500
      DI:=LOC NCONV; SI:=LOC NCONV; IF SC=" " THEN                              00023600
         BEGIN                                                                  00023700
         8(IF SC=" " THEN SI:=SI+1 ELSE                                         00023800
            BEGIN DS:=CHR; TALLY:=TALLY+1; END);                                00023900
         END ELSE TALLY:=8;                                                     00024000
      N:=TALLY; SI:=LOC N; DI:=SIZ; DS:=WDS;                                    00024100
      END PROCEDURE NCONV;                                                      00024200
   %.....................................................................00024300
   REAL STREAM PROCEDURE M1(A,N,SIZ); VALUE N,SIZ;                              00024400
      BEGIN                                                                     00024500
      DI:=A; DS:=7LIT"RECORD "; SI:=LOC N; DS:= SIZ CHR;                        00024600
      DS:=9 LIT " DELETED."; TALLY:=16; M1:=TALLY;                              00024700
      END STREAM PROCEDURE M1;                                                  00024800
   %.....................................................................00024900
   REAL STREAM PROCEDURE M2(A,N,SIZ); VALUE N,SIZ;                              00025000
      BEGIN                                                                     00025100
      DI:=A; DS:=7LIT"RECORD "; SI:=LOC N; DS:=SIZ CHR;                         00025200
      DS:=21LIT" IS NOT IN YOUR FILE."; TALLY:=28; M2:=TALLY;                   00025300
      END STREAM PROCEDURE M2;                                                  00025400
   %.....................................................................00025500
   REAL STREAM PROCEDURE M3(A,N,SIZ); VALUE N,SIZ;                              00025600
      BEGIN                                                                     00025700
      DI:=A; DS:=24LIT"IMPROPER FIX FOR RECORD ";                               00025800
      SI:=LOC N; DS:=SIZ CHR; TALLY:=24; M3:=TALLY;                             00025900
      END STREAM PROCEDURE M3;                                                  00026000
   %.....................................................................00026100
   REAL STREAM PROCEDURE M4(A,N,SIZ); VALUE N,SIZ;                              00026200
      BEGIN                                                                     00026300
      DI:=A; DS:=33LIT"MISSING GROUP MARK IN FIX RECORD ";                      00026400
      SI:=LOC N; DS:=SIZ CHR; TALLY:=33; M4:=TALLY;                             00026500
      END STREAM PROCEDURE M4;                                                  00026600
   %.....................................................................00026700
   REAL STREAM PROCEDURE M5(A,N,SIZ); VALUE N,SIZ;                              00026800
      BEGIN                                                                     00026900
      DI:=A; DS:=41LIT"CANNOT LOCATE YOUR FIX STRING FOR RECORD ";              00027000
```

```
      SI:=LOC N; DS:=SIZ CHR; TALLY:=41; M5:=TALLY;                          00027100
      END STREAM PROCEDURE M5;                                               00027200
    %.....................................................................00027300
    REAL STREAM PROCEDURE M6(A,N,SIZ); VALUE N,SIZ;                          00027400
      BEGIN                                                                  00027500
      DI:=A; DS:=39LIT"NOT ENOUGH ROOM FOR YOUR FIX IN RECORD ";             00027600
      SI:=LOC N; DS:= SIZ CHR; TALLY:=39; M6:=TALLY;                         00027700
      END STREAM PROCEDURE M6;                                               00027800
    %.....................................................................00027900
    SEQ := NCONV(SEQ,SIZ);                                                   00028000
    IF NO=1 THEN SZ:=M1(A,SEQ,SIZ) ELSE                                      00028100
    IF NO=2 THEN SZ:=M2(A,SEQ,SIZ) ELSE                                      00028200
    IF NO=3 THEN SZ:=M3(A,SEQ,SIZ) ELSE                                      00028300
    IF NO=4 THEN SZ:=M4(A,SEQ,SIZ) ELSE                                      00028400
    IF NO=5 THEN SZ:=M5(A,SEQ,SIZ) ELSE SZ:=M6(A,SEQ,SIZ);                   00028500
    TWXOUT(A[0],(SIZ+SZ),1);                                                 00028600
    END PROCEDURE MSG;                                                       00028700
%*********************************************************************00028800
DEFINE TRB=(IF SB THEN DS:=SET ELSE DS:=RESET;SKIP SB)#;                     00028900
DEFINE FTC=SI:=SI+3; DI:=DI+5; SKIP 3 DB;15TRB#;                             00029000
DEFINE CTF=SI:=SI+5; SKIP 3 SB; DI:=DI+3;15TRB#;                             00029100
%*********************************************************************00029200
STREAM PROCEDURE RETURN(A,B);                                                00029300
%*********************************************************************00029400
    BEGIN LOCAL AT,BT;                                                       00029500
    SI:=LOC A; DI:=LOC AT; FTC;                                              00029600
    SI:=LOC B; DI:=LOC BT; FTC;                                              00029700
    SI:=LOC BT; DI:=LOC A; CTF;                                              00029800
    SI:=LOC AT; DI:=LOC B; CTF;                                              00029900
    DI:=AT; SI:=LOC B; DS:=WDS;                                              00030000
    DI:=BT; SI:=LOC A; DS:=WDS;                                              00030100
    END;                                                                     00030200
%*********************************************************************00030300
PROCEDURE SWAP;                                                              00030400
%*********************************************************************00030500
    BEGIN                                                                    00030600
    ARRAY TS,TM[0:1];                                                        00030700
    RETURN(SORTA,TS); RETURN(MERGE,TM);                                      00030800
    END PROCEDURE SWAP;                                                      00030900
%*********************************************************************00031000
STREAM PROCEDURE MAKE300(FIIL,SPECS); VALUE SPECS;                           00031100
%*********************************************************************00031200
    BEGIN                                                                    00031300
    SI:=FIIL;  3(SI:=SI-8); DI:=LOC FIIL; DS:=WDS;                           00031400
    SI:=FIIL; 14(SI:=SI+8); DI:=LOC FIIL; DS:=WDS;                           00031500
    SI:=LOC SPECS; DI:=FIIL; DS:=WDS; % BLOCKING FACTORS                     00031600
    END STREAM PROCEDURE MAKE300;                                            00031700
%*********************************************************************00031800
REAL STREAM PROCEDURE DECONV(X); VALUE X;                                    00031900
    BEGIN SI:=LOC X; DI:=LOC DECONV; DS:= 8 DEC; END;                        00032000
%*********************************************************************00032100
REAL STREAM PROCEDURE WHEREIS(X); % GET ADDRESS OF "X"                       00032200
    BEGIN SI:=X; WHEREIS:=SI; END;                                           00032300
%*********************************************************************00032400
REAL PROCEDURE SIZ(A1,A2); VALUE A1,A2; REAL A1,A2; % GET SIZE FROM ADRS00032500
SIZ:=(A2.[33:15]-A1.[33:15])x8+A2.[30:3]-A1.[30:3];                          00032600
%*********************************************************************00032700
STREAM PROCEDURE REFORMAT(A2,B1,B4,SEQ,IFLG,SFLG,EADRS);                     00032800
%*********************************************************************00032900
VALUE A2,B1,B4,SEQ,IFLG,SFLG,EADRS;                                          00033000
```

```
COMMENT A2 IS STARTING ADDRESS OF STRING IN ARRAY "A",                  00033100
B1 IS ADDRESS OF B[0],                                                  00033200
B4 IS ADDRESS OF B[9] IF NOT TYPE DATA, OF B[10] IF TYPE DATA           00033300
SEQ IS OCTAL VALUE OF SEQUENCE NUMBER IF NOT TYPE DATA,                 00033400
IFLG IS TRUE IF SEQUENCE NUMBER APPEARS IN INPUT RECORD,                00033500
SFLG IS TRUE IF NOT TYPE DATA FILE.                                     00033600
EADRS IS ADDRESS OF LAST LEFT ARROW IN USERS INPUT RECORD               00033700
ROUTINE IS USED TO CONVERT INPUT FROM TANK TO STANDARD (80 CHR) FORM;    00033800
   BEGIN LABEL L; LOCAL SV,DV; DI:=EADRS; DS:=LIT LEFTARROW;            00033900
   DI:=B1; DS:=8LIT" "; SI:=B1; DS:=9WDS; SI:=A2;                       00034000
   IFLG( % SEQUENCE NUMBER IN RECORD                                    00034100
   63(IF SC=" " THEN SI:=SI+1 ELSE JUMP OUT);                           00034200
   08(IF SC LSS "0" THEN JUMP OUT;IF SC GTR "9" THEN JUMP OUT;SI:=SI+1);00034300
   );                                                                   00034400
   DI:=B1;                                                              00034500
   2(40(                                                                00034600
   IF SC=LEFTARROW THEN % CHECK FOR END OF RECORD                       00034700
      BEGIN                                                             00034800
      SV:=SI; SI:=LOC SV; SI:=SI+5;                                     00034900
      DV:=DI; DI:=LOC EADRS; DI:=DI+5;                                  00035000
      IF 3SC=DC THEN JUMP OUT 2 TO L;                                   00035100
      SI:=SV; DI:=DV;                                                   00035200
      END;                                                              00035300
   DS:=CHR));                                                           00035400
L: SFLG(SI:=LOC SEQ; DI:=B4; DS:=8DEC);                                 00035500
   END STREAM PROCEDURE REFORMAT;                                       00035600
%*********************************************************************00035700
BOOLEAN STREAM PROCEDURE FIXCHK(A2,O1,O2,N1,N2,EADRS); VALUE A2,EADRS;  00035800
%*********************************************************************00035900
COMMENT A2 IS STARTING ADDRESS OF STRING IN ARRAY "A",                  00036000
O1,O2 ARE START/FINISH ADDRESS OF "OLD STRING",                         00036100
N1,N2 ARE START/FINISH ADDRESS OF "NEW STRING",                         00036200
EADRS IS ADDRESS OF LAST LEFT ARROW IN USERS INPUT RECORD               00036300
ROUTINE EXAMINES "FIX" SYNTAX AND ESTABLISHES VALUES FOR                00036400
O1,O2,N1 AND N2;                                                        00036500
   BEGIN LOCAL SV,D; LABEL EXIT;                                        00036600
   DI:=EADRS; DS:=LIT LEFTARROW; SI:=A2; DI:=LOC D;                     00036700
   20(IF SC LSS "0" THEN SI:=SI+1 ELSE JUMP OUT), % SCAN TO DIGIT       00036800
   IF TOGGLE THEN GO TO EXIT; % NO DIGITS                               00036900
   08(IF SC LSS "0" THEN JUMP OUT;IF SC GTR "9" THEN JUMP OUT;SI:=SI+1);00037000
   20(IF SC EQL " " THEN SI:=SI+1 ELSE JUMP OUT); % SCAN TO CHAR,       00037100
   IF TOGGLE THEN GO TO EXIT; % EMPTY FIELD                             00037200
   DI:=LOC D; DI:=DI+7; DS:=CHR; % SAVE DELIMITER                       00037300
   IF SC=D THEN GO TO EXIT; % NO STRING                                 00037400
   IF SC = LEFTARROW THEN                                               00037500
      BEGIN                                                             00037600
      SV:=SI; SI:=LOC SV; SI:=SI+5;                                     00037700
      DI:=LOC EADRS; DI:=DI+5;                                          00037800
      IF 3SC=DC THEN % LEFT ARROW REPLACEMENT                           00037900
         BEGIN                                                          00038000
         SI:=SV; SI:=SI-1; SV:=SI; SI:=LOC SV; DI:=O1; DS:=WDS; SI:=SV; 00038100
         SI:=SI+1; SV:=SI; SI:=LOC SV; DI:=O2; DS:=WDS;                 00038200
         SI:=LOC SV; DI:=N1; DS:=WDS; SI:=SV;                           00038300
         SI:=SI+1; SV:=SI; SI:=LOC SV; DI:=N2; DS:=WDS;                 00038400
         TALLY:=1; FIXCHK:=TALLY; GO TO EXIT;                           00038500
         END LEFT ARROW REPLACEMENT;                                    00038600
      SI:=SV;                                                           00038700
      END; % IF LEFTARROW                                               00038800
   SV:=SI; SI:=LOC SV; DI:=O1; DS:=WDS; SI:=SV; % START OF OLD STRING   00038900
   63(                                                                  00039000
```

```
   IF SC=LEFTARROW THEN % CHECK FOR END OF RECORD                          00039100
      BEGIN                                                               00039200
      SV:=SI; SI:=LOC SV; SI:=SI+5;                                       00039300
      DI:=LOC EADRS; DI:=DI+5;                                            00039400
      IF 3SC=DC THEN JUMP OUT TO EXIT ELSE SI:=SV;                        00039500
      END;                                                                00039600
   IF SC NEQ D THEN SI:=SI+1 ELSE JUMP OUT);                              00039700
   IF TOGGLE THEN GO TO EXIT; % MISSING 2ND DELIMITER                     00039800
   SV:=SI; SI:=LOC SV; DI:=02; DS:=WDS; SI:=SV; %END OLD STRING           00039900
   SI:=SI+1; SV:=SI; SI:=LOC SV; DI:=N1; DS:=WDS; SI:=SV; % NEW STRING    00040000
   63(IF SC=LEFTARROW THEN                                                00040100
        BEGIN                                                             00040200
        SV:=SI; SI:=LOC SV; SI:=SI+5; DI:=LOC EADRS; DI:=DI+5;            00040300
        IF 3SC=DC THEN JUMP OUT ELSE SI:=SV;                              00040400
        END;                                                              00040500
   SI:=SI+1);                                                             00040600
   SI:=LOC SV; DI:=N2; DS:=WDS; % END NEW STRING                          00040700
   TALLY:=1; FIXCHK:=TALLY;                                               00040800
EXIT:                                                                     00040900
   END STREAM PROCEDURE FIXCHK;                                           00041000
%*********************************************************************** 00041100
BOOLEAN STREAM PROCEDURE LOCATE(B1,B2,B3,B4,L,01,SZ0);                    00041200
%*********************************************************************** 00041300
COMMENT B1 IS ADDRESS OF B[0],                                            00041400
B2 IS ADDRESS OF START OF "EXISTING" STRING,                              00041500
B3 IS ADDRESS OF CHARACTER FOLLOWING LAST NON-BLANK CHR. IN RECORD,       00041600
B4 IF ADDRESS OF B[9] IF NOT TYPE DATA, OF B[10] IF TYPE DATA             00041700
L IS RECORD SIZE,(2X)36 CHR. IF NOT TYPE DATA, (2X)40 CHR. IF TYPE DATA.  00041800
ROUTINE MATCHES "OLD STRING" AT ADDRESS 01,LENGTH SZ0 WITH                00041900
CONTENTS OF RECORD FOR "FIX" STATEMENT,AND SETS B2 AND B3 IF MATCHED;     00042000
VALUE B1,B4,L,01,SZ0;                                                     00042100
   BEGIN LOCAL SV,D; LABEL L1,L2,EXIT;                                    00042200
   SI:=01; DI:=LOC D; D1:=DI+7; DS:=CHR; % SAVE FIRST CHARACTER           00042300
   SI:=B1;                                                                00042400
   2(L(IF SC=D THEN                                                       00042500
      BEGIN                                                               00042600
      SV:=SI; DI:=01; IF SZ0 SC = DC THEN                                 00042700
         BEGIN                                                            00042800
         SI:=LOC SV; DI:=B2; DS:=WDS; JUMP OUT 2 TO L1;                   00042900
         END;                                                             00043000
      SI:=SV;                                                             00043100
      END;                                                                00043200
   SI:=SI+1));                                                            00043300
   GO TO EXIT;                                                            00043400
L1: SI:=B4; SI:=SI-1;                                                     00043500
   2(L(IF SC=" " THEN SI:=SI-1 ELSE JUMP OUT 2 TO L2));                   00043600
L2: SI:=SI+1; SV:=SI; SI:=LOC SV; DI:=B3; DS:=WDS;                        00043700
   TALLY:=1; LOCATE:=TALLY;                                               00043800
EXIT:                                                                     00043900
   END STREAM PROCEDURE LOCATE;                                           00044000
%*********************************************************************** 00044100
STREAM PROCEDURE EDIT(B2,B4,B5,SZ0,N1,SZN,SFLG);                          00044200
%*********************************************************************** 00044300
COMMENT B2 IS STARTING ADDRES OF "EXISTING" STRING,                       00044400
B4 IS ADDRESS OF B[9] IF NOT TYPE DATA, OF B[10] IF TYPE DATA,            00044500
B5 IS ADDRESS OF B[20],                                                   00044600
SZ0 IS LENGTH OF "EXISTING" STRING,                                       00044700
N1 IS STARTING ADDRESS OF REPLACEMENT STRING,                             00044800
SZN IS LENGTH OF REPLACEMENT STRING,                                      00044900
SFLG IS TRUE IF NOT TYPE DATA FILE.                                       00045000
```

```
ROUTINE REPLACES "OLD" STRING WITH "NEW" STRING FOR "FIX";                    00045100
VALUE B2,B4,B5,SZO,N1,SZN,SFLG;                                               00045200
   BEGIN LOCAL SEQ;                                                           00045300
   SFLG(SI:=B4; DI:=LOC SEQ; DS:=WDS); % SAVE SEQUENCE NO.                    00045400
   DI:=B4; 2(DS:=40LIT" ");                                                   00045500
   SI:=B2; BI:=SI+SZO; DI:=B5; 2(DS:=40CHR);                                  00045600
   SI:=N1; DI:=B2; DS:=SZN CHR;                                               00045700
   SI:=B5; 2(DS:=40CHR);                                                      00045800
   SFLG(SI:=LOC SEQ; DI:=B4; DS:=WDS);                                        00045900
   END STREAM PROCEDURE EDIT;                                                 00046000
%*************************************************************************00046100
STREAM PROCEDURE OUTFORMAT(A1,B1,B4,SQSH,SFLG,PRNT);                          00046200
%*************************************************************************00046300
COMMENT A1 IS ADDRESS OF A[0], B1 IS ADDRESS OF B[0],                         00046400
B4 IS ADDRESS OF B[9] IF NUI TYPE DATA, OF BL10] IF TYPE DATA,                 00046500
SQSH IS TRUE IF "SQUASHED" OPTION SET.                                        00046600
SFLG=TRUE IF NOT TYPE DATA FILE, PRNT=TRUE IF NOT "DISPLAY" VERB              00046700
ROUTINE REFORMATS RECORD FOR OUTPUT TO REMOTE STATION;                        00046800
VALUE A1,B1,B4,SQSH,SFLG,PRNT;                                                00046900
   BEGIN LOCAL SV,SEQ; LABEL EXIT;                                            00047000
      DI:=A1; DS:=8LIT" "; SI:=A1; DS:=10WDS; DI:=A1; % BLANK RCRD            00047100
   SFLG(SI:=B4; DI:=LOC SEQ; DS:=WDS; % SAVE SEQ. NUMBER                      00047200
   PRNT(DI:=LOC SEQ; DS:=7FILL); SI:=LOC SEQ; DI:=A1;                         00047300
   8(IF SC=" " THEN SI:=SI+1 ELSE DS:=CHR);                                   00047400
   PRNT(DS:=LIT" "); SV:=DI; DI:=B4; DS:=8LIT" ", DI:=SV);                    00047500
   SI:=B1;                                                                    00047600
   SQSH(                                                                      00047700
   2(40( IF SC=" " THEN                                                       00047800
      BEGIN SI:=SI+1; IF SC NEQ " " THEN DS:=LIT" "; END                      00047900
   ELSE DS:=CHR));                                                            00048000
   JUMP OUT TO EXIT);                                                         00048100
   2(DS:=40 CHR); DS:=LIT" ";                                                 00048200
EXIT:                                                                         00048300
   END STREAM PROCEDURE OUTFORMAT;                                            00048400
%*************************************************************************00048500
STREAM PROCEDURE REKNUMBER(A1,LREC); VALUE A1,LREC;                           00048600
%*************************************************************************00048700
   BEGIN                                                                      00048800
   SI:=LOC LREC; DI:=A1; DS:=8DEC; DI:=A1; DS:=7FILL;                         00048900
   SI:=A1; DI:=A1; 8(IF SC=" " THEN SI:=SI+1 ELSE DS:=CHR);                   00049000
   DS:=8LIT" ";                                                               00049100
   END STREAM PROCEDURE REKNUMBER;                                            00049200
%*************************************************************************00049300
PROCEDURE NEXTPATCH;                                                          00049400
%*************************************************************************00049500
   BEGIN                                                                      00049600
   IF NPTR:=NPTR+1 LEQ NCHG THEN                                              00049700
      BEGIN                                                                   00049800
      NWORD:=CHNGS[NPTR.[38:5],NPTR.[43:5]];                                  00049900
      N1:=NWORD.[32:16]; % SOURCE RECORD LOCATION                             00050000
      END                                                                     00050100
   ELSE FINTOG:=TRUE;                                                         00050200
   END PROCEDURE NEXTPATCH;                                                   00050300
%*************************************************************************00050302
STREAM PROCEDURE BLANKLINE(A,SEQN); VALUE SEQN;                               00050304
%*************************************************************************00050306
   BEGIN                                                                      00050308
   DI:=A; DS:=8LIT" "; SI:=A; DS:=8WDS; SI:=LOC SEQN; DS:=8DEC;               00050310
   END;                                                                       00050312
%*************************************************************************00050314
```

```
PROCEDURE MONITORHEADER;                                                    00050316
%*************************************************************************00050318
   BEGIN INTEGER TIM,HRS,MIN,AMPM;                                          00050320
   STREAM PROCEDURE MESG(A,NAM,HRS,MIN,AMPM); VALUE NAM,HRS,MIN,AMPM;       00050322
      BEGIN                                                                 00050324
      DI:=A; DS:=8LIT" "; SI:=A; DS:=8WDS; DS:=8LIT"*";                     00050326
      DI:=A; DS:=8LIT"MONITOR "; SI:=LOC NAM; SI:=SI+1; DS:=7CHR;           00050328
      DS:=LIT" "; DS:=2DEC; DS:=LIT":"; DS:=2DEC;                           00050330
      DS:=LIT" "; SI:=SI+4; DS:=4CHR;                                       00050332
      END STREAM PROCEDURE MESG;                                            00050334
   HRS:=(TIM:=TIME(1)) DIV 216000; MIN:=(TIM DIV 3600) MOD 60;              00050336
   AMPM:=IF HRS GTR 11 THEN "P.M." ELSE "A.M.";                             00050338
   IF HRS GTR 12 THEN HRS := HRS - 12;                                      00050340
   MESG(B,A[3],HRS,MIN,AMPM); WRITE(MONFIL,10,B[*]);                        00050342
   END PROCEDURE MONITORHEADER;                                             00050344
%*************************************************************************00050400
%*************************************************************************00050500
BEGIN % INNER BLOCK                                                         00050510
LABEL  COMPARE, ENDMERGE, NEXTP, PCYCLE, MONEOF, SKIP,                      00050520
STARTMERGE,  TLOOP,  WRITENEW,  CYCLE,  READIN,  EOF,  ENDLOOP;            00050530
EOFMARK:=100000000;                                                         00050600
A[0]:=0; A1:=WHEREIS(A[0]); A4:=WHEREIS(A[29]); % SET ADDRESSES             00050700
DISKWAIT(1,A,30,COMMON); % GET ESP RECORD                                   00050800
LINE:=DECONV(A[1].[40:8]); % DECIMAL VALUE OF LINE NO                       00050900
CHANGES :=BOOLEAN(A[1].[1:1]);                                              00051000
SQUASHED:=BOOLEAN(A[1].[2:1]);                                              00051100
SFLG := A[1].[3:1]=0; % ON IF TYPE DATA FILE                                00051200
NUMBERED := A[1].[4:1]=1 AND NOT SFLG;                                      00051300
PRNT := A[1].[7:1] = 0; % ON, IF "DISPLAY" VERB                             00051400
B[0]:=0; B1:=WHEREIS(B[0]); % "B" ADDRESSES                                 00051500
B4:=WHEREIS(B[10-REAL(SFLG)]); B5:=WHEREIS(B[20]);                          00051600
RECSZ:= 40 - 4 × REAL(SFLG); % 1/2 RECORD LENGTH                            00051700
PRNTCHK := (NPARAMS:=A[1].[27:6]) GTR 0; % PARAMETER COUNT                  00051800
USER := A[2];                                                               00051900
FILL WRKFIL WITH "  "&"1S"[6:36:12]&LINE[18:30:18],USER;                    00052000
FILL OLDTAB WITH "  "&"1T"[6:36:12]&LINE[18:30:18],USER;                    00052100
FILL NEWTAB WITH "  "&"1T"[6:36:12]&LINE[18:30:18],USER;                    00052200
FILL PFILE  WITH "  "&"1P"[6:36:12]&LINE[18:30:18],USER;                    00052300
FILL PATCH  WITH "  "&"2P"[6:36:12]&LINE[18:30:18],USER;                    00052400
FILL SOURCE WITH A[3],USER; % RANDOM SOURCE FILE                            00052500
FILL SRCE   WITH A[3],USER; % SERIAL SOURCE FILE                            00052600
IF A[3].[6:12] NEQ "1S" THEN COPYTOG:=TRUE; % COPY REQD IF NOT "1S" FILE    00052700
MOVE(16,A[5],BASE); % TANK FILE ADDRESSES                                   00052800
IF PRNTCHK THEN % PRINTING PARAMETERS GIVEN                                 00052900
   BEGIN % GET SEQUENCE PARAMETERS                                          00053000
   MOVE(9,A[21],PARAMS); NPARAMS:=NPARAMS-1;                                00053100
   LRANGE:=HRANGE:=PARAMS[PC:=0];                                           00053200
   IF PC LSS NPARAMS THEN                                                   00053300
   IF PARAMS[PC+1].[1:1]=1 THEN                                             00053400
   HRANGE:=PARAMS[PC:=PC+1].[21:27];                                        00053500
   ENDTOG:=LRANGE=EOFMARK; % PRINT "END"                                    00053600
   END;                                                                     00053700
IF MONITR:=BOOLEAN(A[1].[9:1]) THEN % MONITOR REQUESTED                     00053710
   BEGIN                                                                    00053712
   DISKWAIT(1,B,30,A[4]); % USERS/CANDE RECORD                              00053714
   FILL MONFIL WITH B[12],USER; % MONITOR FILE NAME                         00053716
   SEARCH(MONFIL,B[*]);                                                     00053718
   IF MONITR := B[0] GTR 0 THEN % FILE IS PRESENT                           00053720
      BEGIN                                                                 00053722
      READ SEEK (MONFIL[ B[5]+1 ]); % SET POINTER TO NEXT RECORD            00053724
```

```
        MONITORHEADER; % PUT CONTROL RECORD IN MONITOR FILE                    00053726
        END; % IF FILE IS PRESENT                                              00053728
    END; % IF MONITOR REQUESTED                                                00053730
% TRANSFER "1P" RECORDS TO ARRAY SORTA                                         00053800
READ(PFILE,30,A[*]); % 1ST RECORD                                              00053900
N1:=N2:=0;                                                                     00054000
WHILE A[N1] NEQ EOFMARK DO                                                     00054100
    BEGIN                                                                      00054200
    SORTA[N1+N2]:=A[N1];                                                       00054300
    IF N1:=N1+1 GTR 29 THEN                                                    00054400
        BEGIN                                                                  00054500
        N2:=N2+30; N1:=0;                                                      00054600
        READ(PFILE,30,A[*]);                                                   00054700
        END;                                                                   00054800
    END;                                                                       00054900
SORTA[N2:=N2+N1]:=EOFMARK;                                                     00055000
SORT(0,N2-1);                                                                  00055100
FOR N1:=0 STEP 32 UNTIL N2 DO                                                  00055200
MOVE(32,SORTA[N1],SORTAA[N1.[38:5],*]);                                        00055300
SWAP; % RETURN CORE SPACE                                                      00055400
%.........                                                                     00055500
STARTMERGE:                                                                    00055600
%.........                                                                     00055700
PSEQ:=(PWORD:=SORTAA[0,0]).[21:27]; % 1ST SEQ. NUMBER IN "1P" FILE             00055800
CODE:=PWORD.[1:2]; PPTR:=0;                                                    00055900
PCT:=NCHG:=NCT:=TCT:=-1;                                                       00056000
IF SFLG THEN % NOT TYPE DATA FILE                                              00056100
    BEGIN                                                                      00056200
    READ(OLDTAB,30,T[*]); TPTR:=NPTR:=0;                                       00056300
    END                                                                        00056400
ELSE                                                                           00056500
    BEGIN                                                                      00056600
    READ SEEK(SOURCE[0]); MAX:=EOFCOUNT(SOURCE)+1;                             00056700
    TSEQ := 0;                                                                 00056800
    END;                                                                       00056900
%.....                                                                         00057000
TLOOP:                                                                         00057100
%.....                                                                         00057200
IF SFLG THEN % SEQUENCED FILE                                                  00057300
    BEGIN                                                                      00057400
    IF TPTR:=TPTR+1 GTR 29 THEN % SEGMENT IS EXHAUSTED                         00057500
        BEGIN                                                                  00057600
        READ(OLDTAB,30,T[*]); % GET NEW SEGMENT                                00057700
        TPTR:=0;                                                               00057800
        END;                                                                   00057900
    TSEQ:=T[TPTR].[21:27]; TCT:=TCT+1; % OLD SEQ. NO. AND LOCATION             00058000
    IF TSEQ=EOFMARK THEN CLOSE(OLDTAB);                                        00058010
    END                                                                        00058100
ELSE                                                                           00058200
    BEGIN % NON-SEQUENCED TYPE FILE                                            00058300
    IF TSEQ := (TCT:=TSEQ) + 1 GTR MAX THEN                                    00058400
        BEGIN                                                                  00058500
        TSEQ := EOFMARK; CLOSE(OLDTAB);                                        00058600
        TCT := MAX;                                                            00058700
        END;                                                                   00058800
    END;                                                                       00058900
%.......                                                                       00059000
COMPARE:                                                                       00059100
%.......                                                                       00059200
IF TSEQ LSS PSEQ THEN % NO CHANGE TO THIS RECORD                               00059300
```

```
      BEGIN                                                                    00059400
      EQLTOG:=TRUE; SEQ:=TSEQ; GO TO WRITENEW;                                 00059500
      END;                                                                     00059600
   EQLTOG := TSEQ = PSEQ; % NEW RECORD SEQ. MATCHES OLD                        00059700
   %......                                                                     00059800
   PCYCLE:                                                                     00059900
   %......                                                                     00060000
   IF PSEQ=EOFMARK THEN GO TO ENDMERGE;                                        00060100
   IF CODE=1 THEN % DELETE CODE                                                00060200
      BEGIN                                                                    00060300
      BFULL:=FALSE;                                                            00060400
      PREV:=PSEQ; % SAVE CURRENT PSEQ NUMBER                                   00060500
      PPTR:=PPTR+1; % GET THE NEXT PSEQ NUMBER                                 00060600
      PSEQ:=(PWORD:=SORTAA[PPTR.[38:5],PPTR.[43:5]]).[21:27];                  00060700
      CODE:=PWORD.[1:2];                                                       00060800
      IF PSEQ=PREV AND CODE.[47:1]=0 THEN GO PCYCLE; % NEW RECORD FOLLOWS      00060900
      IF CHANGES THEN IF NOT BRAAK THEN MSG(1,PREV); % "DELETED" MESSAGE       00061000
      IF EQLTOG THEN % "OLD" RECORD IS DELETED                                 00061100
         BEGIN                                                                 00061200
         COPYTOG:=TRUE; % COPY WILL BE REQUIRED;                               00061300
         NCHG:=NCHG+1; % STORE "DELETE" INSTRUCTIONS FOR THIS RECORD           00061400
         CHNGS[NCHG.[38:5],NCHG.[43:5]]:=TCT&1[1:47:1]; % DELETE CODE          00061500
         IF MONITR THEN % PUT BLANK RECORD IN MONITOR FILE                     00061550
            BEGIN                                                              00061560
            BLANKLINE(A,PREV);                                                 00061570
            WRITE(MONFIL,10,A[*])[TLOOP];                                      00061580
            END;                                                               00061590
         GO TO TLOOP; % ADVANCE "OLDTAB" POINTER                               00061600
         END; % IF EQLTOG                                                      00061700
      GO TO COMPARE; % IF NOT EQLTOG                                           00061800
      END; % IF CODE = 1                                                       00061900
   IF CODE=3 THEN % FIX CODE, CHECK FOR RECORD FIRST                           00062000
      BEGIN                                                                    00062100
      IF NOT (EQLTOG OR BFULL) THEN % NO SUCH RECORD                           00062200
         BEGIN                                                                 00062300
         MSG(2,PSEQ); % "NOT IN YOUR FILE"                                     00062400
         GO TO NEXTP;                                                          00062500
         END;                                                                  00062600
      END;                                                                     00062700
   TANKADRS:=BASE[PWORD.[4:4]] + PWORD.[8:8]; % TANK ROW ADDRESS               00062800
   DISKWAIT(1,A,30,TANKADRS); % GET INPUT FROM TANK FILE                       00062900
   A2 := A1 + PWORD.[16:5]; % STARTING ADDRESS OF INPUT                        00063000
   NCHRS := SIZEOFINPUT(A2)-1; % POSITION OF LAST CHARACTER IN USERS INPUT     00063100
   EADRS := (A2 + NCHRS.[40:5]) & NCHRS[30:45:3]; % END OF RCRD ADDRESS        00063200
   IF CODE NEQ 3 THEN % NOT A FIX COMMAND                                      00063300
      BEGIN                                                                    00063400
      REFORMAT(A2,B1,B4,PSEQ,(PWORD.[1:2]=0),SFLG,EADRS);                      00063500
      BFULL:=TRUE; % NEW RECORD IN "B"                                         00063600
      GO TO NEXTP;                                                           00063700
      END;                                                                     00063800
   IF NOT FIXCHK(A2,O1,C2,N1,N2,EADRS) THEN % "FIX" SYNTAX ERROR               00063900
      BEGIN                                                                    00064000
      MSG(3,PSEQ); % "IMPROPER FIX"                                            00064100
      GO TO NEXTP;                                                           00064200
      END;                                                                     00064300
   IF N2.[33:15] GTR A4 THEN % SHOULD NOT BE HERE                              00064400
      BEGIN                                                                    00064500
      MSG(4,PSEQ); % "MISSING GROUP MARK"                                      00064600
      GO TO NEXTP;                                                           00064700
      END;                                                                     00064800
```

```
IF NOT BFULL THEN % GET RECORD TO BE "FIXED"                                00064900
   BEGIN                                                                    00065000
   READ(SOURCE[ICT],10,B[*]); % PUT IN "B"                                  00065100
   BFULL:=TRUE;                                                             00065200
   END;                                                                     00065300
IF NOT LOCATE(B1,B2,B3,B4,RECSZ,O1,SZO:=SIZ(O1,O2)) THEN                    00065400
   BEGIN                                                                    00065500
   MSG(5,PSEQ); % "CANNOT FIND FIX STRING"                                  00065600
   GO TO NEXTP;                                                             00065700
   END;                                                                     00065800
IF SIZ(B3,B4) LSS (SZN:=SIZ(N1,N2))-SZO THEN                                00065900
   BEGIN                                                                    00066000
   MSG(6,PSEQ); % "NOT ENOUGH ROOM FOR FIX"                                 00066100
   GO TO NEXTP;                                                             00066200
   END;                                                                     00066300
EDIT(B2,B4,B5,SZO,N1,SZN,SFLG); % REPLACE OLD STRING WITH NEW STRING        00066400
%.....                                                                      00066500
NEXTP: % LOOK AT NEXT "1P" WORD                                             00066600
%.....                                                                      00066700
PREV:=PSEQ; % SAVE CURRENT "1P" SEQUENCE NUMBER                             00066800
PPTR:=PPTR+1; % CHECK NEXT "1P" WORD                                        00066900
PSEQ:=(PWORD:=SORTAA[PPTR,[38:5],PPTR,[43:5]]).[21:27]; % NEXT "1P" WD.     00067000
CODE:=PWORD.[1:2];                                                          00067100
IF PREV=PSEQ THEN GO TO PCYCLE; % DUPLICATE, MORE TO BE DONE                00067200
IF NOT BFULL THEN GO TO COMPARE; % IGNORE LAST INPUT                        00067300
SEQ:=PREV; % USING THIS SEQ. NUMBER                                         00067400
WRITE(PATCH,10,B[*]); PCT:=PCT+1; % MOVE RECORD TO PATCH FILE AND COUNT     00067500
IF NOT EQLTOG THEN COPYTOG:=TRUE; % NEW ADDITION IF NOT EQLTOG              00067600
NCHG:=NCHG+1; % SAVE "ADDITION" INSTRUCTION                                 00067700
CHNGS[NCHG,[38:5],NCHG,[43:5]]:=TCT&(REAL(EQLTOG))[2:47:1];                 00067800
IF MONITR THEN WRITE(MONFIL,10,B[*])[MONEOF];                               00067810
GO TO SKIP; MONEOF: MONITR:=FALSE; SKIP:                                    00067820
IF CHANGES THEN IF NOT BRAAK THEN % PRINT THIS RECORD                       00067900
   BEGIN                                                                    00068000
   IF NUMBERED THEN                                                         00068100
      BEGIN                                                                 00068200
      RENUMBER(A1,NCT+2);                                                   00068300
      TWXOUT(A[0],9,1);                                                     00068400
      END;                                                                  00068500
   OUTFORMAT(A1,B1,B4,SQUASHED,SFLG,PRNT);                                  00068600
   TWXOUT(A[0],81,1);                                                       00068700
   END;                                                                     00068800
%........                                                                   00068900
WRITENEW:                                                                   00069000
%........                                                                   00069100
NCT:=NCT+1; % OUTPUT FILE RECORD COUNT                                      00069200
IF SFLG THEN % WRITE TAB FILE IF NOT TYPE DATA                              00069300
   BEGIN                                                                    00069400
   IF NPTR:=NPTR+1 GTR 29 THEN % SEGMENT IS FILLED                          00069500
      BEGIN                                                                 00069600
      WRITE(NEWTAB,30,N[*]); % OUTPUT TO DISK                               00069700
      NPTR:=0;                                                              00069800
      END;                                                                  00069900
   N[NPTR]:=0&SEQ[21:21:27]&NCT[4:32:16];                                   00070000
   END;                                                                     00070100
IF NOT BRAAK AND PRNTCHK THEN % PRINTING STILL OK                           00070200
   BEGIN                                                                    00070300
   IF SEQ GTR HRANGE THEN % GET NEW PARAMETERS                              00070400
   IF PRNTCHK := PC LSS NPARAMS THEN                                        00070500
      BEGIN                                                                 00070600
```

```
            LRANGE:=HRANGE:=PARAMS[PC:=PC+1];                                  00070700
            ENDTOG:=LRANGE=EOFMARK; % WANTS "END" PRINTED                      00070800
            IF PC LSS NPARAMS THEN                                             00070900
            IF PARAMS[PC+1].[1:1]=1 THEN                                       00071000
            HRANGE:=PARAMS[PC:=PC+1].[21:27];                                  00071100
            END;                                                               00071200
         IF NOT (BFULL AND CHANGES) THEN % NOT ALREADY PRINTED                 00071300
         IF SEQ GEQ LRANGE THEN IF SEQ LEQ HRANGE THEN % IN RANGE              00071400
            BEGIN                                                              00071500
            IF NOT BFULL THEN % GET RECORD                                     00071600
            READ(SOURCE[TCT],10,B[*]);                                         00071700
            IF NUMBERED THEN                                                   00071800
               BEGIN                                                           00071900
               REKNUMBER(A1,NCT+1);                                            00072000
               TWXOUT(A[0],9,1);                                               00072100
               END;                                                            00072200
            OUTFORMAT(A1,B1,B4,SQUASHED,SFLG,PRNT);                            00072300
            TWXOUT(A[0],81,1);                                                 00072400
            END;                                                               00072500
      END; % IF NOT BRAAK AND PRNTCHK                                          00072600
   BFULL:=FALSE;                                                               00072700
   IF EQLTOG THEN GO TO TLOOP ELSE GO TO COMPARE;                              00072800
   %........                                                                   00072900
   ENDMERGE:                                                                   00073000
   %........                                                                   00073100
   IF ENDTOG THEN IF NOT BRAAK THEN % PRINT "END"                              00073200
      BEGIN                                                                    00073300
      N1 := NCHG; N2 := TCT-1;                                                 00073400
      IF NCHG GEQ 0 THEN                                                       00073405
         BEGIN                                                                 00073410
   ENDLOOP:                                                                    00073415
         IF BOOLEAN( (I:=CHNGS[N1.[38:5],N1.[43:5]]).[1:1]) THEN % DELETION    00073420
            BEGIN                                                              00073425
            IF I.[32:16]=N2 THEN N2:=N2-1; % DELETED SOURCEFILE RECORD         00073430
            IF (N1:=N1-1) GEQ 0 THEN GO TO ENDLOOP;                            00073435
            END;                                                               00073440
         IF BOOLEAN(I.[1:1]) THEN I:= -1 ELSE I:=I.[32:16];                    00073445
         END ELSE I := -1;                                                     00073450
      IF (N2 GEQ 0) OR (I GEQ 0) THEN % THERE ARE RECORDS IN THE FILE          00073455
      IF NOT (I GEQ TCT AND CHANGES) THEN % NOT ALREADY PRINTED                00073500
         BEGIN                                                                 00073600
         IF I GTR N2 THEN READ(PATCH[PCT],10,B[*]) % GET FROM PATCH            00073700
         ELSE READ(SOURCE[N2],10,B[*]);                                        00073800
         IF NUMBERED THEN                                                      00073900
            BEGIN                                                              00074000
            REKNUMBER(A1,NCT+1);                                               00074100
            TWXOUT(A[0],9,1);                                                  00074200
            END;                                                               00074300
         OUTFORMAT(A1,B1,B4,SQUASHED,SFLG,PRNT);                               00074400
         TWXOUT(A[0],81,1);                                                    00074500
         END;                                                                  00074600
      END;                                                                     00074700
   IF SFLG THEN % FINISH UP TAB FILES                                          00074800
      BEGIN                                                                    00074900
      IF NPTR:=NPTR+1 GTR 29 THEN % SEGMENT IS FILLED                          00075000
         BEGIN                                                                 00075100
         WRITE(NEWTAB,30,N[*]); % OUTPUT TO DISK                               00075200
         NPTR:=0;                                                              00075300
         END;                                                                  00075400
      FOR I:=NPTR STEP 1 UNTIL 29 DO N[I]:=EOFMARK;                            00075500
```

```
      WRITE(NEWTAB,30,N[*]);                                          00075600
      END;                                                            00076100
   REWIND(PATCH);                                                     00076200
   IF NOT COPYTOG THEN % COPY NOT REQUIRED                            00076300
      BEGIN                                                           00076400
      FOR I:=0 STEP 1 UNTIL NCHG DO                                   00076500
         BEGIN                                                        00076600
         N1:=CHNGS[I,[38:5],I,[43:5]],[32:16];                        00076700
         READ(PATCH,10,B[*]); % REPLACEMENT RECORD                    00076800
         WRITE(SOURCE[N1],10,B[*]); % INSERT RECORD                   00076900
         END;                                                         00077000
      IF NCHG GEQ 0 THEN % RE-ADJUST BLOCKING FACTORS                 00077100
         BEGIN                                                        00077200
         MAKE300(SOURCE,(10&30[30:36:12]&300[15:33:15]&10[1:34:14])); 00077300
         CLOSE(SOURCE); LOCK(NEWTAB,*);                               00077400
         END;                                                         00077500
      END % IF COPY NOT REQUIRED                                      00077600
   ELSE                                                               00077700
      BEGIN                                                           00077800
      CLOSE(SOURCE); NPTR:=I:=-1; FINTOG:=FALSE; MT:=TRUE; NEXTPATCH; 00077900
   %......                                                            00078000
   READIN:                                                            00078100
   %......                                                            00078200
      READ(SRCE,10,B[*])[EOF]; I:=I+1; % READ SERIAL FILE             00078300
   %.....                                                             00078400
   CYCLE:                                                             00078500
   %.....                                                             00078600
      IF NOT FINTOG THEN % MORE CHANGES                               00078700
      IF I=N1 THEN % INSERTION,REPLACEMENT OR DELETION                00078800
         BEGIN                                                        00078900
         IF NWORD.[1:1]=1 THEN % DELETION                             00079000
            BEGIN NEXTPATCH; GO TO READIN; END;                       00079100
         IF NWORD.[2:1]=0 THEN % INSERTION, NOT REPLACEMENT           00079200
            BEGIN                                                     00079300
            READ(PATCH,10,A[*]);                                      00079400
            WRITE(WRKFIL,10,A[*]); MT:=FALSE;                         00079500
            NEXTPATCH; GO TO CYCLE;                                   00079600
            END;                                                      00079700
         % REPLACEMENT                                               00079800
         READ(PATCH,10,B[*]);                                         00079900
         NEXTPATCH;                                                   00080000
         END; % IF I=N1                                               00080100
      WRITE(WRKFIL,10,B[*]); MT:=FALSE;                               00080200
      GO TO READIN;                                                   00080300
   %...                                                               00080400
   EOF:                                                               00080500
   %...                                                               00080600
      WHILE NOT FINTOG DO                                             00080700
         BEGIN                                                        00080800
         READ(PATCH,10,B[*]);                                         00080900
         WRITE(WRKFIL,10,B[*]); MT:=FALSE;                            00081000
         NEXTPATCH;                                                   00081100
         END;                                                         00081200
      IF MT THEN READ SEEK(WRKFIL[0]); % OPEN OUTPUT FILE             00081250
      CLOSE(SRCE); LOCK(WRKFIL,*); LOCK(NEWTAB,*);                    00081300
      END; % IF COPYTOG                                               00081400
   END BLOCK;                                                         00081410
   END PROGRAM.                                                       00081500
   END;END.          LAST CARD ON OCRDING TAPE                        99999999
                        CNR ← CNR + DL ;% LITERAL.                    28480000
```

LABEL 000000000PRINTER0017510OCC cX OBJECT/READ;FILE SOURCEFILE=SYMBOL/LIST;END←

OBJECT /READ